
Pyto

Feb 11, 2020

Contents

1	Automation with Shortcuts and x-callback URLs	3
1.1	Shortcuts	3
1.2	x-callback URLs	3
2	OpenCV	5
2.1	Pyto specific API	5
2.2	Face detection example	5
3	Third Party Libraries	7
3.1	Modules	7
3.2	Acknowledgments	8
4	API Reference	85
4.1	pyto_ui	85
4.2	notification_center	117
4.3	sound	118
4.4	sharing	119
4.5	mainthread	120
4.6	UIKit	120
4.7	notifications	120
4.8	music	121
4.9	photos	127
4.10	location	128
4.11	motion	128
4.12	multipeer	130
4.13	xcallback	130
4.14	userkeys	131
4.15	pasteboard	132
	Python Module Index	133
	Index	135



Pyto is an open source app to code and run Python code locally on an iPad or iPhone. The app uses the Python C API to run Python code in the same process of the app, due to iOS restrictions. Third party pure Python modules can be installed from PyPi and some modules are bundled in the app, like Numpy, Matplotlib, Pandas, SciPy, SciKit-Learn, SciKit-Image, OpenCV, Pillow and Biopython.

Automation with Shortcuts and x-callback URLs

1.1 Shortcuts

Pyto provides Shortcuts for running scripts and code. Shortcuts will open Pyto to execute the code and it's not possible to retrieve the result. See *x-callback URLs* for getting callbacks.

`Run Code with Arguments` will execute the given code with the given arguments. Code can be executed in app or without leaving Shortcuts. If you uncheck `Run In App`, the Shortcut will return the output of the code (`stdout + stderr`). When `Run In App` is unchecked, the code will only have access to the standard library and `stdin` cannot be sent.

`Run Script with Arguments` will execute the given script with the given arguments. Note that `Script` is a file, and Shortcuts doesn't provide a way to pick a file as a value. So the only ways to have a valid value passed to this Shortcut are by sharing a file with the Share Sheet and passing the input or by picking each time a script with the `Get file system` Shortcut.

Pyto also provides a Shortcut for each script used with the app.

1.2 x-callback URLs

Pyto supports *x-callback URLs* for running scripts from other apps and getting the result.

Use *x-callback URLs* instead of Shortcuts if you need to get the output of the executed code. With this method you can only run code and not a script but you can use the `runpy` module for running scripts in a location like `~/Documents` or `~/Documents/site-packages`.

This is the structure that should be used for running code:

```
pyto://x-callback/?code=[code]&x-success=[x-success]&x-error=[x-error]&x-cancel=[x-cancel]
```

`code` is the code to execute.

`x-success` is the URL to open when a script was executed successfully. Passes the output (`stdout + stderr`) to a `result` parameter.

`x-error` is the URL to open when a script raised an exception. Passes the exception message to a `errorMessage` parameter.

`x-cancel` is the URL to open when a script was stopped by the user or when the script raised `SystemExit` or `KeyboardInterrupt`.

The Shortcuts app supports opening x-callback URLs. [Here](#) is an example of a Shortcut that shows the current Python version.

Pyto can also open external x-callback URLs with the `xcallback` module.

Pyto includes OpenCV. However, video output does not work the exact same way that on computers. Frames are displayed on the console and no other window is presented. So `cv2.destroyAllWindows` and `cv2.waitKey` will throw errors.

2.1 Pyto specific API

`cv2.autorotate` (*frame, cam*)

Returns an auto rotated frame captured with hardware camera.

By default, frames captured by hardware camera will be rotated correctly only if the device is in portrait mode. By calling this function, you can make sure things like face detection will work on any device orientation.

Parameters

- **frame** – The frame captured by the camera.
- **cam** – The camera used to capture the frame. 0 for back and 1 for front.

2.2 Face detection example

```
"""
An example of face detection using OpenCV.
"""

import cv2
import sys

casc_path = cv2.data.harcascades+"haarcascade_frontalface_default.xml"
face_cascade = cv2.CascadeClassifier(casc_path)

device = 1 # Front camera
```

(continues on next page)

```
try:
    device = int(sys.argv[1]) # 0 for back camera
except IndexError:
    pass

cap = cv2.VideoCapture(device)

while cap.isOpened():

    # Capture frame-by-frame
    ret, frame = cap.read()

    # Check if frame is not empty
    if not ret:
        continue

    # Auto rotate camera
    frame = cv2.autorotate(frame, device)

    # Convert from BGR to RGB
    frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)

    faces = face_cascade.detectMultiScale(
        frame,
        scaleFactor=1.1,
        minNeighbors=5,
        minSize=(30, 30),
        flags=cv2.CASCADE_SCALE_IMAGE
    )

    # Draw a rectangle around the faces
    for (x, y, w, h) in faces:
        cv2.rectangle(frame, (x, y), (x+w, y+h), (0, 255, 0), 2)

    # Display the resulting frame
    cv2.imshow('frame', frame)
```

Pyto includes the following third party modules:

3.1 Modules

- asncrypto
- bcrypt
- black
- bs4
- cffi
- chardet
- cloudpickle
- colorama
- cryptography
- dask
- dateutil
- decorator
- distlib
- idna
- imageio
- joblib
- networkx
- pip - stash

- progress
- py-filelock
- python-certifi
- pytz
- python-webencodings
- requests
- urllib3
- nacl
- numpy
- pandas
- PIL
- pywt
- matplotlib
- kiwisolver
- sklearn
- skimage
- scipy
- cv2
- lxml
- Bio
- rubicon-objc
- jedi

Other pure Python modules can be installed from PyPi. (Three points icon > PyPi). **Only pure Python modules. It will not work for SciPy for example.**

3.2 Acknowledgments

Thanks to [Christian Clauss](#) and [goodclass](#) for contributions! `SciPy`, `OpenCV` and `Lxml` were taken from [goodclass](#).

Thanks to [Daniel Shane](#) for alternate icons!

This application makes use of the following third party libraries:

- appdirs
- asn1crypto
- attrs
- bcrypt
- black

- beautifulsoup4
- Biopython
- cffi
- chardet
- click
- cloudpickle
- Color-Picker-for-iOS
- colorama
- Cpython
- cycler
- dask
- dateutil
- decorator
- distlib
- Down
- FileBrowser
- filelock
- freetype
- idna
- imageio
- jedi
- joblib
- kiwisolver
- lipjpeg
- libpng
- lxml
- matplotlib
- MultiPeer
- networkx
- numpy
- openblas
- opencv
- opencv_arm64
- opencv-python
- OpenTerm
- pandas

- Pillow
- progress
- PyNaCl
- pyparsing
- Python Apple Support
- python-certifi
- Python3_iOS
- pytz
- python-webencodings
- pywt
- requests
- rubicon-objc
- scipy
- scipy_arm64
- scikit-learn
- scikit-image
- six
- soupsieve
- Source Editor
- StaSh
- stopit
- toml
- urllib3
- xlrd

appdirs

Copyright (c) 2010 ActiveState Software Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY

(continues on next page)

(continued from previous page)

CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

asn1crypto

Copyright (c) 2015-2019 Will Bond <will@wbond.net>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

attrs

The MIT License (MIT)

Copyright (c) 2015 Hynek Schlawack

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

bcrypt

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

(continues on next page)

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity

(continues on next page)

(continued from previous page)

- on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
 3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
 4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided

(continues on next page)

that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

(continues on next page)

(continued from previous page)

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

black

The MIT License (MIT)

Copyright (c) 2018 Łukasz Langa

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

beautifulsoup4

Beautiful Soup is made available under the MIT license:

Copyright (c) 2004–2012 Leonard Richardson

Permission is hereby granted, free of charge, to any person obtaining

(continues on next page)

(continued from previous page)

a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE, DAMMIT.

Beautiful Soup incorporates code from the html5lib library, which is also made available under the MIT license.

Biopython

Biopython is currently released under the "Biopython License Agreement" (given in full below). Unless stated otherwise in individual file headers, all Biopython's files are under the "Biopython License Agreement".

Some files are explicitly dual licensed under your choice of the "Biopython License Agreement" or the "BSD 3-Clause License" (both given in full below). This is with the intention of later offering all of Biopython under this dual licensing approach.

Biopython License Agreement

Permission to use, copy, modify, and distribute this software and its documentation with or without modifications and for any purpose and without fee is hereby granted, provided that any copyright notices appear in all copies and that both those copyright notices and this permission notice appear in supporting documentation, and that the names of the contributors or copyright holders not be used in advertising or publicity pertaining to distribution of the software without specific prior permission.

THE CONTRIBUTORS AND COPYRIGHT HOLDERS OF THIS SOFTWARE DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL THE CONTRIBUTORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

BSD 3-Clause License

Copyright (c) 1999-2019, The Biopython Contributors All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

(continues on next page)

(continued from previous page)

Redistributions of source code must retain the above copyright notice, this list of
 ↳ conditions and the following disclaimer.
 Redistributions in binary form must reproduce the above copyright notice, this list
 ↳ of conditions and the following disclaimer in the documentation and/or other
 ↳ materials provided with the distribution.
 Neither the name of the copyright holder nor the names of its contributors may be
 ↳ used to endorse or promote products derived from this software without specific
 ↳ prior written permission.
 THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY
 ↳ EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
 ↳ WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 ↳ IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT,
 ↳ INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 ↳ NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
 ↳ PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
 ↳ WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 ↳ ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
 ↳ POSSIBILITY OF SUCH DAMAGE.

cff

The MIT License

Permission is hereby granted, free of charge, to any person
 obtaining a copy of this software and associated documentation
 files (the "Software"), to deal in the Software without
 restriction, including without limitation the rights to use,
 copy, modify, merge, publish, distribute, sublicense, and/or
 sell copies of the Software, and to permit persons to whom the
 Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included
 in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
 OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
 FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL
 THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
 LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
 FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
 DEALINGS IN THE SOFTWARE.

charset

Python 2/3 compatible character encoding detector.

This library is free software; you can redistribute it and/or
 modify it under the terms of the GNU Lesser General Public
 License as published by the Free Software Foundation; either
 version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
 but WITHOUT ANY WARRANTY; without even the implied warranty of
 MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
 Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public

(continues on next page)

(continued from previous page)

License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

click

Copyright © 2014 by the Pallets team.

Some rights reserved.

Redistribution and use in source and binary forms of the software as well as documentation, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE AND DOCUMENTATION, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Click uses parts of optparse written by Gregory P. Ward and maintained by the Python Software Foundation. This is limited to code in parser.py.

Copyright © 2001-2006 Gregory P. Ward. All rights reserved.
Copyright © 2002-2006 Python Software Foundation. All rights reserved.

cloudpickle

This module was extracted from the `cloud` package, developed by PiCloud, Inc.

Copyright (c) 2015, Cloudpickle contributors.
Copyright (c) 2012, Regents of the University of California.
Copyright (c) 2009 PiCloud, Inc. <http://www.picloud.com>.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions

(continues on next page)

(continued from previous page)

are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of the University of California, Berkeley nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Color-Picker-for-iOS

Copyright (c) 2011 Ryota Hayashi
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR(S) ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR(S) BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

\$FreeBSD\$

colorama

Copyright (c) 2010 Jonathan Hartley
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

(continues on next page)

(continued from previous page)

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the copyright holders, nor those of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

cryptography

Copyright (c) Individual contributors.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of PyCA Cryptography nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License
Version 2.0, January 2004

(continues on next page)

(continued from previous page)

<https://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise

(continues on next page)

(continued from previous page)

designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and

(continues on next page)

(continued from previous page)

do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason

(continues on next page)

of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<https://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Cpython

A. HISTORY OF THE SOFTWARE
=====

Python was created in the early 1990s by Guido van Rossum at Stichting Mathematisch Centrum (CWI, see <http://www.cwi.nl>) in the Netherlands as a successor of a language called ABC. Guido remains Python's principal author, although it includes many contributions from others.

In 1995, Guido continued his work on Python at the Corporation for National Research Initiatives (CNRI, see <http://www.cnri.reston.va.us>) in Reston, Virginia where he released several versions of the software.

In May 2000, Guido and the Python core development team moved to BeOpen.com to form the BeOpen PythonLabs team. In October of the same year, the PythonLabs team moved to Digital Creations, which became Zope Corporation. In 2001, the Python Software Foundation (PSF, see <https://www.python.org/psf/>) was formed, a non-profit organization created specifically to own Python-related Intellectual Property. Zope Corporation was a sponsoring member of the PSF.

All Python releases are Open Source (see <http://www.opensource.org> for the Open Source Definition). Historically, most, but not all, Python releases have also been GPL-compatible; the table below summarizes the various releases.

(continued from previous page)

Release from	Derived	Year	Owner	GPL-compatible? (1)
0.9.0 thru 1.2		1991-1995	CWI	yes
1.3 thru 1.5.2	1.2	1995-1999	CNRI	yes
1.6	1.5.2	2000	CNRI	no
2.0	1.6	2000	BeOpen.com	no
1.6.1	1.6	2001	CNRI	yes (2)
2.1	2.0+1.6.1	2001	PSF	no
2.0.1	2.0+1.6.1	2001	PSF	yes
2.1.1	2.1+2.0.1	2001	PSF	yes
2.1.2	2.1.1	2002	PSF	yes
2.1.3	2.1.2	2002	PSF	yes
2.2 and above	2.1.1	2001-now	PSF	yes

Footnotes:

(1) GPL-compatible doesn't mean that we're distributing Python under the GPL. All Python licenses, unlike the GPL, let you distribute a modified version without making your changes open source. The GPL-compatible licenses make it possible to combine Python with other software that is released under the GPL; the others don't.

(2) According to Richard Stallman, 1.6.1 is not GPL-compatible, because its license has a choice of law clause. According to CNRI, however, Stallman's lawyer has told CNRI's lawyer that 1.6.1 is "not incompatible" with the GPL.

Thanks to the many outside volunteers who have worked under Guido's direction to make these releases possible.

B. TERMS AND CONDITIONS FOR ACCESSING OR OTHERWISE USING PYTHON

PYTHON SOFTWARE FOUNDATION LICENSE VERSION 2

1. This LICENSE AGREEMENT is between the Python Software Foundation ("PSF"), and the Individual or Organization ("Licensee") accessing and otherwise using this software ("Python") in source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, PSF hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Python alone or in any derivative version, provided, however, that PSF's License Agreement and PSF's notice of copyright, i.e., "Copyright (c) 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018 Python Software Foundation; All Rights Reserved" are retained in Python alone or in any derivative version prepared by Licensee.

3. In the event Licensee prepares a derivative work that is based on or incorporates Python or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of

(continues on next page)

the changes made to Python.

4. PSF is making Python available to Licensee on an "AS IS" basis. PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between PSF and Licensee. This License Agreement does not grant permission to use PSF trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By copying, installing or otherwise using Python, Licensee agrees to be bound by the terms and conditions of this License Agreement.

BEOPEN.COM LICENSE AGREEMENT FOR PYTHON 2.0

BEOPEN PYTHON OPEN SOURCE LICENSE AGREEMENT VERSION 1

1. This LICENSE AGREEMENT is between BeOpen.com ("BeOpen"), having an office at 160 Saratoga Avenue, Santa Clara, CA 95051, and the Individual or Organization ("Licensee") accessing and otherwise using this software in source or binary form and its associated documentation ("the Software").

2. Subject to the terms and conditions of this BeOpen Python License Agreement, BeOpen hereby grants Licensee a non-exclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use the Software alone or in any derivative version, provided, however, that the BeOpen Python License is retained in the Software, alone or in any derivative version prepared by Licensee.

3. BeOpen is making the Software available to Licensee on an "AS IS" basis. BEOPEN MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, BEOPEN MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

4. BEOPEN SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF THE SOFTWARE FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THE SOFTWARE, OR ANY

(continued from previous page)

DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

5. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

6. This License Agreement shall be governed by and interpreted in all respects by the law of the State of California, excluding conflict of law provisions. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between BeOpen and Licensee. This License Agreement does not grant permission to use BeOpen trademarks or trade names in a trademark sense to endorse or promote products or services of Licensee, or any third party. As an exception, the "BeOpen Python" logos available at <http://www.pythonlabs.com/logos.html> may be used according to the permissions granted on that web page.

7. By copying, installing or otherwise using the software, Licensee agrees to be bound by the terms and conditions of this License Agreement.

CNRI LICENSE AGREEMENT FOR PYTHON 1.6.1

1. This LICENSE AGREEMENT is between the Corporation for National Research Initiatives, having an office at 1895 Preston White Drive, Reston, VA 20191 ("CNRI"), and the Individual or Organization ("Licensee") accessing and otherwise using Python 1.6.1 software in source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, CNRI hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Python 1.6.1 alone or in any derivative version, provided, however, that CNRI's License Agreement and CNRI's notice of copyright, i.e., "Copyright (c) 1995-2001 Corporation for National Research Initiatives; All Rights Reserved" are retained in Python 1.6.1 alone or in any derivative version prepared by Licensee. Alternately, in lieu of CNRI's License Agreement, Licensee may substitute the following text (omitting the quotes): "Python 1.6.1 is made available subject to the terms and conditions in CNRI's License Agreement. This Agreement together with Python 1.6.1 may be located on the Internet using the following unique, persistent identifier (known as a handle): 1895.22/1013. This Agreement may also be obtained from a proxy server on the Internet using the following URL: <http://hdl.handle.net/1895.22/1013>".

3. In the event Licensee prepares a derivative work that is based on or incorporates Python 1.6.1 or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to Python 1.6.1.

4. CNRI is making Python 1.6.1 available to Licensee on an "AS IS" basis. CNRI MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, CNRI MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS

(continues on next page)

FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON 1.6.1 WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. CNRI SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON 1.6.1 FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON 1.6.1, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

7. This License Agreement shall be governed by the federal intellectual property law of the United States, including without limitation the federal copyright law, and, to the extent such U.S. federal law does not apply, by the law of the Commonwealth of Virginia, excluding Virginia's conflict of law provisions. Notwithstanding the foregoing, with regard to derivative works based on Python 1.6.1 that incorporate non-separable material that was previously distributed under the GNU General Public License (GPL), the law of the Commonwealth of Virginia shall govern this License Agreement only as to issues arising under or with respect to Paragraphs 4, 5, and 7 of this License Agreement. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between CNRI and Licensee. This License Agreement does not grant permission to use CNRI trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By clicking on the "ACCEPT" button where indicated, or by copying, installing or otherwise using Python 1.6.1, Licensee agrees to be bound by the terms and conditions of this License Agreement.

ACCEPT

CWI LICENSE AGREEMENT FOR PYTHON 0.9.0 THROUGH 1.2

Copyright (c) 1991 - 1995, Stichting Mathematisch Centrum Amsterdam, The Netherlands. All rights reserved.

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Stichting Mathematisch Centrum or CWI not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

STICHTING MATHEMATISCH CENTRUM DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL STICHTING MATHEMATISCH CENTRUM BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

cycler

Copyright (c) 2015, matplotlib project
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the matplotlib project nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

dask

Copyright (c) 2014–2018, Anaconda, Inc. and contributors
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Anaconda nor the names of any contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)

(continues on next page)

(continued from previous page)

ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

dateutil

```
Copyright 2017- Paul Ganssle
Copyright 2017- dateutil contributors (see AUTHORS file)

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

The above license applies to all contributions after 2017-12-01, as well as
all contributions that have been re-licensed (see AUTHORS file for the list of
contributors who have re-licensed their code).
-----
dateutil - Extensions to the standard Python datetime module.

Copyright (c) 2003-2011 - Gustavo Niemeyer
Copyright (c) 2012-2014 - Tomi Pieviläinen
Copyright (c) 2014-2016 - Yaron de Leeuw
Copyright (c) 2015-      - Paul Ganssle
Copyright (c) 2015-      - dateutil contributors (see AUTHORS file)

All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice,
this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.
* Neither the name of the copyright holder nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

(continues on next page)

(continued from previous page)

The above BSD License Applies to all code, even that also covered by Apache 2.0.

decorator

Copyright (c) 2005-2018, Michele Simionato
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
Redistributions in bytecode form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

distlib

A. HISTORY OF THE SOFTWARE

=====

Python was created in the early 1990s by Guido van Rossum at Stichting Mathematisch Centrum (CWI, see <http://www.cwi.nl>) in the Netherlands as a successor of a language called ABC. Guido remains Python's principal author, although it includes many contributions from others.

In 1995, Guido continued his work on Python at the Corporation for National Research Initiatives (CNRI, see <http://www.cnri.reston.va.us>) in Reston, Virginia where he released several versions of the software.

In May 2000, Guido and the Python core development team moved to BeOpen.com to form the BeOpen PythonLabs team. In October of the same year, the PythonLabs team moved to Digital Creations (now Zope Corporation, see <http://www.zope.com>). In 2001, the Python Software Foundation (PSF, see <http://www.python.org/psf/>) was formed, a non-profit organization created specifically to own Python-related Intellectual Property. Zope Corporation is a sponsoring member of the PSF.

All Python releases are Open Source (see <http://www.opensource.org> for

(continues on next page)

(continued from previous page)

the Open Source Definition). Historically, most, but not all, Python releases have also been GPL-compatible; the table below summarizes the various releases.

Release from	Derived	Year	Owner	GPL-compatible? (1)
0.9.0 thru 1.2		1991-1995	CWI	yes
1.3 thru 1.5.2	1.2	1995-1999	CNRI	yes
1.6	1.5.2	2000	CNRI	no
2.0	1.6	2000	BeOpen.com	no
1.6.1	1.6	2001	CNRI	yes (2)
2.1	2.0+1.6.1	2001	PSF	no
2.0.1	2.0+1.6.1	2001	PSF	yes
2.1.1	2.1+2.0.1	2001	PSF	yes
2.2	2.1.1	2001	PSF	yes
2.1.2	2.1.1	2002	PSF	yes
2.1.3	2.1.2	2002	PSF	yes
2.2.1	2.2	2002	PSF	yes
2.2.2	2.2.1	2002	PSF	yes
2.2.3	2.2.2	2003	PSF	yes
2.3	2.2.2	2002-2003	PSF	yes
2.3.1	2.3	2002-2003	PSF	yes
2.3.2	2.3.1	2002-2003	PSF	yes
2.3.3	2.3.2	2002-2003	PSF	yes
2.3.4	2.3.3	2004	PSF	yes
2.3.5	2.3.4	2005	PSF	yes
2.4	2.3	2004	PSF	yes
2.4.1	2.4	2005	PSF	yes
2.4.2	2.4.1	2005	PSF	yes
2.4.3	2.4.2	2006	PSF	yes
2.4.4	2.4.3	2006	PSF	yes
2.5	2.4	2006	PSF	yes
2.5.1	2.5	2007	PSF	yes
2.5.2	2.5.1	2008	PSF	yes
2.5.3	2.5.2	2008	PSF	yes
2.6	2.5	2008	PSF	yes
2.6.1	2.6	2008	PSF	yes
2.6.2	2.6.1	2009	PSF	yes
2.6.3	2.6.2	2009	PSF	yes
2.6.4	2.6.3	2009	PSF	yes
2.6.5	2.6.4	2010	PSF	yes
3.0	2.6	2008	PSF	yes
3.0.1	3.0	2009	PSF	yes
3.1	3.0.1	2009	PSF	yes
3.1.1	3.1	2009	PSF	yes
3.1.2	3.1	2010	PSF	yes
3.2	3.1	2010	PSF	yes

Footnotes:

(1) GPL-compatible doesn't mean that we're distributing Python under the GPL. All Python licenses, unlike the GPL, let you distribute a modified version without making your changes open source. The GPL-compatible licenses make it possible to combine Python with other software that is released under the GPL; the others don't.

(continues on next page)

(continued from previous page)

(2) According to Richard Stallman, 1.6.1 is not GPL-compatible, because its license has a choice of law clause. According to CNRI, however, Stallman's lawyer has told CNRI's lawyer that 1.6.1 is "not incompatible" with the GPL.

Thanks to the many outside volunteers who have worked under Guido's direction to make these releases possible.

B. TERMS AND CONDITIONS FOR ACCESSING OR OTHERWISE USING PYTHON

PYTHON SOFTWARE FOUNDATION LICENSE VERSION 2

1. This LICENSE AGREEMENT is between the Python Software Foundation ("PSF"), and the Individual or Organization ("Licensee") accessing and otherwise using this software ("Python") in source or binary form and its associated documentation.
2. Subject to the terms and conditions of this License Agreement, PSF hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Python alone or in any derivative version, provided, however, that PSF's License Agreement and PSF's notice of copyright, i.e., "Copyright (c) 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010 Python Software Foundation; All Rights Reserved" are retained in Python alone or in any derivative version prepared by Licensee.
3. In the event Licensee prepares a derivative work that is based on or incorporates Python or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to Python.
4. PSF is making Python available to Licensee on an "AS IS" basis. PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.
5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.
6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.
7. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between PSF and Licensee. This License Agreement does not grant permission to use PSF trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.
8. By copying, installing or otherwise using Python, Licensee

(continues on next page)

agrees to be bound by the terms and conditions of this License Agreement.

BEOPEN.COM LICENSE AGREEMENT FOR PYTHON 2.0

BEOPEN PYTHON OPEN SOURCE LICENSE AGREEMENT VERSION 1

1. This LICENSE AGREEMENT is between BeOpen.com ("BeOpen"), having an office at 160 Saratoga Avenue, Santa Clara, CA 95051, and the Individual or Organization ("Licensee") accessing and otherwise using this software in source or binary form and its associated documentation ("the Software").

2. Subject to the terms and conditions of this BeOpen Python License Agreement, BeOpen hereby grants Licensee a non-exclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use the Software alone or in any derivative version, provided, however, that the BeOpen Python License is retained in the Software, alone or in any derivative version prepared by Licensee.

3. BeOpen is making the Software available to Licensee on an "AS IS" basis. BEOPEN MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, BEOPEN MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

4. BEOPEN SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF THE SOFTWARE FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THE SOFTWARE, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

5. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

6. This License Agreement shall be governed by and interpreted in all respects by the law of the State of California, excluding conflict of law provisions. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between BeOpen and Licensee. This License Agreement does not grant permission to use BeOpen trademarks or trade names in a trademark sense to endorse or promote products or services of Licensee, or any third party. As an exception, the "BeOpen Python" logos available at <http://www.pythonlabs.com/logos.html> may be used according to the permissions granted on that web page.

7. By copying, installing or otherwise using the software, Licensee agrees to be bound by the terms and conditions of this License Agreement.

CNRI LICENSE AGREEMENT FOR PYTHON 1.6.1

(continued from previous page)

1. This LICENSE AGREEMENT is between the Corporation for National Research Initiatives, having an office at 1895 Preston White Drive, Reston, VA 20191 ("CNRI"), and the Individual or Organization ("Licensee") accessing and otherwise using Python 1.6.1 software in source or binary form and its associated documentation.
2. Subject to the terms and conditions of this License Agreement, CNRI hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Python 1.6.1 alone or in any derivative version, provided, however, that CNRI's License Agreement and CNRI's notice of copyright, i.e., "Copyright (c) 1995-2001 Corporation for National Research Initiatives; All Rights Reserved" are retained in Python 1.6.1 alone or in any derivative version prepared by Licensee. Alternately, in lieu of CNRI's License Agreement, Licensee may substitute the following text (omitting the quotes): "Python 1.6.1 is made available subject to the terms and conditions in CNRI's License Agreement. This Agreement together with Python 1.6.1 may be located on the Internet using the following unique, persistent identifier (known as a handle): 1895.22/1013. This Agreement may also be obtained from a proxy server on the Internet using the following URL: <http://hdl.handle.net/1895.22/1013>".
3. In the event Licensee prepares a derivative work that is based on or incorporates Python 1.6.1 or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to Python 1.6.1.
4. CNRI is making Python 1.6.1 available to Licensee on an "AS IS" basis. CNRI MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, CNRI MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON 1.6.1 WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.
5. CNRI SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON 1.6.1 FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON 1.6.1, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.
6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.
7. This License Agreement shall be governed by the federal intellectual property law of the United States, including without limitation the federal copyright law, and, to the extent such U.S. federal law does not apply, by the law of the Commonwealth of Virginia, excluding Virginia's conflict of law provisions. Notwithstanding the foregoing, with regard to derivative works based on Python 1.6.1 that incorporate non-separable material that was previously distributed under the GNU General Public License (GPL), the law of the Commonwealth of Virginia shall govern this License Agreement only as to issues arising under or with respect to Paragraphs 4, 5, and 7 of this License Agreement. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between CNRI and Licensee. This

(continues on next page)

(continued from previous page)

License Agreement does not grant permission to use CNRI trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By clicking on the "ACCEPT" button where indicated, or by copying, installing or otherwise using Python 1.6.1, Licensee agrees to be bound by the terms and conditions of this License Agreement.

ACCEPT

CWI LICENSE AGREEMENT FOR PYTHON 0.9.0 THROUGH 1.2

Copyright (c) 1991 - 1995, Stichting Mathematisch Centrum Amsterdam, The Netherlands. All rights reserved.

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Stichting Mathematisch Centrum or CWI not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

STICHTING MATHEMATISCH CENTRUM DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL STICHTING MATHEMATISCH CENTRUM BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Down

The MIT License (MIT)

Copyright (c) 2016 Rob Phillips.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(continues on next page)

(continued from previous page)

cmark

Copyright (c) 2014, John MacFarlane

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

houdini.h, houdini_href_e.c, houdini_html_e.c, houdini_html_u.c,
html_unescape.gperf, html_unescape.h

derive from <https://github.com/vmg/houdini> (with some modifications)

Copyright (C) 2012 Vicent Martí

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE

(continues on next page)

(continued from previous page)

```
SOFTWARE.
-----

buffer.h, buffer.c, chunk.h

are derived from code (C) 2012 Github, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is furnished to do
so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

-----

utf8.c and utf8.c

are derived from utf8proc
(),
    (C) 2009 Public Software Group e. V., Berlin, Germany.

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the "Software"),
to deal in the Software without restriction, including without limitation
the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN THE SOFTWARE.

-----

The normalization code in runtests.py was derived from the
markdowntest project, Copyright 2013 Karl Dubost:
```

(continues on next page)

(continued from previous page)

The MIT License (MIT)

Copyright (c) 2013 Karl Dubost

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The CommonMark spec (test/spec.txt) is

Copyright (C) 2014-15 John MacFarlane

Released under the Creative Commons CC-BY-SA 4.0 license:

.

The test software in test/ is

Copyright (c) 2014, John MacFarlane

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT

(continues on next page)

(continued from previous page)

LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The normalization code in `runtests.py` was derived from the `markdowntest` project, Copyright 2013 Karl Dubost:

The MIT License (MIT)

Copyright (c) 2013 Karl Dubost

Permission is hereby granted, free of charge, to any person obtaining a copy of ↵
↵this software and associated documentation files (the "Software"), to deal in the ↵
↵Software without restriction, including without limitation the rights to use, copy, ↵
↵modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, ↵
↵and to permit persons to whom the Software is furnished to do so, subject to the ↵
↵following conditions:

The above copyright notice and this permission notice shall be included in all ↵
↵copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

FileBrowser

The MIT License (MIT)

Copyright (c) 2016 Roy Marmelstein

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

freetype

```
Portions of this software are copyright © 2018 The FreeType
Project (www.freetype.org). All rights reserved.
```

idna

```
Copyright (c) 2013-2018, Kim Davies. All rights reserved.
```

```
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
```

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holder nor the names of the contributors may be used to endorse or promote products derived from this software without specific prior written permission.
4. THIS SOFTWARE IS PROVIDED BY THE CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
Portions of the codec implementation and unit tests are derived from the Python_
↳standard library, which carries the Python Software Foundation License:
```

```
Copyright (c) 2001-2014 Python Software Foundation; All Rights Reserved
```

```
Portions of the unit tests are derived from the Unicode standard, which is subject to_
↳the Unicode, Inc. License Agreement:
```

```
Copyright (c) 1991-2014 Unicode, Inc. All rights reserved. Distributed under the_
↳Terms of Use in .
```

```
Permission is hereby granted, free of charge, to any person obtaining a copy of_
↳the Unicode data files and any associated documentation (the "Data Files") or_
↳Unicode software and any associated documentation (the "Software") to deal in the_
↳Data Files or Software without restriction, including without limitation the rights_
↳to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data_
↳Files or Software, and to permit persons to whom the Data Files or Software are_
↳furnished to do so, provided that
```

```
(a) this copyright and permission notice appear with all copies of the Data Files_
↳or Software,
```

(continues on next page)

(continued from previous page)

(b) this copyright and permission notice appear in associated documentation, and

(c) there is clear notice in each modified Data File or in the Software as well,
 ↳as in the documentation associated with the Data File(s) or Software that the data,
 ↳or software has been modified.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
 ↳EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
 ↳FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
 ↳EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR
 ↳ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES
 ↳WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF
 ↳CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH
 ↳THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be
 ↳used in advertising or otherwise to promote the sale, use or other dealings in
 ↳these Data Files or Software without prior written authorization of the copyright
 ↳holder.

imageio

Copyright (c) 2014-2018, imageio developers
 All rights reserved.

Redistribution and use in source and binary forms, with or without
 modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this
 list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice,
 this list of conditions and the following disclaimer in the documentation
 and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
 AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
 FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
 DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
 SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
 CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
 OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
 OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

jedi

All contributions towards Jedi are MIT licensed.

 The MIT License (MIT)

Copyright (c) <2013>

Permission is hereby granted, free of charge, to any person obtaining a copy

(continues on next page)

(continued from previous page)

of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

joblib

BSD 3-Clause License

Copyright (c) 2008-2016, The joblib developers.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

networkx

Copyright (C) 2004-2019, NetworkX Developers
Aric Hagberg <hagberg@lanl.gov>
Dan Schult <dschult@colgate.edu>
Pieter Swart <swart@lanl.gov>
All rights reserved.

(continues on next page)

(continued from previous page)

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of the NetworkX Developers nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

kiwisolver

=====
The Kiwi licensing terms
=====

Kiwi is licensed under the terms of the Modified BSD License (also known as New or Revised BSD), as follows:

Copyright (c) 2013, Nucleic Development Team

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the Nucleic Development Team nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE

(continues on next page)

(continued from previous page)

DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

About Kiwi

Chris Colbert began the Kiwi project in December 2013 in an effort to create a blisteringly fast UI constraint solver. Chris is still the project lead.

The Nucleic Development Team is the set of all contributors to the Nucleic project and its subprojects.

The core team that coordinates development on GitHub can be found here: <http://github.com/nucleic>. The current team consists of:

- * Chris Colbert

Our Copyright Policy

Nucleic uses a shared copyright model. Each contributor maintains copyright over their contributions to Nucleic. But, it is important to note that these contributions are typically only changes to the repositories. Thus, the Nucleic source code, in its entirety is not the copyright of any single person or institution. Instead, it is the collective copyright of the entire Nucleic Development Team. If individual contributors want to maintain a record of what changes/contributions they have specific copyright on, they should indicate their copyright in the commit message of the change, when they commit the change to one of the Nucleic repositories.

With this in mind, the following banner should be used in any source code file to indicate the copyright and license terms:

```
#-----
# Copyright (c) 2013, Nucleic Development Team.
#
# Distributed under the terms of the Modified BSD License.
#
# The full license is in the file COPYING.txt, distributed with this software.
#-----
```

libjpeg

LICENSE TERMS (ships as a part of the libjpeg package in the README file)
=====

1. We don't promise that this software works. (But if you find any bugs, please let us know!)
2. You can use this software for whatever you want. You don't have to pay us.
3. You may not pretend that you wrote this software. If you use it in a program, you must acknowledge somewhere in your documentation that you've used the IJG code.

(continues on next page)

(continued from previous page)

In legalese:

The authors make NO WARRANTY or representation, either express or implied, with respect to this software, its quality, accuracy, merchantability, or fitness for a particular purpose. This software is provided "AS IS", and you, its user, assume the entire risk as to its quality and accuracy.

This software is copyright (C) 1991-2016, Thomas G. Lane, Guido Vollbeding. All Rights Reserved except as specified below.

Permission is hereby granted to use, copy, modify, and distribute this software (or portions thereof) for any purpose, without fee, subject to these conditions:

- (1) If any part of the source code for this software is distributed, then this README file must be included, with this copyright and no-warranty notice unaltered; and any additions, deletions, or changes to the original files must be clearly indicated in accompanying documentation.
- (2) If only executable code is distributed, then the accompanying documentation must state that "this software is based in part on the work of the Independent JPEG Group".
- (3) Permission for use of this software is granted only if the user accepts full responsibility for any undesirable consequences; the authors accept NO LIABILITY for damages of any kind.

These conditions apply to any software derived from or based on the IJG code, not just to the unmodified library. If you use our work, you ought to acknowledge us.

Permission is NOT granted for the use of any IJG author's name or company name in advertising or publicity relating to this software or products derived from it. This software may be referred to only as "the Independent JPEG Group's software".

We specifically permit and encourage the use of this software as the basis of commercial products, provided that all warranty or liability claims are assumed by the product vendor.

libpng

COPYRIGHT NOTICE, DISCLAIMER, and LICENSE
=====

PNG Reference Library License version 2

- * Copyright (c) 1995-2018 The PNG Reference Library Authors.
- * Copyright (c) 2018 Cosmin Truta.
- * Copyright (c) 2000-2002, 2004, 2006-2018 Glenn Randers-Pehrson.
- * Copyright (c) 1996-1997 Andreas Dilger.
- * Copyright (c) 1995-1996 Guy Eric Schalnat, Group 42, Inc.

The software is supplied "as is", without warranty of any kind, express or implied, including, without limitation, the warranties of merchantability, fitness for a particular purpose, title, and non-infringement. In no even shall the Copyright owners, or anyone distributing the software, be liable for any damages or other liability, whether in contract, tort or otherwise, arising

(continues on next page)

(continued from previous page)

from, out of, or in connection with the software, or the use or other dealings in the software, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this software, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated, but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This Copyright notice may not be removed or altered from any source or altered source distribution.

PNG Reference Library License version 1 (for libpng 0.5 through 1.6.35)

libpng versions 1.0.7, July 1, 2000 through 1.6.35, July 15, 2018 are Copyright (c) 2000-2002, 2004, 2006-2018 Glenn Randers-Pehrson, are derived from libpng-1.0.6, and are distributed according to the same disclaimer and license as libpng-1.0.6 with the following individuals added to the list of Contributing Authors:

Simon-Pierre Cadieux
Eric S. Raymond
Mans Rullgard
Cosmin Truta
Gilles Vollant
James Yu
Mandar Sahastrabudde
Google Inc.
Vadim Barkov

and with the following additions to the disclaimer:

There is no warranty against interference with your enjoyment of the library or against infringement. There is no warranty that our efforts or the library will fulfill any of your particular purposes or needs. This library is provided with all faults, and the entire risk of satisfactory quality, performance, accuracy, and effort is with the user.

Some files in the "contrib" directory and some configure-generated files that are distributed with libpng have other copyright owners, and are released under other open source licenses.

libpng versions 0.97, January 1998, through 1.0.6, March 20, 2000, are Copyright (c) 1998-2000 Glenn Randers-Pehrson, are derived from libpng-0.96, and are distributed according to the same disclaimer and license as libpng-0.96, with the following individuals added to the list of Contributing Authors:

(continues on next page)

Tom Lane
Glenn Randers-Pehrson
Willem van Schaik

libpng versions 0.89, June 1996, through 0.96, May 1997, are Copyright (c) 1996-1997 Andreas Dilger, are derived from libpng-0.88, and are distributed according to the same disclaimer and license as libpng-0.88, with the following individuals added to the list of Contributing Authors:

John Bowler
Kevin Bracey
Sam Bushell
Magnus Holmgren
Greg Roelofs
Tom Tanner

Some files in the "scripts" directory have other copyright owners, but are released under this license.

libpng versions 0.5, May 1995, through 0.88, January 1996, are Copyright (c) 1995-1996 Guy Eric Schalnat, Group 42, Inc.

For the purposes of this copyright and license, "Contributing Authors" is defined as the following set of individuals:

Andreas Dilger
Dave Martindale
Guy Eric Schalnat
Paul Schmidt
Tim Wegner

The PNG Reference Library is supplied "AS IS". The Contributing Authors and Group 42, Inc. disclaim all warranties, expressed or implied, including, without limitation, the warranties of merchantability and of fitness for any purpose. The Contributing Authors and Group 42, Inc. assume no liability for direct, indirect, incidental, special, exemplary, or consequential damages, which may result from the use of the PNG Reference Library, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this source code, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this source code must not be misrepresented.
2. Altered versions must be plainly marked as such and must not be misrepresented as being the original source.
3. This Copyright notice may not be removed or altered from any source or altered source distribution.

The Contributing Authors and Group 42, Inc. specifically permit, without fee, and encourage the use of this source code as a component to supporting the PNG file format in commercial products. If you use

(continued from previous page)

this source code in a product, acknowledgment is not required but would be appreciated.

lxml

`lxml` is copyright Infrae and distributed under the BSD license (see `doc/licenses/BSD.txt`), with the following exceptions:

Some code, such as `selftest.py`, `selftest2.py` and `src/lxml/_elementpath.py` are derived from `ElementTree` and `cElementTree`. See `doc/licenses/elementtree.txt` for the license text.

`lxml.cssselect` and `lxml.html` are copyright Ian Bicking and distributed under the BSD license (see `doc/licenses/BSD.txt`).

`test.py`, the test-runner script, is GPL and copyright Shuttleworth Foundation. See `doc/licenses/GPL.txt`. It is believed the unchanged inclusion of `test.py` to run the unit test suite falls under the "aggregation" clause of the GPL and thus does not affect the license of the rest of the package.

The `isoschematron` implementation uses several XSL and RelaxNG resources:

- * The (XML syntax) RelaxNG schema for `schematron`, copyright International Organization for Standardization (see `src/lxml/isoschematron/resources/rng/iso-schematron.rng` for the license text)
- * The skeleton `iso-schematron-xml pure-xslt schematron` implementation xsl stylesheets, copyright Rick Jelliffe and Academia Sinica Computing Center, Taiwan (see the xsl files here for the license text: `src/lxml/isoschematron/resources/xsl/iso-schematron-xslt1/`)
- * The `xsd/rng` schema `schematron` extraction xsl transformations are unlicensed and copyright the respective authors as noted (see `src/lxml/isoschematron/resources/xsl/RNG2Schtrn.xsl` and `src/lxml/isoschematron/resources/xsl/XSD2Schtrn.xsl`)

matplotlib

License agreement for `matplotlib` versions 1.3.0 and later

- =====
1. This LICENSE AGREEMENT is between the Matplotlib Development Team ("MDT"), and the Individual or Organization ("Licensee") accessing and otherwise using `matplotlib` software in source or binary form and its associated documentation.
 2. Subject to the terms and conditions of this License Agreement, MDT hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use `matplotlib` alone or in any derivative version, provided, however, that MDT's License Agreement and MDT's notice of copyright, i.e., "Copyright (c) 2012- Matplotlib Development Team; All Rights Reserved" are retained in `matplotlib` alone or in any derivative version prepared by Licensee.
 3. In the event Licensee prepares a derivative work that is based on or incorporates `matplotlib` or any part thereof, and wants to

(continues on next page)

(continued from previous page)

make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to matplotlib .

4. MDT is making matplotlib available to Licensee on an "AS IS" basis. MDT MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, MDT MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF MATPLOTLIB WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. MDT SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF MATPLOTLIB FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING MATPLOTLIB , OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between MDT and Licensee. This License Agreement does not grant permission to use MDT trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By copying, installing or otherwise using matplotlib , Licensee agrees to be bound by the terms and conditions of this License Agreement.

License agreement for matplotlib versions prior to 1.3.0

=====

1. This LICENSE AGREEMENT is between John D. Hunter ("JDH"), and the Individual or Organization ("Licensee") accessing and otherwise using matplotlib software in source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, JDH hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use matplotlib alone or in any derivative version, provided, however, that JDH's License Agreement and JDH's notice of copyright, i.e., "Copyright (c) 2002-2011 John D. Hunter; All Rights Reserved" are retained in matplotlib alone or in any derivative version prepared by Licensee.

3. In the event Licensee prepares a derivative work that is based on or incorporates matplotlib or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to matplotlib.

4. JDH is making matplotlib available to Licensee on an "AS IS" basis. JDH MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, JDH MAKES NO AND

(continues on next page)

(continued from previous page)

DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF MATPLOTLIB WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. JDH SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF MATPLOTLIB FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING MATPLOTLIB , OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between JDH and Licensee. This License Agreement does not grant permission to use JDH trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By copying, installing or otherwise using matplotlib, Licensee agrees to be bound by the terms and conditions of this License Agreement.

MultiPeer

MIT License

Copyright (c) 2018 Wilson Ding

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

numpy

Copyright (c) 2005–2019, NumPy Developers.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright

(continues on next page)

(continued from previous page)

notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the NumPy Developers nor the names of any contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The NumPy repository and source distributions bundle several libraries that are compatibly licensed. We list these here.

Name: Numpydoc
Files: doc/sphinxext/numpydoc/*
License: 2-clause BSD
For details, see doc/sphinxext/LICENSE.txt

Name: scipy-sphinx-theme
Files: doc/scipy-sphinx-theme/*
License: 3-clause BSD, PSF and Apache 2.0
For details, see doc/scipy-sphinx-theme/LICENSE.txt

Name: lapack-lite
Files: numpy/linalg/lapack_lite/*
License: 3-clause BSD
For details, see numpy/linalg/lapack_lite/LICENSE.txt

Name: tempita
Files: tools/npv_tempita/*
License: BSD derived
For details, see tools/npv_tempita/license.txt

Name: dragon4
Files: numpy/core/src/multiarray/dragon4.c
License: One of a kind
For license text, see numpy/core/src/multiarray/dragon4.c

openblas

Copyright (c) 2011-2014, The OpenBLAS Project
All rights reserved.

(continues on next page)

(continued from previous page)

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the OpenBLAS project nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

opencv

By downloading, copying, installing or using the software you agree to this license. If you do not agree to this license, do not download, install, copy or use the software.

License Agreement
For Open Source Computer Vision Library
(3-clause BSD License)

Copyright (C) 2000-2019, Intel Corporation, all rights reserved.
 Copyright (C) 2009-2011, Willow Garage Inc., all rights reserved.
 Copyright (C) 2009-2016, NVIDIA Corporation, all rights reserved.
 Copyright (C) 2010-2013, Advanced Micro Devices, Inc., all rights reserved.
 Copyright (C) 2015-2016, OpenCV Foundation, all rights reserved.
 Copyright (C) 2015-2016, Itseez Inc., all rights reserved.
 Third party copyrights are property of their respective owners.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

(continues on next page)

(continued from previous page)

* Neither the names of the copyright holders nor the names of the contributors may be used to endorse or promote products derived from this software without specific prior written permission.

This software is provided by the copyright holders and contributors "as is" and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall copyright holders or contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

opencv_arm64

Copyright [2019] [goodclass]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

opencv-python

Copyright (c) 2016-2018 Olli-Pekka Heinisuo and contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

OpenTerm

This software is bi-licensed under the GPLv2 (or later) and the MPLv2

(continues on next page)

(continued from previous page)

Any commit to this repository implicitly allows Louis D'hauwe to relicense this software to any OSI-approved license, without prior consent.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free

(continues on next page)

(continued from previous page)

program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under

(continues on next page)

(continued from previous page)

these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

(continues on next page)

(continued from previous page)

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing

(continues on next page)

(continued from previous page)

to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

(continues on next page)

END OF TERMS AND CONDITIONS

Mozilla Public License Version 2.0
=====

1. Definitions

1.1. "Contributor"

means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. "Contributor Version"

means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"

means Covered Software of a particular Contributor.

1.4. "Covered Software"

means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. "Incompatible With Secondary Licenses"

means

(a) that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

(b) that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. "Executable Form"

means any form of the work other than Source Code Form.

1.7. "Larger Work"

means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. "License"

means this document.

1.9. "Licensable"

means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. "Modifications"

means any of the following:

(continues on next page)

(continued from previous page)

(a) any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

(b) any new file in Source Code Form that contains any Covered Software.

1.11. "Patent Claims" of a Contributor

means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. "Secondary License"

means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. "Source Code Form"

means the form of the work preferred for making modifications.

1.14. "You" (or "Your")

means an individual or a legal entity exercising rights under this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

- (a) under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and
- (b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first

(continues on next page)

distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License.

Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

- (a) for any code that a Contributor has removed from Covered Software;
or
- (b) for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or
- (c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source

(continues on next page)

(continued from previous page)

Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

- (a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and
- (b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

(continues on next page)

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

* 6. Disclaimer of Warranty *
* ----- *
* Covered Software is provided under this License on an "as is" *
* basis, without warranty of any kind, either expressed, implied, or *
* statutory, including, without limitation, warranties that the *
* Covered Software is free of defects, merchantable, fit for a *
* particular purpose or non-infringing. The entire risk as to the *
* quality and performance of the Covered Software is with You. *
* Should any Covered Software prove defective in any respect, You *
* (not any Contributor) assume the cost of any necessary servicing, *
* repair, or correction. This disclaimer of warranty constitutes an *
* essential part of this License. No use of any Covered Software is *
* authorized under this License except under this disclaimer. *
* *****

(continued from previous page)

```

*****
*
* 7. Limitation of Liability
* -----
*
* Under no circumstances and under no legal theory, whether tort
* (including negligence), contract, or otherwise, shall any
* Contributor, or anyone who distributes Covered Software as
* permitted above, be liable to You for any direct, indirect,
* special, incidental, or consequential damages of any character
* including, without limitation, damages for lost profits, loss of
* goodwill, work stoppage, computer failure or malfunction, or any
* and all other commercial damages or losses, even if such party
* shall have been informed of the possibility of such damages. This
* limitation of liability shall not apply to liability for death or
* personal injury resulting from such party's negligence to the
* extent applicable law prohibits such limitation. Some
* jurisdictions do not allow the exclusion or limitation of
* incidental or consequential damages, so this exclusion and
* limitation may not apply to You.
*
*****

```

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version

(continues on next page)

(continued from previous page)

of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at <http://mozilla.org/MPL/2.0/>.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.

pandas

BSD 3-Clause License

Copyright (c) 2008-2012, AQR Capital Management, LLC, Lambda Foundry, Inc. and PyData Development Team
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

(continues on next page)

(continued from previous page)

* Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Pillow

The Python Imaging Library (PIL) is

Copyright © 1997–2011 by Secret Labs AB

Copyright © 1995–2011 by Fredrik Lundh

Pillow is the friendly PIL fork. It is

Copyright © 2010–2019 by Alex Clark and contributors

Like PIL, Pillow is licensed under the open source PIL Software License:

By obtaining, using, and/or copying this software and/or its associated documentation,
 ↳ you agree that you have read, understood, and will comply with the following terms,
 ↳ and conditions:

Permission to use, copy, modify, and distribute this software and its associated
 ↳ documentation for any purpose and without fee is hereby granted, provided that the
 ↳ above copyright notice appears in all copies, and that both that copyright notice
 ↳ and this permission notice appear in supporting documentation, and that the name of
 ↳ Secret Labs AB or the author not be used in advertising or publicity pertaining to
 ↳ distribution of the software without specific, written prior permission.

SECRET LABS AB AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE,
 ↳ INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL
 ↳ SECRET LABS AB OR THE AUTHOR BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL
 ↳ DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS,
 ↳ WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT
 ↳ OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

progress

Copyright (c) 2012 Giorgos Verigakis

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR

(continues on next page)

(continued from previous page)

ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

PyNaCl

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally

(continues on next page)

(continued from previous page)

submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must

(continues on next page)

(continued from previous page)

include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

(continues on next page)

(continued from previous page)

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

pyparsing

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Python Apple support

Copyright (c) 2014-2017 Russell Keith-Magee.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

python-certifi

```
This packge contains a modified version of ca-bundle.crt:

ca-bundle.crt -- Bundle of CA Root Certificates

Certificate data from Mozilla as of: Thu Nov  3 19:04:19 2011#
This is a bundle of X.509 certificates of public Certificate Authorities
(CA). These were automatically extracted from Mozilla's root certificates
file (certdata.txt). This file can be found in the mozilla source tree:
http://mxr.mozilla.org/mozilla/source/security/nss/lib/ckfw/builtins/certdata.txt?
↳raw=1#
It contains the certificates in PEM format and therefore
can be directly used with curl / libcurl / php_curl, or with
an Apache+mod_ssl webserver for SSL client authentication.
Just configure this file as the SSLCACertificateFile.#

***** BEGIN LICENSE BLOCK *****
This Source Code Form is subject to the terms of the Mozilla Public License,
v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain
one at http://mozilla.org/MPL/2.0/.

***** END LICENSE BLOCK *****
@(#) $RCSfile: certdata.txt,v $ $Revision: 1.80 $ $Date: 2011/11/03 15:11:58 $
```

Python3_iOS

```
BSD 3-Clause License

Copyright (c) 2019, Nicolas Holzschuch
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this
  list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.

* Neither the name of the copyright holder nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

pywt

Copyright (c) 2006-2012 Filip Wasilewski <<http://en.ig.ma/>>
 Copyright (c) 2012-2019 The PyWavelets Developers <<https://github.com/PyWavelets/pywt>>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The PyWavelets repository and source distributions bundle some code that is adapted from compatibly licensed projects. We list these here.

pytz

MIT license.

python-webencodings

Copyright (c) 2012 by Simon Sapin.

Some rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * The names of the contributors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,

(continues on next page)

(continued from previous page)

SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

requests

Copyright 2018 Kenneth Reitz

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<https://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Rubicon ObjC

Copyright (c) 2014 Russell Keith-Magee.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of Rubicon nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This project uses code derived from Pyglet, which, in turn, derived its code from objective-ctypes. The license for objective-ctypes is as follows:

(continues on next page)

(continued from previous page)

Copyright (c) 2011, Phillip Nguyen
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of objective-ctypes nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

scipy

Copyright (c) 2001-2002 Enthought, Inc. 2003-2019, SciPy Developers.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY

(continues on next page)

(continued from previous page)

THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

scipy_arm64

Copyright 2019 goodclass

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

scikit-learn

New BSD License

Copyright (c) 2007-2019 The scikit-learn developers.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- a. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- b. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- c. Neither the name of the Scikit-learn Developers nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

scikit-image

Copyright (C) 2019, the scikit-image team
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of skimage nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

skimage/_shared/version_requirements.py:_check_version

Copyright (c) 2013 The IPython Development Team
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE

(continues on next page)

OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

skimage/_shared/version_requirements.py:is_installed:

Original Copyright (C) 2009-2011 Pierre Raybaut

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

skimage/external/tiffiffile:

Copyright (c) 2008-2014, Christoph Gohlke
Copyright (c) 2008-2014, The Regents of the University of California
Produced at the Laboratory for Fluorescence Dynamics
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

six

Copyright (c) 2010–2018 Benjamin Peterson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

soupsieve

MIT License

Copyright (c) 2018 Isaac Muse

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

SourceEditor

MIT License

Copyright (c) 2018 Louis D'hauwe

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

(continues on next page)

(continued from previous page)

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

StaSh

The MIT License (MIT)

Copyright (c) 2014 ywangd

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

toml

The MIT License

Copyright 2013-2018 William Pearson
Copyright 2015-2016 Julien Enselme
Copyright 2016 Google Inc.
Copyright 2017 Samuel Vasko
Copyright 2017 Nate Prewitt
Copyright 2017 Jack Evans

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR

(continues on next page)

(continued from previous page)

```
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
```

urllib3

```
This is the MIT license: http://www.opensource.org/licenses/mit-license.php
```

```
Copyright 2008-2016 Andrey Petrov and contributors (see CONTRIBUTORS.txt)
```

```
Permission is hereby granted, free of charge, to any person obtaining a copy of this
software and associated documentation files (the "Software"), to deal in the Software
without restriction, including without limitation the rights to use, copy, modify,
↳merge,
publish, distribute, sublicense, and/or sell copies of the Software, and to permit
↳persons
to whom the Software is furnished to do so, subject to the following conditions:
```

```
The above copyright notice and this permission notice shall be included in all copies
↳or
substantial portions of the Software.
```

```
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
↳PARTICULAR
PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
↳LIABLE
FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR
OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN THE SOFTWARE.
```

xlrd

```
There are two licenses associated with xlrd. This one relates to the bulk of
the work done on the library::
```

```
Portions copyright © 2005-2009, Stephen John Machin, Lingfo Pty Ltd
All rights reserved.
```

```
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
```

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. None of the names of Stephen John Machin, Lingfo Pty Ltd and any contributors may be used to endorse or promote products derived from this software without specific prior written permission.

```
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
```

(continues on next page)

```
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO,
THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS
BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
THE POSSIBILITY OF SUCH DAMAGE.
```

This one covers some earlier work::

```
/*-
 * Copyright (c) 2001 David Giffin.
 * All rights reserved.
 *
 * Based on the the Java version: Andrew Khan Copyright (c) 2000.
 *
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the
 * distribution.
 *
 * 3. All advertising materials mentioning features or use of this
 * software must display the following acknowledgment:
 * "This product includes software developed by
 * David Giffin ."
 *
 * 4. Redistributions of any form whatsoever must retain the following
 * acknowledgment:
 * "This product includes software developed by
 * David Giffin ."
 *
 * THIS SOFTWARE IS PROVIDED BY DAVID GIFFIN ``AS IS'' AND ANY
 * EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
 * PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL DAVID GIFFIN OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
 * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
 * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
 * OF THE POSSIBILITY OF SUCH DAMAGE.
 */
```

Zip

The MIT License (MIT)

Copyright (c) 2015 Roy Marmelstein

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

- paste by Arthur Shlain from the Noun Project

The following modules are integrated with Pyto.

4.1 `pyto_ui`

UI for scripts

The `pyto_ui` module contains classes for building and presenting a native UI, in app or in the Today Widget. This library's API is very similar to UIKit.

Warning: This library requires iOS / iPadOS 13.

This library may have a lot of similarities with `UIKit`, but subclassing isn't supported very well. Instead of overriding methods, you will often need to set properties to a function. For properties, setters are what makes the passed value take effect, so instead of override the getter, you should just set properties. If you really want to subclass a `View`, you can set properties from the initializer.

(Many docstrings are quoted from the Apple Documentation)

4.1.1 Getting Started

Each item presented on the screen is a `View` object. This module contains many `View` subclasses.

We can initialize a view like that:

```
import pyto_ui as ui

view = ui.View()
```

You can modify the view's attributes, like `background_color` for example:

```
view.background_color = ui.COLOR_SYSTEM_BACKGROUND
```

Then, call the `show_view()` function to show the view:

```
ui.show_view(view, ui.PRESENTATION_MODE_SHEET)
```

A view will be presented, with the system background color, white or black depending on if the device has dark mode enabled or not. It's important to set our view's background color because it will be transparent if it's not set. That looks great on widgets, but not in app.

NOTE: The thread will be blocked until the view is closed, but you can run code on another thread and modify the UI from there:

```
ui.show_view(view)
print("Closed") # This line will be called after the view is closed.
```

Now we have an empty view, the root view, we can add other views inside it, like a *Button*:

```
button = ui.Button(title="Hello World!")
button.size = (100, 50)
button.center = (view.width/2, view.height/2)
button.flex = [
    ui.FLEXIBLE_TOP_MARGIN,
    ui.FLEXIBLE_BOTTOM_MARGIN,
    ui.FLEXIBLE_LEFT_MARGIN,
    ui.FLEXIBLE_RIGHT_MARGIN
]
view.add_subview(button)
```

We are creating a button with title “Hello World!”, with 100 as width and 50 as height. We place it at center, and we set *flex* to have flexible margins so the button will always stay at center even if the root view will change its size.

To add an action to the button:

```
def button_pressed(sender):
    sender.superview.close()

button.action = button_pressed
```

We define a function that takes the button as parameter and we pass it to the button's *action* property. The *superview* property of the button is the view that contains it. With the *close()* function, we close it.

So we have this code:

```
import pyto_ui as ui

def button_pressed(sender):
    sender.superview.close()

view = ui.View()
view.background_color = ui.COLOR_SYSTEM_BACKGROUND

button = ui.Button(title="Hello World!")
button.size = (100, 50)
button.center = (view.width/2, view.height/2)
button.flex = [
    ui.FLEXIBLE_TOP_MARGIN,
    ui.FLEXIBLE_BOTTOM_MARGIN,
```

(continues on next page)

(continued from previous page)

```

    ui.FLEXIBLE_LEFT_MARGIN,
    ui.FLEXIBLE_RIGHT_MARGIN
]
button.action = button_pressed
view.add_subview(button)

ui.show_view(view, ui.PRESENTATION_MODE_SHEET)

print("Hello World!")

```

When the button is clicked, the UI will be closed and “Hello World!” will be printed. UIs can be presented on the Today widget if you set the widget script.

4.1.2 API Reference

Classes

This page contains a list of classes used by the `pyto_ui` API.

Views

A list of UI elements to be displayed on screen.

`class pyto_ui.View`

An object that manages the content for a rectangular area on the screen.

add_gesture_recognizer (*gesture_recognizer: pyto_ui.GestureRecognizer*)

Adds a gesture recognizer.

Parameters `gesture_recognizer` – The gesture recognizer to be added.

add_subview (*view: pyto_ui.View*)

Adds the given view to the receiver’s hierarchy.

Parameters `view` – The view to add.

alpha

The opacity of the view.

Return type float

appearance

The appearance of the view. See [Appearance](#) constants for possible values.

Return type

[Appearance](#)

background_color

The background color of the view.

Return type [Color](#)

become_first_responder () → bool

Becomes the first responder. On [TextView](#) and [TextField](#) objects, the keyboard will be shown. Returns a boolean indicating the success.

Return type bool

border_color

The color of the view's border

Return type *Color*

border_width

The width of the view's border.

Return type float

button_items

A list of *ButtonItem* objects to be displayed on the top bar. Works only if the view is the root view presented with *show_view()* or *push()*.

Return type List[*ButtonItem*]

center

The center point of the view's frame rectangle. Setting this value updates *frame* property appropriately. This value is a tuple with X and Y coordinates.

Return type Tuple[float, float]

center_x

The center x-coordinate of the view's frame rectangle. Setting this value updates *frame* property appropriately.

Return type float

center_y

The center y-coordinate of the view's frame rectangle. Setting this value updates *frame* property appropriately.

Return type float

clips_to_bounds

A boolean value that determines whether subviews are confined to the bounds of the view.

Return type bool

close()

Closes the view, if the receiver object is the root view presented to the user.

content_mode

A flag used to determine how a view lays out its content when its bounds change. See *Content Mode* <constants.html#content-mode>'_ constants for possible values.

Return type *Content Mode* <constants.html#content-mode>'_

corner_radius

The radius to use when drawing rounded corners for the view's background.

Return type float

first_responder

(Read Only) A boolean indicating the view is first responder. *UIKit* dispatches some types of events, such as motion events, to the first responder initially.

Return type bool

flex

A list that determines how the receiver resizes itself when its superview's bounds change. See *Auto Resizing* constants for possible values.

Return typeList[[Auto Resizing](#)]**frame**

The frame rectangle, which describes the view's location and size in its superview's coordinate system. This value is a tuple with X, Y, Width and Height values.

Return type Tuple[float, float, float, float]**gesture_recognizers**

(Read Only) Returns all gesture recognizers.

See `add_gesture_recognizer()`.**Return type** List[[GestureRecognizer](#)]**height**

The height of the view.

Return type float**hidden**

A boolean indicating whether the view is visible or not.

Return type bool**insert_subview** (*view: pyto_ui.View, index: int*)

Inserts the given view to the receiver's hierarchy at the given index.

Parameters

- **view** – The view to insert.
- **index** – The index where the view should be inserted.

insert_subview_above (*view: pyto_ui.View, above_view: pyto_ui.View*)

Inserts the given view to the receiver's hierarchy above another given view.

Parameters

- **view** – The view to insert.
- **above_view** – The view bellow the inserted view.

insert_subview_bellow (*view: pyto_ui.View, bellow_view: pyto_ui.View*)

Inserts the given view to the receiver's hierarchy bellow another given view.

Parameters

- **view** – The view to insert.
- **bellow_view** – The view above the inserted view.

layout

A function called when the view is resized. Takes the view as parameter.

Return type Callable[[[View](#)], None]**name**

The name identifying the view. To access a subview with its name, you can use the `subview_with_name()` function. `View` is also subscriptable, so you can do something like that:

```
import pyto_ui as ui

button = ui.Button()
```

(continues on next page)

(continued from previous page)

```
button.name = "Button"

view = ui.View()
view.add_subview(button)
view["Button"] # -> Button object
```

Return type str

navigation_bar_hidden

A boolean indicating whether the Navigation Bar of the View should be hidden.

Return type bool

opaque

A boolean indicating whether the view is opaque or not. Setting to `True` should prevent the view from having a transparent background.

Return type bool

origin

A point that specifies the coordinates of the view's rectangle's origin. This value is a tuple with X and Y coordinates.

Return type Tuple[float, float]

pop()

Pops the visible view controller from the navigation controller.

push (*view*: *pyto_ui.View*)

Presents the given additional view on top of the receiver.

Parameters *view* – The view to present.

remove_from_superview()

Removes the view from the parent's hierarchy.

remove_gesture_recognizer (*gesture_recognizer*: *pyto_ui.GestureRecognizer*)

Removes a gesture recognizer.

Parameters *gesture_recognizer* – The gesture recognizer to be removed.

resign_first_responder() → bool

Stops being the first responder. On *TextView* and *TextField* objects, the keyboard will be hidden. Returns a boolean indicating the success.

Return type bool

size

A size that specifies the height and width of the rectangle. This value is a tuple with height and width values.

Return type Tuple[float, float]

size_to_fit()

Sizes the view to fit its content.

subview_with_name (*name*) → *pyto_ui.View*

Returns the subview with the given name.

Raises `NameError` if no view is found.

Return type *View*

subviews

(Read Only) A list of the view's children.

See also `add_subview()`.

Return type List[View]

superview

(Read Only) The parent view containing the receiver view.

Return type View

tint_color

The tint color of the view. If set to `None`, the tint color will be inherited from the superview. The tint color affects some views like `Button` for title color, `TextView` for cursor color, etc.

Return type Color

title

If this view is directly presented, the top bar will show this view's title.

Return type str

user_interaction_enabled

A boolean indicating whether the view responds to touches.

Return type bool

width

The width of the view.

Return type float

x

The x-coordinate of the view.

Return type float

y

The y-coordinate of the point.

Return type float

class `pyto_ui.ImageView` (*image: Optional[PIL.Image.Image] = None, url: Optional[str] = None*)

A view displaying an image. The displayed image can be a PIL image, an UIKit UIImage (see `image_with_system_name()`) or can be directly downloaded from an URL.

image

The image displayed on screen. Can be a PIL image or an UIKit UIImage. See `image_with_system_name()` for more information about how to get a symbol image.

Return type Image.Image

load_from_url (*url*)

Loads and display the image at given url.

Parameters `url` – The URL of the image.

class `pyto_ui.Label` (*text: str = ""*)

A view displaying not editable and not selectable text.

adjusts_font_size_to_fit_width

A boolean indicating whether the label adjusts its font size to fit its size.

Return type bool

font

The font of the text.

Return type *Font*

line_break_mode

The line break mode.

Return type *Line Break Mode*

load_html (*html*)

Loads HTML in the Label.

Parameters **html** – The HTML code to load.

number_of_lines

The numbers of lines displayed in the label. Set to 0 to show all the text.

Return type *int*

text

The text to be displayed on the view.

Return type *str*

text_alignment

The text's alignment. For possible values, see *Text Alignment* constants.

Return type

Text Alignment

text_color

The color of the text.

Return type *Color*

```
class pyto_ui.TableViewCell (style:          ui_constants.TABLE_VIEW_STYLE      =   TA-  
                                     BLE_VIEW_CELL_STYLE_DEFAULT)
```

A cell contained in a *TableView*. Can have a title, a subtitle, an image and an accessory view.

For a list of supported style, see *Table View Cell Style* constants.

accessory_type

The type of accessory view placed to the right of the cell. See *Accessory Type* constants for possible values.

Return type

Accessory Type.

content_view

(Read Only) The view contained in the cell. Custom views should be added inside it.

Return type *View*

detail_text_label

(Read Only) The label containing secondary text. May return *None* for some *Table View Cell Style* values.

Return type *Label*

image_view

(Read Only) The view containing an image. May return *None* for some *Table View Cell Style* values.

Return type *Image View*

movable

A boolean indicating whether the cell is movable. If set to `True`, the container `TableViewSection` object should handle the move.

Return type `bool`

removable

A boolean indicating the cell is removable. If set to `True`, the container `TableViewSection` object should handle the removal.

Return type `bool`

text_label

(Read Only) The label containing the main text of the cell.

Return type `Label`

```
class pyto_ui.TableView (style:          ui_constants.TABLE_VIEW_STYLE      =      TA-
                          BLE_VIEW_STYLE_PLAIN, sections:  List[pyto_ui.TableViewSection]
                          = [])
```

A view containing a list of cells.

A Table View has a list of `TableViewSection` objects that represent groups of cells. A Table View has two possible styles. See [Table View Style](#).

deselect_row()

Deselects the current selected row.

edit_button_item

Returns a bar button item that toggles its title and associated state between Edit and Done. The button item is setup to edit the Table View.

Return type `ButtonItem`

reload_action

A function called when the button item is pressed. Takes the button item as parameter.

Return type `Callable[[TableView], None]`

sections

A list of `TableViewSection` containing cells to be displayed on the Table View. Setting a new value will reload automatically the contents of the Table View.

Return type `List[TableViewSection]`

```
class pyto_ui.TextView (text="")
```

An editable, multiline and scrollable view containing text.

autocapitalization_type

The type of autocapitalization to use while editing th text. See [Auto Capitalization](#) constants for possible values.

Return type

`Auto Capitalization`

autocorrection

A boolean indicating whether autocorrection is enabled.

Return type `bool`

did_begin_editing

A function called when the Text View begins editing. Takes the sender Text View as parameter.

Return type `Callable[[TextView], None]`

did_change

A function called when the Text View's text changes. Takes the sender Text View as parameter.

Return type Callable[[*TextView*], None]

did_end_editing

A function called when the Text View ends editing. Takes the sender Text View as parameter.

Return type Callable[[*TextView*], None]

editable

A boolean indicating whether the text is editable.

Return type bool

font

The font of the text displayed on screen.

Return type *Font*

keyboard_appearance

The appearance of the keyboard used while editing the text. See [Keyboard Appearance](#) constants for possible values.

Return type

[Keyboard Appearance](#)

keyboard_type

The type of keyboard to use while editing the text. See [Keyboard Type](#) constants for possible values.

Return type

[Keyboard Type](#)

load_html (*html*)

Loads HTML in the Text View.

Parameters *html* – The HTML code to load.

return_key_type

The type of return key to show on the keyboard used to edit the text. See [Return Key Type](#) constants for possible values.

Return type

[Return Key Type](#)

secure

A boolean indicating whether the keyboard should be configured to enter sensitive data.

Return type bool

selectable

A boolean indicating whether the text is selectable.

Return type bool

smart_dashes

A boolean indicating whether smart dashes are enabled.

Return type bool

smart_quotes

A boolean indicating whether smart quotes are enabled.

Return type bool

text

The text contained in the view.

Return type str

text_alignment

The alignment of the text displayed on screen. See [Text Alignment](#) constants for possible values.

Return type

[Text Alignment](#)

text_color

The color of the text displayed on screen.

Return type [Color](#)

class `pyto_ui.WebView` (*url: Optional[str] = None*)

A View that displays web content.

exception JavaScriptException

An exception while running JavaScript code. Raised by `evaluate_js()`.

can_go_back

(Read Only) A boolean indicating whether `go_back()` can be performed.

Return type bool

can_go_forward

(Read Only) A boolean indicating whether `go_forward()` can be performed.

Return type bool

did_fail_loading

A function called when the Web View failed to load contents. Takes the sender Web View and a string describing the error as parameters.

Return type Callable[[[WebView](#), str], None]

did_finish_loading

A function called when the Web View finished loading contents. Takes the sender Web View as parameter.

Return type Callable[[[WebView](#)], None]

did_start_loading

A function called when the Web View starts loading contents. Takes the sender Web View as parameter.

Return type Callable[[[WebView](#)], None]

evaluate_js (*code*) → str

Runs JavaScript code and returns a String representation of the evaluation result. Raises a [JavaScriptException](#).

Parameters `code` – JavaScript code to run.

Return type str

go_back ()

Goes back.

go_forward ()

Goes forward.

is_loading

(Read Only) A boolean indicating whether the Web View is loading content.

Return type bool

load_html (*html: str, base_url: Optional[str] = None*)

Loads an HTML string.

Parameters

- **html** – The HTML code to load.
- **base_url** – An optional URL used to resolve relative paths.

load_url (*url: str*)

Loads an URL.

Parameters **url** – The URL to load. Can be 'http://', 'https://' or 'file://'.

reload ()

Reloads the Web View.

stop ()

Stops loading content.

url

(Read Only) The current URL loaded into the Web View.

Return type str

Controls

class `pyto_ui.Control`

The base class for controls, which are visual elements that convey a specific action or intention in response to user interactions.

Inherited by `Button`, `SegmentedControl`, `Slider`, `Switch` and `TextField`

action

A function called when the control triggers its action. For example, a `Button` object calls this function when it's pressed.

Takes the `Control` object as parameter.

Return type Callable[[`Control`], None]

enabled

A boolean indicating whether the control is enabled.

Return type bool

horizontal_alignment

The horizontal alignment of the view's contents. See [Horizontal Alignment](#) constants for possible values.

Return type

[Horizontal Alignment](#)

vertical_alignment

The vertical alignment of the view's contents. See [Vertical Alignment](#) constants for possible values.

Return type [Vertical Alignment](#)

class `pyto_ui.SegmentedControl` (*segments: List[str] = []*)

A horizontal control made of multiple segments, each segment functioning as a discrete button. The function passed to `action` will be called when the segmented control changes its selection.

segments

A list of strings representing segments titles.

Return type List[str]

selected_segment

The index of selected segment.

Return type int

class `pyto_ui.Slider` (*value: float = 0.5*)

A control used to select a single value from a continuous range of values. The default range is located between 0 and 1. The function passed to *action* will be called when the slider changes its value.

maximum_track_color

The color used to tint the default maximum track.

Return type *Color*

maximum_value

The maximum value of the slider.

Return type float

minimum_track_color

The color used to tint the default minimum track.

Return type *Color*

minimum_value

The minimum value of the slider.

Return type float

set_value_with_animation (*value: float*)

Sets the value of the slider with an animation.

Parameters *value* – The value of the slider.

thumb_color

The color used to tint the default thumb.

Return type *Color*

value

The value of the slider between its range.

Return type float

class `pyto_ui.Switch` (*on=False*)

A control that offers a binary choice, such as On/Off. The function passed to *action* will be called when the switch changes its value.

on

A boolean indicating whether the switch is On.

Return type bool

on_color

The color used to tint the appearance of the switch when it is turned on.

Return type *Color*

set_on_with_animation (*on: bool*)

Sets the state of the switch to On or Off with an animation.

Parameters `on` – A boolean indicating whether the switch should be On.

thumb_color

The color used to tint the appearance of the thumb.

Return type *Color*

class `pyto_ui.Button` (*type: ui_constants.BUTTON_TYPE = BUTTON_TYPE_SYSTEM, title: str = "", image: Optional[PIL.Image.Image] = None*)

A control that executes your custom code in response to user interactions. To add an action, set `action`.

For types of buttons, see [Button Type](#) constants.

font

The font to be applied to the text.

Return type *Font*

image

The image displayed on the button. Can be a PIL image or an UIKit symbol image. For more information about symbols, see `image_with_system_name()`.

Return type `PIL.Image.Image`

title

The title of the button

Return type `str`

title_color

The color of the title.

Return type *Color*

class `pyto_ui.TextField` (*text: str = "", placeholder: Optional[str] = None*)

A field to type single line text. The function passed to `action` will be called when the text field changes its text.

autocapitalization_type

The type of autocapitalization to use while editing th text. See [Auto Capitalization](#) constants for possible values.

Return type

Auto Capitalization

autocorrection

A boolean indicating whether autocorrection is enabled.

Return type `bool`

did_begin_editing

A function called when the Text Field begins editing. Takes the sender Text Field as parameter.

Return type `Callable[[TextField], None]`

did_end_editing

A function called when the Text Field ends editing. Takes the sender Text Field as parameter.

Return type `Callable[[TextField], None]`

font

The font of the text displayed on screen.

Return type *Font*

keyboard_appearance

The appearance of the keyboard used while editing the text. See [Keyboard Appearance](#) constants for possible values.

Return type

[Keyboard Appearance](#)

keyboard_type

The type of keyboard to use while editing the text. See [Keyboard Type](#) constants for possible values.

Return type

[Keyboard Type](#)

placeholder

A gray text shown when there is no text.

Return type str**return_key_type**

The type of return key to show on the keyboard used to edit the text. See [Return Key Type](#) constants for possible values.

Return type

[Return Key Type](#)

secure

A boolean indicating whether the keyboard should be configured to enter sensitive data. The text entered by the user will be hidden.

Return type bool**smart_dashes**

A boolean indicating whether smart dashes are enabled.

Return type bool**smart_quotes**

A boolean indicating whether smart quotes are enabled.

Return type bool**text**

The text contained in the Text Field.

Return type str**text_alignment**

The alignment of the text displayed on screen. See [Text Alignment](#) constants for possible values.

Return type

[Text Alignment](#)

text_color

The color of the text displayed on screen.

Return type *Color***Other classes**

A list of classes that don't represent views.

class `pyto_ui.Color` (*py_color*)

A `Color` object represents a color to be displayed on screen.

Example:

```
import pyto_ui as ui

# RGB
black = ui.Color.rgb(0, 0, 0, 1)

# White
white = ui.Color.white(1, 1)

# Dynamic
background = ui.Color.dynamic(light=white, dark=black)
```

For pre-defined colors, see [Color constants](#).

alpha () → float

Returns the alpha value of the color.

Return type float

blue () → float

Returns the blue value of the color.

Return type float

classmethod **dynamic** (*light: pyto_ui.Color, dark: pyto_ui.Color*) → `pyto_ui.Color`

Initializes and returns a color that dynamically changes in dark or light mode.

Parameters

- **light** – `Color` object to be displayed in light mode.
- **dark** – `Color` object to be displayed in dark mode.

Return type `Color`

green () → float

Returns the green value of the color.

Return type float

red () → float

Returns the red value of the color.

Return type float

classmethod **rgb** (*red: float, green: float, blue, alpha: float*) → `pyto_ui.Color`

Initializes a color from RGB values.

All values should be located between 0 and 1, not between 0 and 255.

Parameters

- **red** – The red value.
- **green** – The green value.
- **blue** – The blue value.
- **alpha** – The opacity value.

Return type `Color`

classmethod `white` (*white: float, alpha: float*) → `pyto_ui.Color`
 Initializes and returns a color from white value.

All values should be located between 0 and 1, not between 0 and 255.

Parameters

- **white** – The grayscale value.
- **alpha** – The opacity value.

Return type *Color*

class `pyto_ui.Font` (*name: str, size: float*)

A `Font` object represents a font (with name and size) to be used on labels, buttons, text views etc.

classmethod `bold_system_font_of_size` (*size: float*) → `pyto_ui.Font`

Returns the font object used for standard interface items that are rendered in boldface type in the specified size

Parameters **size** – The size (in points) for the font. This value must be greater than 0.0.

Return type *Font*

classmethod `font_names_for_family_name` (*name: str*) → `List[str]`

Returns an array of font names available in a particular font family.

Parameters **name** – The name of the font family. Use the `font_family_names()` function to get an array of the available font family names on the system.

Return type `List[str]`

classmethod `font_with_style` (*style: ui_constants.FONT_TEXT_STYLE*) → `pyto_ui.Font`

Returns an instance of the system font for the specified text style and scaled appropriately for the user's selected content size category.

Parameters **style** – The text style for which to return a font. See [Font Text Style constants](#) for possible values.

Return type *Font*

classmethod `italic_system_font_of_size` (*size: float*) → `pyto_ui.Font`

Returns the font object used for standard interface items that are rendered in italic type in the specified size.

Parameters **size** – The size (in points) for the font. This value must be greater than 0.0.

Return type *Font*

classmethod `system_font_of_size` (*size: float*) → `pyto_ui.Font`

Returns the font object used for standard interface items in the specified size.

Parameters **size** – The size (in points) to which the font is scaled. This value must be greater than 0.0.

Return type *Font*

with_size (*size: float*) → `pyto_ui.Font`

Returns a font object that is the same as the receiver but which has the specified size instead.

Parameters **size** – The desired size (in points) of the new font object. This value must be greater than 0.0.

Return type *Font*

```
class pyto_ui.GestureRecognizer (type: ui_constants.GESTURE_TYPE, action: Optional[Callable[[pyto_ui.GestureRecognizer], None]] = None)
```

A gesture-recognizer object—or, simply, a gesture recognizer—decouples the logic for recognizing a sequence of touches (or other input) and acting on that recognition. When one of these objects recognizes a common gesture or, in some cases, a change in the gesture, it sends an action message to each designated target object.

This class represents the type of gesture passed to the `type` initializer parameter. See [Gesture Type](#) constants for possible values. When the gesture is starting, cancelling or changing, `action` is called with the gesture recognizer as parameter. You can then access the location and the state from it.

Example:

```
'''
Move a circle with finger.
'''

import pyto_ui as ui

view = ui.View()
view.background_color = ui.COLOR_SYSTEM_BACKGROUND

circle = ui.View()
circle.size = (50, 50)
circle.center = (view.width/2, view.height/2)
circle.flexible_margins = True
circle.corner_radius = 25
circle.background_color = ui.COLOR_LABEL
view.add_subview(circle)

def move(sender: ui.GestureRecognizer):
    if sender.state == ui.GESTURE_STATE_CHANGED:
        circle.center = sender.location

gesture = ui.GestureRecognizer(ui.GESTURE_TYPE_PAN)
gesture.action = move
view.add_gesture_recognizer(gesture)

ui.show_view(view)
```

action

A function called to handle the gesture. Takes the sender gesture recognizer as parameter.

Return type Callable[[[GestureRecognizer](#)], None]

allowedTouchTypes

An array of touch types used to distinguish type of touches. For possible values, see [Touch Type](#) constants.

Return type List[[Touch Type](#)]

allowed_touch_types

An array of touch types used to distinguish type of touches. For possible values, see [Touch Type](#) constants.

Return type

List[[Touch Type](#)]

cancels_touches_in_view

A Boolean value affecting whether touches are delivered to a view when a gesture is recognized.

Return type bool

delays_touches_began

A Boolean value determining whether the receiver delays sending touches in a begin phase to its view.

Return type bool

delays_touches_ended

A Boolean value determining whether the receiver delays sending touches in an end phase to its view.

Return type bool

enabled

A boolean indicating whether the gesture recognizer is enabled.

Return type bool

location

(Read Only) Returns a tuple with the X and the Y position of the gesture in its container view.

Return type Tuple[float, float]

number_of_touches

(Read Only) Returns the number of touches involved in the gesture represented by the receiver.

Return type int

requires_exclusive_touch_type

A Boolean indicating whether the gesture recognizer considers touches of different types simultaneously.

Return type bool

state

(Read Only) The current state of the gesture recognizer.

Return type *Gesture State*

view

(Read Only) Returns the view associated with the gesture.

Return type *View*

x

(Read Only) Returns the X position of the gesture in its container view.

Return type float

y

(Read Only) Returns the Y position of the gesture in its container view.

class `pyto_ui.TableViewSection` (*title: str, cells: List[TableViewCell]*)

An object representing a section in a Table View. A section has a title and a list of cells contained in.

cells

Cells contained in the section. After setting a value, the section will be reloaded automatically.

Return type *TableViewCell*

did_delete_cell

A function called when a cell contained in the section is deleted. Takes the sender section and the selected deleted cell's index as parameters. This function should be used to remove the data corresponding to the cell from the database.

Return type Callable[[*TableViewCell*, int], None]

did_move_cell

A function called when a cell contained in the section is moved. Takes the sender section, the moved deleted cell's index and the destination index as parameters. This function should be used to move the data corresponding to the cell from the database.

Return type Callable[[*TableViewSection*, int, int], None]

did_select_cell

A function called when a cell contained in the section is selected. Takes the sender section and the selected cell's index as parameters.

Return type Callable[[*TableViewSection*, int], None]

did_tap_cell_accessory_button

A function called when the accessory button of a cell contained in the section is pressed. Takes the sender section and the cell's index as parameters.

Return type Callable[[*TableViewSection*, int], None]

table_view

(Read Only) Returns the Table view associated with the section.

Return type *TableView*

title

The title of the section displayed on screen.

Return type str

```
class pyto_ui.ButtonItem(title: Optional[str] = None, image: Optional[PIL.Image.Image]
                        = None, system_item: Optional[ui_constants.SYSTEM_ITEM]
                        = None, style: ui_constants.BUTTON_ITEM_STYLE = BUT-
                        TON_ITEM_STYLE_PLAIN)
```

A special kind of button that can be placed on the view's navigation bar. Can have a title, and image or a system type.

action

A function called when the button item is pressed. Takes the button item as parameter.

Return type Callable[[*ButtonItem*], None]

enabled

A boolean indicating whether the button is enabled.

Return type bool

image

A PIL image object displayed on screen. May also be an UIKit UIImage symbol. See *image_with_system_name()*.

Return type PIL.Image.Image

style

The button item style. See *Button Item Style* constants for possible values.

Return type

Button Item Style

title

The title of the button displayed on screen.

Return type str

class `pyto_ui.Alert` (*title: str, message: str*)
 A class representing an alert.

Example:

```
from pyto_ui import Alert

alert = Alert("Hello", "Hello World!")
alert.add_action("Ok")
alert.add_cancel_action("Cancel")
if (alert.show() == "Ok"):
    print("Good Bye!")
```

add_action (*title: str*)

Adds an action with given title.

Parameters **title** – The title of the action.

add_cancel_action (*title: str*)

Adds a cancel action with given title. Can only be added once.

Parameters **title** – The title of the action.

add_destructive_action (*title: str*)

Adds a destructive action with given title.

Parameters **title** – The title of the action.

show () → str

Shows alert.

Returns the title of the selected action.

Return type str

Functions

This page contains a list of functions defined in the `pyto_ui` module.

`pyto_ui.font_family_names` () → List[str]

Returns all font family names that can be used to initialize a font.

Return type List[str]

`pyto_ui.image_with_system_name` (*name: str*) → None

Returns a system symbol image from given name. The return value is an UIKit UIImage object, so it can only be used on the `pyto_ui` library. More info about symbols on [Apple's Web Site](#) .

Parameters **name** – The name of the SF Symbol.

Return type UIImage

`pyto_ui.show_view` (*view: pyto_ui.View, mode: ui_constants.PRESENTATION_MODE*)

Presents the given view.

This function doesn't return until the view is closed. You can use another thread to perform background tasks and modify the UI after it's presented.

On iPad, if the view has a custom size, it will be used for the presentation.

Parameters

- **view** – The *View* object to present.

- **mode** – The presentation mode to use. The value will be ignored on a widget. See [Presentation Mode](#) constants for possible values.

Constants

This page contains a list of constants used by the `pyto_ui` API.

UI Elements Colors

`pyto_ui.COLOR_LABEL`

The color for text labels containing primary content.

`pyto_ui.COLOR_SECONDARY_LABEL`

The color for text labels containing secondary content.

`pyto_ui.COLOR_TERTIARY_LABEL`

The color for text labels containing tertiary content.

`pyto_ui.COLOR_QUATERNARY_LABEL`

The color for text labels containing quaternary content.

`pyto_ui.COLOR_SYSTEM_FILL`

An overlay fill color for thin and small shapes.

`pyto_ui.COLOR_SECONDARY_SYSTEM_FILL`

An overlay fill color for medium-size shapes.

`pyto_ui.COLOR_TERTIARY_SYSTEM_FILL`

An overlay fill color for large shapes.

`pyto_ui.COLOR_QUATERNARY_SYSTEM_FILL`

An overlay fill color for large areas containing complex content.

`pyto_ui.COLOR_PLACEHOLDER_TEXT`

The color for placeholder text in controls or text views.

`pyto_ui.COLOR_SYSTEM_BACKGROUND`

The color for the main background of your interface.

`pyto_ui.COLOR_SECONDARY_SYSTEM_BACKGROUND`

The color for content layered on top of the main background.

`pyto_ui.COLOR_TERTIARY_SYSTEM_BACKGROUND`

The color for content layered on top of secondary backgrounds.

`pyto_ui.COLOR_SYSTEM_GROUPED_BACKGROUND`

The color for the main background of your grouped interface.

`pyto_ui.COLOR_SECONDARY_GROUPED_BACKGROUND`

The color for content layered on top of the main background of your grouped interface.

`pyto_ui.COLOR_TERTIARY_GROUPED_BACKGROUND`

The color for content layered on top of secondary backgrounds of your grouped interface.

`pyto_ui.COLOR_SEPARATOR`

The color for thin borders or divider lines that allows some underlying content to be visible.

`pyto_ui.COLOR_OPAQUE_SEPARATOR`

The color for borders or divider lines that hide any underlying content.

`pyto_ui.COLOR_LINK`

The color for links.

`pyto_ui.COLOR_DARK_TEXT`

The nonadaptable system color for text on a light background.

`pyto_ui.COLOR_LIGHT_TEXT`

The nonadaptable system color for text on a dark background.

Standard Colors

`pyto_ui.COLOR_SYSTEM_BLUE`

A blue color that automatically adapts to the current trait environment.

`pyto_ui.COLOR_SYSTEM_GREEN`

A green color that automatically adapts to the current trait environment.

`pyto_ui.COLOR_SYSTEM_INDIGO`

An indigo color that automatically adapts to the current trait environment.

`pyto_ui.COLOR_SYSTEM_ORANGE`

An orange color that automatically adapts to the current trait environment.

`pyto_ui.COLOR_SYSTEM_PINK`

A pink color that automatically adapts to the current trait environment.

`pyto_ui.COLOR_SYSTEM_PURPLE`

A purple color that automatically adapts to the current trait environment.

`pyto_ui.COLOR_SYSTEM_RED`

A red color that automatically adapts to the current trait environment.

`pyto_ui.COLOR_SYSTEM_TEAL`

A teal color that automatically adapts to the current trait environment.

`pyto_ui.COLOR_SYSTEM_YELLOW`

A yellow color that automatically adapts to the current trait environment.

`pyto_ui.COLOR_SYSTEM_GRAY`

The base gray color.

`pyto_ui.COLOR_SYSTEM_GRAY2`

A second-level shade of grey.

`pyto_ui.COLOR_SYSTEM_GRAY3`

A third-level shade of grey.

`pyto_ui.COLOR_SYSTEM_GRAY4`

A fourth-level shade of grey.

`pyto_ui.COLOR_SYSTEM_GRAY5`

A fifth-level shade of grey.

`pyto_ui.COLOR_SYSTEM_GRAY6`

A sixth-level shade of grey.

Fixed Colors

`pyto_ui.COLOR_CLEAR`

A color object with grayscale and alpha values that are both 0.0.

`pyto_ui.COLOR_BLACK`

A color object in the sRGB color space with a grayscale value of 0.0 and an alpha value of 1.0.

`pyto_ui.COLOR_BLUE`

A color object with RGB values of 0.0, 0.0, and 1.0 and an alpha value of 1.0.

`pyto_ui.COLOR_BROWN`

A color object with RGB values of 0.6, 0.4, and 0.2 and an alpha value of 1.0.

`pyto_ui.COLOR_CYAN`

A color object with RGB values of 0.0, 1.0, and 1.0 and an alpha value of 1.0.

`pyto_ui.COLOR_DARK_GRAY`

A color object with a grayscale value of 1/3 and an alpha value of 1.0.

`pyto_ui.COLOR_GRAY`

A color object with a grayscale value of 0.5 and an alpha value of 1.0.

`pyto_ui.COLOR_GREEN`

A color object with RGB values of 0.0, 1.0, and 0.0 and an alpha value of 1.0.

`pyto_ui.COLOR_LIGHT_GRAY`

A color object with a grayscale value of 2/3 and an alpha value of 1.0.

`pyto_ui.COLOR_MAGENTA`

A color object with RGB values of 1.0, 0.0, and 1.0 and an alpha value of 1.0.

`pyto_ui.COLOR_ORANGE`

A color object with RGB values of 1.0, 0.5, and 0.0 and an alpha value of 1.0.

`pyto_ui.COLOR_PURPLE`

A color object with RGB values of 0.5, 0.0, and 0.5 and an alpha value of 1.0.

`pyto_ui.COLOR_RED`

A color object with RGB values of 1.0, 0.0, and 0.0 and an alpha value of 1.0.

`pyto_ui.COLOR_WHITE`

A color object with a grayscale value of 1.0 and an alpha value of 1.0.

`pyto_ui.COLOR_YELLOW`

A color object with RGB values of 1.0, 1.0, and 0.0 and an alpha value of 1.0.

Keyboard Appearance

`pyto_ui.KEYBOARD_APPEARANCE_DEFAULT`

Specifies the default keyboard appearance for the current input method.

`pyto_ui.KEYBOARD_APPEARANCE_LIGHT`

Specifies a keyboard appearance suitable for a light UI look.

`pyto_ui.KEYBOARD_APPEARANCE_DARK`

Specifies a keyboard appearance suitable for a dark UI look.

Keyboard Type

`pyto_ui.KEYBOARD_TYPE_DEFAULT`

Specifies the default keyboard for the current input method.

`pyto_ui.KEYBOARD_TYPE_ASCII_CAPABLE`

Specifies a keyboard that displays standard ASCII characters.

`pyto_ui.KEYBOARD_TYPE_ASCII_CAPABLE_NUMBER_PAD`

Specifies a number pad that outputs only ASCII digits.

`pyto_ui.KEYBOARD_TYPE_DECIMAL_PAD`

Specifies a keyboard with numbers and a decimal point.

`pyto_ui.KEYBOARD_TYPE_EMAIL_ADDRESS`

Specifies a keyboard optimized for entering email addresses. This keyboard type prominently features the at (“@”), period (“.”) and space characters.

`pyto_ui.KEYBOARD_TYPE_NAME_PHONE_PAD`

Specifies a keypad designed for entering a person’s name or phone number. This keyboard type does not support auto-capitalization.

`pyto_ui.KEYBOARD_TYPE_NUMBER_PAD`

Specifies a numeric keypad designed for PIN entry. This keyboard type prominently features the numbers 0 through 9. This keyboard type does not support auto-capitalization.

`pyto_ui.KEYBOARD_TYPE_NUMBERS_AND_PUNCTUATION`

Specifies the numbers and punctuation keyboard.

`pyto_ui.KEYBOARD_TYPE_PHONE_PAD`

Specifies a keypad designed for entering telephone numbers. This keyboard type prominently features the numbers 0 through 9 and the “*” and “#” characters. This keyboard type does not support auto-capitalization.

`pyto_ui.KEYBOARD_TYPE_TWITTER`

Specifies a keyboard optimized for Twitter text entry, with easy access to the at (“@”) and hash (“#”) characters.

`pyto_ui.KEYBOARD_TYPE_URL`

Specifies a keyboard optimized for URL entry. This keyboard type prominently features the period (“.”) and slash (“/”) characters and the “.com” string.

`pyto_ui.KEYBOARD_TYPE_WEB_SEARCH`

Specifies a keyboard optimized for web search terms and URL entry. This keyboard type prominently features the space and period (“.”) characters.

Return Key Type

`pyto_ui.RETURN_KEY_TYPE_DEFAULT`

Specifies that the visible title of the Return key is “return”.

`pyto_ui.RETURN_KEY_TYPE_CONTINUE`

Specifies that the visible title of the Return key is “Continue”.

`pyto_ui.RETURN_KEY_TYPE_DONE`

Specifies that the visible title of the Return key is “Done”.

`pyto_ui.RETURN_KEY_TYPE_EMERGENCY_CALL`

Specifies that the visible title of the Return key is “Emergency Call”.

`pyto_ui.RETURN_KEY_TYPE_GO`

Specifies that the visible title of the Return key is “Go”.

`pyto_ui.RETURN_KEY_TYPE_GOOGLE`

Specifies that the visible title of the Return key is “Google”.

`pyto_ui.RETURN_KEY_TYPE_JOIN`

Specifies that the visible title of the Return key is “Join”.

`pyto_ui.RETURN_KEY_TYPE_NEXT`

Specifies that the visible title of the Return key is “Next”.

`pyto_ui.RETURN_KEY_TYPE_ROUTE`
Specifies that the visible title of the Return key is “Route”.

`pyto_ui.RETURN_KEY_TYPE_SEARCH`
Specifies that the visible title of the Return key is “Search”.

`pyto_ui.RETURN_KEY_TYPE_SEND`
Specifies that the visible title of the Return key is “Send”.

`pyto_ui.RETURN_KEY_TYPE_YAHOO`
Specifies that the visible title of the Return key is “Yahoo”.

Auto Capitalization

`pyto_ui.AUTO_CAPITALIZE_NONE`
Specifies that there is no automatic text capitalization.

`pyto_ui.AUTO_CAPITALIZE_ALL`
Specifies automatic capitalization of all characters, such as for entry of two-character state abbreviations for the United States.

`pyto_ui.AUTO_CAPITALIZE_SENTENCES`
Specifies automatic capitalization of the first letter of each sentence.

`pyto_ui.AUTO_CAPITALIZE_WORDS`
Specifies automatic capitalization of the first letter of each word.

Font Text Style

`pyto_ui.FONT_TEXT_STYLE_BODY`
The font used for body text.

`pyto_ui.FONT_TEXT_STYLE_CALLOUT`
The font used for callouts.

`pyto_ui.FONT_TEXT_STYLE_CAPTION_1`
The font used for standard captions.

`pyto_ui.FONT_TEXT_STYLE_CAPTION_2`
The font used for alternate captions.

`pyto_ui.FONT_TEXT_STYLE_FOOTNOTE`
The font used in footnotes.

`pyto_ui.FONT_TEXT_STYLE_HEADLINE`
The font used for headings.

`pyto_ui.FONT_TEXT_STYLE_SUBHEADLINE`
The font used for subheadings.

`pyto_ui.FONT_TEXT_STYLE_LARGE_TITLE`
The font style for large titles.

`pyto_ui.FONT_TEXT_STYLE_TITLE_1`
The font used for first level hierarchical headings.

`pyto_ui.FONT_TEXT_STYLE_TITLE_2`
The font used for second level hierarchical headings.

`pyto_ui.FONT_TEXT_STYLE_TITLE_3`
The font used for third level hierarchical headings.

Font Size

`pyto_ui.FONT_LABEL_SIZE`
Returns the standard font size used for labels.

`pyto_ui.FONT_BUTTON_SIZE`
Returns the standard font size used for buttons.

`pyto_ui.FONT_SMALL_SYSTEM_SIZE`
Returns the size of the standard small system font.

`pyto_ui.FONT_SYSTEM_SIZE`
Returns the size of the standard system font.

Button Type

`pyto_ui.BUTTON_TYPE_SYSTEM`
A system style button, such as those shown in navigation bars and toolbars.

`pyto_ui.BUTTON_TYPE_CONTACT_ADD`
A contact add button.

`pyto_ui.BUTTON_TYPE_CUSTOM`
No button style.

`pyto_ui.BUTTON_TYPE_DETAIL_DISCLOSURE`
A detail disclosure button.

`pyto_ui.BUTTON_TYPE_INFO_DARK`
An information button that has a dark background.

`pyto_ui.BUTTON_TYPE_INFO_LIGHT`
An information button that has a light background.

Gesture Type

`pyto_ui.GESTURE_TYPE_LONG_PRESS`
A long press gesture.

`pyto_ui.GESTURE_TYPE_PAN`
A dragging gesture.

`pyto_ui.GESTURE_TYPE_TAP`
A tap gesture.

Gesture State

`pyto_ui.GESTURE_STATE_POSSIBLE`
The gesture recognizer has not yet recognized its gesture, but may be evaluating touch events. This is the default state.

`pyto_ui.GESTURE_STATE_BEGAN`

The gesture recognizer has received touch objects recognized as a continuous gesture. It sends its action message (or messages) at the next cycle of the run loop.

`pyto_ui.GESTURE_STATE_CHANGED`

The gesture recognizer has received touches recognized as a change to a continuous gesture. It sends its action message (or messages) at the next cycle of the run loop.

`pyto_ui.GESTURE_STATE_ENDED`

The gesture recognizer has received touches recognized as the end of a continuous gesture. It sends its action message (or messages) at the next cycle of the run loop and resets its state to possible.

`pyto_ui.GESTURE_STATE_CANCELLED`

The gesture recognizer has received touches resulting in the cancellation of a continuous gesture. It sends its action message (or messages) at the next cycle of the run loop and resets its state to possible.

`pyto_ui.GESTURE_STATE_RECOGNIZED`

The gesture recognizer has received a multi-touch sequence that it recognizes as its gesture. It sends its action message (or messages) at the next cycle of the run loop and resets its state to possible.

Touch Type

`pyto_ui.TOUCH_TYPE_DIRECT`

A touch resulting from direct contact with the screen.

`pyto_ui.TOUCH_TYPE_INDIRECT`

A touch that did not result from contact with the screen.

`pyto_ui.TOUCH_TYPE_PENCIL`

A touch from Apple Pencil.

Auto Resizing

`pyto_ui.FLEXIBLE_WIDTH`

Resizing performed by expanding or shrinking a view's width.

`pyto_ui.FLEXIBLE_HEIGHT`

Resizing performed by expanding or shrinking a view's height.

`pyto_ui.FLEXIBLE_TOP_MARGIN`

Resizing performed by expanding or shrinking a view in the direction of the top margin.

`pyto_ui.FLEXIBLE_BOTTOM_MARGIN`

Resizing performed by expanding or shrinking a view in the direction of the bottom margin.

`pyto_ui.FLEXIBLE_LEFT_MARGIN`

Resizing performed by expanding or shrinking a view in the direction of the left margin.

`pyto_ui.FLEXIBLE_RIGHT_MARGIN`

Resizing performed by expanding or shrinking a view in the direction of the right margin.

Content Mode

`pyto_ui.CONTENT_MODE_SCALE_TO_FILL`

The option to scale the content to fit the size of itself by changing the aspect ratio of the content if necessary.

`pyto_ui.CONTENT_MODE_SCALE_ASPECT_FIT`

The option to scale the content to fit the size of the view by maintaining the aspect ratio. Any remaining area of the view's bounds is transparent.

`pyto_ui.CONTENT_MODE_SCALE_ASPECT_FILL`

The option to scale the content to fill the size of the view. Some portion of the content may be clipped to fill the view's bounds.

`pyto_ui.CONTENT_MODE_REDRAW`

The option to redisplay the view when the bounds change by invoking the `setNeedsDisplay()` method.

`pyto_ui.CONTENT_MODE_CENTER`

The option to center the content in the view's bounds, keeping the proportions the same.

`pyto_ui.CONTENT_MODE_TOP`

The option to center the content aligned at the top in the view's bounds.

`pyto_ui.CONTENT_MODE_BOTTOM`

The option to center the content aligned at the bottom in the view's bounds.

`pyto_ui.CONTENT_MODE_LEFT`

The option to align the content on the left of the view.

`pyto_ui.CONTENT_MODE_RIGHT`

The option to align the content on the right of the view.

`pyto_ui.CONTENT_MODE_TOP_LEFT`

The option to align the content in the top-left corner of the view.

`pyto_ui.CONTENT_MODE_TOP_RIGHT`

The option to align the content in the top-right corner of the view.

`pyto_ui.CONTENT_MODE_BOTTOM_LEFT`

The option to align the content in the bottom-left corner of the view.

`pyto_ui.CONTENT_MODE_BOTTOM_RIGHT`

The option to align the content in the bottom-right corner of the view.

Horizontal Alignment

`pyto_ui.HORIZONTAL_ALIGNMENT_CENTER`

Aligns the content horizontally in the center of the control.

`pyto_ui.HORIZONTAL_ALIGNMENT_FILL`

Aligns the content horizontally to fill the content rectangles; text may wrap and images may be stretched.

`pyto_ui.HORIZONTAL_ALIGNMENT_LEADING`

Aligns the content horizontally from the leading edge of the control.

`pyto_ui.HORIZONTAL_ALIGNMENT_LEFT`

Aligns the content horizontally from the left of the control (the default).

`pyto_ui.HORIZONTAL_ALIGNMENT_RIGHT`

Aligns the content horizontally from the right of the control.

`pyto_ui.HORIZONTAL_ALIGNMENT_TRAILING`

Aligns the content horizontally from the trailing edge of the control.

Vertical Alignment

`pyto_ui.VERTICAL_ALIGNMENT_BOTTOM`

Aligns the content vertically at the bottom in the control.

`pyto_ui.VERTICAL_ALIGNMENT_CENTER`

Aligns the content vertically in the center of the control.

`pyto_ui.VERTICAL_ALIGNMENT_FILL`

Aligns the content vertically to fill the content rectangle; images may be stretched.

`pyto_ui.VERTICAL_ALIGNMENT_TOP`

Aligns the content vertically at the top in the control (the default).

Text Alignment

`pyto_ui.TEXT_ALIGNMENT_LEFT`

Text is visually left aligned.

`pyto_ui.TEXT_ALIGNMENT_RIGHT`

Text is visually right aligned.

`pyto_ui.TEXT_ALIGNMENT_CENTER`

Text is visually center aligned.

`pyto_ui.TEXT_ALIGNMENT_JUSTIFIED`

Text is justified.

`pyto_ui.TEXT_ALIGNMENT_NATURAL`

Use the default alignment associated with the current localization of the app. The default alignment for left-to-right scripts is left, and the default alignment for right-to-left scripts is right.

Line Break Mode

`pyto_ui.LINE_BREAK_MODE_BY_WORD_WRAPPING`

Wrapping occurs at word boundaries, unless the word itself doesn't fit on a single line.

`pyto_ui.LINE_BREAK_MODE_BY_CHAR_WRAPPING`

Wrapping occurs before the first character that doesn't fit.

`pyto_ui.LINE_BREAK_MODE_BY_CLIPPING`

Lines are simply not drawn past the edge of the text container.

`pyto_ui.LINE_BREAK_MODE_BY_TRUNCATING_HEAD`

The line is displayed so that the end fits in the container and the missing text at the beginning of the line is indicated by an ellipsis glyph. Although this mode works for multiline text, it is more often used for single line text.

`pyto_ui.LINE_BREAK_MODE_BY_TRUNCATING_TAIL`

The line is displayed so that the beginning fits in the container and the missing text at the end of the line is indicated by an ellipsis glyph. Although this mode works for multiline text, it is more often used for single line text.

`pyto_ui.LINE_BREAK_MODE_BY_TRUNCATING_MIDDLE`

The line is displayed so that the beginning and end fit in the container and the missing text in the middle is indicated by an ellipsis glyph. This mode is used for single-line layout; using it with multiline text truncates the text into a single line.

Table View Cell Style

`pyto_ui.TABLE_VIEW_CELL_STYLE_DEFAULT`

A simple style for a cell with a text label (black and left-aligned) and an optional image view.

`pyto_ui.TABLE_VIEW_CELL_STYLE_SUBTITLE`

A style for a cell with a left-aligned label across the top and a left-aligned label below it in smaller gray text.

`pyto_ui.TABLE_VIEW_CELL_STYLE_VALUE1`

A style for a cell with a label on the left side of the cell with left-aligned and black text; on the right side is a label that has smaller blue text and is right-aligned. The Settings application uses cells in this style.

`pyto_ui.TABLE_VIEW_CELL_STYLE_VALUE2`

A style for a cell with a label on the left side of the cell with text that is right-aligned and blue; on the right side of the cell is another label with smaller text that is left-aligned and black. The Phone/Contacts application uses cells in this style.

Accessory Type

`pyto_ui.ACCESSORY_TYPE_NONE`

No accessory view.

`pyto_ui.ACCESSORY_TYPE_CHECKMARK`

A checkmark image.

`pyto_ui.ACCESSORY_TYPE_DETAIL_BUTTON`

An information button.

`pyto_ui.ACCESSORY_TYPE_DETAIL_DISCLOSURE_BUTTON`

An information button and a disclosure (chevron) control.

`pyto_ui.ACCESSORY_TYPE_DISCLOSURE_INDICATOR`

A chevron-shaped control for presenting new content.

Table View Style

`pyto_ui.TABLE_VIEW_STYLE_PLAIN`

A plain table view.

`pyto_ui.TABLE_VIEW_STYLE_GROUPED`

A table view whose sections present distinct groups of rows.

Text Field Border Style

`pyto_ui.TEXT_FIELD_BORDER_STYLE_NONE`

The text field does not display a border.

`pyto_ui.TEXT_FIELD_BORDER_STYLE_BEZEL`

Displays a bezel-style border for the text field. This style is typically used for standard data-entry fields.

`pyto_ui.TEXT_FIELD_BORDER_STYLE_LINE`

Displays a thin rectangle around the text field.

`pyto_ui.TEXT_FIELD_BORDER_STYLE_ROUNDED_RECT`

Displays a rounded-style border for the text field.

Button Item Style

`pyto_ui.BUTTON_ITEM_STYLE_PLAIN`
Glows when tapped. The default item style.

`pyto_ui.BUTTON_ITEM_STYLE_DONE`
The style for a done button—for example, a button that completes some task and returns to the previous view.

System Item

`pyto_ui.SYSTEM_ITEM_ACTION`
The system action button.

`pyto_ui.SYSTEM_ITEM_ADD`
The system plus button containing an icon of a plus sign.

`pyto_ui.SYSTEM_ITEM_BOOKMARKS`
The system bookmarks button.

`pyto_ui.SYSTEM_ITEM_CAMERA`
The system camera button.

`pyto_ui.SYSTEM_ITEM_CANCEL`
The system Cancel button, localized.

`pyto_ui.SYSTEM_ITEM_COMPOSE`
The system compose button.

`pyto_ui.SYSTEM_ITEM_DONE`
The system Done button, localized.

`pyto_ui.SYSTEM_ITEM_EDIT`
The system Edit button, localized.

`pyto_ui.SYSTEM_ITEM_FAST_FORWARD`
The system fast forward button.

`pyto_ui.SYSTEM_ITEM_FLEXIBLE_SPACE`
Blank space to add between other items. The space is distributed equally between the other items. Other item properties are ignored when this value is set.

`pyto_ui.SYSTEM_ITEM_ORGANIZE`
The system organize button.

`pyto_ui.SYSTEM_ITEM_PAUSE`
The system pause button.

`pyto_ui.SYSTEM_ITEM_PLAY`
The system play button.

`pyto_ui.SYSTEM_ITEM_REDO`
The system redo button.

`pyto_ui.SYSTEM_ITEM_REFRESH`
The system refresh button.

`pyto_ui.SYSTEM_ITEM_REPLY`
The system reply button.

`pyto_ui.SYSTEM_ITEM_REWIND`
The system rewind button.

`pyto_ui.SYSTEM_ITEM_SAVE`
The system Save button, localized.

`pyto_ui.SYSTEM_ITEM_SEARCH`
The system search button.

`pyto_ui.SYSTEM_ITEM_STOP`
The system stop button.

`pyto_ui.SYSTEM_ITEM_TRASH`
The system trash button.

`pyto_ui.SYSTEM_ITEM_UNDO`
The system undo button.

Presentation Mode

`pyto_ui.PRESENTATION_MODE_SHEET`
A presentation style that displays the content centered in the screen.

`pyto_ui.PRESENTATION_MODE_FULLSCREEN`
A presentation style in which the presented view covers the screen.

`pyto_ui.PRESENTATION_MODE_WIDGET`
A presentation mode style which simulates a Today Widget. Should be used in app to preview how a widget will look.

Appearance

`pyto_ui.APPEARANCE_UNSPECIFIED`
An unspecified interface style.

`pyto_ui.APPEARANCE_LIGHT`
The light interface style.

`pyto_ui.APPEARANCE_DARK`
The dark interface style.

4.2 notification_center

Today Widgets

This module is used to manage Today Widget expansion / collapsion.

`notification_center.expandable = False`
If set to True, the widget will be expandable.

`notification_center.maximum_height = 280`
The maximum height of an expandable widget.

4.3 sound

4.3.1 Playing sounds

class `sound.AudioPlayer` (*path: str*)

A wrapper of the `AVAudioPlayer` class of the `AVFoundation` framework. Use this class for playing long sounds with the ability to pause, to stop and to set the time.

current_time

The current time of the sound in seconds.

Return type float

pause ()

Pauses the audio.

play ()

Plays the audio asynchronously.

playing

A boolean indicating whether the sound is playing. (read only)

Return type bool

stop ()

Stops the audio.

volume

The volume of the sound. (From 0 to 1)

Return type float

`sound.play_file` (*path: str*)

Plays a file at given path.

Warning: Only use this function for sounds under 30 seconds. Use `AudioPlayer` for longer sounds.

Parameters `path` – The relative path of the file to play.

4.3.2 Playing system sounds

`sound.play_system_sound` (*id: int*)

Plays a system sound with given ID.

For a list of sounds: github.com/TUNER88/iOSSystemSoundsLibrary.

Parameters `id` – The ID of the system sound to play.

`sound.play_beep` ()

Plays a beep sound.

4.3.3 Frameworks

`sound.AVFoundation`

The `AVFoundation` system framework.

sound.**AudioToolbox**

The AudioToolbox system framework.

4.4 sharing

Sharing Items

This module allows you to share items, to import files and to open URLs.

class sharing.**FilePicker**

A class representing a file picker. Presents an UIDocumentPickerViewController.

A class representing a file picker.

Example:

```
filePicker = sharing.FilePicker()
filePicker.file_types = ["public.data"]
filePicker.allows_multiple_selection = True

def files_picked() -> None:
    files = sharing.picked_files()
    sharing.share_items(files)

filePicker.completion = files_picked
sharing.pick_documents(filePicker)
```

allows_multiple_selection = False

Allow multiple selection or not.

completion = None

The code to execute when files where picked.

file_types = []

Document types that can be opened.

urls = []

Picked URLs.

sharing.**open_url** (*url: str*)

Opens the given URL.

Parameters **url** – URL to open. Can be a String or an Objective-C NSURL.

sharing.**pick_documents** (*filePicker: sharing.FilePicker*)

Pick documents with given parameters as a FilePicker.

Parameters **filePicker** – The parameters of the file picker to be presented.

sharing.**picked_files** () → List[str]

Returns paths of files picked with pickDocumentsWithFilePicker.

sharing.**quick_look** (*path: str*)

Previews given file.

This function doesn't block the current thread. You can call this function multiple times and the file path will be appended to the current Preview controller. Thread safe.

Parameters **path** – Path of the image to preview.

`sharing.share_items (items: object)`

Opens a share sheet with given items.

Parameters `items` – Items to be shared with the sheet.

4.5 mainthread

This module allows you to run code on the main thread easily. This can be used for modifying the UI.

Example:

```
from UIKit import UIScreen
import mainthread

def set_brightness():
    inverted = int(not int(UIScreen.mainScreen.brightness))
    UIScreen.mainScreen.setBrightness(inverted)

mainthread.run_async(set_brightness)
```

`mainthread.run_async (code)`

Runs the given code asynchronously on the main thread.

Parameters `code` – Code to execute in the main thread.

4.6 UIKit

UIKit classes

This module contains all UIKit classes. You should not use this module to present an UI, there is the `pyto_ui` module for that. However, you can safely use classes like `UIDevice`, `UIImage`, `UIFont` etc.

4.7 notifications

Schedule notifications

Use the `notifications` to schedule notifications that can be delivered even if Pyto isn't opened.

class `notifications.Notification`

A class representing a notification.

actions = None

Additional actions on the notification.

A dictionary with the name of the action and the URL to open.

message = None

The body of the notification.

url = None

The URL to open when the notification is opened.

`notifications.UNUserNotificationCenter = None`

The 'UNUserNotificationCenter' class from the `UserNotifications` framework.

`notifications.cancel_all()`
Cancels all pending notifications.

`notifications.cancel_notification(notification: notifications.Notification)`
Cancels a pending notification.

Parameters `notification` – The `Notification` object returned from `get_pending_notifications()`.

`notifications.get_pending_notifications()` → List[notifications.Notification]
Returns a list of pending notifications. Notifications cannot be modified after being scheduled.

Return type List[`Notification`]

`notifications.remove_delivered_notifications()`
Removes all delivered notifications from the Notification Center.

`notifications.schedule_notification(notification: notifications.Notification, delay: float, repeat: bool)`
Schedules a notification.

Parameters

- **Notification** – The `Notification` object representing the notification content.
- **delay** – The time interval in seconds until the notification is delivered.
- **repeat** – A boolean indicating whether the notification delivery should be repeated indefinitely.

4.8 music

Access to the media library

Use the `music` module to access the Music library of the user and to play items.

Example:

```
import music
music.set_queue_with_items(music.pick_music())
music.play()
```

4.8.1 Playing music

`music.set_queue_with_items(items: List[music.MediaItem])`
Sets a music player's playback queue using a media item collection.

Parameters `items` – An array of `MediaItem` that you want as the playback queue.

`music.set_queue_with_store_ids(ids: List[str])`
Sets a music player's playback queue using with media items identified by the store identifiers.

Parameters `ids` – An array of store identifiers associated with the media items to be added to the queue.

`music.play()`
Initiates playback of the current item.

`music.stop()`
Ends playback of the current item.

`music.restart()`

Restarts playback at the beginning of the currently playing media item.

`music.next()`

Starts playback of the next media item in the playback queue; or, the music player is not playing, designates the next media item as the next to be played.

`music.previous()`

Starts playback of the previous media item in the playback queue; or, the music player is not playing, designates the previous media item as the next to be played.

`music.set_shuffle_mode(mode: int)`

Sets the shuffle mode of the music player.

Parameters `mode` – The shuffle mode to set. See [Shuffle Mode](#) for possible values.

`music.set_repeat_mode(mode: int)`

Sets the repeat mode of the music player.

Parameters `mode` – The repeat mode to set. See [Repeat Mode](#) for possible values.

`music.playback_state()` → int

The current playback state of the music player.

Return type [Playback State](#)

4.8.2 Now playing item

`music.now_playing_item()` → [music.MediaItem](#)

The currently-playing media item, or the media item, within a queue, that you have designated to begin playback with.

Return type [MediaItem](#)

`music.index_of_now_playing_item()` → int

The index of the now playing item in the current playback queue.

Return type int

4.8.3 Picking music

`music.pick_music()` → List[[music.MediaItem](#)]

Picks music with a UI.

Returns a list of selected items.

Return type List[[MediaItem](#)]

4.8.4 Classes

class `music.MediaItem` (*item: [MPMediaItem](#)*)

A collection of properties that represents a single item contained in the media library.

album_artist

The primary performing artist for an album as a whole.

Return type str

album_artist_persistent_id

The persistent identifier for the primary performing artist for an album as a whole.

Return type str

album_persistent_id

The persistent identifier for an album.

Return type str

album_title

The title of an album, such as “Live On Mars”, as opposed to the title of an individual song on the album, such as “Crater Dance (radio edit)”.

Return type str

album_track_count

The number of tracks in the album that contains the media item.

Return type int

artist

The performing artist(s) for a media item—which may vary from the primary artist for the album that a media item belongs to.

Return type str

artist_persistent_id

The persistent identifier for an artist.

Return type int

artwork

The artwork image for the media item.

Return type PIL.Image

beats_per_minute

The number of musical beats per minute for the media item.

Return type int

bookmark_time

The user’s place in the media item the most recent time it was played.

Return type float

comments

Textual information about the media item.

Return type str

composer

The musical composer for the media item.

Return type str

composer_persistent_id

The persistent identifier for a composer.

Return type str

date_added

The date the item was added to the library.

Return type str

disc_count

The number of discs in the album that contains the media item.

Return type int

disc_number

The disc number of the media item, for a media item that is part of a multi-disc album.

Return type int

genre

The musical or film genre of the media item.

Return type str

genre_persistent_id

The persistent identifier for a genre.

Return type str

has_protected_asset

A Boolean indicating whether the media item has a protected asset.

Return type bool

is_cloud_item

A Boolean value indicating whether the media item is an iCloud Music Library item.

Return type bool

is_compilation

A Boolean value indicating whether the media item is part of a compilation.

Return type bool

is_explicit_item

A Boolean value that indicates whether the item has explicit (adult) lyrics or language.

Return type bool

last_played_date

The date a media item was last played.

Return type str

lyrics

The lyrics for the media item.

Return type str

persistent_id

The persistent identifier for the media item.

Return type str

play_count

The number of times the user has played the media item.

Return type int

playback_duration

The playback duration of the media item.

Return type float

playback_store_id

The non-library identifier for a media item.

Return type str

podcast_persistent_id

The persistent identifier for an audio podcast.

Return type str

podcast_title

The title of a podcast, such as “This Martian Drudgery”, as opposed to the title of an individual episode of a podcast such as “Episode 12: Another Cold Day At The Pole”.

Return type str

rating

The user-specified rating of the object in the range [0..5], where a value of 5 indicates the most favorable rating.

Return type int

release_date

The date on which the media item was first publicly released.

Return type str

skip_count

The number of times the user has skipped playing the item.

Return type int

title

The title (or name) of the media item.

Return type str

user_grouping

Grouping information for the media item.

Return type str

class `music.MediaQuery` (*query: MPMediaQuery*)

A query that specifies a set of media items from the device’s media library by way of a filter and a grouping type.

classmethod `albums` () → `music.MediaQuery`

Creates a media query that matches music items and that groups and sorts collections by album name.

Return type *MediaQuery*

classmethod `artists` () → `music.MediaQuery`

Creates a media query that matches music items and that groups and sorts collections by artist name.

Return type *MediaQuery*

classmethod `audiobook` () → `music.MediaQuery`

Creates a media query that matches audio book items and that groups and sorts collections by audio book name.

Return type *MediaQuery*

classmethod `compilations` () → `music.MediaQuery`

Creates a media query that matches compilation items and that groups and sorts collections by album name.

Return type *MediaQuery*

classmethod composers () → music.MediaQuery
Creates a media query that matches all media items and that groups and sorts collections by composer name.

Return type *MediaQuery*

classmethod genres () → music.MediaQuery
Creates a media query that matches all media items and that groups and sorts collections by genre name.

Return type *MediaQuery*

items

An array of media items that match the media query's predicate.

Return type List[*MediaItem*]

classmethod playlists () → music.MediaQuery
Creates a media query that matches the entire library and that groups and sorts collections by playlist name.

Return type *MediaQuery*

classmethod podcasts () → music.MediaQuery
Creates a media query that matches podcast items and that groups and sorts collections by podcast name.

Return type *MediaQuery*

classmethod songs () → music.MediaQuery
Creates a media query that matches music items and that groups and sorts collections by song name.

Return type *MediaQuery*

4.8.5 Objective-C classes

`music.MPMusicPlayerController = 'MPMusicPlayerController'`

The 'MPMusicPlayerController' class from the 'MediaPlayer' framework.

`music.MPMediaItem = 'MPMediaItem'`

The 'MPMediaItem' class from the 'MediaPlayer' framework.

`music.MPMediaItemCollection = 'MPMediaItemCollection'`

The 'MPMediaItemCollection' class from the 'MediaPlayer' framework.

`music.MPMediaQuery = 'MPMediaQuery'`

The 'MPMusicMPMediaQueryPlayerController' class from the 'MediaPlayer' framework.

4.8.6 Constants

Repeat mode

`music.REPEAT_MODE_DEFAULT = 0`

The user's preferred repeat mode.

`music.REPEAT_MODE_NONE = 1`

The music player will not repeat the current song or playlist.

`music.REPEAT_MODE_ONE = 2`

The music player will repeat the current song.

`music.REPEAT_MODE_ALL = 3`

The music player will repeat the current playlist.

Shuffle mode

`music.SHUFFLE_MODE_DEFAULT = 0`

The user's preferred shuffle mode.

`music.SHUFFLE_MODE_OFF = 1`

The playlist is not shuffled.

`music.SHUFFLE_MODE_SONGS = 2`

The playlist is shuffled by song.

`music.SHUFFLE_MODE_ALBUMS = 3`

The playlist is shuffled by album.

Playback state

`music.PLAYBACK_STATE_STOPPED = 0`

The music player is stopped.

`music.PLAYBACK_STATE_PLAYING = 1`

The music player is playing.

`music.PLAYBACK_STATE_PAUSED = 2`

The music player is paused.

`music.PLAYBACK_STATE_INTERRUPTED = 3`

The music player has been interrupted, such as by an incoming phone call.

`music.PLAYBACK_STATE_SEEKING_FORWARD = 4`

The music player is seeking forward.

`music.PLAYBACK_STATE_SEEKING_BACKWARD = 5`

The music player is seeking backward.

4.9 photos

Accessing photos and the camera

Use this library to pick and take photos.

`photos.pick_photo()` → `PIL.Image.Image`

Pick a photo from the photos library. Returns the picked image as a PIL Image.

Return type `PIL.Image.Image`

`photos.save_image(image: PIL.Image.Image)`

Saves the given image to the photos library.

Parameters `image` – A PIL image to save.

`photos.take_photo()` → `PIL.Image.Image`

Take a photo from the camera. Returns the taken image as a PIL Image.

Return type `PIL.Image.Image`

4.10 location

Accessing location

This module gives access to the devices's location.

`location.LOCATION_ACCURACY_BEST = -1`

The best level of accuracy available.

`location.LOCATION_ACCURACY_BEST_FOR_NAVIGATION = -2`

The highest possible accuracy that uses additional sensor data to facilitate navigation apps.

`location.LOCATION_ACCURACY_HUNDRED_METERS = 100`

Accurate to within one hundred meters.

`location.LOCATION_ACCURACY_KILOMETER = 1000`

Accurate to the nearest kilometer.

`location.LOCATION_ACCURACY_NEAREST_TEN_METERS = 10`

Accurate to within ten meters of the desired target.

`location.LOCATION_ACCURACY_THREE_KILOMETERS = 3000`

Accurate to the nearest three kilometers.

class `location.Location` (*longitude, latitude, altitude*)

A tuple containing data about longitude, latitude and altitude.

altitude

Alias for field number 2

latitude

Alias for field number 1

longitude

Alias for field number 0

`location.accuracy = -1`

The number of meters from the original geographic coordinate that could yield the user's actual location.

`location.get_location()` → `location.Location`

Returns a tuple with current longitude, latitude and altitude.

Return type *Location*

`location.start_updating()`

Starts receiving location updates. Call this before calling `get_location()`.

`location.stop_updating()`

Stops receiving location updates.

4.11 motion

Motion sensors

The `motion` module gives access to the device's accelerometer, gyroscope and magnetometer data.

4.11.1 Functions

`motion.start Updating()`

Starts receiving information from the sensors.

`motion.stop Updating()`

Stops receiving information from the sensors

`motion.get_acceleration()` → `motion.Acceleration`

Returns a tuple with information about acceleration (x, y, z).

Return type *Acceleration*

`motion.get_attitude()` → `motion.Attitude`

Returns a tuple with information about the attitude (roll, pitch, yaw).

Return type *Attitude*

`motion.get_gravity()` → `motion.Gravity`

Returns a tuple with information about gravity (x, y, z).

Return type *Gravity*

`motion.get_magnetic_field()` → `motion.MagneticField`

Returns a tuple with information about the magnetic field (x, y, z).

Return type *MagneticField*

`motion.get_rotation()` → `motion.Rotation`

Returns a tuple with information about rotation (x, y, z).

Return type *Rotation*

4.11.2 Data Types

class `motion.Acceleration(x, y, z)`

A tuple containing data about acceleration (x, y, z).

x

Alias for field number 0

y

Alias for field number 1

z

Alias for field number 2

class `motion.Attitude(roll, pitch, yaw)`

A tuple containing data about attitude (roll, pitch, yaw).

pitch

Alias for field number 1

roll

Alias for field number 0

yaw

Alias for field number 2

class `motion.Gravity(x, y, z)`

A tuple containing data about gravity (x, y, z).

x Alias for field number 0

y Alias for field number 1

z Alias for field number 2

class `motion.MagneticField(x, y, z)`
A tuple containing data about magnetic field (x, y, z).

x Alias for field number 0

y Alias for field number 1

z Alias for field number 2

class `motion.Rotation(x, y, z)`
A tuple containing data about rotation (x, y, z).

x Alias for field number 0

y Alias for field number 1

z Alias for field number 2

4.12 multipeer

Peer to peer wireless connection

Use this module to trade data with other devices running Pyto. Works without WiFi.

`multipeer.connect()`
Starts connecting to other devices.

`multipeer.disconnect()`
Disconnects from all connected devices.

`multipeer.get_data()` → str
Returns available data. Returns once per available data.

Return type str

`multipeer.send(data: str)`
Sends the given string to other connected devices.

Parameters `data` – The string to send.

4.13 xcallback

Opening x-callback URLs

This module is used to interact with other apps with x-callback URLs.

`xcallback.open_url(url: str) → str`

Opens the given x-callback URL. The function will return only if the request was successfully completed.

Raises: `RuntimeError` if there was an error and `SystemExit` if the request was cancelled. Returns: The result sent by the opened app.

Parameters `url` – The URL to open.

Return type `str`

4.13.1 Example with Shortcuts

```
"""
Opens a Shortcut and retrieves the result.
"""

import xcallback
from urllib.parse import quote

shortcut_name = input("The name of the shortcut to open: ")
shortcut_input = input("What would you like to send to the Shortcut? ")

# https://support.apple.com/guide/shortcuts/apdcd7f20a6f/ios
url = f"shortcuts://x-callback-url/run-shortcut?name={quote(shortcut_name)}&
↳input=text&text={quote(shortcut_input)}"

try:
    res = xcallback.open_url(url) # If succeeded, returns the result
    print("Result:\n"+res)
except RuntimeError as e:
    print("Error: "+str(e)) # If failed, raises ``RuntimeError``
except SystemExit:
    print("Cancelled") # If cancelled, raises ``SystemExit``
```

4.14 userkeys

Save values on disk

This module makes possible to save values on disk. Values are shared between the Today Widget and the main app. Values are stored in a JSON dictionary, so it's not possible to save every type of data.

`userkeys.delete(key: str)`

Deletes the value stored with the given key.

Parameters `key` – The key identifying the value to delete.

`userkeys.get(key: str)`

Returns the value stored with the given key.

Parameters `key` – The key identifying the value.

`userkeys.set(value, key: str)`

Adds the given value to the database with the given key.

Parameters

- `value` – A JSON compatible value.

- **key** – The key identifying the value.

4.15 pasteboard

Pasteboard Access

This module gives access to the pasteboard.

`pasteboard.image()` → None

Returns the image contained in the pasteboard as an Objective-C UIImage.

`pasteboard.images()` → List[None]

Returns all images contained in the pasteboard as Objective-C UIImage s.

`pasteboard.set_image(image: None)`

Copy the given image to the pasteboard.

Parameters `image` – The image to copy.

`pasteboard.set_images(array: List[None])`

Copy the given images to the pasteboard.

Parameters `array` – A list of images to copy.

`pasteboard.set_string(text: str)`

Copy the given text to the pasteboard.

Parameters `text` – The text to copy.

`pasteboard.set_strings(array: List[str])`

Copy the given strings to the pasteboard.

Parameters `array` – A list of strings to copy.

`pasteboard.set_url(url: None)`

Copy the given URL to the pasteboard.

Parameters `url` – The Objective-C NSURL to copy.

`pasteboard.set_urls(array: List[None])`

Copy the given URLs to the pasteboard.

Parameters `array` – A list of Objective-C NSURL s to copy.

`pasteboard.string()` → str

Returns the text contained in the pasteboard.

`pasteboard.strings()` → List[str]

Returns all strings contained in the pasteboard.

`pasteboard.url()` → None

Returns the URL contained in the pasteboard as an Objective-C NSURL.

`pasteboard.urls()` → List[None]

Returns all URLs contained in the pasteboard as Objective-C NSURL s.

I

location, 128

M

mainthread, 120

motion, 128

multipeer, 130

music, 121

N

notification_center, 117

notifications, 120

P

pasteboard, 132

photos, 127

pyto_ui, 85

S

sharing, 119

U

UIKit, 120

userkeys, 131

X

xcallback, 130

A

- Acceleration (*class in motion*), 129
- accessory_type (*pyto_ui.TableViewCell attribute*), 92
- ACCESSORY_TYPE_CHECKMARK (*in module pyto_ui*), 115
- ACCESSORY_TYPE_DETAIL_BUTTON (*in module pyto_ui*), 115
- ACCESSORY_TYPE_DETAIL_DISCLOSURE_BUTTON (*in module pyto_ui*), 115
- ACCESSORY_TYPE_DISCLOSURE_INDICATOR (*in module pyto_ui*), 115
- ACCESSORY_TYPE_NONE (*in module pyto_ui*), 115
- accuracy (*in module location*), 128
- action (*pyto_ui.ButtonItem attribute*), 104
- action (*pyto_ui.Control attribute*), 96
- action (*pyto_ui.GestureRecognizer attribute*), 102
- actions (*notifications.Notification attribute*), 120
- add_action() (*pyto_ui.Alert method*), 105
- add_cancel_action() (*pyto_ui.Alert method*), 105
- add_destructive_action() (*pyto_ui.Alert method*), 105
- add_gesture_recognizer() (*pyto_ui.View method*), 87
- add_subview() (*pyto_ui.View method*), 87
- adjusts_font_size_to_fit_width (*pyto_ui.Label attribute*), 91
- album_artist (*music.MediaItem attribute*), 122
- album_artist_persistent_id (*music.MediaItem attribute*), 122
- album_persistent_id (*music.MediaItem attribute*), 123
- album_title (*music.MediaItem attribute*), 123
- album_track_count (*music.MediaItem attribute*), 123
- albums() (*music.MediaQuery class method*), 125
- Alert (*class in pyto_ui*), 104
- allowed_touch_types (*pyto_ui.GestureRecognizer attribute*), 102
- allowedTouchTypes (*pyto_ui.GestureRecognizer attribute*), 102
- allows_multiple_selection (*sharing.FilePicker attribute*), 119
- alpha (*pyto_ui.View attribute*), 87
- alpha() (*pyto_ui.Color method*), 100
- altitude (*location.Location attribute*), 128
- appearance (*pyto_ui.View attribute*), 87
- APPEARANCE_DARK (*in module pyto_ui*), 117
- APPEARANCE_LIGHT (*in module pyto_ui*), 117
- APPEARANCE_UNSPECIFIED (*in module pyto_ui*), 117
- artist (*music.MediaItem attribute*), 123
- artist_persistent_id (*music.MediaItem attribute*), 123
- artists() (*music.MediaQuery class method*), 125
- artwork (*music.MediaItem attribute*), 123
- Attitude (*class in motion*), 129
- audiobook() (*music.MediaQuery class method*), 125
- AudioPlayer (*class in sound*), 118
- AudioToolbox (*in module sound*), 118
- AUTO_CAPITALIZE_ALL (*in module pyto_ui*), 110
- AUTO_CAPITALIZE_NONE (*in module pyto_ui*), 110
- AUTO_CAPITALIZE_SENTENCES (*in module pyto_ui*), 110
- AUTO_CAPITALIZE_WORDS (*in module pyto_ui*), 110
- autocapitalization_type (*pyto_ui.TextField attribute*), 98
- autocapitalization_type (*pyto_ui.TextView attribute*), 93
- autocorrection (*pyto_ui.TextField attribute*), 98
- autocorrection (*pyto_ui.TextView attribute*), 93
- autorotate() (*in module cv2*), 5
- AVFoundation (*in module sound*), 118

B

- background_color (*pyto_ui.View attribute*), 87
- beats_per_minute (*music.MediaItem attribute*), 123
- become_first_responder() (*pyto_ui.View method*), 87

- blue() (*pyto_ui.Color* method), 100
 bold_system_font_of_size() (*pyto_ui.Font* class method), 101
 bookmark_time (*music.MediaItem* attribute), 123
 border_color (*pyto_ui.View* attribute), 87
 border_width (*pyto_ui.View* attribute), 88
 Button (class in *pyto_ui*), 98
 BUTTON_ITEM_STYLE_DONE (in module *pyto_ui*), 116
 BUTTON_ITEM_STYLE_PLAIN (in module *pyto_ui*), 116
 button_items (*pyto_ui.View* attribute), 88
 BUTTON_TYPE_CONTACT_ADD (in module *pyto_ui*), 111
 BUTTON_TYPE_CUSTOM (in module *pyto_ui*), 111
 BUTTON_TYPE_DETAIL_DISCLOSURE (in module *pyto_ui*), 111
 BUTTON_TYPE_INFO_DARK (in module *pyto_ui*), 111
 BUTTON_TYPE_INFO_LIGHT (in module *pyto_ui*), 111
 BUTTON_TYPE_SYSTEM (in module *pyto_ui*), 111
 ButtonItem (class in *pyto_ui*), 104
- ## C
- can_go_back (*pyto_ui.WebView* attribute), 95
 can_go_forward (*pyto_ui.WebView* attribute), 95
 cancel_all() (in module *notifications*), 120
 cancel_notification() (in module *notifications*), 121
 cancels_touches_in_view (*pyto_ui.GestureRecognizer* attribute), 102
 cells (*pyto_ui.tableViewSection* attribute), 103
 center (*pyto_ui.View* attribute), 88
 center_x (*pyto_ui.View* attribute), 88
 center_y (*pyto_ui.View* attribute), 88
 clips_to_bounds (*pyto_ui.View* attribute), 88
 close() (*pyto_ui.View* method), 88
 Color (class in *pyto_ui*), 99
 COLOR_BLACK (in module *pyto_ui*), 107
 COLOR_BLUE (in module *pyto_ui*), 108
 COLOR_BROWN (in module *pyto_ui*), 108
 COLOR_CLEAR (in module *pyto_ui*), 107
 COLOR_CYAN (in module *pyto_ui*), 108
 COLOR_DARK_GRAY (in module *pyto_ui*), 108
 COLOR_DARK_TEXT (in module *pyto_ui*), 107
 COLOR_GRAY (in module *pyto_ui*), 108
 COLOR_GREEN (in module *pyto_ui*), 108
 COLOR_LABEL (in module *pyto_ui*), 106
 COLOR_LIGHT_GRAY (in module *pyto_ui*), 108
 COLOR_LIGHT_TEXT (in module *pyto_ui*), 107
 COLOR_LINK (in module *pyto_ui*), 106
 COLOR_MAGENTA (in module *pyto_ui*), 108
 COLOR_OPAQUE_SEPARATOR (in module *pyto_ui*), 106
 COLOR_ORANGE (in module *pyto_ui*), 108
 COLOR_PLACEHOLDER_TEXT (in module *pyto_ui*), 106
 COLOR_PURPLE (in module *pyto_ui*), 108
 COLOR_QUATERNARY_LABEL (in module *pyto_ui*), 106
 COLOR_QUATERNARY_SYSTEM_FILL (in module *pyto_ui*), 106
 COLOR_RED (in module *pyto_ui*), 108
 COLOR_SECONDARY_GROUPED_BACKGROUND (in module *pyto_ui*), 106
 COLOR_SECONDARY_LABEL (in module *pyto_ui*), 106
 COLOR_SECONDARY_SYSTEM_BACKGROUND (in module *pyto_ui*), 106
 COLOR_SECONDARY_SYSTEM_FILL (in module *pyto_ui*), 106
 COLOR_SEPARATOR (in module *pyto_ui*), 106
 COLOR_SYSTEM_BACKGROUND (in module *pyto_ui*), 106
 COLOR_SYSTEM_BLUE (in module *pyto_ui*), 107
 COLOR_SYSTEM_FILL (in module *pyto_ui*), 106
 COLOR_SYSTEM_GRAY (in module *pyto_ui*), 107
 COLOR_SYSTEM_GRAY2 (in module *pyto_ui*), 107
 COLOR_SYSTEM_GRAY3 (in module *pyto_ui*), 107
 COLOR_SYSTEM_GRAY4 (in module *pyto_ui*), 107
 COLOR_SYSTEM_GRAY5 (in module *pyto_ui*), 107
 COLOR_SYSTEM_GRAY6 (in module *pyto_ui*), 107
 COLOR_SYSTEM_GREEN (in module *pyto_ui*), 107
 COLOR_SYSTEM_GROUPED_BACKGROUND (in module *pyto_ui*), 106
 COLOR_SYSTEM_INDIGO (in module *pyto_ui*), 107
 COLOR_SYSTEM_ORANGE (in module *pyto_ui*), 107
 COLOR_SYSTEM_PINK (in module *pyto_ui*), 107
 COLOR_SYSTEM_PURPLE (in module *pyto_ui*), 107
 COLOR_SYSTEM_RED (in module *pyto_ui*), 107
 COLOR_SYSTEM_TEAL (in module *pyto_ui*), 107
 COLOR_SYSTEM_YELLOW (in module *pyto_ui*), 107
 COLOR_TERTIARY_GROUPED_BACKGROUND (in module *pyto_ui*), 106
 COLOR_TERTIARY_LABEL (in module *pyto_ui*), 106
 COLOR_TERTIARY_SYSTEM_BACKGROUND (in module *pyto_ui*), 106
 COLOR_TERTIARY_SYSTEM_FILL (in module *pyto_ui*), 106
 COLOR_WHITE (in module *pyto_ui*), 108
 COLOR_YELLOW (in module *pyto_ui*), 108
 comments (*music.MediaItem* attribute), 123
 compilations() (*music.MediaQuery* class method), 125
 completion (*sharing.FilePicker* attribute), 119
 composer (*music.MediaItem* attribute), 123
 composer_persistent_id (*music.MediaItem* attribute), 123
 composers() (*music.MediaQuery* class method), 125

connect () (in module *multipeer*), 130
 content_mode (*pyto_ui.View* attribute), 88
 CONTENT_MODE_BOTTOM (in module *pyto_ui*), 113
 CONTENT_MODE_BOTTOM_LEFT (in module *pyto_ui*), 113
 CONTENT_MODE_BOTTOM_RIGHT (in module *pyto_ui*), 113
 CONTENT_MODE_CENTER (in module *pyto_ui*), 113
 CONTENT_MODE_LEFT (in module *pyto_ui*), 113
 CONTENT_MODE_REDRAW (in module *pyto_ui*), 113
 CONTENT_MODE_RIGHT (in module *pyto_ui*), 113
 CONTENT_MODE_SCALE_ASPECT_FILL (in module *pyto_ui*), 113
 CONTENT_MODE_SCALE_ASPECT_FIT (in module *pyto_ui*), 112
 CONTENT_MODE_SCALE_TO_FILL (in module *pyto_ui*), 112
 CONTENT_MODE_TOP (in module *pyto_ui*), 113
 CONTENT_MODE_TOP_LEFT (in module *pyto_ui*), 113
 CONTENT_MODE_TOP_RIGHT (in module *pyto_ui*), 113
 content_view (*pyto_ui.TableViewController* attribute), 92
 Control (class in *pyto_ui*), 96
 corner_radius (*pyto_ui.View* attribute), 88
 current_time (*sound.AudioPlayer* attribute), 118

D

date_added (*music.MediaItem* attribute), 123
 delays_touches_began (*pyto_ui.GestureRecognizer* attribute), 103
 delays_touches_ended (*pyto_ui.GestureRecognizer* attribute), 103
 delete () (in module *userkeys*), 131
 deselect_row () (*pyto_ui.TableView* method), 93
 detail_text_label (*pyto_ui.TableViewController* attribute), 92
 did_begin_editing (*pyto_ui.TextField* attribute), 98
 did_begin_editing (*pyto_ui.TextView* attribute), 93
 did_change (*pyto_ui.TextView* attribute), 93
 did_delete_cell (*pyto_ui.TableViewSection* attribute), 103
 did_end_editing (*pyto_ui.TextField* attribute), 98
 did_end_editing (*pyto_ui.TextView* attribute), 94
 did_fail_loading (*pyto_ui.WebView* attribute), 95
 did_finish_loading (*pyto_ui.WebView* attribute), 95
 did_move_cell (*pyto_ui.TableViewSection* attribute), 103
 did_select_cell (*pyto_ui.TableViewSection* attribute), 104
 did_start_loading (*pyto_ui.WebView* attribute), 95
 did_tap_cell_accessory_button (*pyto_ui.TableViewSection* attribute), 104

disc_count (*music.MediaItem* attribute), 123
 disc_number (*music.MediaItem* attribute), 124
 disconnect () (in module *multipeer*), 130
 dynamic () (*pyto_ui.Color* class method), 100

E

edit_button_item (*pyto_ui.TableView* attribute), 93
 editable (*pyto_ui.TextView* attribute), 94
 enabled (*pyto_ui.ButtonItem* attribute), 104
 enabled (*pyto_ui.Control* attribute), 96
 enabled (*pyto_ui.GestureRecognizer* attribute), 103
 evaluate_js () (*pyto_ui.WebView* method), 95
 expandable (in module *notification_center*), 117

F

file_types (*sharing.FilePicker* attribute), 119
 FilePicker (class in *sharing*), 119
 first_responder (*pyto_ui.View* attribute), 88
 flex (*pyto_ui.View* attribute), 88
 FLEXIBLE_BOTTOM_MARGIN (in module *pyto_ui*), 112
 FLEXIBLE_HEIGHT (in module *pyto_ui*), 112
 FLEXIBLE_LEFT_MARGIN (in module *pyto_ui*), 112
 FLEXIBLE_RIGHT_MARGIN (in module *pyto_ui*), 112
 FLEXIBLE_TOP_MARGIN (in module *pyto_ui*), 112
 FLEXIBLE_WIDTH (in module *pyto_ui*), 112
 Font (class in *pyto_ui*), 101
 font (*pyto_ui.Button* attribute), 98
 font (*pyto_ui.Label* attribute), 91
 font (*pyto_ui.TextField* attribute), 98
 font (*pyto_ui.TextView* attribute), 94
 FONT_BUTTON_SIZE (in module *pyto_ui*), 111
 font_family_names () (in module *pyto_ui*), 105
 FONT_LABEL_SIZE (in module *pyto_ui*), 111
 font_names_for_family_name () (*pyto_ui.Font* class method), 101
 FONT_SMALL_SYSTEM_SIZE (in module *pyto_ui*), 111
 FONT_SYSTEM_SIZE (in module *pyto_ui*), 111
 FONT_TEXT_STYLE_BODY (in module *pyto_ui*), 110
 FONT_TEXT_STYLE_CALLOUT (in module *pyto_ui*), 110
 FONT_TEXT_STYLE_CAPTION_1 (in module *pyto_ui*), 110
 FONT_TEXT_STYLE_CAPTION_2 (in module *pyto_ui*), 110
 FONT_TEXT_STYLE_FOOTNOTE (in module *pyto_ui*), 110
 FONT_TEXT_STYLE_HEADLINE (in module *pyto_ui*), 110
 FONT_TEXT_STYLE_LARGE_TITLE (in module *pyto_ui*), 110
 FONT_TEXT_STYLE_SUBHEADLINE (in module *pyto_ui*), 110

- FONT_TEXT_STYLE_TITLE_1 (in module *pyto_ui*), 110
- FONT_TEXT_STYLE_TITLE_2 (in module *pyto_ui*), 110
- FONT_TEXT_STYLE_TITLE_3 (in module *pyto_ui*), 110
- font_with_style() (*pyto_ui.Font* class method), 101
- frame (*pyto_ui.View* attribute), 89
- ## G
- genre (*music.MediaItem* attribute), 124
- genre_persistent_id (*music.MediaItem* attribute), 124
- genres() (*music.MediaQuery* class method), 126
- gesture_recognizers (*pyto_ui.View* attribute), 89
- GESTURE_STATE_BEGAN (in module *pyto_ui*), 111
- GESTURE_STATE_CANCELLED (in module *pyto_ui*), 112
- GESTURE_STATE_CHANGED (in module *pyto_ui*), 112
- GESTURE_STATE_ENDED (in module *pyto_ui*), 112
- GESTURE_STATE_POSSIBLE (in module *pyto_ui*), 111
- GESTURE_STATE_RECOGNIZED (in module *pyto_ui*), 112
- GESTURE_TYPE_LONG_PRESS (in module *pyto_ui*), 111
- GESTURE_TYPE_PAN (in module *pyto_ui*), 111
- GESTURE_TYPE_TAP (in module *pyto_ui*), 111
- GestureRecognizer (class in *pyto_ui*), 101
- get() (in module *userkeys*), 131
- get_acceleration() (in module *motion*), 129
- get_attitude() (in module *motion*), 129
- get_data() (in module *multipeer*), 130
- get_gravity() (in module *motion*), 129
- get_location() (in module *location*), 128
- get_magnetic_field() (in module *motion*), 129
- get_pending_notifications() (in module *notifications*), 121
- get_rotation() (in module *motion*), 129
- go_back() (*pyto_ui.WebView* method), 95
- go_forward() (*pyto_ui.WebView* method), 95
- Gravity (class in *motion*), 129
- green() (*pyto_ui.Color* method), 100
- ## H
- has_protected_asset (*music.MediaItem* attribute), 124
- height (*pyto_ui.View* attribute), 89
- hidden (*pyto_ui.View* attribute), 89
- horizontal_alignment (*pyto_ui.Control* attribute), 96
- HORIZONTAL_ALIGNMENT_CENTER (in module *pyto_ui*), 113
- HORIZONTAL_ALIGNMENT_FILL (in module *pyto_ui*), 113
- HORIZONTAL_ALIGNMENT_LEADING (in module *pyto_ui*), 113
- HORIZONTAL_ALIGNMENT_LEFT (in module *pyto_ui*), 113
- HORIZONTAL_ALIGNMENT_RIGHT (in module *pyto_ui*), 113
- HORIZONTAL_ALIGNMENT_TRAILING (in module *pyto_ui*), 113
- ## I
- image (*pyto_ui.Button* attribute), 98
- image (*pyto_ui.ButtonItem* attribute), 104
- image (*pyto_ui.ImageView* attribute), 91
- image() (in module *pasteboard*), 132
- image_view (*pyto_ui.TableViewController* attribute), 92
- image_with_system_name() (in module *pyto_ui*), 105
- images() (in module *pasteboard*), 132
- ImageView (class in *pyto_ui*), 91
- index_of_now_playing_item() (in module *music*), 122
- insert_subview() (*pyto_ui.View* method), 89
- insert_subview_above() (*pyto_ui.View* method), 89
- insert_subview_bellow() (*pyto_ui.View* method), 89
- is_cloud_item (*music.MediaItem* attribute), 124
- is_compilation (*music.MediaItem* attribute), 124
- is_explicit_item (*music.MediaItem* attribute), 124
- is_loading (*pyto_ui.WebView* attribute), 95
- italic_system_font_of_size() (*pyto_ui.Font* class method), 101
- items (*music.MediaQuery* attribute), 126
- ## K
- keyboard_appearance (*pyto_ui.TextField* attribute), 98
- keyboard_appearance (*pyto_ui.TextView* attribute), 94
- KEYBOARD_APPEARANCE_DARK (in module *pyto_ui*), 108
- KEYBOARD_APPEARANCE_DEFAULT (in module *pyto_ui*), 108
- KEYBOARD_APPEARANCE_LIGHT (in module *pyto_ui*), 108
- keyboard_type (*pyto_ui.TextField* attribute), 99
- keyboard_type (*pyto_ui.TextView* attribute), 94
- KEYBOARD_TYPE_ASCII_CAPABLE (in module *pyto_ui*), 108
- KEYBOARD_TYPE_ASCII_CAPABLE_NUMBER_PAD (in module *pyto_ui*), 108

- KEYBOARD_TYPE_DECIMAL_PAD (in module *pyto_ui*), 109
- KEYBOARD_TYPE_DEFAULT (in module *pyto_ui*), 108
- KEYBOARD_TYPE_EMAIL_ADDRESS (in module *pyto_ui*), 109
- KEYBOARD_TYPE_NAME_PHONE_PAD (in module *pyto_ui*), 109
- KEYBOARD_TYPE_NUMBER_PAD (in module *pyto_ui*), 109
- KEYBOARD_TYPE_NUMBERS_AND_PUNCTUATION (in module *pyto_ui*), 109
- KEYBOARD_TYPE_PHONE_PAD (in module *pyto_ui*), 109
- KEYBOARD_TYPE_TWITTER (in module *pyto_ui*), 109
- KEYBOARD_TYPE_URL (in module *pyto_ui*), 109
- KEYBOARD_TYPE_WEB_SEARCH (in module *pyto_ui*), 109
- ## L
- Label (class in *pyto_ui*), 91
- last_played_date (*music.MediaType* attribute), 124
- latitude (*location.Location* attribute), 128
- layout (*pyto_ui.View* attribute), 89
- line_break_mode (*pyto_ui.Label* attribute), 92
- LINE_BREAK_MODE_BY_CHAR_WRAPPING (in module *pyto_ui*), 114
- LINE_BREAK_MODE_BY_CLIPPING (in module *pyto_ui*), 114
- LINE_BREAK_MODE_BY_TRUNCATING_HEAD (in module *pyto_ui*), 114
- LINE_BREAK_MODE_BY_TRUNCATING_MIDDLE (in module *pyto_ui*), 114
- LINE_BREAK_MODE_BY_TRUNCATING_TAIL (in module *pyto_ui*), 114
- LINE_BREAK_MODE_BY_WORD_WRAPPING (in module *pyto_ui*), 114
- load_from_url () (*pyto_ui.ImageView* method), 91
- load_html () (*pyto_ui.Label* method), 92
- load_html () (*pyto_ui.TextView* method), 94
- load_html () (*pyto_ui.WebView* method), 96
- load_url () (*pyto_ui.WebView* method), 96
- Location (class in *location*), 128
- location (module), 128
- location (*pyto_ui.GestureRecognizer* attribute), 103
- LOCATION_ACCURACY_BEST (in module *location*), 128
- LOCATION_ACCURACY_BEST_FOR_NAVIGATION (in module *location*), 128
- LOCATION_ACCURACY_HUNDRED_METERS (in module *location*), 128
- LOCATION_ACCURACY_KILOMETER (in module *location*), 128
- LOCATION_ACCURACY_NEAREST_TEN_METERS (in module *location*), 128
- LOCATION_ACCURACY_THREE_KILOMETERS (in module *location*), 128
- longitude (*location.Location* attribute), 128
- lyrics (*music.MediaType* attribute), 124
- ## M
- MagneticField (class in *motion*), 130
- mainthread (module), 120
- maximum_height (in module *notification_center*), 117
- maximum_track_color (*pyto_ui.Slider* attribute), 97
- maximum_value (*pyto_ui.Slider* attribute), 97
- MediaItem (class in *music*), 122
- MediaQuery (class in *music*), 125
- message (*notifications.Notification* attribute), 120
- minimum_track_color (*pyto_ui.Slider* attribute), 97
- minimum_value (*pyto_ui.Slider* attribute), 97
- motion (module), 128
- movable (*pyto_ui.TableViewController* attribute), 92
- MPMediaItem (in module *music*), 126
- MPMediaItemCollection (in module *music*), 126
- MPMediaQuery (in module *music*), 126
- MPMusicPlayerController (in module *music*), 126
- multipeer (module), 130
- music (module), 121
- ## N
- name (*pyto_ui.View* attribute), 89
- navigation_bar_hidden (*pyto_ui.View* attribute), 90
- next () (in module *music*), 122
- Notification (class in *notifications*), 120
- notification_center (module), 117
- notifications (module), 120
- now_playing_item () (in module *music*), 122
- number_of_lines (*pyto_ui.Label* attribute), 92
- number_of_touches (*pyto_ui.GestureRecognizer* attribute), 103
- ## O
- on (*pyto_ui.Switch* attribute), 97
- on_color (*pyto_ui.Switch* attribute), 97
- opaque (*pyto_ui.View* attribute), 90
- open_url () (in module *sharing*), 119
- open_url () (in module *xcallback*), 130
- origin (*pyto_ui.View* attribute), 90
- ## P
- pasteboard (module), 132
- pause () (*sound.AudioPlayer* method), 118
- persistent_id (*music.MediaType* attribute), 124
- photos (module), 127
- pick_documents () (in module *sharing*), 119

- pick_music() (in module music), 122
 pick_photo() (in module photos), 127
 picked_files() (in module sharing), 119
 pitch (motion.Attitude attribute), 129
 placeholder (pyto_ui.TextField attribute), 99
 play() (in module music), 121
 play() (sound.AudioPlayer method), 118
 play_beep() (in module sound), 118
 play_count (music.MediaItem attribute), 124
 play_file() (in module sound), 118
 play_system_sound() (in module sound), 118
 playback_duration (music.MediaItem attribute), 124
 playback_state() (in module music), 122
 PLAYBACK_STATE_INTERRUPTED (in module music), 127
 PLAYBACK_STATE_PAUSED (in module music), 127
 PLAYBACK_STATE_PLAYING (in module music), 127
 PLAYBACK_STATE_SEEKING_BACKWARD (in module music), 127
 PLAYBACK_STATE_SEEKING_FORWARD (in module music), 127
 PLAYBACK_STATE_STOPPED (in module music), 127
 playback_store_id (music.MediaItem attribute), 124
 playing (sound.AudioPlayer attribute), 118
 playlists() (music.MediaQuery class method), 126
 podcast_persistent_id (music.MediaItem attribute), 125
 podcast_title (music.MediaItem attribute), 125
 podcasts() (music.MediaQuery class method), 126
 pop() (pyto_ui.View method), 90
 PRESENTATION_MODE_FULLSCREEN (in module pyto_ui), 117
 PRESENTATION_MODE_SHEET (in module pyto_ui), 117
 PRESENTATION_MODE_WIDGET (in module pyto_ui), 117
 previous() (in module music), 122
 push() (pyto_ui.View method), 90
 pyto_ui (module), 85
- ## Q
- quick_look() (in module sharing), 119
- ## R
- rating (music.MediaItem attribute), 125
 red() (pyto_ui.Color method), 100
 release_date (music.MediaItem attribute), 125
 reload() (pyto_ui.WebView method), 96
 reload_action (pyto_ui.TableView attribute), 93
 removable (pyto_ui.TableViewCell attribute), 93
 remove_delivered_notifications() (in module notifications), 121
 remove_from_superview() (pyto_ui.View method), 90
 remove_gesture_recognizer() (pyto_ui.View method), 90
 REPEAT_MODE_ALL (in module music), 126
 REPEAT_MODE_DEFAULT (in module music), 126
 REPEAT_MODE_NONE (in module music), 126
 REPEAT_MODE_ONE (in module music), 126
 requires_exclusive_touch_type (pyto_ui.GestureRecognizer attribute), 103
 resign_first_responder() (pyto_ui.View method), 90
 restart() (in module music), 121
 return_key_type (pyto_ui.TextField attribute), 99
 return_key_type (pyto_ui.TextView attribute), 94
 RETURN_KEY_TYPE_CONTINUE (in module pyto_ui), 109
 RETURN_KEY_TYPE_DEFAULT (in module pyto_ui), 109
 RETURN_KEY_TYPE_DONE (in module pyto_ui), 109
 RETURN_KEY_TYPE_EMERGENCY_CALL (in module pyto_ui), 109
 RETURN_KEY_TYPE_GO (in module pyto_ui), 109
 RETURN_KEY_TYPE_GOOGLE (in module pyto_ui), 109
 RETURN_KEY_TYPE_JOIN (in module pyto_ui), 109
 RETURN_KEY_TYPE_NEXT (in module pyto_ui), 109
 RETURN_KEY_TYPE_ROUTE (in module pyto_ui), 109
 RETURN_KEY_TYPE_SEARCH (in module pyto_ui), 110
 RETURN_KEY_TYPE_SEND (in module pyto_ui), 110
 RETURN_KEY_TYPE_YAHOO (in module pyto_ui), 110
 rgb() (pyto_ui.Color class method), 100
 roll (motion.Attitude attribute), 129
 Rotation (class in motion), 130
 run_async() (in module mainthread), 120
- ## S
- save_image() (in module photos), 127
 schedule_notification() (in module notifications), 121
 sections (pyto_ui.TableView attribute), 93
 secure (pyto_ui.TextField attribute), 99
 secure (pyto_ui.TextView attribute), 94
 SegmentedControl (class in pyto_ui), 96
 segments (pyto_ui.SegmentedControl attribute), 96
 selectable (pyto_ui.TextView attribute), 94
 selected_segment (pyto_ui.SegmentedControl attribute), 97
 send() (in module multipeer), 130
 set() (in module userkeys), 131
 set_image() (in module pasteboard), 132
 set_images() (in module pasteboard), 132

- set_on_with_animation() (*pyto_ui.Switch method*), 97
 set_queue_with_items() (*in module music*), 121
 set_queue_with_store_ids() (*in module music*), 121
 set_repeat_mode() (*in module music*), 122
 set_shuffle_mode() (*in module music*), 122
 set_string() (*in module pasteboard*), 132
 set_strings() (*in module pasteboard*), 132
 set_url() (*in module pasteboard*), 132
 set_urls() (*in module pasteboard*), 132
 set_value_with_animation() (*pyto_ui.Slider method*), 97
 share_items() (*in module sharing*), 119
 sharing (*module*), 119
 show() (*pyto_ui.Alert method*), 105
 show_view() (*in module pyto_ui*), 105
 SHUFFLE_MODE_ALBUMS (*in module music*), 127
 SHUFFLE_MODE_DEFAULT (*in module music*), 127
 SHUFFLE_MODE_OFF (*in module music*), 127
 SHUFFLE_MODE_SONGS (*in module music*), 127
 size (*pyto_ui.View attribute*), 90
 size_to_fit() (*pyto_ui.View method*), 90
 skip_count (*music.MediaItem attribute*), 125
 Slider (*class in pyto_ui*), 97
 smart_dashes (*pyto_ui.TextField attribute*), 99
 smart_dashes (*pyto_ui.TextView attribute*), 94
 smart_quotes (*pyto_ui.TextField attribute*), 99
 smart_quotes (*pyto_ui.TextView attribute*), 94
 songs() (*music.MediaQuery class method*), 126
 start_updating() (*in module location*), 128
 start_updating() (*in module motion*), 129
 state (*pyto_ui.GestureRecognizer attribute*), 103
 stop() (*in module music*), 121
 stop() (*pyto_ui.WebView method*), 96
 stop() (*sound.AudioPlayer method*), 118
 stop_updating() (*in module location*), 128
 stop_updating() (*in module motion*), 129
 string() (*in module pasteboard*), 132
 strings() (*in module pasteboard*), 132
 style (*pyto_ui.ButtonItem attribute*), 104
 subview_with_name() (*pyto_ui.View method*), 90
 subviews (*pyto_ui.View attribute*), 90
 superview (*pyto_ui.View attribute*), 91
 Switch (*class in pyto_ui*), 97
 system_font_of_size() (*pyto_ui.Font class method*), 101
 SYSTEM_ITEM_ACTION (*in module pyto_ui*), 116
 SYSTEM_ITEM_ADD (*in module pyto_ui*), 116
 SYSTEM_ITEM_BOOKMARKS (*in module pyto_ui*), 116
 SYSTEM_ITEM_CAMERA (*in module pyto_ui*), 116
 SYSTEM_ITEM_CANCEL (*in module pyto_ui*), 116
 SYSTEM_ITEM_COMPOSE (*in module pyto_ui*), 116
 SYSTEM_ITEM_DONE (*in module pyto_ui*), 116
 SYSTEM_ITEM_EDIT (*in module pyto_ui*), 116
 SYSTEM_ITEM_FAST_FORWARD (*in module pyto_ui*), 116
 SYSTEM_ITEM_FLEXIBLE_SPACE (*in module pyto_ui*), 116
 SYSTEM_ITEM_ORGANIZE (*in module pyto_ui*), 116
 SYSTEM_ITEM_PAUSE (*in module pyto_ui*), 116
 SYSTEM_ITEM_PLAY (*in module pyto_ui*), 116
 SYSTEM_ITEM_REDO (*in module pyto_ui*), 116
 SYSTEM_ITEM_REFRESH (*in module pyto_ui*), 116
 SYSTEM_ITEM_REPLY (*in module pyto_ui*), 116
 SYSTEM_ITEM_REWIND (*in module pyto_ui*), 116
 SYSTEM_ITEM_SAVE (*in module pyto_ui*), 116
 SYSTEM_ITEM_SEARCH (*in module pyto_ui*), 117
 SYSTEM_ITEM_STOP (*in module pyto_ui*), 117
 SYSTEM_ITEM_TRASH (*in module pyto_ui*), 117
 SYSTEM_ITEM_UNDO (*in module pyto_ui*), 117
- ## T
- table_view (*pyto_ui.TableViewSection attribute*), 104
 TABLE_VIEW_CELL_STYLE_DEFAULT (*in module pyto_ui*), 115
 TABLE_VIEW_CELL_STYLE_SUBTITLE (*in module pyto_ui*), 115
 TABLE_VIEW_CELL_STYLE_VALUE1 (*in module pyto_ui*), 115
 TABLE_VIEW_CELL_STYLE_VALUE2 (*in module pyto_ui*), 115
 TABLE_VIEW_STYLE_GROUPED (*in module pyto_ui*), 115
 TABLE_VIEW_STYLE_PLAIN (*in module pyto_ui*), 115
 TableView (*class in pyto_ui*), 93
 TableViewCell (*class in pyto_ui*), 92
 TableViewSection (*class in pyto_ui*), 103
 take_photo() (*in module photos*), 127
 text (*pyto_ui.Label attribute*), 92
 text (*pyto_ui.TextField attribute*), 99
 text (*pyto_ui.TextView attribute*), 94
 text_alignment (*pyto_ui.Label attribute*), 92
 text_alignment (*pyto_ui.TextField attribute*), 99
 text_alignment (*pyto_ui.TextView attribute*), 95
 TEXT_ALIGNMENT_CENTER (*in module pyto_ui*), 114
 TEXT_ALIGNMENT_JUSTIFIED (*in module pyto_ui*), 114
 TEXT_ALIGNMENT_LEFT (*in module pyto_ui*), 114
 TEXT_ALIGNMENT_NATURAL (*in module pyto_ui*), 114
 TEXT_ALIGNMENT_RIGHT (*in module pyto_ui*), 114
 text_color (*pyto_ui.Label attribute*), 92
 text_color (*pyto_ui.TextField attribute*), 99
 text_color (*pyto_ui.TextView attribute*), 95
 TEXT_FIELD_BORDER_STYLE_BEZEL (*in module pyto_ui*), 115

TEXT_FIELD_BORDER_STYLE_LINE (in module *pyto_ui*), 115
TEXT_FIELD_BORDER_STYLE_NONE (in module *pyto_ui*), 115
TEXT_FIELD_BORDER_STYLE_ROUNDED_RECT (in module *pyto_ui*), 115
text_label (*pyto_ui*.*TableViewCell* attribute), 93
TextField (class in *pyto_ui*), 98
TextView (class in *pyto_ui*), 93
thumb_color (*pyto_ui*.*Slider* attribute), 97
thumb_color (*pyto_ui*.*Switch* attribute), 98
tint_color (*pyto_ui*.*View* attribute), 91
title (*music*.*MediaItem* attribute), 125
title (*pyto_ui*.*Button* attribute), 98
title (*pyto_ui*.*ButtonItem* attribute), 104
title (*pyto_ui*.*TableViewSection* attribute), 104
title (*pyto_ui*.*View* attribute), 91
title_color (*pyto_ui*.*Button* attribute), 98
TOUCH_TYPE_DIRECT (in module *pyto_ui*), 112
TOUCH_TYPE_INDIRECT (in module *pyto_ui*), 112
TOUCH_TYPE_PENCIL (in module *pyto_ui*), 112

U

UIKit (module), 120
UNUserNotificationCenter (in module *notifications*), 120
url (*notifications*.*Notification* attribute), 120
url (*pyto_ui*.*WebView* attribute), 96
url () (in module *pasteboard*), 132
urls (*sharing*.*FilePicker* attribute), 119
urls () (in module *pasteboard*), 132
user_grouping (*music*.*MediaItem* attribute), 125
user_interaction_enabled (*pyto_ui*.*View* attribute), 91
userkeys (module), 131

V

value (*pyto_ui*.*Slider* attribute), 97
vertical_alignment (*pyto_ui*.*Control* attribute), 96
VERTICAL_ALIGNMENT_BOTTOM (in module *pyto_ui*), 114
VERTICAL_ALIGNMENT_CENTER (in module *pyto_ui*), 114
VERTICAL_ALIGNMENT_FILL (in module *pyto_ui*), 114
VERTICAL_ALIGNMENT_TOP (in module *pyto_ui*), 114
View (class in *pyto_ui*), 87
view (*pyto_ui*.*GestureRecognizer* attribute), 103
volume (*sound*.*AudioPlayer* attribute), 118

W

WebView (class in *pyto_ui*), 95

WebView.*JavaScriptException*, 95
white () (*pyto_ui*.*Color* class method), 100
width (*pyto_ui*.*View* attribute), 91
with_size () (*pyto_ui*.*Font* method), 101

X

x (*motion*.*Acceleration* attribute), 129
x (*motion*.*Gravity* attribute), 129
x (*motion*.*MagneticField* attribute), 130
x (*motion*.*Rotation* attribute), 130
x (*pyto_ui*.*GestureRecognizer* attribute), 103
x (*pyto_ui*.*View* attribute), 91
xcallback (module), 130

Y

y (*motion*.*Acceleration* attribute), 129
y (*motion*.*Gravity* attribute), 130
y (*motion*.*MagneticField* attribute), 130
y (*motion*.*Rotation* attribute), 130
y (*pyto_ui*.*GestureRecognizer* attribute), 103
y (*pyto_ui*.*View* attribute), 91
yaw (*motion*.*Attitude* attribute), 129

Z

z (*motion*.*Acceleration* attribute), 129
z (*motion*.*Gravity* attribute), 130
z (*motion*.*MagneticField* attribute), 130
z (*motion*.*Rotation* attribute), 130