**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+2-3=4

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\*2= 10
* 5/5= 1

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
* 14/4=3.5

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 (14/4) 4

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

* “==” is an equality check so you could say 1 == 1
* “=” just gives you an answer to an equation

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.

* 2+2+5+4+7+1>=21 => True
* 3+5+3+2+2+1>=44=> 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.

* Apple works with quotes because you are telling python that it is part of a string
  1. Also explain why “2 + 5” does not equal 7.
* It wont work because it has quotations around it so it thinks its part of a string

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 works because you can only increase the string and not decrease it
  1. Also explain why “Hello” \* 10 works but why “Hello” / 10 does work.
* It only works because you can only add or multiply not divide or subtract

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.

* Liam L is 0 I is 1 a is 2 m is 3

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 letter H is 0 and does not start at 1
  1. What does print(“Hay, Bob!”[4]) print? For a hint try print(“Hay, Bob!”[3]) and print(“Hay, Bob!”[5])
* It prints ‘ ‘ for each letter

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.

* Because there is not enough letters for there to be seven

**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 a traceback error
  1. Why doesn’t typing kittens / 3 work?
* Because you did not give kittens a value

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
      * your telling the computer that the word puppies equals 36
      * puppies = puppies / 6
      * your telling the computer to divide the number that you told it before
      * puppies
      * now when you type puppies the value will be 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”
      * your telling the computer that the word colour is red
      * puppies = 36
      * you telling the computer that the word puppies is equal to 36
      * colour + puppies
      * your telling the computer to combine the value of colour and 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
      * Python did the multiplication first because it goes by order of operations so I got the word Monday three times
      * ( Color + day ) \* fishes
      * If you put brackets around color and day python knows that it needs to add yellow and Monday first then multiply the whole thing by three
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”?

* The index of r in watermelon is 4
  1. Write an expression using mynumber to return ‘r’
* Mynumber=7
* Fruit[mynumber-3]

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 you use “==” you are comparing values and when you use “=” when you are assigning a value
  1. Create your own mnemonic to remember this difference.
* = is “this equals that” and == is “is this equal to that?”

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?

* Because one is a string and one is a integer
  1. Wht is the difference between int and str?
* “int” is a integer and “str” 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 a Type Error

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

* Print(“liam”,”Noakes”)

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 a string
  1. What is the value of: type( True )
* The value is a boolean
  1. Why is the result different?
* Because when you use quotes python thinks it is a string but if you don’t it considers it 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?

* They are important because we use them to make decisions in are 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?

* Because in python there is only true and false and if there was maybe it would not make sense

**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 is no other combination
  1. Explain how the AND operator is similar to a math operator and how it is different.
* Because it compares things but also gives an answer

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.
* With OR as long as one comparison is true the whole expression is true but with and if only one is true it is false

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 is similar because NOT reverses the answer that python gives and with the OR as long as one is still true it is still true

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)

* it gives you false because it thinks it’s a string
  + 1. not True or True
* because as long as one of them is true it stays true
  1. Explain why the following two Python statements give the same results.
     1. not (True and True)
     2. not True and True
* it gives you false because NOT reverses the normal answer which would normally be 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.

* True and false
* Not True or True
* True and 2==2
  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.

* Toronto raptors
* Blue jays
  1. Assign your list to a variable.
* Team
  1. Confirm that your variable and your list are the same.
* Team=[“Toronto raptors”,”blue jays”]

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.

* Blue jays 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?
* It’s a 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.

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

123=alfred

if 123 == "alfred":

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”).

* myname="liamnoakes"
* if myname == "liamnoakes":
* print("Hi liamnoakes!")
* else:
* print("liamnoakes!")

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 myname == "liam":
* print("Hi liam!")
* elif myname == "kashish":
* print("Hi kashish !")
* else:
* print("Who are you?!?")
* if myname == "liam":
* print("Hi liam!")
* elif myname == "myles":
* print("Hi myles!")
* else:
* print("Who are you?!?")
* if myname == "liam":
* print("Hi liam!")
* elif myname == "rohan":
* print("Hi rohan!")
* else:
* print("Who are you?!?")