Python for Economics

Richard Lawrence
Morning Session, 9/10/2021
“Introduction”
Table of Contents

(This course is divided into numbered Lessons)

- Orientation
  1. Introduction to Python
  2. Variables
  3. Files (a preview)
  4. Data Types
  5. User Input
- Lunch
  (... more after lunch)
Orientation

A description of the course and some exercises for getting started.
Orientation Learning objectives

1. Motivate the use of Python for Economics
2. Set expectations for course structure
3. Get started with Google Classroom
4. Get started with Google Colab
5. (Maybe) Complete pre-course diagnostic
Why Learn Python

- Programming takes time to master; start early to get ahead
- Most career paths include some amount of programming
- Python is a flexible language that you can take with you wherever you go
- Python is used in Economics for *machine learning* tasks and *web scraping*.
This presentation is the first of 6-part series: a morning and an afternoon on each of three consecutive fridays. You are expected to attend all of them (100%).

- 9/10 am “Introduction” ➙ (you are here)
- 9/10 pm “Algorithms”
- 9/17 am “Data Structures”
- 9/17 pm “Data Tools”
- 9/24 am “Data Analysis”
- 9/24 pm “Data Scavenging”
Learn Python

Python has many topics.

They build on each other.
Learn Python

The goal of this course is to complete a project.

We will learn only a few topics to make that possible.
Course Destination

The end goal for this course is to build a multi-stage data project:

- Get data from the internet
- Put the data into a structure
- Visualize and Analyze the data

and do all of that in Python.
Google Classroom

We use this to host the assignments and course materials.

This makes the materials accessible outside of class.
Find your Classroom

- Start from the link classroom.google.com
- You must use your TAMU google account to log in.
- Find this button (or use Class code 5lf4jn5k)

- If you have never used your TAMU google account before, follow instructions from TAMU IT at google.tamu.edu.
The Classroom

Welcome to Google Classroom.

To find your assignments, click on “classwork”.

![Google Classroom Interface](image.png)
An Assignment

Welcome to Google Classroom.

To work on an assignment, click the **name of the file** in the box labeled “your work”
Your Work

A copy of the file is now in your Google drive.

This is your copy to edit, and it is what we see when you turn it in.
Interactive Learning

- The Jupyter Notebook is a **file** you can edit.
- Colaboratory (provided Free from Google) is the **editor**.
- In a Notebook, you can try out new code right away!
Classroom Exercise!

Find “Example Assignment” and open the file. You should get a new tab like this one.

Return to the classroom tab and turn in the assignment.

CONGRATULATIONS 🎉
Classwork and Homework Assignments

Flexible course format:

- in-person, virtually, or hybrid

Classwork covers fundamentals. **Participation** is required.
Homework covers additional practice and is **mandatory**.

Expected: mastery of today’s fundamentals by the next Friday.
Pass: 80% completion
Getting Help

There are office hours during the week where staff will be available to assist you. (details later)

We can view your work remotely to provide feedback. (any time)

Problems? help@hprc.tamu.edu & put ECON in the subject line
Lesson 1
Introduction to Python

An introduction to programming concepts and characteristics of Python language
Lesson 1 learning objectives

1. Understand general programming concepts

2. Know what Python is, where it comes from, how to use it
   a. Demonstrate different ways of working with Python

3. Write and Execute your first Python code
Programming Concept

- Computers don’t think. They only follow instructions.
- Computers don’t understand what you want. They do what you say, literally.
- Programming means: organizing your thoughts to make what you mean very clear.
- A programming language is a way for programmers to communicate with computers and each other without being misunderstood.
Among Programming Languages

Python is a High-level language, which means:

- Syntax is similar to human language syntax
- Supports abstract concepts
- Takes care of simple hardware tasks for you

Python is designed to simplify the development process so you can focus on what matters.
Python Usefulness

No programming language is perfect for every task. Python is best for research and rapid development.

Pros
- Easy to use
- Modules are readily available
- Portable
- Flexible

Cons
- Not the fastest
- Source code style is mandatory
- Few built-in features
Using Python 3.7.10

Python versions are not all the same. Python3 and Python2 are *not compatible*. Python3 is better, so we will be staying away from Python2.

Colaboratory is set up with 3.7.10 so that’s what we are using.
Where to get Python

Python is an Open Source project administered by the Python Software Foundation. As such, it is both freely available and distributed by multiple package managers.

- Windows offers Python in the app store, with Visual Studio integration.
- Mac and Linux often come with Python pre-installed, but it is most likely the older Python 2. An update to Python 3 is strongly recommended. It is available through your OS’s repository manager.
- Google offers Python though the web browser app Colaboratory.
How do we use it?

Python is an interpreted language, which means Python code is both parsed and executed by Python’s interpreter (also named Python).

Broadly speaking, there are three ways to use the Python interpreter.

1. **Interactive:** Launch the interpreter, and type in Python commands. Each command is executed as soon as it’s entered.

2. **Script:** Save your code in a file, and tell the interpreter to read it.

3. **Integrated Development Environment:** combines the above two strategies: edit your files and execute them in the same window.
Example program Hello World: three methods

Method 1. Interactive

```bash
$ python
>>> print("hello world")
>>> exit()
```

Method 2. Script

File ‘hello.py’ contains:

```python
print("hello world")
```

```bash
$ python hello.py
```

Method 3. IDE (Jupyter notebook)

Click of a button!
Example program Hello World: ...again

Website replit.com offers all three methods together in a single window.

Method 1. Interactive
Method 2. Script
Method 3. IDE (click!)
Assignment “Hello World”

Go to google classroom assignment “Hello World”

Tasks:
- Create text cells, code cells
- Use the `print` statement
- Use the `for` statement with the `print` statement
- Use the `import` statement
- Save and clean your work
- Connect to Google drive
Discussion

What was that `for` statement?

How would you `print` 100 times? Copy and paste?

At what point is it easier to use the `for` statement?
Break time reminder slide

break 10 minutes
Lesson 2
Variables

The most important concept in programming.
Lesson Learning Objectives

- Understand what Variables are in Python
- Use the assignment operator
- Inspect variables with print()
Lesson Primer

Computers are able to store data in memory (a hardware component). This data is stored in the form of binary bits (0s and 1s) and each location in memory has an address which is also in the form of binary bits.

A programming language allows the programmer to refer to the data by a *label* and think abstractly. This is called a Variable. It is closely related to the math concept with the same name.
Return to Google classroom

Don’t forget to **turn in** your work on Lesson 1 Hello world!

Next assignment: Lesson 2 - Your first variables
Assignment “Your first variables”

Go to google classroom assignment “Your first Variables”

Tasks:
- Read the examples and try them to learn variables
- Exercises: create variables and inspect them with print
- Exercises: fix some bugs in code
Note about The Quiz (Homework)

Lesson 2 has a Quiz! Locate Assignment “Lesson 2 Quiz”.

Some Lessons will have one or more quizzes. These are homework, so people working at home get instant feedback on their learning.
Lesson 3
Files (a preview)

A quick preview of working with Files in Python
Lesson Learning Objectives

- Work with files in Google Colab
- Read and write a simple file
Lesson Primer

- Python variables live in Memory which is **temporary**
- Google Colab’s local file system is also **temporary**
- Files in Google Drive are **permanent**
Assignment “Text files”

Go to google classroom assignment “Text Files”

Tasks: Observe a demonstration of:
● Create a text file by hand
● Read in the text file
● Write out a text file
● Write out a text file to Google Drive

Completion of this will be homework.
Lesson 4
Data Types

Handling different types of data appropriately
Lesson Learning Objectives

- Understand how Python handles different types of data
- Inspect variables with type()
- Practice using common number and text variable types
  - Integer
  - Float
  - String
  - Boolean
- Learn how to use type-specific functions
- Practice an advanced variable type
  - Datetime
Lesson Primer

Different kinds of data are treated differently in Python. For example, you can do math with numbers, but you cannot do math with text.

Python keeps track of which kind of data is which so that if you accidentally tell it to do something nonsensical, it will stop and tell you that you made a mistake.

Each variable comes with an extra bit of information called “type”.
Assignment “Common Variable Types”

Go to google classroom assignment “Common Variable Types”

Tasks:
- Read the examples and try them
- Exercise: create variables of different types
- Exercise: convert from one type to another
- Exercise: print data in formatted strings
- Bonus: investigate memory usage
Break time reminder slide

10 minutes break
Assignment “How to use functions”

Go to google classroom assignment “How to use functions”

Tasks:
● Read the examples and try them
● Exercises: figure out how some functions work
Assignment “Datetime”

Go to google classroom assignment “Datetime”

Tasks:
● Read the examples and try them
● Exercises: convert between text and datetime format
● Exercises: compute using dates and time intervals
A Note about Homework

Lesson 4 has another assignment that is homework.

“The droid”

This is a fun activity which provides more practice with data types and functions.
Lesson 5
User Input

Interacting with an already-running program
Lesson Learning Objectives

- Use multiple methods to get user input
  - Python built-in input() function
  - Jupyter built-in Forms feature
- Process user input data type and length
- Practice designing reusable code
Lesson Primer

- Code in general is predictable, so a program executed more than once will usually result in the same outcome.

- To make a program more versatile, it helps to let the user make changes to it. The user shouldn’t be allowed to inject code (that would just be a new program) but they could inject data.

- Separation of code and data makes code reusable.
Multiple Homework Assignments

This lesson will not be covered in class; the assignments are homework.

- Assignment “User Input”
  - Lecture and Exercises
- Assignment “Story Generator”
  - Fun Activity
- Assignment “Lesson 5 Quiz”
Lunch break reminder slide

Lunch break 12 pm - 1 pm

Continue to Lesson 6 when we return