PSG COLLEGE OF TECHNOLOGY, COIMBATORE – 641004
DEPARTMENT OF INFORMATION TECHNOLOGY

22/6/2022

Tutorial - 2

1. Define a class Alien which contains the <x_coordinate> <y_coordinate> and <health>
Further the class must support following functions:
(i) Constructor to initialize the coordinates using the arguments passed and assigning a value of 3 for health.
(ii) Hit() If the alien is hit by the object , the health must be reduced by 1
(iii) Is_alive() method to be checked often to verify whether the alien is alive (i.e) health value is greater than zero
(iv) Teleport() method will allow the alien coordinates to be changed based on the passed values.
(v) Collision_detection() will check whether the distance between the alien and other passed object is less than three. If so , It invokes the hit() method to reduce the health value of the alien object.
(vi) Provide a function which can print the coordinates of the alien object when it is passed to print() function
It should also support the following tasks.
(i) Keep track of the number of alien objects created
(ii) Allows to create a list of alien objects given a list of coordinates

2. Triangle, pentagonal, and hexagonal numbers are generated by the following formulae:

Triangle		$T_n=n(n+1)/2$		1, 3, 6, 10, 15, ...
Pentagonal		$P_n=n(3n-1)/2$		1, 5, 12, 22, 35, ...
Hexagonal		$H_n=n(2n-1)$		1, 6, 15, 28, 45, ...

(i) Create a class Numbers which is used for representing these specific numbers
(ii) Include functions which generate the triangle, pentagonal and hexagonal number given the value of n
(iii) Also include a function Find_n() which accepts two arguments: a numeric value and a marker ‘T’, ‘H’ (or) ‘P’ to denote the type of the number , Find the value of n for the given numeric value of the mentioned category.

3. Create a class to validate the credit card numbers using Luhn Algorithm.
It should support functions
(i) is_valid_format
It should check whether the strings should not contain any non-digit character other than space.
(ii) is_valid_length
It should check that the string should contain 16 digits in it.
(iii) is_valid_number
Should apply the Luhn algorithm and validate the number. Here the first step is to double the alternate digits in the number. If the doubled value is greater than 9, subtract 9 from it.

Ex: 4132 6521 3195 0343
The number after modification would be 8162 3561 9195 0383
The second step is to compute the sum of digits in the number
Sum of digits in the number = 8+1+6+2+3+5+6+1+9+1+9+5+0+3+8+3
= 70
Finally, check whether the number is divisible by 10, If so, the number is a valid number.

4. Create a class which supports to-do-list. The object allows you to maintain a list of items.

It should have the following basic functionality

- Add an item to the to-do list
- Delete a selected item from the to-do list
- Print out the list

5. Design an **automatic grading** system which computes the average marks secured by the student. The class which models students must allow the user to store the student name and marks secured (list) and average marks. It should provide support for computing the average of the student and the class average

The user will provide the input in the following format:
The first line will contain N (number of students) and M (number of assignments)
In the next N lines, student name followed by the marks separated by spaces

Sample Input:

```
3 5
JON 19 14 15 15 16
JEREMY 15 11 10 15 16
JESSE 19 17 20 19 18
```

The output should be class average(in the first line). In subsequent lines, print the student name and the student average marks separated by space.

Sample Output:

```
15.93
JON 15.80
JEREMY 13.40
JESSE 18.60
```