**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.
   2. List your expression and the result below.

5 + 5

10

5 - 5

0

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.
   2. List your expression and the result below.

5 \* 5

25

5 / 5

1.0

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

25 / 5

5.0

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

10 / 4

2.5

* 1. List your expressions and the results below.

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(10/3)

3

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

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

I think it is “==” because “=” is already used to do calculations

* 1. What does “=” mean?

“=” functions as a calculator for the numbers.

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

3 + 6 / 3 \* 4 - 1 >= 10

True

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

5 + 2 - 3 / 2 \* 4 >= 20

False

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

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.

Python only reads strings inside quotation marks.

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

It does not equal 7 because the quotation marks surrounding it make it a quote.

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.

The operand - was not supported.

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

“Hello” / 10 is an unsupported operand.

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 K I R A T

0 1 2 3 4 5 6 7

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”.

It does not print “l” because the index starts at 0.

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

“Hay, Bob!”[4] prints a blank space because [5] prints B and [3] is supposed to print the comma.

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 are not 8 letters in “Hello!”

**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?

12.0

* 1. Why doesn’t typing kittens / 3 work?  
     There was no value assigned to 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

Assigning the value 36 to puppies.

* + - puppies = puppies / 6

assigning the value of the puppies as the current value divided by 6

* + - puppies  
      Showing the value assigned to puppies.

1. Read through “Lesson 5: Variables – Rules”.
2. 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
3. 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 first multiplies day and fishes and then adds yellow giving the result ‘yellowMondayMondayMonday’. The second adds color + day and then multiplies that by fishes giving the result ‘yellowMondayyellowMondayyellowMonday’
4. 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”?

4 because it is the fifth letter in watermelon and the index starts at 0

* 1. Write an expression using mynumber to return ‘r’  
     mynumber = 2

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 “==”?

“=” assigns a value to a variable while “==” compares two values.

* 1. Create your own mnemonic to remember this difference.  
     “=” is shorter because it’s “this equals that”. “==” is two signs because “is this thing equal to this thing?” is a longer sentence.

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?

One is an integer and one is a string. These cannot be added together.

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

An integer is a number while a string is a series of letters.

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?

“friend” + 5 is an example of a syntax error.

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

Print(“Harkirat”, “Mann”)

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”)

“True” is a string.

* 1. What is the value of: type( True )

True is class ‘bool’

* 1. Why is the result different?

Python thinks “True” is a variable

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?

Booleans are important because they say if something is true or false.

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?

Booleans are either True or False with no inbetween.

**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

* + 1. True and False

False

* + 1. False and True

False

* + 1. False and False

True

* 1. Explain if there are any other combinations of True / False.

No

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

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
      2. True or False
      3. False or True
      4. False or 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.

The OR operator is correct even when one of the things being compared is wrong while AND only works when both are correct. NOT reverses the answer to a comparison.

1. 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

not (True or True) is incorrect because of the not command reversing the answer to false. not True or True is correct because the not command reversing the answer to True.

* 1. Explain why the following two Python statements give the same results.
     1. not (True and True)
     2. not True and True

Both are normally correct but both are incorrect because of the not operator before it.

1. 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.

5 == 5 and 3 == 3, not 4 == 6 and 3 == 3, not False and True

* 1. Provide the results for your practice expressions

True, True, True

1. 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.

[“Raptors”, “Blue Jays”, “Leafs”]

* 1. Assign your list to a variable.

Sports = [“Raptors”, “Blue Jays”, “Leafs”]

* 1. Confirm that your variable and your list are the same.

Sports

['Raptors', 'Blue Jays', 'Leafs']

1. 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.

Sports[2]

* 1. 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?

Runtime error.

1. Complete “Lesson 8: Lists – Practice” and “Lesson 8: Lists – Practice Answers” by typing the sample commands in the black area of the IDE.

colors = ["blue", "red","white"]

print(colors[0])

blue

print(colors[1])

red

print(colors[2])

white

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

if 6 == 6:

print("Hi, Alfred")

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 6 == 6:

print("Harkirat")

else:

print("Mann")

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 6 == 6:

print("Harkirat")

elif 6 == 7:

print("Hi, Gurtaj")

elif 6 == 5:

print("Hi, Karman")

elif 6 == 4:

print("Hi, Ivan")

else:

print("Hi, Ziyaad")