



Check this page for updates on upcoming classes, our learning goals, and the lesson modules we'll use to get there.

[← return home](#)

## Session Calendar and session guides (Spring 2019)

*CIT-129: Python 2*

Date	Topic	Notes/Status
<b>Wk 1: WED 6 FEB</b>	Data types: Lists, tuples, Strings and dicts	Manipulating the essential elements, including String transformations
<b>Wk 2: WED 13 FEB</b>	Looping, and Methods	Creating, manipulating, looping over dictionaries and other objects Writing code in sensible, bite-sized methods
<b>Wk 3: WED 20 FEB</b>	File I/O	Reading, writing, & transforming
<b>Wk 4: WED 27 FEB</b>	Data encoding	JSON (native to py), XML (foreign to py), HTML (foreign) parsers
<b>Wk 5: WED 6 MAR</b>	Networking: Screen Scraping 101	Using the requests library and parsing HTML Eating BeautifulSoup all night long
<b>Wk 6: WED 13 MAR</b>	Collaborative project	Creating a file tree viewing script for CIT-115 @ west hills
<b>Wk 7: WED 20 MAR</b>	API project	Design and implement a full-scale application in Python that usefully manipulates data from an API. Prepare to share at the start of week 8
<b>Wk 8: WED 27 MAR</b>	Packaging and planning	Transition from Dictionaries and Web into phase II of python II
Wk 9: WED 3 APR	Prep work on mini-lessons	Celebration of completion of core curriculum
Wk 10: WED 10 APR	Peer teaching	<b>Developing peer teaching methods -- topics assigned:</b> Modules: Brandon; Comprehensions: Adam; Regexps: Eric;

		Object-Oriented Python: Jeff; Exceptions: Coral; Advanced functions as objects: Matt; Testing: Dominic
<b>WED 17 APR</b>	SPRING BREAK	NO IN-PERSON MEETING
Wk 11: WED 24 APR	Client project	
Wk 12: WED 1 MAY	Intro to data analytic packages	Exploration of pandas and numpy for data analysis
Wk 13: WED 8 MAY (FINAL)	Final project sharing	Bring a fully-baked final project, please!

## Weekly guides

Guiding questions, objectives, and lesson activities are available for each week of class. Click "toggle full session guide" to view the full guide.

### Wk 1: 6 FEB 19

#### *Reviewing the essential built-in types in Python*

##### Guiding questions

- How do Python's built-in type set compare to other languages you know?
- What are the defining features of tuples, lists, and dictionaries?

##### Learning Objectives

- Create and manipulate small quantities of data in tuples, lists, and dictionaries
- Process Strings with slicing and built-in methods on String objects
- Creatively engineer a program for storing and displaying information about our classmates

##### Resources

- [Official python documentation on the string built-in functions](#)
- [Python documentation on dictionaries](#)

##### Built-in types opening activity

The Python types that we'll review tonight:

1. Numbers