SCIT, University of Wollongong

CSIT110/CSIT810

Autumn Session 2020

Assignment 1 (10%) due on Saturday 24 October 2020 at 00:00AM

Objectives

- Able to write clear code with comments and follow coding convention
- Able to use variables with meaningful names and correct data types
- Able to get user input
- Able to display output

Marking criteria:

- Total mark is 10. Deduct 1 mark for each day late.
- More than 3 days late will result in a zero mark.
- Code must compile: 0 mark for the whole assignment if there is a compile error.
- Correct file format (.py extension): 0 mark for the whole assignment if file submission is not in correct format.
-

Question 1	correctness, completeness and consistency with the assignment specification	1 mark
Question 2	correctness, completeness and consistency with the assignment specification	2 marks
Question 3	correctness, completeness and consistency with the assignment specification	2 marks
Question 4	correctness, completeness and consistency with the assignment specification	2 marks
Question 5	correctness, completeness and consistency with the assignment specification	3 marks
Overall	comments include name, student number, subject code; clear code and follow coding convention; use variables with meaningful names and correct data types	Deduct up to 1 mark

Submission Instruction: Assignment 1 submission is on Moodle. Put all your python code into a single python file (file extension .py) and submit it.

Assignment questions: there are 5 assignment questions.

Write clear code with **comments** and follow **coding conventions**. Comments should include **your name**, **student number** and **subject code** on top of your code. Please also add this information to the variables as stated in the template Your code must work **exactly** like the provided examples.

```
"""Assignment 1

Name: John Snow
Student number: 1234567
Subject code: CSIT110
"""

name = 'John Snow'
student_num = '1234567'  # Student number
subject_code = 'SCIT110'  # SCIT110 or SP420

def question1():
    ...
```

Question 1. Write a program to ask the user to enter 3 integers and then display a multiplication equation. Your code must work exactly like the following example (except the text in bold which indicates the user input):

```
Please enter the 1st integer: 2
Please enter the 2nd integer: -5
Please enter the 3rd integer: 20
Display equation using string addition: 2 x -5 x 20 = -200
```

Important requirement:

Your program must use string addition to produce the exact output as illustrated in the above example. You may assume the input are integers.

Question 2.

Write a program to ask the user to enter 2 subject details and then display this information using **string format - <class 'str'>.format()**. Your code must work exactly like the following example (the text in bold indicates the user input):

```
Enter the 1st subject code: MATH 101
Enter the 1st subject title: Linear Algebra
Enter the 1st subject credit point: 9
Enter the 2nd subject code: CS 203
Enter the 2nd subject title: Data Structure with Java
Enter the 2nd subject credit point: 10
Your chosen subjects:
MATH 101: Linear Algebra
CS 203: Data Structure with Java
Total credit points: 19.
```

Important requirement:

- Your program must use **format - <class 'str'>.format()**. to produce the **exact output** as illustrated in the above example
- You should test your program for different input values

Question 3.

At the Singapore Zoo, each adult ticket costs $ 39, each child (>= 3 years old, <= 12 years old) ticket costs $26.50 and each young child (< 3 years old) ticket is free. Write a program to ask the user to enter the number of tickets for each type and then display the ticket information.

Your code must work exactly like the following example (the text in bold indicates the user input):

```
How many adult tickets you want to order: 2
How many children (3-12 years old) tickets: 3
How many children (<3 years old) tickets: 2
Type                Number of tickets          Cost
Adult                               2         $78.00
Children (3-12y.o.)                 3         $79.50
Children (<3)                       2           free
Total                               7        $157.50
```

Important requirement:

Your program must use string format to produce the exact output as illustrated in the above example (including the alignment - 20 characters, 17 characters, 15 characters, dollar signs and 2 decimal places in cost).

You should test your program for different integer values as input

Question 4. A subject has 3 assessments:

- An assignment whose mark is an integer between 0 and 20;
- A project whose mark is an integer between 0 and 30;
- A final exam whose mark is an integer between 0 and 50.
- There are 4 possible grades: A, B, C and Fail. The grade is determined by the following rules:

Marks	**Grade**
Final exam mark = <20	Fail
Total mark >= 90	A
Total mark >=75	B
Total mark >=60	C
Total mark <60	Fail

```
Question 4.
Enter your assignment mark: 20
Enter your project mark: 17
Enter your final exam mark: 15
Your result:
Assignment:             20
Project:                17
Final exam:             15
Grade:                Fail
```

Important requirement:

- Your program must use string format to produce the exact output as illustrated in the above example. 22 characters for the first column and 4 characters for the results.
- You should test your program for different kind of scenarios, here are some examples for testing:
- Assignment 11, Project 20, Final exam 19, Grade Fail
- Assignment 11, Project 20, Final exam 41, Grade C
- Assignment 13, Project 20, Final exam 28, Grade C
- Assignment 18, Project 20, Final exam 37, Grade B
- Assignment 10, Project 18, Final exam 49, Grade B
- Assignment 18, Project 26, Final exam 48, Grade A

Question 5.

Jimmy downloaded a large number of poorly named files, e.g. "cute_cat[20150203].jpg", "[quick]brown_fox.png", "[x264]lazy_dog[1280x720].mp4"

and would like to automatically clean up these filenames by removing all instances of square brackets and the contents contained within.

Write a Python script that:

1. Prompts for filenames with "Filename?" repeatedly, until an empty string is given

2. Prints a string with all the cleaned filenames, separated by commas

```
Question 5.
Filename? cute_cat[20150203].jpg
Filename? [quick]brown_fox.png
Filename? [x264]lazy_dog[1280x720].mp4
Filename?
cute_cat.jpg,brown_fox.png,lazy_dog.mp4
```

Important requirement:

- Your program must use string format to produce the exact output as illustrated in the above example. With commas separation in between the first and last file names

You should test your program for different kind of scenarios