**lOutline**

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+6-1**

**=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\*6/5**

**=6**

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

**6/3**

**=2**

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

**6/4**

**=1.5**

* 1. List your expressions and the results below.

**6/4**

**=1.5**

**6/3**

**=2**

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(6/3)**

**=3**

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

**== is to “equal to” something , = is “equal”**

* 1. What does “=” mean?

**= is “equal”**

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

**12\*2==24**

**=true**

**5\*5==25**

**=true**

**1\*1==1**

**=true**

**2\*2==4**

**=true**

**3\*3==9**

**=true**

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

**3\*9==4**

**=false**

**5\*5==10**

**=false**

**2\*2==5**

**=false**

**3\*3==10**

**=false**

**1\*1==4**

**=false**

* 1. List your expressions and the results returned below.

**12\*2==24**

**=true**

**5\*5==25**

**=true**

**1\*1==1**

**=true**

**2\*2==4**

**=true**

**3\*3==9**

**=true**

**3\*9==4**

**=false**

**5\*5==10**

**=false**

**2\*2==5**

**=false**

**3\*3==10**

**=false**

**1\*1==4**

**=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.

**Because it is a word and words needs quotation marks.**

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

**Because it is a number, and numbers don’t need quotation marks.**

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.

**Typing “appl” + “e” works, because you are adding a letter and why typing “apple” - “e” gives an error, because your subtracting a letter that’s not there.**

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

**“Hello” \* 10 works because you multiplying hello 10 times, but why “Hello” / 10 does work, because your dividing 10 hellos which are not there.**

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.

**“G” + “o” + “b” + “I” + “n” + “a”**

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”.

**(“Hello!”[4]) does not print “l”, because l is not the fourth index, “o” is**.

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

**It prints the space between the comma and bob.**

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 not a 7th index’s.**

**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?

**I would get the amount of puppies I have divided by 3.**

* 1. Why doesn’t typing kittens / 3 work?  
     **There is no amount of kittens to divide by 3.**

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 was given the amount of 36**
      * puppies = puppies / 6 **the amount of 36 puppies was then divided by 6**
      * puppies **the answer to what 36/6 is, to is how many puppies are there.**
2. Read through “Lesson 5: Variables – Rules”.
3. 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

**I did not work because you can’t add strings and indexes.**

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 gives different result because of the order of operation it is in. Since ( Color + day ) is in bracket so it does that part of the equation first.**
2. 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-9]**

**‘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 an answer to the question being asked**

**== is asking the question.**

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

**==question**

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 cannot add 5 friends. You can multiple friends with, 5 to duplicate friends 5 times.

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

**int is an integer**

**str is a string**

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

**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 (“Gobina, Mathyaparana”)**

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”)

**<classs ‘str’>**

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

**Trace back error**

* 1. Why is the result different?

**Because the commands are different. One has a quotation marks the other does not.**

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?

**It is important because without Boolean you cannot decipher if a problem is true or false.**

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 there is no maybe answer it either has to be a true or false for the Booleans to work.**

**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 is True**
      2. **True and False is False**
      3. **False and True is False**
      4. **False and False is False**
   2. Explain if there are any other combinations of True / False.

**There are no other combination because if it’s false its false, and if it’s true its true.**

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

**It is similar because it is comparing both equations to find the answer, it is different because you use “<” and “>” to determine if it is similar or not.**

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 is true**
      2. **True or False is true**
      3. **False or True is true**
      4. **False or False is false**
   2. Explain how the OR operator is similar to the AND operator and how it is different.

**The OR operator is similar to the AND operator because if both condition are false/true then “or” “and” will have the same answer, it is different because “or “is true when one of the conditions is true and “and” is true when both conditions 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) is False**
      2. **not (True or False) is False**
      3. **not (False or True) is False**
      4. **not (False or False) is True**
   2. Explain how the combination of the NOT & OR operators is similar to the AND operator by itself and how it is different.

**The NOT & OR operators is similar to the AND operator because by itself because “and” has “not (True or False) is False” “not (False or True) is False”. it is different** because when both condition are false then the answer for the not operator is true, unlike the “and” operator it will be false.

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

**It gives different results because there are two different operators that uses “or” are doing different steps, because the first statement is in brackets, so it will execute that step first, unlike the second statement.**

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

**It gives the same result because it uses a “and” operator, for the first statement it is in brackets and will do that step first, which will make the answer false. While the second statement will be false since the first condition (not true) is false.**

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.

**“Won” == ”Won”**

**“New”==”New”**

**“Loss”==”Loss”**

* 1. Provide the results for your practice expressions

**“Won” ==”Won” False**

**“New”==”New” False**

**“Loss”==”Loss” False**

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.

**Blue jays, raptors, maple leaves**

* 1. Assign your list to a variable.

**Team = [“blue jays”, “raptors”, “maple leaves”]**

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

**They are the same.**

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 last index is “S”.**

* 1. In the tutorial, the error produced by typing “fruit[3]” is an example of:

**Name 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

myname = "Alfred"

if myname=="Alfred" :

print("Hi Alfred!")

myname = "Alfred"

if myname=="Alfred" :

print("Hi " + myname + "!")

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 = "FIRSTNAME"

if myname == "FIRSTNAME":

print("Hi GOBINA!")

else:

print("GOBINA")

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 = "HASRAT"

if myname == "HASRAT":

print("Hi HASRAT!")

elif myname == "SHARLENE":

print("Hi SHARLENE!")

elif myname == "ABIRA":

print("Hi ABIRA!")

elif myname == "MUTIKA":

print("Hi MUTIKA!")

else:

print("Who are you?!?")