**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+2=9

False

5+9=14

True

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\*6=30

True

5\*10=60

False

5/5=2

False

50/5=10

True

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.

10/2

5.0

10/3

3.3333333333333335

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 (10/2)

5

round (10/3)

3

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

== is used to ask the computer if the numbers or anything on both of the sides means the same thing. The computer will answer by saying if it is “True” or “False” only if the comparison is types correctly.

* 1. What does “=” mean?

= is used to define a variable. An example of this is name = Jasjot.

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+2/2<=9-1\*1

True

5+6-2/2>10\*9-2

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.

Typing apple with quotation works because the syntax is correct and the computer recognizes it. Typing apple without quotations doesn’t work because the syntax is incorrect and it is not a string.

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

2+5 with quotations doesn’t equal 7 because we are concatenating, or putting 2 and 5 side by side.

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.

We can put together strings, but not subtract them.

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

Strings can be multiplied and repeated but they can’t be divided.

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 j o t – 0 1 2 3 4 5

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”.

“l” has an index of 2 and 3 while the index of 4 is “o”.

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

It prints an empty 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.

It gives an error because there is no character for index 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?

We get 12 because puppies is 36.

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

It doesn’t work because kittens is not defined as a variable.

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

Puppies is given a value of 36, then puppies is assigned a new value of puppies/6 which is 36/6. SO when we type puppies, the answer is 6.0.

1. Read through “Lesson 5: Variables – Rules”.

I have read through it and I understand it. Calculate once, keep the result to use later. Keep the name same, change the value.

1. 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

The result is that there is an error because we cannot add a string with an integer variable.

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

This gives different results because python follows the order of operations. Brackets change the order because color and day are added together and then multiplied to fishes. In the first one, day and fishes are multiplied first and then they are added with day.

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 is 4.

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

mynumber = 5

fruit = "watermelon"

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 “==”?

The difference is that “=” means it is something used to define a variable while “==” means that we asking the computer if both of the sides have the same value.

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

Jasjot = 16

School = 12

Jasjot == School

False

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 doesn’t work because we can’t combine an integer with a string.

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

Int means it is an integer while str means it 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 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("Jasjot" + " Benipal")

Jasjot Benipal

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”)

The value is that it is a string.

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

The value is that it is a Boolean.

* 1. Why is the result different?

The result is different because when we put quotations around True, it becomes a string while without quotations, python thinks of it as a Boolean.

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?

We use them a lot when we need to make decisions about what to do in our 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?

This is because Python only thinks of a Booleans as having a true or false value.

**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

False

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

There are no other possible combinations.

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

It is similar to a math operator because it compares and combines the two terms. It is different because in python, it is strictly used for true or false answers while the math operator is used for adding the two terms. It is used as a word form of add.

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.

The or operator can be used for asking python if at least one part of the comparisons is true while the and operators is used for asking if both comparisons are true.

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)

True

* + 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 the different because when we use and with the same on each side, it gives that opposite answer as using or and not together. It is same because when we use and with different things on each side, it always gives the opposite answer as or and not together.

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

They give different results because the operator can only be used with brackets. Without brackets, it gives an error.

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

They give the same result because the value for both of them 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.

False and False

Not ( True and False)

1 == 1 or 2 == 3

* 1. Provide the results for your practice expressions

False

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.

[“Fc Barcelona”, “LA Lakers”, “Toronto Raptors”, “PSG”]

* 1. Assign your list to a variable.

Myteam = [“Fc Barcelona”, “LA Lakers”, “Toronto Raptors”, “PSG”]

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

Myteam

[“Fc Barcelona”, “LA Lakers”, “Toronto Raptors”, “PSG”]

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.

The list index is 3 for “PSG”.

* 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 is an Index 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.

colours = [“Yellow”, “Red”, “Green”, “Orange”]

Print(colours[0])

Yellow

Print(colours[1])

Red

Print(colours[2])

Green

Print(colours[3])

Orange

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 = 2

if myname == 2:

print(“Hi Alfred!”)

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 = “Jasjot”

if myname == “Jasjot”:

print(“Jasjot”)

else:

print(“Benipal”)

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.

Friend 1 = “Tejveer”

Friend 2 = “Jagdev”

Friend 3 = “Harjap”

Friend 4 = “Malhar”

If Friend 1 == “Tejveer”

print(“Tejveer”)

elif Friend 2 == “Jagdev”

print(“Jagdev”)

elif Friend 3 == “Harjap”

print(“Harjap”)

else Friend 4 == “Malhar”

print(“Malhar”)