

CPSC 1301, Computer Science I Lab Assignment

Lab 02a

Run the following commands in your Python interpreter. Submit a transcript of your session as your deliverable for this lab. This should be a plain text file named `lab02a_lastname.txt`.

1 Reading and writing command line arguments

Python automatically creates an array named `argv`. It placed the command line arguments one by one into this array. The first command line argument is always the name of the program, that is, `argv[0]`. The first argument will be at `argv[1]`, the second at `argv[2]`, and so on.

To use the command line arguments, you must first `import sys`. Then, you can access each one in turn by `sys.argv[1]`, and so on.

2 Writing and Running your First Python Program

Use this template for writing your Python labs and other quick and dirty scripts. Use any text editor of your choice. I recommend Notepad++ (<https://notepad-plus-plus.org/>).

2.1 Hello World

```

1  #-----
2  # week2_1.py
3  #-----
4
5  print("hello_world")

```

2.2 Hello name

```

1  #-----
2  # week2_2.py
3  #-----
4
5  name = "Your_Name"
6  print(name)

```

2.3 Print sum

```

1  #-----
2  # week2_3.py
3  #-----
4
5  lhs = 4
6  rhs = 4
7  sum = lhs + rhs
8  print(sum)

```

2.4 Getting data from users

```
1  #-----
2  # week2_4.py
3  #-----
4
5  print("Enter an integer:")
6  lhs = int(input())
7  print("Enter another integer:")
8  rhs = int(input())
9  product = lhs * rhs
10 print(product)
```

2.5 A void function

```
1  #-----
2  # week2_5.py
3  #-----
4
5  def my_fun():
6      print("You have called my_fun")
7
8  my_fun()
```

2.6 A void function with parameter

```
1  #-----
2  # week2_6.py
3  #-----
4
5  print("Enter your name:", end = '')
6  name = input()
7  def my_fun(name):
8      print("Hello", name)
9
10 my_fun()
```

2.7 A value returning function

```
1  #-----
2  # week2_7.py
3  #-----
4
5  def my_fun():
6      print("Enter your first name:", end = '')
7      first = input()
8      print("Enter your last name:", end = '')
9      last = input()
10     fullname = first + " " + last
11     return fullname
12
13 turnip = my_fun()
14 print(turnip)
```

2.8 A function that has input and returns a value

```
1  #-----
2  # week2_8.py
3  #-----
4
```

```

5     def my_fun(lhs, rhs):
6         produce = lhs * rhs
7         return product
8
9     print("Enter the lhs:", end = '')
10    lhs = int(input())
11    print("Enter the rhs:", end = '')
12    rhs = int(input())
13
14    out = my_fun(lhs, rhs)
15    print("The product of %d and %d is %d", (lhs, rhs, out))

```

3 At the Python prompt

```

C:\Users\ccc31\cols-st\cpssc1301>python
Python 3.8.5 (tags/v3.8.5:580fbb0, Jul 20 2020, 15:43:08) [MSC v.1926 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> # VARIABLES #####
>>> print ("Hello, CSU!");
>>> w = print("Hello")
>>> w
>>> print(w)
>>> name = "Student"
>>> print("Hello", name, "!")
>>> c = "Columbus"
>>> s = "State"
>>> u = "University"
>>> print("Hello", c, s, u)
>>> h = "Hello"
>>> print(h, c, s, u)
>>> greeting = "Hello, CSU!"
>>> print(greeting)
>>> # assign your name to a variable and greet yourself
>>> # ----- your code here -----
>>> # ARITHMETIC #####
>>> x = 5
>>> y = 7
>>> z = 9
>>> x + y
>>> x * z
>>> z - y
>>> z / x
>>> z % x
>>> x % z
>>> x // z
>>> z // x
>>> x / z
>>> z / x
>>> print(x + y, x - z, z * y, z / x, z % x, y // x)
>>> x + y * z
>>> (x + y) * z
>>> x + (y * z)
>>> x + y * z - x
>>> (x + y) * (z - x)
>>> x + (y * z) - x

```

```

>>> x * y / z
>>> (x * y) / z
>>> x * (y / z)
>>> x / y * z
>>> (x / y) * z
>>> x / (y * z)
>>> # COMPARISON OPERATORS #####
>>> x == 5
>>> x != 5
>>> x == y
>>> x != y
>>> x > y
>>> x < y
>>> x >= 5
>>> x >= y
>>> 5 >= x
>>> 5 > x
>>> # LOGICAL OPERATORS #####
>>> 0 and 0
>>> 1 and 1
>>> 2 and 4
>>> 2 or 4
>>> True
>>> not True
>>> False
>>> not False
>>> not (1 and 1)
>>> not (1 or 12)
>>> not (0 and 0)
>>> not (0 or 0)
>>> True and True
>>> True and False
>>> False and True
>>> False and False
>>> True or True
>>> True or False
>>> False or True
>>> False and False
>>> # DOING MATH #####
>>> import math
>>> math.pi
>>> math.sqrt(4)
>>> math.sin(0)
>>> math.cos(0)
>>> math.tan(0)
>>> exit()

```