## Practice Assignment No. 3

**DSCI 15310 – Computational Thinking and Programming**

# September 12, 2017

**Due Date: September 19, 2017**

**My name is .**

Below is a table of two columns. The first column provides you with information that you should type into your IDLE window (shell). After you input the given instructions, put (you may copy and paste your result from the IDLE window) your answer into the corresponding cell in the right-hand column.

You should be observing and thinking about the results that you get. You should be asking yourself, “Why did this happen?” Study the environment of the statements given. Also, notice that statements can be broken into two or more lines, with the latter part going onto the next line(s). The first line prompt is “>>>” , and the second line prompt (as well as of subsequent lines) is “…”. If you have any questions or don’t understand something, ask for assistance. E-mail me for help.

|  |  |
| --- | --- |
| **Use the Interactive Development Environment (IDLE) to input the following information** | **What is your result?** |
|  |  |
| >>> -17 + 4 | -13 |
| >>> studentLoan = 100000  >>> interestRate = 0.25  >>> interest = studentLoan \* \  ... interestRate  >>> costOfStudentLoan = studentLoan + \ ... interest  >>> costOfStudentLoan  N.B. Do not type the characters ‘\’ and the ellipsis. They just mean that the line continues on the next line. | 125000.0 |
| >>> 25 \* 13 | 325 |
| >>> 25.0 \* 13 | 325.0 |
| >>> 25.0 \* 13.0 | 325.0 |
| >>> 0 / 1 | 0.0 |
| >>> 1 / 0 | Traceback (most recent call last):  File "<pyshell#10>", line 1, in <module>  1 / 0  ZeroDivisionError: division by zero |
| >>> 654321 \* 123456 | 6706253376 |
| >>> import sys  >>> sys.maxint | Traceback (most recent call last):  File "<pyshell#13>", line 1, in <module>  sys.maxint  AttributeError: module 'sys' has no attribute 'maxint' |
| >>> 2 \*\* 5 | 32 |
| >>> 5 \*\* 2 | 25 |
| >>> square = 2  >>> cube = 3  >>> sideLength = 4  >>> area = sideLength \*\* square  >>> area | 16 |
| >>> volume = sideLength \*\* cube  >>> volume | 64 |
| >>> area = pow(sideLength, square)  >>> area | 16 |
| >>> area = pow(5, 2)  >>> area | 25 |
| >>> volume = pow(sideLength, cube)  >>> volume | 64 |
| >>> volume = pow(5, 3)  >>> volume | 125 |
| >>> 2 \*\* -3 | 0.125 |
| >>> 2 \*\* -4 | 0.0625 |
| >>> 4 \*\* 0.5 | 2.0 |
| >>> 4 \*\* -0.5 | 0.5 |
| >>> pow(2, -3) | 0.125 |
| >>> pow(2, -4) | 0.0625 |
| >>> pow(4, 0.5) | 2.0 |
| >>> pow(4, -0.5) | 0.5 |
| >>> 25 % 6 | 1 |
| >>> 2001 % 4 | 1 |
| >>> 2000 % 4 | 0 |
| >>> 1000 % 0 | Traceback (most recent call last):  File "<pyshell#42>", line 1, in <module>  1000 % 0  ZeroDivisionError: integer division or modulo by zero |
| >>> 0 % 1000 | 0 |
| >>> 21 % 3, 21 % 4, 21 % 5, 21 % 6 | (0, 1, 1, 3) |
| >>> 30 / 7 | 4.285714285714286 |
| >>> -30 / -7 | 4.285714285714286 |
| >>> -3 / 2 | -1.5 |
| >>> 3 / 2 | 1.5 |
| >>> -3.0 / 2 | -1.5 |
| >>> 3.0 / 2 | 1.5 |
| >>> 3 / float(2) | 1.5 |
| >>> 3 / int(1.25) | 3.0 |
| >>> 3 / int(1.5) | 3.0 |
| >>> 3 / int(1.75) | 3.0 |
| >>> 3 / int(2.0) | 1.5 |
| >>> 3 / int(2.15) | 1.5 |
| >>> quotient, remainder = divmod(100,3)  >>> quotient, remainder | (33, 1) |
| >>> q, r = divmod(100, 3)  >>> q, r | (33, 1) |
| >>> a = b = c = d = e = f = 3  >>> x = 2  >>> a += x  >>> a | 5 |
| >>> b -= x  >>> b | 1 |
| >>> c \*= x  >>> c | 6 |
| >>> d /= x  >>> d | 1.5 |
| >>> d /= float(x)  >>> d | 0.75 |
| >>> e \*\*= x  >>> e | 9 |
| >>> f %= x  >>> f | 1 |
| >>> abs(-444) | 444 |
| >>> abs(444) | 444 |
| >>> do = 1  >>> re = 2  >>> mi = 3  >>> fa = 4  >>> so = 5  >>> la = 6  >>> ti = 7  >>> do = 8  >>> array = [do, re, mi, fa]  >>> ofGoldenSun = [so, la, ti, do]  >>> max(array), max(ofGoldenSun), \  ... max(array, ofGoldenSun)  N.B. Do not type the characters ‘\’ and the ellipsis. They just mean that the line continues on the next line. | (8, 8, [8, 2, 3, 4]) |
| >>> min(array), min(ofGoldenSun), \  ... min(array, ofGoldenSun)  N.B. Do not type the characters ‘\’ and the ellipsis. They just mean that the line continues on the next line. |  |
| >>> max(-444, 1, abs(-444)) | 444 |
| >>> min(-222, 22, abs(-222)) | -222 |
| >>> do = "Doe, a dear, a female \ ... ... dear,\n"  >>> re = "Ray, a drop of golden sun,\n"  >>> mi = "Me, a name I call myself,\n"  >>> fa = "Far, a long, long way to \ ... run,\n"  >>> so = "Sew, a needle pulling \  ... thread,\n"  >>> la = "La, a note to follow sew,\n"  >>> ti = "Tea, a drink with jam and \ ... bread!\n"  >>> doh = "That will bring us back to \ ... do oh-oh-oh!"  >>> print (do, re, mi, fa, so, la, ti, \ ... doh)  N.B. Do not type the characters ‘\’ and the ellipsis. They just mean that the line continues on the next line. | Doe, a dear, a female dear,  Ray, a drop of golden sun,  Me, a name I call myself,  Far, a long, long way to run,  Sew, a needle pulling run,  La, a note to follow sew,  Tea, a drink with jam and bread!  That will bring us back to do oh-oh-oh! |
| >>> array = [do, re, mi, fa]  >>> ofGoldenSun = [so, la, ti, doh]  >>> array, ofGoldenSun | (['Doe, a dear, a female dear,\n', 'Ray, a drop of golden sun,\n', 'Me, a name I call myself,\n', 'Far, a long, long way to run,\n'], ['Sew, a needle pulling run,\n', 'La, a note to follow sew,\n', 'Tea, a drink with jam and bread!\n', 'That will bring us back to do oh-oh-oh!']) |
| >>> print (array, ofGoldenSun) | ['Doe, a dear, a female dear,\n', 'Ray, a drop of golden sun,\n', 'Me, a name I call myself,\n', 'Far, a long, long way to run,\n'] ['Sew, a needle pulling run,\n', 'La, a note to follow sew,\n', 'Tea, a drink with jam and bread!\n', 'That will bring us back to do oh-oh-oh!'] |
| >>> max(array) | 'Ray, a drop of golden sun,\n' |
| >>> min(array) | 'Doe, a dear, a female dear,\n' |
| >>> max(ofGoldenSun) | 'That will bring us back to do oh-oh-oh!' |
| >>> min(ofGoldenSun) | 'La, a note to follow sew,\n' |
| >>> max(array), max(ofGoldenSun), \  ... max(array, ofGoldenSun)  N.B. Do not type the characters ‘\’ and the ellipsis. They just mean that the line continues on the next line. | ('Ray, a drop of golden sun,\n', 'That will bring us back to do oh-oh-oh!', ['Sew, a needle pulling run,\n', 'La, a note to follow sew,\n', 'Tea, a drink with jam and bread!\n', 'That will bring us back to do oh-oh-oh!']) |
| >>> x = 2222.49  >>> y = 2222.51  >>> round(x), round(x, 2), round(x, -3) | (2222, 2222.49, 2000.0) |
| >>> round(y), round(y, 2), round(y, -3) | (2223, 2222.51, 2000.0) |
| >>> xx = -2222.49  >>> yy = -2222.51  >>> round(xx), round(xx, 2), \  ... round(xx, -3)  N.B. Do not type the characters ‘\’ and the ellipsis. They just mean that the line continues on the next line. | (-2222, -2222.49, -2000.0) |
| >>> round(yy), round(yy, 2), \  ... round(yy, -3)  N.B. Do not type the characters ‘\’ and the ellipsis. They just mean that the line continues on the next line. | (-2223, -2222.51, -2000.0) |
| >>> import math  >>> math.ceil(-1.49), math.ceil(1.51) | (-1, 2) |
| >>> math.ceil(-1.51), math.ceil(1.49) | (-1, 2) |
| >>> math.floor(-1.49), math.floor(1.51) | (-2, 1) |
| >>> math.floor(-1.51), math.floor(1.49) | (-2, 1) |
| Open up a script window by clicking  File | New File at the top of the IDLE screen.  Save this file as ‘My Template.py’.  You will use this to start all of your programs.  Enter the following information into the template:  #First Name Last Name  #DSCI 15310 -- Computational Thinking and Programming  #Program No.  #Name of Program --  #Due Date --  These are your header comments. The last three lines will have to be changed to reflect each program that you write. This practice assignment is No. 3, and the program is No. 0. Also, notice above, the second comment line is wrapped around. When you type this line, do not break it up after the word ‘Thinking.’ |  |
| Save the same file now as ‘Practice Assignment No. 3, Program No. 0’. The file ‘My Template’ will not be lost.  Enter the following information below your header comments:  string1 = "Hello World!"  string2 = "My name is ."  print (string1, string2)  print ("\n", string1, "\n", string2)  When you finish entering the given block of code, save your file, and click Run | Run Module. You will now be taken to the IDLE window (shell), where you will see the results of your labor. Save the script file. Be sure to enter your output in the adjacent cell. | Hello World! My name is Andrew Tennant.  Hello World!  My name is Andrew Tennant. |