**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+5-5

=>10

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

=>5.0

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

5/1

=>5.0

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

8/5

=>1.6

* 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 (8/5)

=>2

* 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 “=”?

Comparison Operator

* 1. What does “=” mean?

Assignment Operator

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

5+5\*5+5-5/5==34

=>True

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

100/100\*100-100+100==0

=> 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 likes to use quotes to represent a variable.

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

It is using 2 + 5 as a variable.

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 goes for the most recent callback

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

Same error in part a.

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.

‘s’ + ‘a’ + ‘n’ + ‘v’ + ‘I’ + ‘r’

=>’sanvir’

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 is printing the letter in the index which is o. Also, python starts with the number 0.

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

It prints nothing.

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 not 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?

You get an error message.

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

You get an error message as well

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

=>6.0

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

There is a error message

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

It is following the rules of algebra.

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’

Student = mynumber

Student[7]

=>’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 “==”?

When we're assigning a value, we're saying "this equals that". That's a short sentence, so it only gets one equal sign: =

But when we're comparing values, we're asking "is this thing equal to that thing?". And that's a longer sentence, so it gets two equal signs: ==

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

Fruit = ‘watermelon’

Fruit == ‘watermelon’

=>True

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 need to put the multiplication sign instead

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

One is a integer while the other is a string.

1. Read through “Lesson 6: Errors – Parts of an Error Message”.
   1. Is “friend” + 5 an example of:

It is a type error.

* + 1. A Syntax Error?
    2. A Runtime Error?
    3. A Logic Error?

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

Print( ‘sanvir’ , ‘rana’)

Sanvir rana

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 )

Error message.

* 1. Why is the result different?

Thye first one is a string while the other is a string with no quotation.

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 help computer programming by deciding what to do in coding

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?

It only uses true or false with the first letter capital

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

If they say the same thing.

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

It is basically kind of the equal sign.

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 false
      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.

The or operator is the opposite of the and operator

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
      2. not (True or False) true
      3. not (False or True) true
      4. not (False or False) false
   2. 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 opposite thing of and comparisons

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

True and 6==6

True

True and 5==4

False

False and 4==4

True

* 1. Provide the results for your practice expressions

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

Los Angeles Lakers

Philadelphia 76ers

* 1. Assign your list to a variable.

sportTeam = [‘TorontoRaptors’ , ‘LosAngelesLakers’ , ‘Philadelphia76ers’]

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

sportTeam = [‘TorontoRaptors’ , ‘LosAngelesLakers’ , ‘Philadelphia76ers’]

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.
   2. In the tutorial, the error produced by typing “fruit[3]” is an example of:
      1. A Syntax Error? Syntax error
      2. A Runtime Error?
      3. A Logic Error?
2. Complete “Lesson 8: Lists – Practice” and “Lesson 8: Lists – Practice Answers” by typing the sample commands in the black area of the IDE.

Colors = [‘black’ , ‘blue’ , ‘white’]

Print (colors[0])

black

Print (colors[1])

Blue

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

myName = ‘Alfred’

if myName == ‘Alfred’ :

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 = ‘Sanvir’

if myName == ‘Sanvir’ :

print (‘hi sanvir’)

else:

print(‘who r u’)

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.

myName = ‘Sanvir’

if myName == ‘Sanvir’ :

print (‘hi sanvir’)

elif myName == ‘Alfred’:

print (‘hi alfred’)

elif myName == ‘Singh’:

print (‘hi singh’)

else:

print (‘who r u’)