

Description

This exercise is designed to give you exposure to the Python programming environment. In general, if you finish with the lab early, you are free to leave or start working on the next lab or programming assignment. Feel free to get help from other students as well as from your Instructor/TA on the lab assignment.

Note! Although you can help each other on the lab assignments, make sure that you submit work that's entirely yours. For your programming assignments, while discussing them at the conceptual level with others is ok, working together on the same solution is not. Your only source of outside help on the assignments will be your instructor, TA, and PAC consultants. Even a single case of collaboration on coding an assignment may result in an F (for the course).

Setting Up Your Machine

Anaconda Distribution of Python

Anaconda is a freemium, open source distribution of the Python and R programming languages for large-scale data processing, predictive analytics, and scientific computing, that aims to simplify package management and deployment. Its package management system is conda.

The installation will set up an IDE (Integrated Development Environment) on your computer for working on Python programs. The Anaconda distribution is particularly useful since it comes with numerical, plotting and scientific packages which will be used for your course.

Installation, File Organization, IDE environment, Program creation

1. To install Anaconda Python, go to the website: <https://www.anaconda.com/download/> and download the version that's relevant for your platform.
2. Follow instructions to install the 3.x version of Python (on some platforms a higher version may be available which is not recommended at this time since it may include features that are slightly different from version 3.x). The latest version of Python is likely to be 3.7 at this time for most platforms.
3. Locate the Spyder Development environment after installation and create a short-cut to access it easily.
4. Open Spyder and get familiar with the development environment
5. Set up a directory structure on your computer to organize your files for this course. I suggest creating a folder called CS410P somewhere. Underneath this folder create two sub-folders: one for Labs and one for Assignments. For this lab you will create a folder under Labs called 0L – the idea is that every time you start a Lab or an Assignment you go to that particular folder and create a sub-folder with the name of the Lab/Assignment.
6. Create a very basic Python program to display some information about yourself (including why you are taking this course and how much exposure to programming you've had). Use multiple `print()` statements (see screenshots below to see how the print statement is used). Also try to use the newline character (`\n`) for printing to continue on to the next line. Before you run the program you will need to save it to a file. Make sure you save it under the 0L folder under Labs and choose a name that's meaningful to the program. Spyder will use the default extension of `.py` for all your Python files. If you want to see a sample Python program, look in the Public folder on the course web site under the 0L folder (best viewed in Chrome, in some browsers you may see all the lines appearing together).

7. Run your program and see if it produces the desired results. You can edit, save and run the program any number of times until you are happy with what it's displaying.
8. Make sure you add some comments at the top of your program, at least in this case your name and the number of the lab (0L). Comments are documentation notes which are not used by the program but are useful in maintaining information about the programmer, program, changes, etc. Single line comments in Python start with the # character. If you want to have multiple lines of comments use a sequence of 3 double quotes like this (all the lines that are enclosed within the lines containing the 3 quotation marks are considered comments):

```
""" Arvind Narayan
    CS 410P
    Lab Assignment 0L
    """
```

9. To submit your program go to myCourses, navigate to CS 410P, find the lab 0L and upload the file that you created for this lab.
10. You can upload the file any number of times for submission, only your latest submission will be graded and all previous submissions will be lost.
11. If you're done with any lab early you can choose to leave or start working on the next lab or assignment if they are already posted.

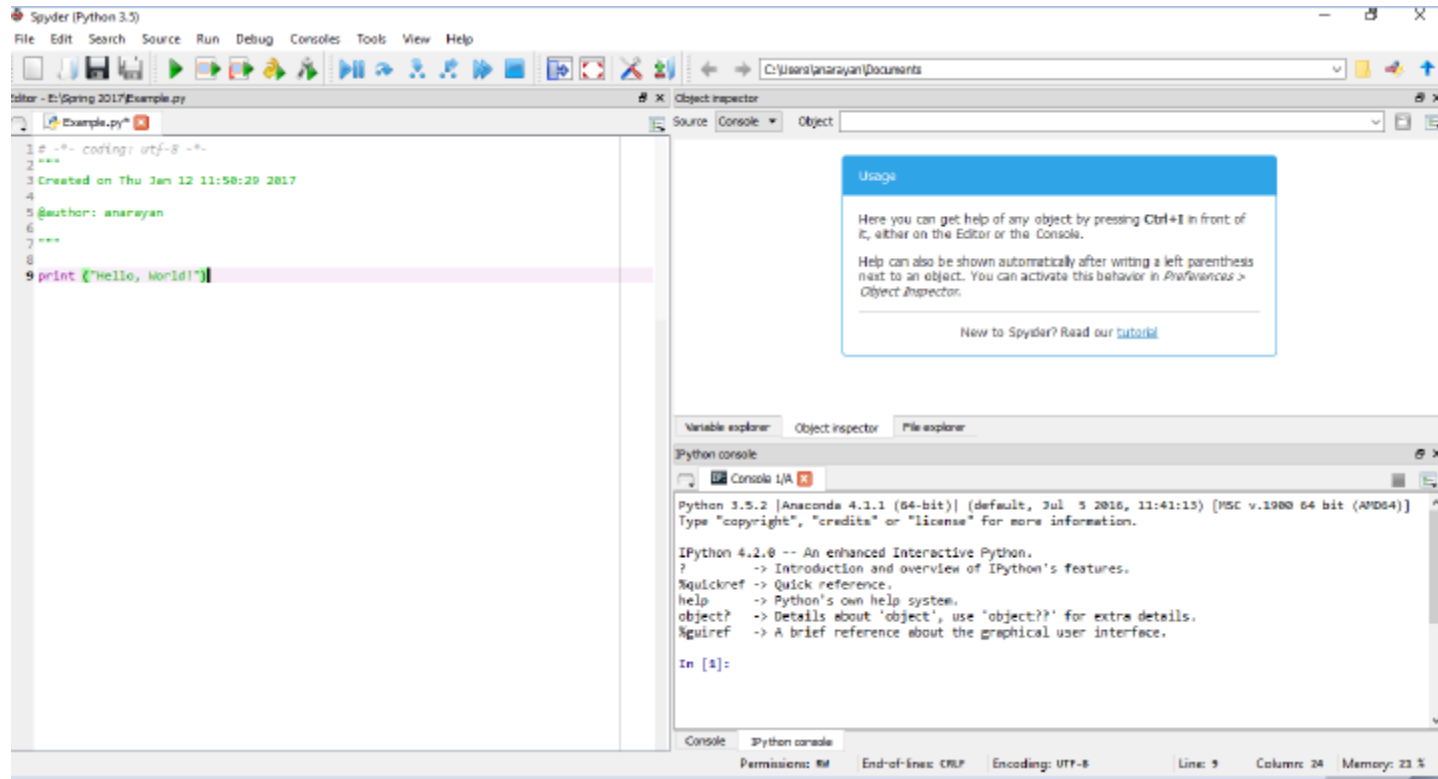
String Literals

- In Python, a string literal is a sequence of characters enclosed in single or double quotation marks
- ' ' and " " represent the **empty string**
- Use ' ' ' and " " " for multi-line paragraphs

```
print("I'm using a single quote on this line")
print("""This is a longer sentence but probably will all be on one line""")
print("This is a longer sentence, but will it actually go on another line or keep going on to the same line?")
print("What do you think will happen here?\nAre there two lines on the output?")
```

- Output from the above lines of code

```
I'm using a single quote on this line
This is a longer sentence but probably will all be on one line
This is a longer sentence, but will it actually go on another line or keep going on
the same line?
What do you think will happen here?
```



Grade Key

A	Name, comments	10
B	Program compiles and executes printing out some basic information about yourself	90