Week 1 Deliverables

Overview: In this week, you have set-up your AWS Educate account, connected to the AWS console, started a Cloud9 IDE environment and connected to the Safari Text Library that contains the digital textbook. The Lab for this week demonstrates your first use of the AWS Cloud9 IDE with a fairly simple Python application.

Be sure to develop and test your Python code in the AWS Cloud9 IDE provided for the class. Also, only use the functionality and Python concepts covered so far in the readings and examples. For example, we have not covered loops, arrays or else if statement. We are limited to simple if statements, print, input, mathematical operators, comparison operators and a few other introductory concepts.

We build these other concepts very quickly over the next couple of weeks.

You should also begin to use the PEP Python Style guide mentioned in the book and found here:

https://www.python.org/dev/peps/pep-0008/

Some examples of Python Coding Style best practices include:

- Limit all lines to a maximum of 79 characters.
- Imports are always put at the top of the file, just after any module comments and before module globals and constants.
- Use 4 spaces for indentation.

Submission requirements include 3 files. (Zipping them into one file is acceptable):

- Python MinMax Application Code
- Python Calculator Application Code
- Word of PDF file containing your test results

Python Applications for Lab1: (total 100 points):

This lab consists of two parts. The first exercise calculates the minimum and maximum of 5 numbers input by a user. The second exercise produces a simple Python calculator where a user selects the math operator and then enters two values as input for the operator selected.

1. Using the AWS Cloud9 IDE write a program that takes 5 input values from a user and prints the minimum and maximum value. Be sure to prompt the user for each of the 5 input values. The following is a possible application interface. Other application interfaces are possible as well. **(40 points)**

****

Welcome to the Python MinMax Application!

This application calculates the minimum and maximum of 5 integer values entered by a user.

Enter your first integer:

4

Enter your second integer:

7

Enter your third integer:

2

Enter your fourth integer:

12

Enter your fifth integer:

1

The minimum integer entered was 1.

The maximum integer entered was 12.

Thanks for trying the Python MinMax application.

****

Hints:

1. Use int() to cast the string entered by the user to an integer
2. Use min() and max() to calculate the minimum and maximum values
3. Test with many combinations. For example, what happens if you enter a non-number?
4. Use comments to document your code

Note: without any exception handling, you should see issues/errors when testing your program.

2. Using the AWS Cloud9 IDE Write a Python calculator application that prompts a user to calculate the sum, difference, modulus, product or quotient of two input integers. Since we can't loop yet, the application should run once and then exit. The application should prompt the user to select the mathematical operator, display what was selected and then prompt the user to enter the two integers.

If the user enters a 0 for the second number when the division operator is selected, the application should warn that divide by zero is not allowed and exit.

The following is a possible application interface. Other application interfaces are possible as well. (**40 points)**

Welcome to the Python Calculator Application.

What calculation do you want to perform?

1) Addition (+)

2) Subtraction (-)

3) Division (/)

4) Multiplication (*)

5) Modulus (%)

Enter 1,2,3,4 or 5 indicating your selection.

2

Subtraction was selected.

Enter your first integer:

12

Enter your second integer

3

The difference of 12 and 3 is 9.

Thanks for trying the Python calculator

****

Hints:

1. Use int() to cast the string entered by the user to an integer
2. Use the mathematical operators, if statements and comparison operators as needed
3. Test with many combinations. For example, what happens if you enter a non-number?
4. Use comments to document your code

3. Document your test results for each application within the AWS Cloud9 classroom environment. The test document should include a test table that includes the input values, the expected results and the actual results. A screen capture should be included that shows the actual test results of running the test case. Be sure to include multiple test cases to provide full coverage for all code. For example, if you have you should demonstrate each mathematical operator selected works as expected and every statement in the code is reached through the test cases**. (20 points)**

A possible test table may look like this:

Test Case	Input	Expected Output	Actual Output	Pass?
1a	1,2,3,4,5	Minimum = 1, Maximum =5	Minimum = 1, Maximum =5	Yes
1b	2,10,1,11,11	Minimum = 1, Maximum =11	Minimum = 1, Maximum =11	Yes
1c	3,g,e,s,7	Error	Error	Yes but note in future this needs to be addressed.
…				

Screen capture of test cases:

```
Enter first: 1
Enter second: 2
Enter third: 3
Enter fourth: 4
Enter fifth: 5
Minimum is  1
Maximum is  5


Process exited with code: 0
```

Figure 1. Test Case 1a Execution results

```
Enter first: 2
Enter second: 10
Enter third: 1
Enter fourth: 11
Enter fifth: 11
Minimum is  1
Maximum is  11


Process exited with code: 0
```

Figure 2. Test Case 1b Execution results

```
Enter first: 3
Enter second: g
Enter third: e
Enter fourth: s
Enter fifth: 7
Traceback (most recent call last):
  File "/home/ec2-user/environment/Project1/Hello.py", line 33, in <module>
    print ('Minimum is ' , min(int(x1),int(x2),int(x3),int(x4),int(x5)))
ValueError: invalid literal for int() with base 10: 'g'
```

Figure 3. Test Case 1c Execution results

Any submissions that do not represent work originating from the student will be submitted to the Dean's office and evaluated for possible academic integrity violations and sanctions.