READMEs for Open Projects

Open Life Science • Week 4

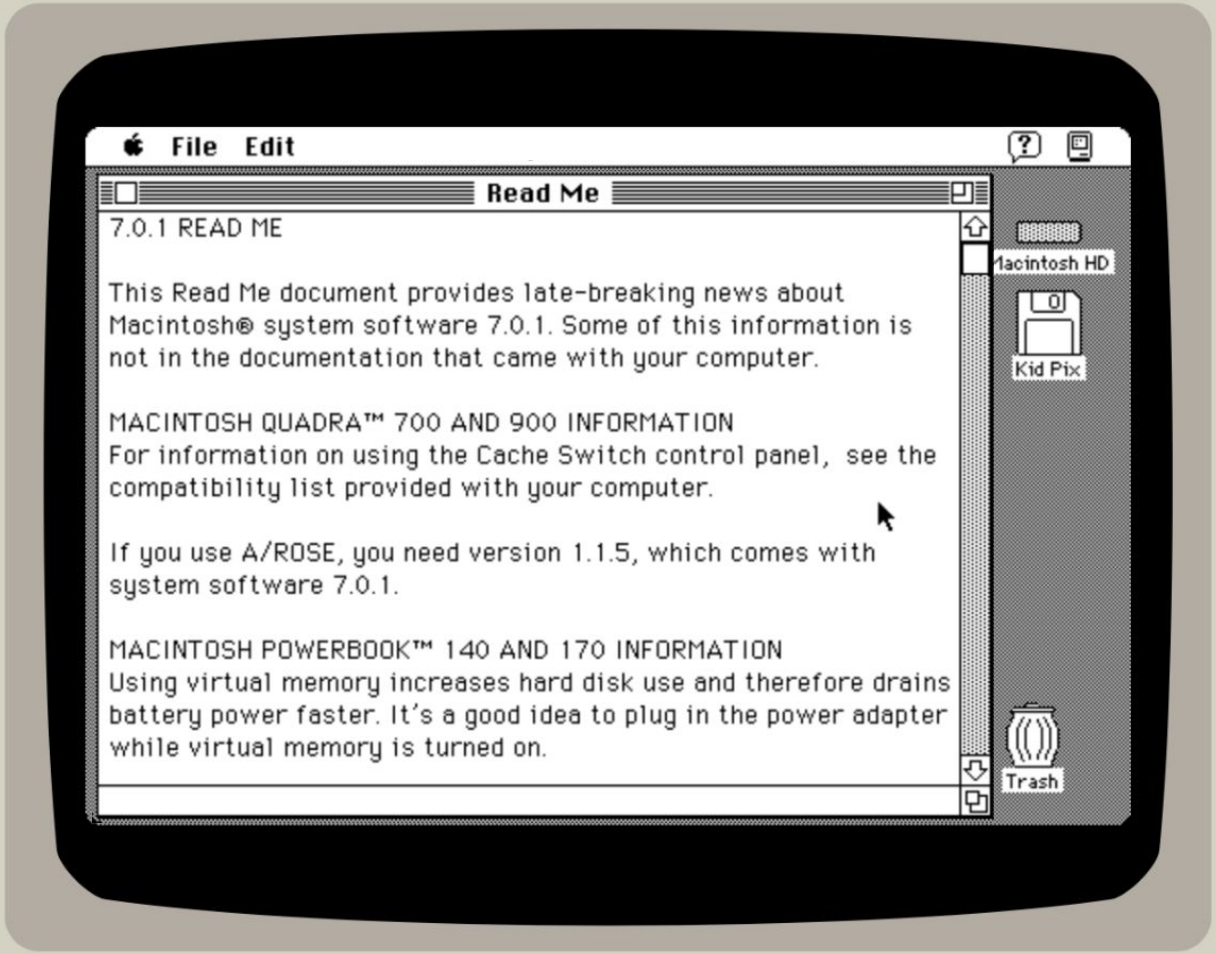
Alex Chan • they/she

Slides: **bit.ly/ols-readme**

THE WELLCOME BUILDING
wellcome collection
Somewhere in Between
Four Collaborations

1995.11
1994.11
1992 11
2003 11
2002 11
2001 11

File
Edit
View
Label
Special
Macintosh HD
7 items
6.8 MB in disk
13.1 MB available
System Folder
Read Me
TeachText
HyperCard 2.1 Player
Macintosh Basics
MacDraw 1.9
MacPaint 2.0
Macintosh HD
Kid Pix
Trash



7.0.1 READ ME

This Read Me document provides late-breaking news about Macintosh® system software 7.0.1. Some of this information is not in the documentation that came with your computer.

MACINTOSH QUADRA™ 700 AND 900 INFORMATION
For information on using the Cache Switch control panel, see the compatibility list provided with your computer.

If you use A/ROSE, you need version 1.1.5, which comes with system software 7.0.1.

MACINTOSH POWERBOOK™ 140 AND 170 INFORMATION
Using virtual memory increases hard disk use and therefore drains battery power faster. It's a good idea to plug in the power adapter while virtual memory is turned on.

File
Edit
View
Label
Special
Macintosh HD
7 items
6.8 MB in disk
13.1 MB available
System Folder
Read Me
TeachText
HyperCard 2.1 Player
Macintosh Basics
MacDraw 1.9
MacPaint 2.0
Macintosh HD
Kid Pix
Trash

What is a README?

README

README.txt

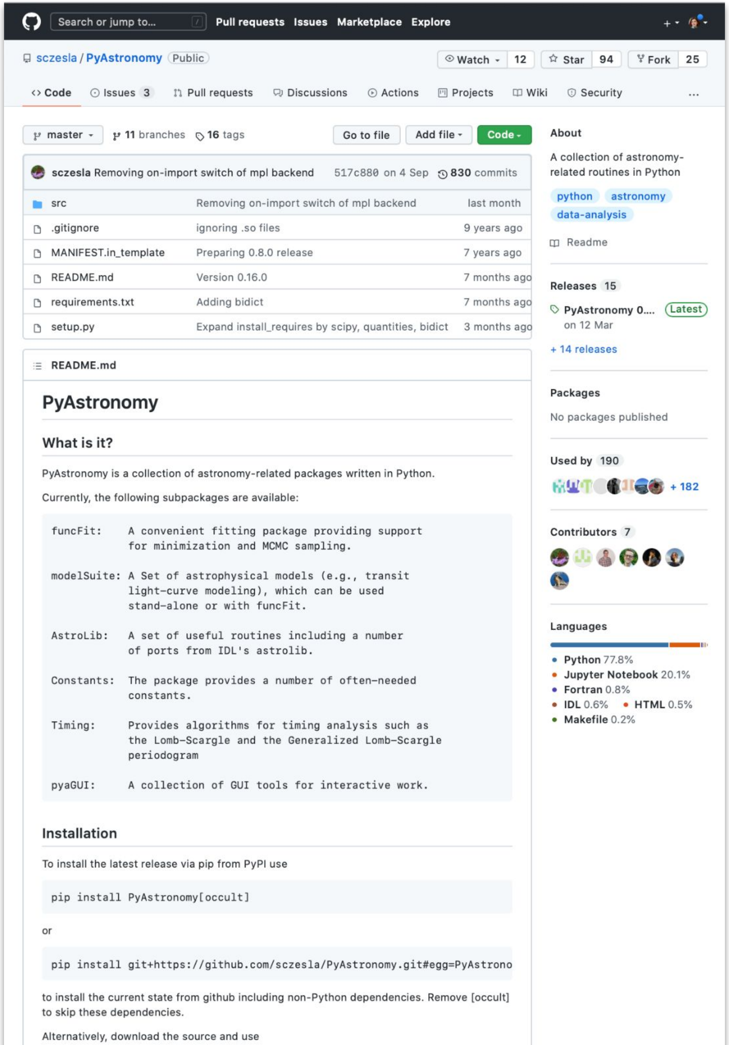
README.md

Open Life Science

What is a README?

What should a README tell me?

- What is this project?
- Who should use it?
- How do they get started?



sczesla Removing on-import switch of mpl backend 517c880 on 4 Sep 830 commits

src	Removing on-import switch of mpl backend	last month
.gitignore	ignoring .so files	9 years ago
MANIFEST.in_template	Preparing 0.8.0 release	7 years ago
README.md	Version 0.16.0	7 months ago
requirements.txt	Adding bidict	7 months ago
setup.py	Expand install_requires by scipy, quantities, bidict	3 months ago

About

A collection of astronomy-related routines in Python

Releases 15

PyAstronomy 0.... Latest
on 12 Mar

+ 14 releases

Packages

No packages published

Used by 190

+ 182

Contributors 7

Languages

- Python 77.8%
- Jupyter Notebook 20.1%
- Fortran 0.8%
- IDL 0.6%
- HTML 0.5%
- Makefile 0.2%

README.md

PyAstronomy

What is it?

PyAstronomy is a collection of astronomy-related packages written in Python.

Currently, the following subpackages are available:

```
funcFit:    A convenient fitting package providing support
            for minimization and MCMC sampling.

modelSuite: A Set of astrophysical models (e.g., transit
            light-curve modeling), which can be used
            stand-alone or with funcFit.

AstroLib:   A set of useful routines including a number
            of ports from IDL's astrolib.

Constants:  The package provides a number of often-needed
            constants.

Timing:     Provides algorithms for timing analysis such as
            the Lomb-Scargle and the Generalized Lomb-Scargle
            periodogram

pyaGUI:     A collection of GUI tools for interactive work.
```

Installation

To install the latest release via pip from PyPI use

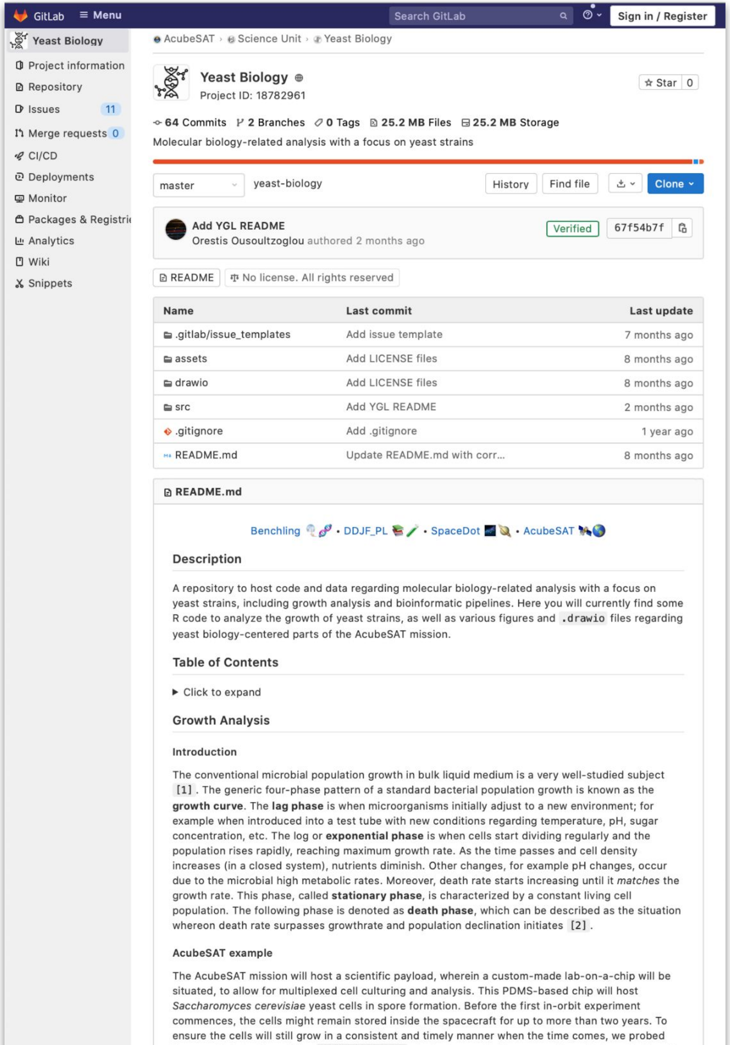
```
pip install PyAstronomy[occult]
```

or

```
pip install git+https://github.com/sczesla/PyAstronomy.git#egg=PyAstrono
```

to install the current state from github including non-Python dependencies. Remove [occult] to skip these dependencies.

Alternatively, download the source and use



Yeast Biology

Project ID: 18782961

64 Commits 2 Branches 0 Tags 25.2 MB Files 25.2 MB Storage

Molecular biology-related analysis with a focus on yeast strains

Add YGL README
Orestis Ousoultzoglou authored 2 months ago Verified 67f54b7f

Name	Last commit	Last update
.gitlab/issue_templates	Add issue template	7 months ago
assets	Add LICENSE files	8 months ago
drawio	Add LICENSE files	8 months ago
src	Add YGL README	2 months ago
.gitignore	Add .gitignore	1 year ago
README.md	Update README.md with corr...	8 months ago

README.md

Benchling 🔬🧬 • DDJF_PL 📚🧪 • SpaceDot 🌌🪐 • AcubeSAT 🛰️🌍

Description

A repository to host code and data regarding molecular biology-related analysis with a focus on yeast strains, including growth analysis and bioinformatic pipelines. Here you will currently find some R code to analyze the growth of yeast strains, as well as various figures and `.drawio` files regarding yeast biology-centered parts of the AcubeSAT mission.

Table of Contents

▶ Click to expand

Growth Analysis

Introduction

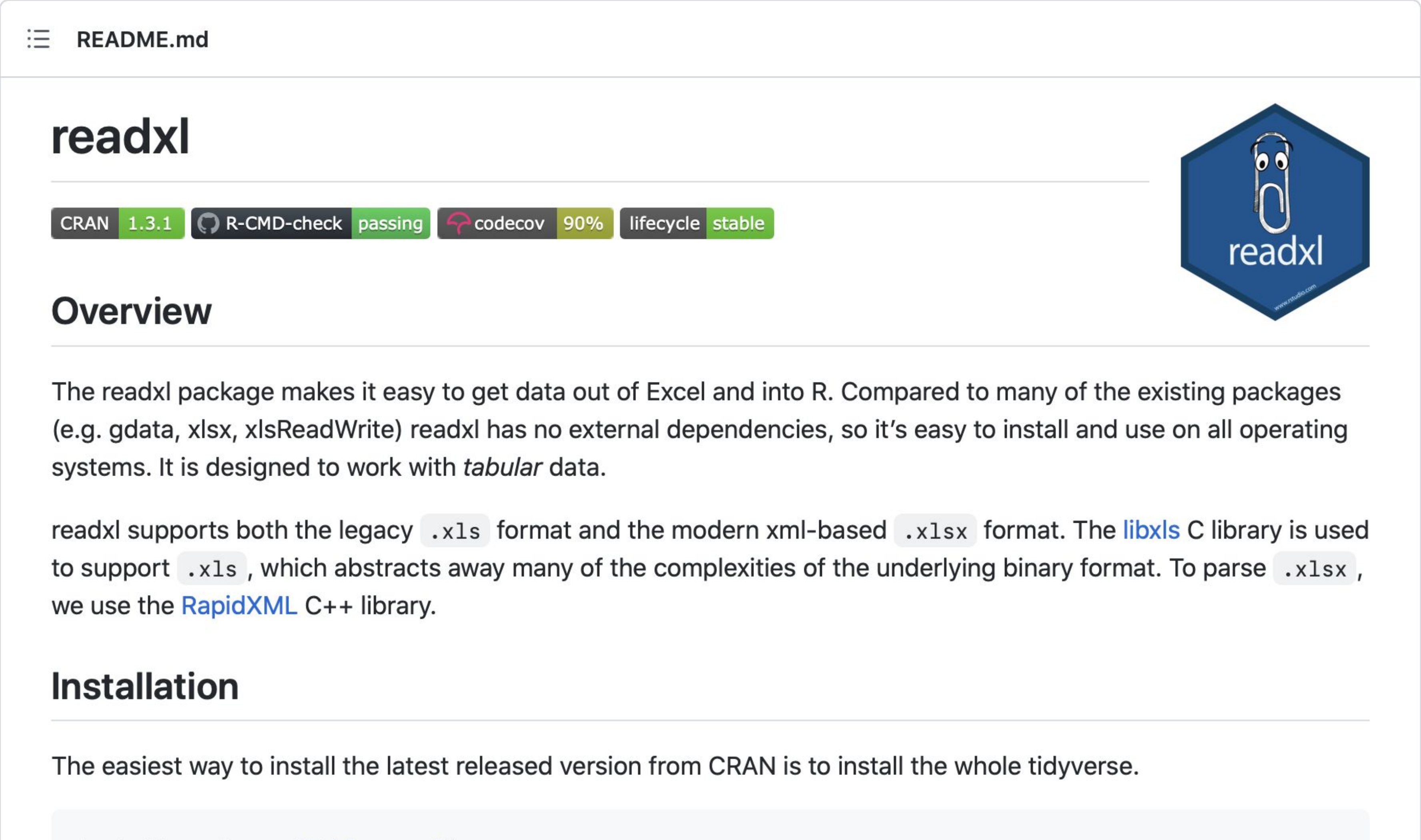
The conventional microbial population growth in bulk liquid medium is a very well-studied subject `[1]`. The generic four-phase pattern of a standard bacterial population growth is known as the **growth curve**. The **lag phase** is when microorganisms initially adjust to a new environment; for example when introduced into a test tube with new conditions regarding temperature, pH, sugar concentration, etc. The log or **exponential phase** is when cells start dividing regularly and the population rises rapidly, reaching maximum growth rate. As the time passes and cell density increases (in a closed system), nutrients diminish. Other changes, for example pH changes, occur due to the microbial high metabolic rates. Moreover, death rate starts increasing until it *matches* the growth rate. This phase, called **stationary phase**, is characterized by a constant living cell population. The following phase is denoted as **death phase**, which can be described as the situation whereon death rate surpasses growthrate and population declination initiates `[2]`.

AcubeSAT example

The AcubeSAT mission will host a scientific payload, wherein a custom-made lab-on-a-chip will be situated, to allow for multiplexed cell culturing and analysis. This PDMS-based chip will host *Saccharomyces cerevisiae* yeast cells in spore formation. Before the first in-orbit experiment commences, the cells might remain stored inside the spacecraft for up to more than two years. To ensure the cells will still grow in a consistent and timely manner when the time comes, we probed

What should a README tell me?

- What is this project?
- Who should use it?
- How do they get started?



README.md

readxl

CRAN 1.3.1 R-CMD-check passing codecov 90% lifecycle stable

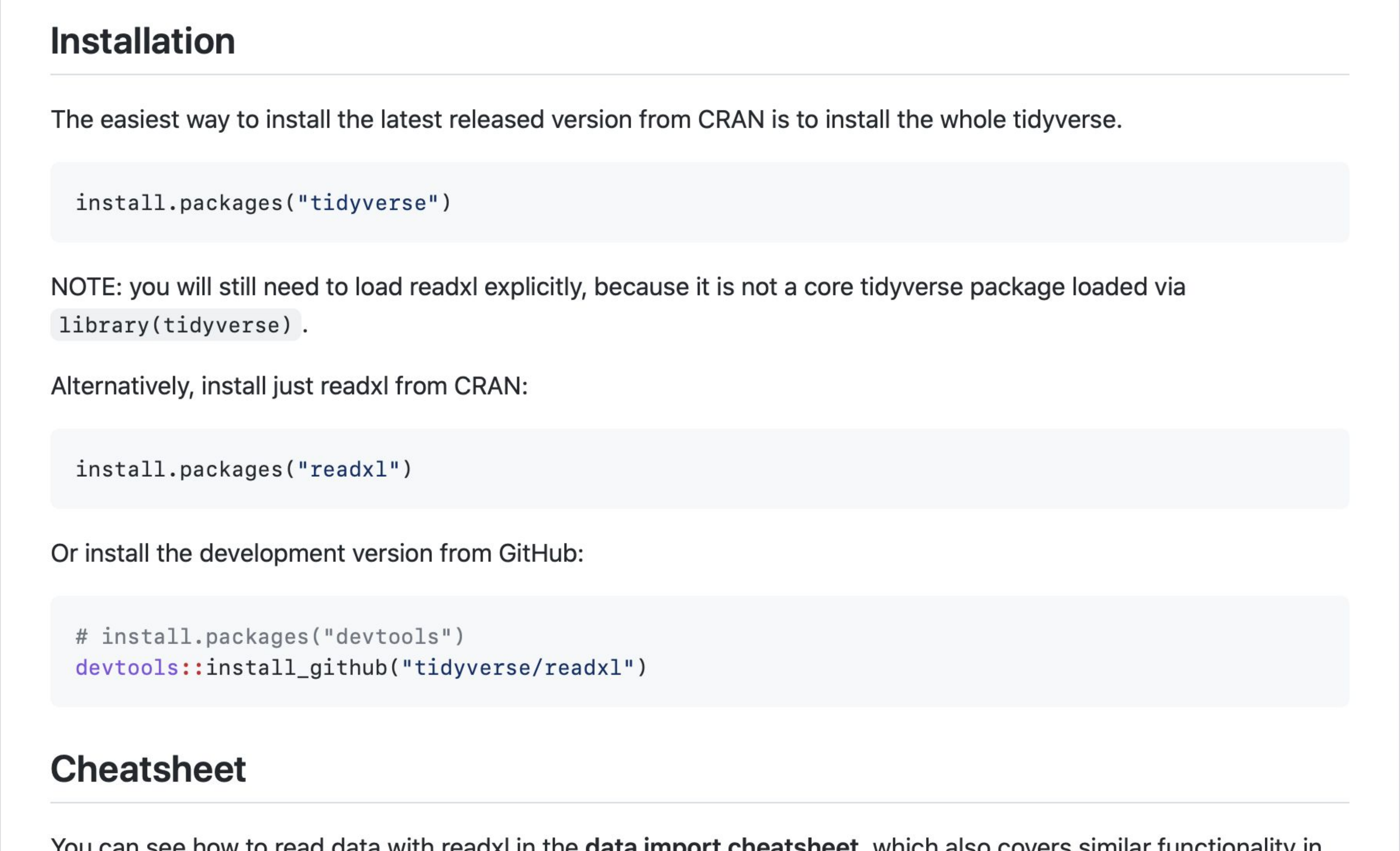
Overview

The readxl package makes it easy to get data out of Excel and into R. Compared to many of the existing packages (e.g. gdata, xlsx, xlsReadWrite) readxl has no external dependencies, so it’s easy to install and use on all operating systems. It is designed to work with *tabular* data.

readxl supports both the legacy `.xls` format and the modern xml-based `.xlsx` format. The libxls C library is used to support `.xls`, which abstracts away many of the complexities of the underlying binary format. To parse `.xlsx`, we use the RapidXML C++ library.

Installation

The easiest way to install the latest released version from CRAN is to install the whole tidyverse.



Installation

The easiest way to install the latest released version from CRAN is to install the whole tidyverse.

```
install.packages("tidyverse")
```

NOTE: you will still need to load readxl explicitly, because it is not a core tidyverse package loaded via `library(tidyverse)`.

Alternatively, install just readxl from CRAN:

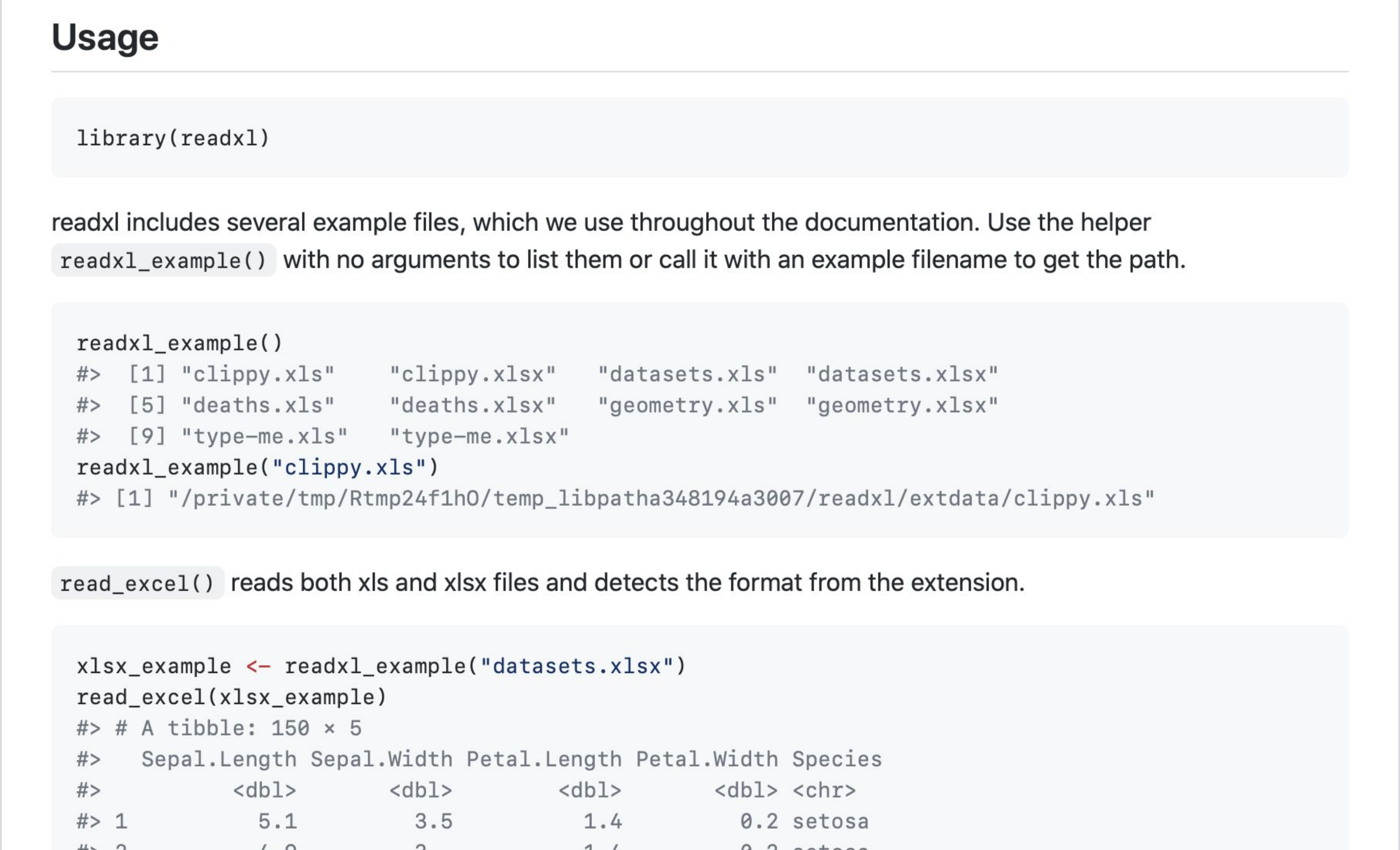
```
install.packages("readxl")
```

Or install the development version from GitHub:

```
# install.packages("devtools")
devtools::install_github("tidyverse/readxl")
```

Cheatsheet

You can see how to read data with readxl in the **data import cheatsheet**, which also covers similar functionality in



Usage

```
library(readxl)
```

readxl includes several example files, which we use throughout the documentation. Use the helper `readxl_example()` with no arguments to list them or call it with an example filename to get the path.

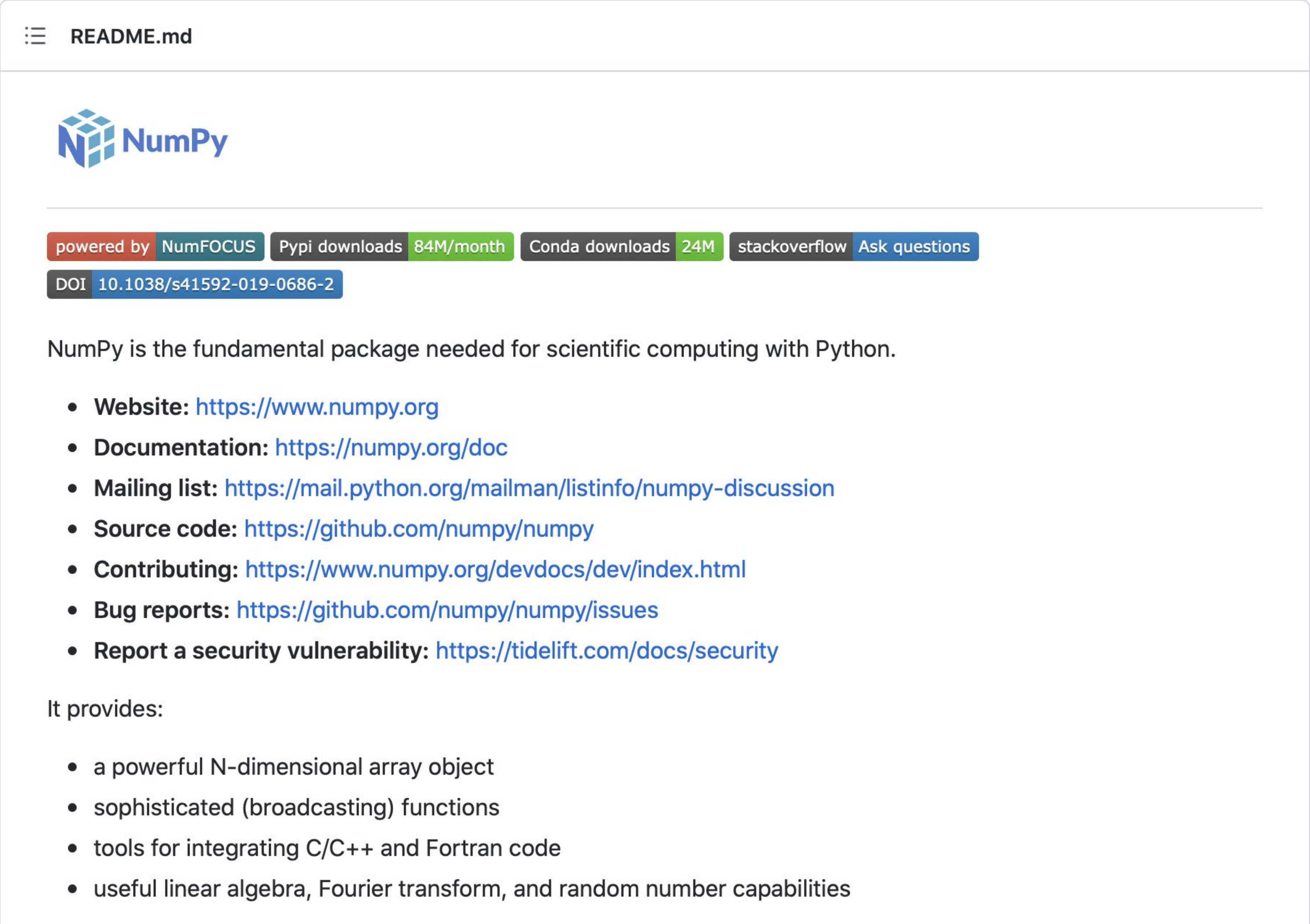
```
readxl_example()
#>  [1] "clippy.xls"    "clippy.xlsx"   "datasets.xls"  "datasets.xlsx"
#>  [5] "deaths.xls"    "deaths.xlsx"   "geometry.xls"  "geometry.xlsx"
#>  [9] "type-me.xls"   "type-me.xlsx"
readxl_example("clippy.xls")
#> [1] "/private/tmp/Rtmp24f1hO/temp_libpatha348194a3007/readxl/extdata/clippy.xls"
```

`read_excel()` reads both xls and xlsx files and detects the format from the extension.

```
xlsx_example <- readxl_example("datasets.xlsx")
read_excel(xlsx_example)
#> # A tibble: 150 × 5
#>   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
#>          <dbl>       <dbl>        <dbl>       <dbl> <chr>
#> 1          5.1         3.5          1.4         0.2 setosa
#> 2          4.9         3            1.4         0.2 setosa
```

What should a README tell me?

- What is this project?
- Who should use it?
- How do they get started?



README.md

NumPy

powered by NumFOCUS | Pypi downloads 84M/month | Conda downloads 24M | stackoverflow Ask questions
DOI 10.1038/s41592-019-0686-2

NumPy is the fundamental package needed for scientific computing with Python.

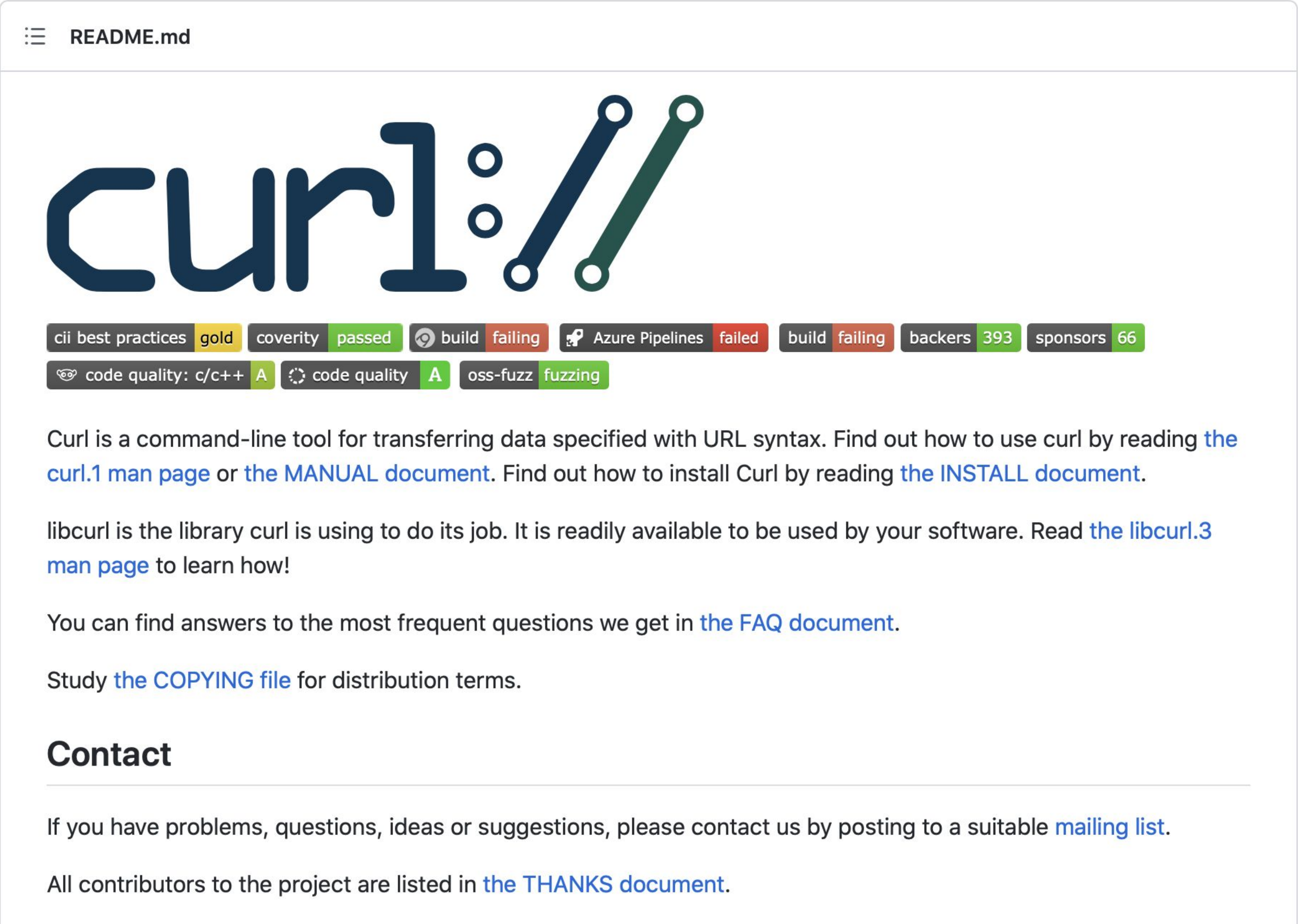
- **Website:** https://www.numpy.org
- **Documentation:** https://numpy.org/doc
- **Mailing list:** https://mail.python.org/mailman/listinfo/numpy-discussion
- **Source code:** https://github.com/numpy/numpy
- **Contributing:** https://www.numpy.org/devdocs/dev/index.html
- **Bug reports:** https://github.com/numpy/numpy/issues
- **Report a security vulnerability:** https://tidelift.com/docs/security

It provides:

- a powerful N-dimensional array object
- sophisticated (broadcasting) functions
- tools for integrating C/C++ and Fortran code
- useful linear algebra, Fourier transform, and random number capabilities

What should a README tell me?

- What is this project?
- Who should use it?
- How do they get started?



README.md

cii best practices gold | coverity passed | build failing | Azure Pipelines failed | build failing | backers 393 | sponsors 66 | code quality: c/c++ A | code quality A | oss-fuzz fuzzing

Curl is a command-line tool for transferring data specified with URL syntax. Find out how to use curl by reading the curl.1 man page or the MANUAL document. Find out how to install Curl by reading the INSTALL document.

libcurl is the library curl is using to do its job. It is readily available to be used by your software. Read the libcurl.3 man page to learn how!

You can find answers to the most frequent questions we get in the FAQ document.

Study the COPYING file for distribution terms.

Contact

If you have problems, questions, ideas or suggestions, please contact us by posting to a suitable mailing list.

All contributors to the project are listed in the THANKS document.

What should a README tell me?

- What is this project?
- Who should use it?
- How do they get started?

Open Life Science

What a README is not:

- The comprehensive documentation for your project
- The only documentation for your project

What should a README tell me?

- What is this project?
- Who should use it?
- How do they use it?

What is a README?

READMEs for Open Projects

Open Life Science • Week 4

Alex Chan • they/she

Slides: **bit.ly/ols-readme**