**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.

10+5+6+4+8+8-4-7-6-11-1

12

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.

10\*2\*4\*6\*5\*5/64/2/2/2/2

11.71875

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
   2. And one division expression that gives a decimal number answer.
   3. List your expressions and the results below.

22/11

2.0

9/8

1.125

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

round(9/8)

1

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

== is asking the question if the number is equal to the variable and = is used to assign a variable with a number or to solve an assignment.

* 1. What does “=” mean?

= means to assign a variable with a number or to solve an assignment.

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
   2. And an expression using 5 different operators that returns a “False” result.
   3. List your expressions and the results returned below.

answer==2\*4\*3/8+10-round(9/8)

True

answer==5\*7\*6/12+4-round(17/2)

False

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

When you add quotes to data that data becomes a string and without the quotes then that data becomes a variable.

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

This does not equal 7 because the data 2+5 in in quotes meaning the data is a string not an assignment. If there were no quotes than the answer would have been 7.

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.

“appl” + “e” works because the 2 strings are not added they are concatenated, meaning putting strings side by side. “apple” - “e” does not work because python cannot find a string in another string and remove it. The string “e” in not in “apple” so “e” cannot be removed from “apple”.

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

“Hello” \* 10 works because the same string is concatenated 10 times, meaning the string is side by side 10 times. “Hello” / 10 does not work because the string is not being concatenated, it is being divided. Strings can only be concatenated.

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.

“j”+”a”+”s”+”k”+”a”+”r”+”a”+”n”

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 “l” is in the number 3 position not the number 4 position. Strings are organize in numbers starting from 0 and 1, 2,3,4etc. In “Hello!”, H is #0, e is #1, l is #2, l is #3, o is #4 and ! is 5.

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

It prints nothing because [4] is a space. A space means nothing is there.

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.

It gives an error because in the [7] there in nothing in that spot. The string “Hello!” only has letters until [5], so ant thing above that will give an error because anything above that does not exist.

**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 12.

* 1. Why doesn’t typing kittens / 3 work?  
     kittens does not work because it is not assigned a number. Kittens in not a variable so it cannot be divided by 3. Puppies was assigned the number 36 so it could be divided, and the answer will be 12.

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

First the variable puppies is assigned the number 36. Then puppies is assigned another variable which is puppies divides by 6. 36 divided by 6 is 6. So the new value of puppies will be 6. The final answer in this sequence would be 6 because puppies equals 6.

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:
      * color = “red”
      * puppies = 36
      * color + puppies

First, the variable color is assigned the String “red” and the variable puppies in assigned an integer 36. A math operator is used to add the 2 variables. The result is an error because you cannot add a string to an integer.

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 following commands give different results because of Bedmas. In Color + day \* fishes, the first function is multiplication so day is multiplied by fishes, the result of that will be added to color to give the final result. In ( Color + day ) \* fishes, the first function is addition because Color + day is in brackets, the result of that will be multiplied by fishes to give the final result.

1. 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

* 1. Write an expression using mynumber to return ‘r’  
     fruit[mynumber+1]

‘r’

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 to a variable and == is used to for comparison if a variable is equal to or not.

* 1. Create your own mnemonic to remember this difference.  
     when assigning a value, we say "this equals that", the equals has an s at the end so s for single equal sign. When comparing we say, "is this thing equal to that thing?", the equals has no s so it is more that one which is 2 equal signs.

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?

It does not work because you are trying to add a string to an integer, which is not possible.

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

Int stands for Integer which is a number. Str stands for String which is a kind of data with a bunch of characters put together.

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?

This is a Runtime error.

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

Saini= “Saini”

Print(“Harjap”, Saini)

Harjap Saini

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”)

<class 'str'>

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

<class 'bool'>

* 1. Why is the result different?

The results are different because one of them has quotations around it, making the data a string. The other one is not a string, it is a Boolean, so the results are 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?

There are important because they are used to make decision in code.

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 because this would confuse the computer and give you an error.

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

There are no other combinations.

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

The and operator is like a math operator because it is a function with an answer. It is different because the answer is not an integer or a float.

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

* + 1. True or False

True

* + 1. False or True

True

* + 1. False or False

False

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

It is similar because the answer for True or True, and True and True is the same. The answer for False and false and False or False was also the same. All other combinations were different between OR and AND.

1. 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)

False

* + 1. not (True or False)

False

* + 1. not (False or True)

False

* + 1. not (False or False)

True

* 1. Explain how the combination of the NOT & OR operators is similar to the AND operator by itself and how it is different.

It is similar because there is only one True combination out of all the possible combination. It is different because NOT and OR is the opposite of AND, the only True combination is False or False and the only true combination for AND was True and True. Theses combinations are a complete opposite.

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

The results are different because on statements has brackets and the other does not. The one with the brackets means that the statement is the reversed form of OR. If the combination is False or False then the answer is True, if any other combination is anything else then the answer will be False. The statement with the bracket is True or True so the answer will be **False**. The statement without the brackets means that one of the Booleans is the opposite because there is a NOT. This means the statement goes from True or True to False or True because of the NOT. The answer to False or True is True. This means the answers are different. This is why the answers are different.

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

The answer for the first statement is false because the NOT means the statement in the brackets is opposite, True and True is True so the opposite of that is False. The second statement is also False because the NOT only applies to only one of the Booleans. The NOT makes the Boolean the opposite Boolean. The stamen goes from True and True to False and True. The answer to that will be false. Therefor the answer to both statements is the same.

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.
   2. Provide the results for your practice expressions

3\*4==2\*6 and 8+4==20-8

True

“day”==”days”

False

True or 12\*4==42\*56

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.
   2. Assign your list to a variable.
   3. Confirm that your variable and your list are the same.

Teams=[“Oilers”, “Leafs”]

Print(Teams)

[‘Oilers’, ‘Leafs’]

Type(Teams)

<class 'list'>

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.

[‘Oilers’, ‘Leafs’] The index is 1.

0 1

* 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?

This is a 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 = ["Green", "Red", "Blue"]

print(Colors[0])

Green

print(Colors[1])

Red

print(Colors[2])

Blue

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

age = 2\*3

Greeting="Hi Alfred!"

if age == 6:

print(Greeting)

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”).

myname="Harjap Saini"

if myname == "Harjap Saini":

print("Harjap")

else:

print("Saini")

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.

F1="Harman"

F2="Mokak"

F3="Tarnpreet"

F4="Amrit"

if F1=="Harman":

print('Hi Harman!')

elif F2=='Mohak':

print('Hi Mohak')

elif F3=='Tarnpreet':

print('Hi Tarnpreet')

elif F4=='Amrit':

print('Hi Amrit')

else:

print('Who are you')