# PIC 16 – Python with Applications – Winter 2018

| | |
|---|---|
| Instructor: | Matt Haberland <haberland@ucla.edu> @ MS 6617A |
| TA: | Wonjun Lee <wlee@math.ucla.edu> @ PIC Lab (MS 2000) |
| Lecture: | MWF 11:00 a.m. – 11:50 a.m. @ Geo 4645 (and PIC Lab as announced) |
| Discussion: | TR, times and places same as lecture |
| Office Hours: | M 10:00 a.m. – 10:45 a.m., W 12:25 p.m. – 2:25 p.m.<br>F 12:00 p.m. – 12:30 p.m. (Haberland @ MS 6617A)<br>T 12:00 p.m. – 1:00 p.m., R 1:00 p.m. – 2:00 p.m. (Lee @ PIC Lab) |

**Format**

Instruction will follow a "flipped classroom" model, that is, students will review readings, code, and short videos at home *before* classroom sessions and will begin activities and/or assignments during class time. Sometimes class will include a lecture to *supplement* the preparatory work, but these lectures will refer to the preparatory work. It is essential that students complete this preparatory work in order to maximize learning during class.

**Requirements and Recommendations**

Prior completion of at least two sequential programming classes is required. Concepts covered in these courses will be built upon without planned review. Students should already be quite comfortable applying common programming elements such as variables, operators, functions, control-flow statements, classes/objects, data structures, etc… to write programs in at least one language.

After learning how to apply these programming elements in Python, we will delve into applications. At this point, the course splits into two: you can elect to study either mathematical applications or "less-mathematical" applications. Topics covered in lecture will alternate between the two "tracks", and you will only be responsible for material from one of the two tracks. (You are welcome to attend class and complete assignments for both, though!) More information is available in the document Regarding Tracks…

No particular mathematical background is assumed for either track; resources for learning (or reviewing) required math will be provided. However, students who elect to study the mathematical applications will be more comfortable if they have a solid understanding of algebra and calculus (e.g. trigonometry, systems of equations, vector and matrix operations, differentiation, integration, unconstrained optimization of a single variable). Nonetheless, students without such a background can succeed in study of the mathematical applications with sufficient interest and work ethic.

**Intended Learning Outcomes**

After successfully completing this course, students will be able to write programs in the Python programming language to accomplish a variety of tasks, including some of the following:

- loading/saving data from/to various formats, including text files, and spreadsheets;
- interacting with the user via a graphical user interface (GUI) and voice user interface (VUI);
- displaying and animating graphics;
- calculating statistics and producing graphs to represent data with more flexibility (and often greater ease) than more common tools, such as Microsoft Excel;
- natural language processing, including tokenization, part-of-speech tagging, and lemmatization;
- scraping data from websites;
- communicating between networked computers;
- solving systems of linear, nonlinear, and ordinary differential equations, linear programming problems, and nonlinear programming problems from real-world examples;
- machine learning, for making decisions based on a model generated from data; and
- image processing and computer vision, including object tracking.

## Expectations

In order to succeed in this course, students are expected to:

- prepare for every class by reviewing the assigned readings, code, and videos,
- engage in all scheduled lectures and discussions,
- participate in class activities and complete assignments on time,
- ask questions and attend office hours as necessary,
- perform additional study and *practice as necessary* to build confidence in the material, and
- maintain academic integrity (see below).

The course staff firmly believes that all students who follow the above guidelines can succeed in this course. Learning can be difficult, but we are committed to ensuring that you do!

## Resources

| | |
|---|---|
| Website: | https://ccle.ucla.edu/course/view/18W-COMPTNG16-1<br>All course information will be posted and all assignments will be submitted here. |
| Textbooks: | None. (No textbook covers all of this cool stuff!) Reading materials will be provided. |
| PIC Lab: | MWF 9 – 6, TR 9 – 9, Sat Closed, Sun 1 – 5 @ MS 2000<br>For updated hours and other information, see http://www.pic.ucla.edu/ |

## Evaluation

Assignments: 60% - Students will be required to complete approximately *half* of all assignments, which will be provided almost every class. Solutions are due on CCLE by the end of the second lecture period *after* the one in which they were assigned. Official solutions will be posted shortly after the assignment is due; work submitted after this cannot be accepted for credit. With each submission, students must also submit an honest evaluation of their solution according to a provided rubric. Assignment credit will typically be granted according to this self-evaluation. However, self-evaluations will be audited, and *dishonest self-assessments will be treated as a breach of academic integrity (see below).*

Quizzes and Participation: 20% - Quizzes on assigned preparation material, administered in class. Additional class, office hour, etc… participation may raise this portion of the grade. Approximately 10% of the lowest assignment and quiz grades will be discarded.

Final Exam: 20% - Sunday, December 10, 11:30a.m. – 2:30p.m. @ TBA

Final letter grades are assigned on an absolute scale: 90-100 A, 80-90 B, 70-80 C, 60-70 D, <60 F.
+/- designations are typically assigned corresponding with the top/bottom 3 points of each range.

## Academic Integrity

"As a student and member of the University community, you are here to get an education and are, therefore, expected to demonstrate integrity in all of your academic endeavors. You are evaluated on your own merits, so be proud of your accomplishments, and protect academic integrity at UCLA."
(Student Guide to Academic Integrity, http://www.deanofstudents.ucla.edu/Portals/16/Documents/StudentGuide.pdf)

Assignments are offered to help students improve their Python programming skills and as a means for evaluation. Some uses of outside resources, such as internet tutorials, office hours, and broad discussion with other students, support the learning process and are encouraged. Use of resources in ways that do not promote learning, such as referring to the work of other students and copying existing code without adaptation and sufficient understanding, are not allowed. When in doubt, ask course staff proactively. Please acknowledge the contributions of all resources (including other students, not including provided resources or course staff) informally on your solutions. You are encouraged to consult with other students on your self-assessments to help ensure that you evaluate yourself honesty.

## Accommodations for Students with Disabilities

If you are registered with the Office for Students with Disabilities, please discuss accommodations with course staff during the first week.

**Schedule**

Preparations posted on CCLE are to be completed *before* the corresponding lecture.
Assignments posted on CCLE are to be completed within two lectures of the corresponding lecture.
See "Regarding Tracks" for more information about the different tracks.

| Week | Date | Day | Topic | Track |
|---|---|---|---|---|
| 1 | 1/08 | Monday | Introduction to PIC 16 | |
| | 1/10 | Wednesday | Python Basics - Getting Started. Basic Data Types | |
| | 1/12 | Friday | Python Basics - Control Flow, Functions | |
| 2 | 1/15 | Monday | Martin Luther King, Jr. Holiday (No Class) | |
| | 1/17 | Wednesday | Python Basics - Data Structures | |
| | 1/19 | Friday | Python Basics - Functional Programming | |
| 3 | 1/22 | Monday | Python Basics - Exception | |
| | 1/24 | Wednesday | Python Basics - Classes and Objects, Magic Methods | |
| | 1/26 | Friday | Python Basics - Iterators and Generators | |
| 4 | 1/29 | Monday | Python Basics - Input/Output (Console, text files, CSV) | |
| | 1/31 | Wednesday | Regular Expressions | |
| | 2/02 | Friday | Inheritance | |
| 5 | 2/05 | Monday | GUI I: Graphics, Animation | |
| | 2/07 | Wednesday | GUI II: Widgets, Signals and Slots, Events | |
| | 2/09 | Friday | GUI III: Layout | |
| 6 | 2/12 | Monday | Sympy (Computer Algebra) | A |
| | 2/14 | Wednesday | Amazon Alexa I | B |
| | 2/16 | Friday | NumPy and Matplotlib (Numerical Computing) | A |
| 7 | 2/19 | Monday | President's Day Holiday (No Class) | |
| | 2/21 | Wednesday | Amazon Alexa II | B |
| | 2/23 | Friday | SciPy I - Input/Output, Linear Algebra, and FFT | A |
| 8 | 2/26 | Monday | Plotly | B |
| | 2/28 | Wednesday | SciPy II - Interpolation and Optimization | A |
| | 3/02 | Friday | NLTK | B |
| 9 | 3/05 | Monday | SciPy III - Numerical Integration and ODEs | A |
| | 3/07 | Wednesday | Scrapy | B |
| | 3/09 | Friday | OpenCV (Computer Vision) | A |
| 10 | 3/12 | Monday | `socket` (Networking) | B |
| | 3/14 | Wednesday | Scikit-learn (Machine Learning) | A |
| | 3/16 | Friday | `threading` (Multi-threading) | B |
| - | 3/22 | Thursday | Final Exam | |