**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.

1+2-3+4-5+6

=5

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\*5/5\*2\*3

=15

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.

A) 36/9=4

B) 65/7=9.285714285714286

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(65/7)=9

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

I think Equals is “==” instead of “=” because it symbolizes that the two numbers are equal in the sense that they are equivalent to each other whereas if only one = is but it can be easily mistaken as there maybe be < or > sign in front of it.

* 1. What does “=” mean?

“=” does not mean anything in this case.

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.

5+6\*2>5+8

=True

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.

If you want Python to read a string it must be inside quotes.

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

This is because by putting 2+5 in inside quotes we have asked Python to read 2+5 not solve.

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.

This is because we can add a letter to appl but it is difficult to subtract a letter to appl as it is not comprehendible.

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

You can multiply “Hello” by 10 meaning write “Hello” 10 times but it is not possible to divide “Hello” or any word by any number.

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” + “s” + “i” + “m” + “r” + “i” + “t”

=’Harsimrit’

012345678

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”.

Because according to the index “o” is 4. As “H” 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 as according to the index 4 is 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.

According to the index there is no “7”.

**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

* 1. Why doesn’t typing kittens / 3 work?

That’s because we haven’t put the value of 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

We have given the value of puppies as 36 but then we divide it by 6 leaving us with the answer of 6. This means that our final answer is puppies = 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:
      * colour = “red”
      * puppies = 36
      * colour + puppies

It gives an error because red and 36 can’t be added.

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

yellowMondayMondayMonday

* + - ( Color + day ) \* fishes

yellowMondayyellowMondayyellowMonday

We get two different answers because in the first command, it only asks to multiply day by fishes. So only the day meaning Monday will be multiplied by 3, but color meaning yellow will remain the same. In the second command, it asks to multiply color AND day by fishes. Therefore, both yellow and Monday are multiplied by 3.

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”?

The index in watermelon is 4.

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

Fruit [mynumber+1]

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 “==”?

“=” means that we are giving a value to a variable. “==” means we are asking if the statement is true/false.

* 1. Create your own mnemonic to remember this difference.

Once

Means

Value

Again

Means

Asking

True/False

OM VAMAT

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?

This is because the string “friend” is a word in which no integer can be added.

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

Int means integer and str means 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?

I think “friend”+5 is an example of a syntax error in which python does not understand what you are saying as they are two different thing: a string and a integer.

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

Print(“Harsimrit Sangha”)

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?

Str means a string which means a word . But bool means a true/false value.

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?

Boolean type data is important because it helps us to make decisions about what to do in our code. Like for example “If 15==15 is True, do this; if it is False, do something else instead.”

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?

I think there is no “Maybe” because a statement can either be true or false there is no in between. For example if the statement is “Anne has apples”, it can be either true or false, Anne can have the apples or not have them there is no maybe.

**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

True

* + 1. False and False

False

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

No there aren’t any other combinations of true/false because they can be put with themself and the other option.

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

Well I think that the AND operator is similar to a math operator because AND actually signifies how without one the other is not possible. But it isn’t technically a math operator because it does not add, subtract, multiply, or divide any of the two options.

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

False

* + 1. False or False

False

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

The OR operator is basically two options but saying this one or that one and preferably the first option is chosen whereas in AND operator the second option is usually chosen.

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.

The combination of the NOT & OR operators is basically the opposite of what the answer should give in case of the OR operator. It’s different because it's not comparing its asking the opposite of the answer to the statement.

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

In order for the not operator to work, the statement must be in (). The first one gives the answer of false and the second one gives the answer of true. This is because in the first statement its being asked to tell the opposite of (“true or true”) which is false. In the second statement it’s being asked to choose between “true” or “true” because for the not operator to initiate we need to put () after the not.

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

The first statement is asking the answer to the opposite of “true” or “true” which is false as the answer to “true” and “true” is true. So the opposite would be false. The second statement is saying that there are two options in which you can choose. According to what I’ve seen, the AND operator chooses the second option which in this case is “true”.

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.

1)”harsimrit” == “harsimrit”

2)”2513” and “1051850”

3)”fruit” or “fruits”

* 1. Provide the results for your practice expressions

1)True

2)1051850

3)’fruit’

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.

[“toronto raptors”, “blue jays”, “toronto maple leafs”]

* 1. Assign your list to a variable.

sports= [“toronto raptors”, “blue jays”, “toronto maple leafs”]

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

Type(sports)

<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.

3

* 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?

It’s a syntax 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.

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

code="1253"

if code=="1253":

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”).

code=input("Enter the code please.")

if code=="1253":

print("Hello Harsimrit")

else:

print("You are not Harsimrit")

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.

name=input("What is your name?")

if name=="Harsimrit":

print("Hi Harsimrit!")

elif name == "Avreeta":

print("Hey Avreeta!")

elif name=="Rajkanwar":

print("Hey Raju")

elif name=="Garima":

print("Hi princess")

else:

print("I don't know you, goodbye")