| **Assignment # 15**  ***Session*: Spring 2022 *Total marks*: 100**  ***Name*** : ***\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Roll number* : \_\_\_\_\_\_\_\_\_\_\_\_\_** |
| --- |

***Submission:***

• *Email instructor or TA if there are any questions. You cannot look at others’ solutions or use others’ solutions, however, you can discuss it with each other. Plagiarism will be dealt with according to the course policy.*

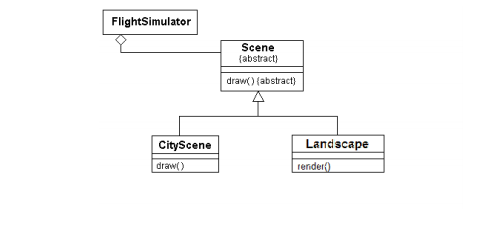
*• Submission after due time will not be accepted.*

**There should be a Report explaining your code and highlighting results. Follow this naming convention for your report RollNumber\_Assignment#.pdf e.g BSCE21001\_Assignment12.pdf.**

**Note:** From this assignment onwards, you will create default and parameterized constructor/s and destructor for every class.

### **Problem**

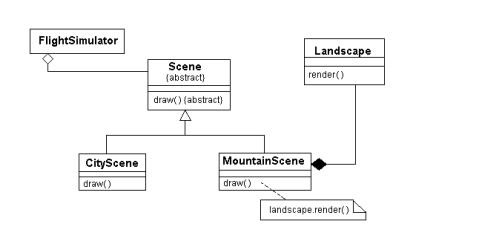
We have a Flight Simulator application that has a Scene of Flight as its sub part. Scene can be of City or Landscape. We have the following scenario.



We want to use an existing Landscape class in our flight simulator application which have a render function, but as you can see, it doesn't have the right interface (Scene interface has no virtual render function). We want to solve this problem.

### **Solution**

We make Landscape compatible with Scene by implementing the following:



**Class Scene:**

capture\_date : int

draw() : void // Abstract method

**Class Landscape:**

area : double

render(): void

**Class FlightSimulator:**

It has aggregation relationship with abstract class

Redefine draw() function that has only a print statement of “Drawing Scene”.

**Class MountainScene:**

Call render function of Landscape as mentioned in diagram.

**TASK 1:**

You are required to code this scenario in C++.

|  |
| --- |

**TASK 2:**

**(a)** Add a function getArea() in 'Scene' class.

**Class Scene:**

capture\_date : int

draw() : void // Abstract method

getArea() : string // Displays 'Area of Scene'

**Class CityScene:**

draw() : void

getArea() : string // Displays 'Area of City Scene'

**Class MountainScene:**

draw() : void

getArea() : string // Displays 'Area of Mountain Scene'

**(b)** 1- Create a pointer of Scene class

2- Assign object of MountainScene class to the pointer created in part 1

3- Call getArea() through the pointer

4- Report the output you get in part 3

|  |
| --- |

**TASK 3:**

Comment on the implementation of functions in Task 1 and Task 2 in terms of overriding.

|  |
| --- |