AST3100: Code development and packaging mini-course

Winter 2020

(last updated: February 23, 2020; rev. 818cfce)

Course description

This graduate-level mini-course provides an overview of the steps involved in developing, publishing, and maintaining a scientific software package. The course focuses on current best practices and the different tools available to build, release, and maintain a software package. The course will involve students building their own small software package, learning how to use git/GitHub to host it, how to test and document it, how to use automatic documentation/building/testing tools, how to release the package, and how to deal with issues and pull requests.

Logistics

- **Meeting times and place:** Tue, 10:30–12:30pm, Room: AB 113

  The class will meet five times: Feb. 25, Mar. 3, Mar. 10, Mar. 24 (note the weeklong break), and the last class is to TBD.

  Sessions will typically consist of about a half-hour/hour of lecture/discussion and in-class work for the remainder of the session

- **Instructor:** Jo Bovy, AB 229.
- **Email:** jo.bovy@utoronto.ca
- **Office hours:** Stop by my office or by appointment.
- **Course website:** `https://github.com/jobovy/code-packaging-minicourse`.

Learning objectives

After this course you should understand

- the dos and don'ts of package development
- how to setup a Python package structure that can be easily installed
- version control with git, hosting code on GitHub
- how to document your code using sphinx
- how to write a good test and how to run a test suite with pytest
- test code coverage and how to measure it
- code licenses

- how to use travis to build and test your code
- how to use readthedocs to host documentation
- how to release your code to PyPI (`pip install X`)
- how to write, compile, and use a C extension in your Python code
- the basics of GitHub workflows
- how to get nifty badges for your GitHub site

Reading

A set of lecture notes will be posted on the course website throughout the semester.

Grading scheme

As part of this course, students will develop a small package in four steps (one for each of the first four lectures); each of these steps will be worth 25% of the final mark.

- **Assignment 1:** Set up a basic package and put it on your GitHub page
- **Assignment 2:** Set up and write documentation for your package and publish it to `readthedocs.io`.
- **Assignment 3:** Write a test suite for your code, set it up to run on `travis`, send code-coverage results to `codecov`.
- **Assignment 4:** Release your code to PyPI, add nifty badges to your GitHub page.

More information on these assignments will be given in class.

Schedule

- **Lecture 1:** Class logistics; Introduction, git/GitHub, basics of Python packages, code licensing, issues and pull requests.
- **Lecture 2:** Documentation: Sphinx, readthedocs
- **Lecture 3:** Code testing, using pytest as a test runner, running your tests on travis, code coverage
- **Lecture 4:** Releasing your code to PyPI (`pip install X`), get nifty badges for your GitHub site
- **Lecture 5:** Advanced topics: C extensions, GitHub workflows, ...

Academic integrity

From Appendix D of the Academic Integrity Handbook:

Academic integrity is one of the cornerstones of the University of Toronto. It is critically important both to maintain our community which honours the values of honesty, trust, respect, fairness, and responsibility and to protect you, the students within this community, and the value of the degree towards which you are all working so diligently.

According to Section B of the University of Toronto's Code of Behaviour on Academic Matter (`http://www.governingcouncil.utoronto.ca/policies/behaveac.htm`) which all students are expected to read and by which they are expected to abide, it is an offence for students to:

- Use someone else's ideas or words in their own work without acknowledging explicitly that those ideas/words are not their own with a citation and quotation marks, i.e. to commit plagiarism.
- Include false, misleading, or concocted citations in their work.
- Obtain unauthorized assistance on any assignment.
- Provide unauthorized assistance to another students. This includes showing another student your own work.
- Submit their own work for credit in more than one course without the permission of the instructors.

There are other offenses covered under the Code, but these are the most common. You are instructed to respect these rules and the values that they protect.