**Advanced Object Oriented Programming**

**Mini Assignment 1**

**(10%)**

**Average Time to complete: 2 and a half hours**

**Time given: 1 Week and 3 days.**

**Due:** Friday 1st October at 11:30 pm

**Submission:** On Blackboard.

Format: Upload this completed document

This project must be done in a group. Group size: 2 to 4 people.

It consists of 4 parts.

|  |  |  |  |
| --- | --- | --- | --- |
| **Group member** | **First name** | **Last name** | **Student id number** |
| **1** | Michael | Noseworthy | 101027533 |
| **2** | Ryan | Lee | 101296633 |
| **3** | Hamza Murat | Istek | 101313801 |
| **4** | Ronal | Rodriguez Arias | 101314540 |

**One member of the group is responsible for uploading the completed document.**

Question1:

What is aggregation with respect to OOP? – (1 mark) In your explanation you must include:

- Differentiate between the two forms aggregation and composition. (1 mark each)

- Explain how they are shown in UML. – (0.5 marks each)

Total for question: [**4 marks**]

(Delete the lines after the question and insert your response)

It’s a “has-a” relationship between two objects where one object has another object. In aggregation, the lifetime of the “part” is not tied to the lifetime of the “whole”, which means if the “whole” is deleted, the “parts” don’t have to be deleted. In composition it’s the opposite because the lifetime of the “part” is tied to the lifetime of the “whole”, which means if the “whole” is deleted, the “parts” have to be deleted too . Aggregation relationship is represented by a hollow diamond which goes to the “whole” side(the side that has another “parts”), whereas composition relationship is represented by a filled black diamond which also goes to the “whole” side.

Question2:

Clearly explain the difference between an object and a class (you may use examples or diagrams to assist)

Total for question [**3 marks**]

(Delete the lines after the question and insert your response)

A class is a template for making objects and an object is an instance of a class. That means when creating an object, you must allocate a memory by using a “new” keyword, whereas no memory allocated when creating a class. Also, objects store its state or values of fields , but classes don’t do that. Another difference is that objects can be manipulated and classes can’t because objects are stored in memory.

Question 3:

What is an access modifier and why is it important? -( 1 mark for its importance and usage)

In your explanations you must also indicate:

- The differences between public and private access modifiers. -(0.5 marks each)

* How are they shown in a UML diagram. –(0.5 marks each)

Total for question: [**3 marks**]

(Delete the lines after the question and insert your response)

It’s a keyword that specifies the accessibility of a class or class members. Access modifier is important because it helps to control the access to certain class members, which allows to encapsulate data and protect the integrity of a program. Public access modifier grants access to a class or a member from everywhere in the program, whereas private access modifier makes class members visible only to the class they were declared. In a UML diagram, private access modifier is shown with a “-“ (minus) sign and public access modifier is shown with a “+” (plus) sign.

Question 4: [10 marks]

A company rents out planes to private pilots. Bookings are made when a client reserves a plane for a specified date. Planes are picked up and dropped off at specified airports. This is usually done on the telephone. Planes can be booked for one (1) day only and are booked for the whole day.

The company would like to computerize their system. The company clerk will be entering information to a computer while speaking to a client on the telephone. The clerk will need to retrieve information (e.g. plane details, airport details, or client details if the clerk is dealing with an existing client) and create new bookings using the system. The clerk must be able to add, delete and update information on planes and airports. The clerk must also be able to add new clients as well as record new bookings and checking the availability of planes for specific dates.

You must come up with **appropriate** state (3 to 4 state values are acceptable) information for all classes involved. All collections must be shown as arrays.

Carry out an initial object-oriented design for the above specification, in which you should identify and show in a UML diagram:

* Classes that you think will be required.
* Their attributes and behaviours
* Any aggregation relationships
* Any other relationships between your classes

The UML Diagram should show all the information of the classes up to and including the coordinator class.

**Please note that you must show or display competency in the techniques taught in the classes.**

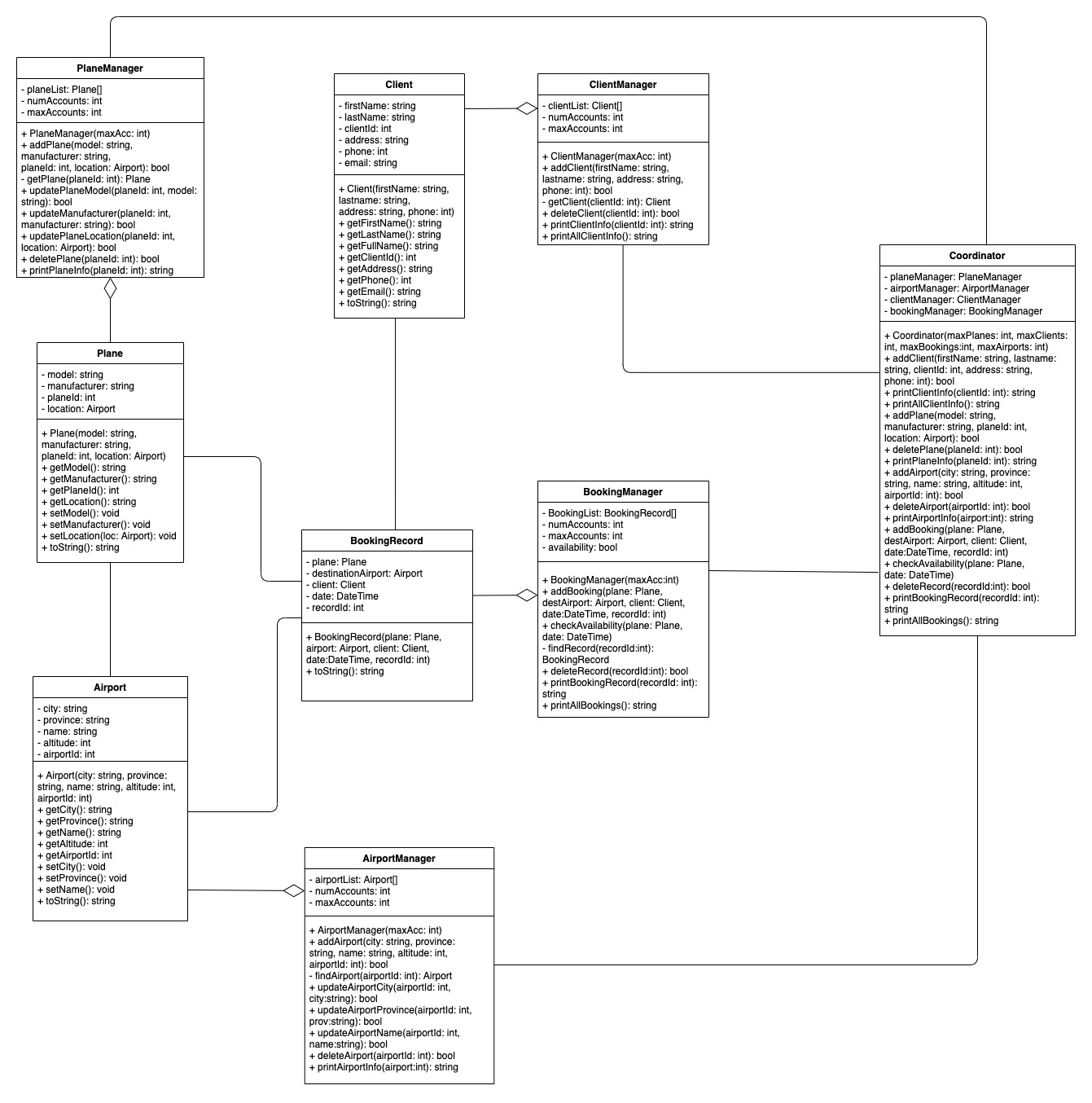
**Proper usage of concepts such as manager classes, coordinator classes, aggregation and responsibility delegation must be displayed in your design.**

**The names and ID numbers of all group members must be in the diagram.**

**You can use the website “draw.io” to do your design and export the image. This was shown in the recorded class.**

Insert your UML diagram on the next page

**Insert your UML diagram after this line.**



CAREFUL NOTE:

- Please safeguard your own team’s work.

- You are to do this project with your group members only.

- Note: Mobi Help members are NOT supposed to do or help you with your assignment.