**Outline**

Access the Python Development environment and follow the tutorial to gain an initial exposure to a programming language. Begin to develop an familiarity with basic programming concepts.

**Objectives**

* Use correct terminology to describe programming concepts;
* Describe the types of data that computers can process and store (e.g., numbers, text);
* Explain the difference between constants and variables used in programming;
* Use variables, expressions, and assignment statements to store and manipulate numbers and text in a program

**Materials**

* Python3 Development Environment at: //repl.it/
* Python Tutorial at: <http://www.letslearnpython.com/learn/>

**Accessing the Python3 Web IDE Environment**

Accessing the IDE

* Go to: <https://repl.it/>
* Select Python3
* Sign-up / Create an account
* Make sure you can remember your account information for the rest of the course.

Using the IDE

* Use the black area like a calculator to try simple statements or commands
* Use the white area to create programs with multiple statements

**Accessing the Tutorial**

Accessing the Tutorial

* Go to: <http://www.letslearnpython.com/learn/>
* Read up to “Lesson 3: Math”

**Level 1: Basic Math & Strings**

Access the Tutorial and start at “Lesson 3: Math”.

Questions

1. Complete “Lesson 3: Math – Math Basics” by typing the sample commands in the black area of the IDE.
   1. Create your own expression using 5 “+” and “-“ operators.

5+9-8+9-4

* 1. List your expression and the result below.

Result was 11.

1. Complete “Lesson 3: Math – More Operators” by typing the sample commands in the black area of the IDE.
   1. Create your own expression using 5 “\*” and “/” operators.

5\*58/7\*9/8

* 1. List your expression and the result below.

Result was 46.60714285714286

1. Complete “Lesson 3: Math – More Division” by typing the sample commands in the black area of the IDE.
   1. Create one division expression that gives a whole number answer

56/8

* 1. And one division expression that gives a decimal number answer.

7/8

* 1. List your expressions and the results below.

56/8=7

7/8=0.875

1. Complete “Lesson 3: Math – Floats” by typing the sample commands in the black area of the IDE.
   1. Use the “round()” function for the expressions you created in question #3 above.

round(56/8)

round(7/8)

* 1. List your “round()” expressions and the results they return below.

Round(56/8)=7

Round(7/8)=1

1. Read through “Lesson 3: Math – Comparison Operators”.
   1. Why do you think Equals is “==” instead of “=”?

Equal is “==” because the actual equal sign may be used for other programs/

* 1. What does “=” mean?

The equal sign means that a word or letter is equal to a certain number or a certain number that is inputted.

1. Complete “Lesson 3: Math – Practice” and “Lesson 3: Math – Practice Answers” by typing the sample commands in the black area of the IDE.
   1. Create an expression using 5 different operators that returns a “True” result

9=7+2

10>=2

9<10

2==2

96>=48+48

* 1. And an expression using 5 different operators that returns a “False” result.

6>9

0>100-50

40+40>50+50

5!=5

6!=6

* 1. List your expressions and the results returned below.

9==7+2 = True

10>=2 = True

9<10 = True

2==2 = True

96>=48+48 = True

6>9 = False

0>100-50 = False

40+40>50+50 = False

5!=5 = False

6!=6 = False

1. Complete “Lesson 4: Strings – Strings” and “Lesson 4: Strings – Examples” by typing the sample commands in the black area of the IDE.
   1. Explain why typing “apple” works and why typing apple without quotes gives an error.

The quotes are an input that pretty much tells the computer to repeat it again while using the quotes. If the quotes aren’t used, it is an invalid entry that does nothing.

* 1. Also explain why “2 + 5” does not equal 7.

The program is repeating the equation. It would only give the answer without the quotes.

1. Complete “Lesson 4: Strings – Operators” by typing the sample commands in the black area of the IDE.
   1. Explain why typing “appl” + “e” works and why typing “apple” - “e” gives an error.

It is because you can’t subtract an “e” from the word, as it can only add letters or words.

* 1. Also explain why “Hello” \* 10 works but why “Hello” / 10 does work.

It is because you can’t divide the word. When “’Hello” \* 10” is used, the word output is the word ten times.

1. Complete “Lesson 4: Strings – Indexes” by typing the sample commands in the black area of the IDE.
   1. List the letters in your first name and the index for each letter in your first name.

H A R M A N G O R A Y A

1. Complete “Lesson 4: Strings – Indexes Examples” by typing the sample commands in the black area of the IDE.
   1. Explain why print(“Hello!”[4]) does not print “l”.

The index is 0.

* 1. What does print(“Hay, Bob!”[4]) print? For a hint try print(“Hay, Bob!”[3]) and print(“Hay, Bob!”[5])

It prints a space. The program counts spaces as characters. The 4th character in this case was a space.

1. Complete “Lesson 4: Strings – Rules” by typing the sample commands in the black area of the IDE.
   1. Explain why print(“Hello!”[7]) gives an error.

There is no 7th letter. In the input.

**Level 2: Booleans & Variables**

Access the Tutorial and start at “Lesson 5: Variables”

Questions

1. Complete “Lesson 5: Variables – Save a Value” by typing the sample commands in the black area of the IDE.
   1. What do you get if you type puppies / 3?

You get the output number 81 three times.

* 1. Why doesn’t typing kittens / 3 work?  
     Typing kittens three times won’t work as there is no input for kittens. For kittens to work, you would have to put kitten=(an number or equation) for “kittens".

1. Complete “Lesson 5: Variables – Assign a New Value” by typing the sample commands in the black area of the IDE.
   1. Explain how the following sequence of commands works:
      * puppies = 36
      * puppies = puppies / 6
      * puppies  
        The “puppies” value was first set to 36. Then the value was changed. The puppies’ value was set to the first value divided by 6. Then when the “puppies” was entered, it gives the value of 36/6, which is 6.
2. Read through “Lesson 5: Variables – Rules”.
3. Complete “Lesson 5: Variables – Math Operators” by typing the sample commands in the black area of the IDE.
   1. Explain what happens for following sequence of commands:
      * colour = “red”
      * puppies = 36
      * colour + puppies

For the first command, the word “red” will be responded with when colour is typed. The second command would set the value of puppies to 36. The third command would add the values of both words together if they were inputted before.

1. Complete “Lesson 5: Variables – String Operators” by typing the sample commands in the black area of the IDE.
   1. Explain why the following commands give different results:
      * Color + day \* fishes
      * ( Color + day ) \* fishes  
        The word that is inputted for “day” would be written out three times and the word for the word “color” would be put in front of it. The second one would put the words for color and day together and multiply it three times.
2. Complete “Lesson 5: Variables – Indexes” by typing the sample commands in the black area of the IDE.
   1. What is the index of ‘r’ in “watermelon”?

The index of ‘r’ in ‘watermelon is 5.

* 1. Write an expression using mynumber to return ‘r’

mynumber=7

fruit[mynumber-2]

1. Complete “Lesson 5: Variables – Assignments or Comparisons” by typing the sample commands in the black area of the IDE.
   1. What is the difference between “=” and “==”?

“=” is used to assign a value. “==” is used to compare values.

* 1. Create your own mnemonic to remember this difference.  
     One is to assign and two is to compare.

1. Complete “Lesson 6: Errors – Examples” by typing the sample commands in the black area of the IDE.
   1. What doesn’t “friend” + 5 work?

You can’t add 5 to a word in programming.

* 1. What is the difference between int and str?

Int is an integer and str is a string.

1. Read through “Lesson 6: Errors – Parts of an Error Message”.
   1. Is “friend” + 5 an example of:
      1. A Syntax Error?
      2. A Runtime Error?
      3. A Logic Error?

It is a TypeError.

1. Read through “Lesson 6: Errors – Fixing Errors”.
   1. Use the ‘print’ command to print your first name and last name.

print(“Harman” + “Goraya”)

1. Complete “Lesson 7: Booleans – Types of Data” by typing the sample commands in the black area of the IDE.
   1. What is the value of: type(“True”)
   2. What is the value of: type( True )
   3. Why is the result different?

1. Complete “Lesson 7: Booleans – What Is A Boolean” by typing the sample commands in the black area of the IDE.
   1. Why do you think that having a Boolean data type is important in computer programming?

Having a Boolean data type is important in computer programming because it tells you whether the statement is true or false. It helps you with math.

1. Complete “Lesson 7: Booleans – Trying Out Booleans” by typing the sample commands in the black area of the IDE.
   1. Why do you think that there is no Maybe” Boolean data value in computer programming?

There is no “Maybe” as there is only true or false in Math. For example, saying 7 is equal to 9 is either true or false, which is false in this case.

**Level 3: Lists & Logic**

Access the Tutorial and start at “Lesson 7: Booleans”

Questions

1. Complete “Lesson 7: Booleans – AND Comparisons” by typing the sample commands in the black area of the IDE.
   1. Try the following Python statements and record the results.
      1. True and True = True
      2. True and False = False
      3. False and True = False
      4. False and False = False
   2. Explain if there are any other combinations of True / False.

No there are no other combinations of True/False.

* 1. Explain how the AND operator is similar to a math operator and how it is different.

It is similar as it tells you what is correct and what is false. It is different as it needs the “and”.

1. Complete “Lesson 7: Booleans – OR Comparisons” by typing the sample commands in the black area of the IDE.
   1. Try the following Python statements and record the results.
      1. True or True = True
      2. True or False = True
      3. False or True = True
      4. False or False = False
   2. Explain how the OR operator is similar to the AND operator and how it is different.
2. Complete “Lesson 7: Booleans – NOT Comparisons” by typing the sample commands in the black area of the IDE.
   1. Try the following Python statements and record the results.
      1. not (True or True)
      2. not (True or False)
      3. not (False or True)
      4. not (False or False)
   2. Explain how the combination of the NOT & OR operators is similar to the AND operator by itself and how it is different.
3. Complete “Lesson 7: Booleans – Expressions” by typing the sample commands in the black area of the IDE.
   1. Explain why the following two Python statements give different results.
      1. not (True or True)
      2. not True or True
   2. Explain why the following two Python statements give the same results.
      1. not (True and True)
      2. not True and True
4. Complete “Lesson 7: Booleans – Practice” by typing the sample commands in the black area of the IDE.
   1. Create three more practice expressions similar to those in the tutorial.
   2. Provide the results for your practice expressions
5. Complete “Lesson 8: Lists – A Collection of Objects” by typing the sample commands in the black area of the IDE.
   1. Create a list of your favorite sports teams.
   2. Assign your list to a variable.
   3. Confirm that your variable and your list are the same.
6. Complete “Lesson 8: Lists – List Indexes” by typing the sample commands in the black area of the IDE.
   1. What is the list index of the last team in your list of favorite sports teams.
   2. In the tutorial, the error produced by typing “fruit[3]” is an example of:
      1. A Syntax Error?
      2. A Runtime Error?
      3. A Logic Error?
7. Complete “Lesson 8: Lists – Practice” and “Lesson 8: Lists – Practice Answers” by typing the sample commands in the black area of the IDE.

NOTE: Starting with Lesson 9 you should use the WHITE area of the IDE for entering example code with multiple statements.

1. Complete “Lesson 9: Logic – Making Decisions” by typing the sample commands in the white area of the IDE.
   1. Modify the tutorial code to print “Hi Alfred!” based on a decision using numbers

myname=”Harman”

if myname==”Harman”:

print(“Hi Harman!”)

1. Complete “Lesson 9: Logic – Adding A Choice” by typing the sample commands in the white area of the IDE.
   1. Modify the tutorial code to print your first name or your last name based on a choice (using “else”).

If myname==”Harman”:

print(“Hi Harman!”)

else

print(“Incorrect”)

1. Complete “Lesson 9: Logic – Adding Many Choices” and “Lesson 9: Logic – Practice” by typing the sample commands in the white area of the IDE.
   1. Modify the tutorial code and “elif” statements to make a choice using at least 4 of your friends names.

If myname==”Harman”

Print(“Hi Harman”)

elif myname==”Harjap”

print(“Hi Harjap”)

elif myname==”Amrit”

print(“Hi Amrit”)

elif myname==”Arnav”

print(“Hi Arnav”)

elif myname==”Manraj”

print(“Hi Manraj”)

else:

print(“Who are you?!?”)