Lab #5: Intro to Python
CSE 3, Winter 2018
In this lab you will be introduced to programming in Python
(variables, simple strings, and for each loop)

Getting Started

The purpose of this tutorial-style self-paced lab is to help you learn the basics of Python. If at any point you get lost or stuck, go back to the beginning of the section and try again.

PyCharm is an Integrated Development Environment (IDE). It allows you to write and run your code all in one place. It also allows you to organize your work into different projects, each able to contain multiple files. The main components of an IDE are labeled below.

Note: In PyCharm you may need to click the Python Console button (circled above) in order to view the console and your output.
Part 1: Using PyCharm (Educational Edition)

Starting PyCharm

To open PyCharm, double click on the “pycharm” shortcut on the desktop. Be sure you are opening version 2017.3. PyCharm may take a minute or two to load.

Once PyCharm has loaded, a start menu will appear. Select Browse Courses and then select the Introduction to Python option. This will create a Python Project which is stored in a folder called PythonIntroduction.

*** Important ***

In order for your progress on this tutorial to be saved, you must first change where your work is saved. To do this, first go to PyCharm and go to File > Close Project.

Then, open your Libraries folder. Select Computer on the left hand side and then UCSD (C:).
Then go to **Users** and find the **folder with your username**. Your file path should look like this, where your user name is in the red box.

![Folder Path](image)

From there, open the folder named **PycharmProjects**. Inside this folder you should see another folder named **PythonIntroduction**.

Now, **open a new Libraries window** and navigate to your CSE3 folder. Once inside your CSE3 folder, drag the entire **PythonIntroduction** into your CSE3 folder. While the folder is copying over you should see a pop-up like this:

![Copy Pop-up](image)

Once this has finished, you have successfully saved the files in the proper location. To verify, open your CSE3 folder > PythonIntroduction and you should see a list of 10 lessons as well as a few other files.

Now, go back to PyCharm and **Select the Open option**. We are going to open the files that we just saved into our CSE3 folder. Be sure to select **H: \**, find your CSE3 folder, and select **PythonIntroduction**, then click **OK**
Part 2: Beginning the Tutorial

Once PyCharm has finished loading, your screen should look similar to this. Again, if you do not see the console, click on the Python Console button at the bottom left of the PyCharm IDE.

The first part of the tutorial you will be completing is the Introduction. To begin the first task, either double click on the Introduction option in the Project Explorer, or the small arrow next to it. Then, to select the first task, double click on the Our first program option.

Each task will include a small description of the task as well as instructions to either write code or fill in a small portion of it. For this task, replace the text in the blue box with your name.

Once you have filled in the code, click the check mark button found above the text for that lesson to compile and run your code.
If you have successfully completed the task, you will see a Congratulations message, your output will be displayed in the console, the task name will be in green, and your number of tasks completed will increase by one. To continue on to the next task, either double click on the task in the Project Explorer menu, or click on the next arrow button (circled above).

If you have not successfully completed the task, the task name will turn red and the console will give you a hint about what is wrong with your code.

You are able to attempt each task as many times as you’d like, so you can start the task over using the refresh button 🔄. If you are still having trouble completing a task, you can click on the lightbulb button to receive a hint.

**Part 3: Completing the Tutorial**

For this lab you are required to complete all the tasks in the Introduction, and Variables section and the For loop and For loop using string tasks in Loop (you don’t need to do While loop, Break keyword, and Continue Keyword). These are the minimum requirements to receive full credit for this assignment, but feel free to complete more if you wish.
****HINTS****

❖ The modulus operator (%) is used to calculate the **remainder** of a division.
   - 6/3 will give you 2
   - 6%3 will give you 0. Do you see why?
❖ Look at the code that is already in the task and take your time to try to figure out what it does. Sometimes that gives you a hint on what to do.
❖ In the For loop task, you can iterate over the elements of a list by using a for each loop and the list’s name
   - Example: to print all elements in list `letters = ["a","b","c"]` you do:

```
for letter in letters:
    print(letter)
```

Once you have completed these sections, you should be able to answer/explain the following:
- What are the two types of numbers in python and what is the difference between them?
- What does the modulus operator (%) calculate?
- What is the difference between `=` and `==`?
- How do you print all the elements of a list?

Your progress should look (at least) as follows:
Part 4: Writing Your First Python Program

Now that you have learned a little bit more about how to code in Python, it’s time to write your first program!

Before we get started, create a Lab5 folder in your CSE3 folder. This is where we’ll be saving your new program.

To start a new file in PyCharm, click on File > New Project… at the top left hand side of your screen.

We want to save this into our Lab5 directory, so to change the location, click on the “…” button to the right of the Location box and the file path. From there you will be able to navigate to your CSE3 folder and then into your Lab5 folder.

Also be sure to have Pure Python selected on the left hand side.

When you have finished your file path should look similar to this:

** Make sure that you have either the New Virtualenv environment or the Python 3.5 Interpreter selected as in the picture!!

- Everytime you open up PyCharm make sure you are using Python 3.5 by going to File > Settings > Project:Your_Project_Name > Project Interpreter
- If you do not see the Python 3.5 Interpreter, use another computer**
Click the **Create** button

Once you have created your new project, we must create a new file inside of your project. To do so, right click on your Lab5 directory in your Project Explorer and select New Python File.

![Image of creating a new Python file](image)

Name your file HelloWorld and click OK. Make sure the file type is “Python file”.

![Image of naming file](image)

Now that we have our file created it’s time to get coding! Because this is your first program, we will provide all of the code, but it is your job to understand what is happening and why. Type the following into your Python file:

```python
# < Your name > < Your lab day and time >
# < Today's date > Lab5

print("Hello World!")  #Print Hello World!

my_name = input("What is your name? ") # Ask the user for their name and store it in the my_name variable

print("Hello " + my_name) # Print Hello and the user's name
```

This piece of code is prompting the user to input their name and then it prints a message with the user’s name. Be sure to replace anything in < > with your own information. Once you have completed typing this in, run your program by clicking the green arrow found next to the line 1 of your code, or right clicking on your file name and selecting Run.
You should see the following in the console:

```
Hello World!
What is your name?
```

Now, **type in your name** (the text you type will be green) and **click enter**. Your output in the console will look similar to this:

```
Hello World!
What is your name? Sierra
Hello Sierra!
```

If your program does not compile the first time you run it, don’t worry! It is normal to have to debug your code after you write it. Look at the errors the console displays and try to fix it yourself, and then go back and try to spot the difference between your code and the given code.

Next, we are going to ask the user what is their year in college and major, and then we are going to display a message containing the year and major that the user inputed. Write the following under what you wrote before:

```python
print("Hello World!")                      # Print Hello World!
my_name = input("What is your name? ")    # Ask the user for their name and store it in the my_name variable
print("Hello " + my_name + ")")            # Print Hello and the user’s name

my_grad_year = input("What year are you graduating in? ")  # Ask for their graduation year
my_major = input("What is your major? ")     # Ask for major

print("You will graduate in " + my_grad_year + " with a " + my_major + " degree.")
```

Run it. Your output should look similar to the following:

```
Hello World!
What is your name? Sierra
Hello Sierra!
What year are you graduating in? 2019
What is your major? Computer Engineering
You will graduate in 2019 with a Computer Engineering degree.
```

You can use print() to print an empty line before we write code for next section.

Finally, we are going to do a list that contains three courses that you want to take before you graduate. Then, we are going to iterate through this list with a for each loop (remember those from the tutorial?).

Additionally, we are going to create a variable called `num_courses` that keeps track of the number of courses in your list. In the example below, there are 3 courses in my list (“course1”, “course2”, and
“course3”). In order to do this, we initialize the value of \texttt{num\_courses} to 0. Then, every time we iterate through the for loop we increment the value of \texttt{num\_courses} by 1. By doing this, we are counting the number of courses stored in the list called \texttt{courses}.

Type in the following code at the bottom of your file:

```python
print("Three courses I want to take before I graduate are: ")
courses = ["course1", "course2", "course3"] # Create a list of courses
num_courses = 0 # Counts how many courses are in the list
for course in courses:
    num_courses = num_courses + 1 # Increment variable for every course in the list
    print(course) # Print the name of each course in the list
print("I want to take these \, \text{num\_courses}, \, \text{ courses before I graduate.}")
```

Replace course 1, course 2, and course 3 with the courses that you want to take before graduation and run your program.

Once your program compiles and runs, you should be able to enter in your own information into the console and your program should print it out. Your output should look similar to this now:

```
Hello World!
What is your name? Sierra
Hello Sierra!
What year are you graduating in? 2019
What is your major? Computer Engineering
You will graduate in 2019 with a Computer Engineering degree.

Three courses I want to take before I graduate are:
Computer Networking
Web Client Language
Human Computer Interaction
I want to take these 3 courses before I graduate.

Process finished with exit code 0
```

**Note:** Remember the \textbf{green text} is entered by the user while the program is running (You type this in!).

As a bonus try adding in another question for the user to answer and printing that information out at the end of your program!

***Important***: This lab does not need to be posted online. \textbf{DO NOT} drag the entire python project to your CSE 3 folder on the web server; otherwise it will take very long for it to be uploaded.
Check-off Instructions:

1. Open PyCharm > Open File > PythonIntroduction
2. Make sure progress bar can be seen
3. Be able to answer questions about Python
4. Run your HelloWorld program with your tutor’s information

Conditional Statements - Optional (but recommended)
We will be doing Conditional Expressions in Python later on, so consider doing the Conditional Expressions section from the PyCharm tutorial as well or referencing it later. If you decide to do it, here is a hint that you might find useful:

❖ In the Boolean expressions section, for the third task titled “If Statement” you are asked to check if a list is empty. **An empty list has a length of zero and you can find the length of a list using len(ListName).**
   How could you use an if statement to check if the given list is empty?

If you have questions, remember that you can visit Homework Hours. A tutor will help you out!