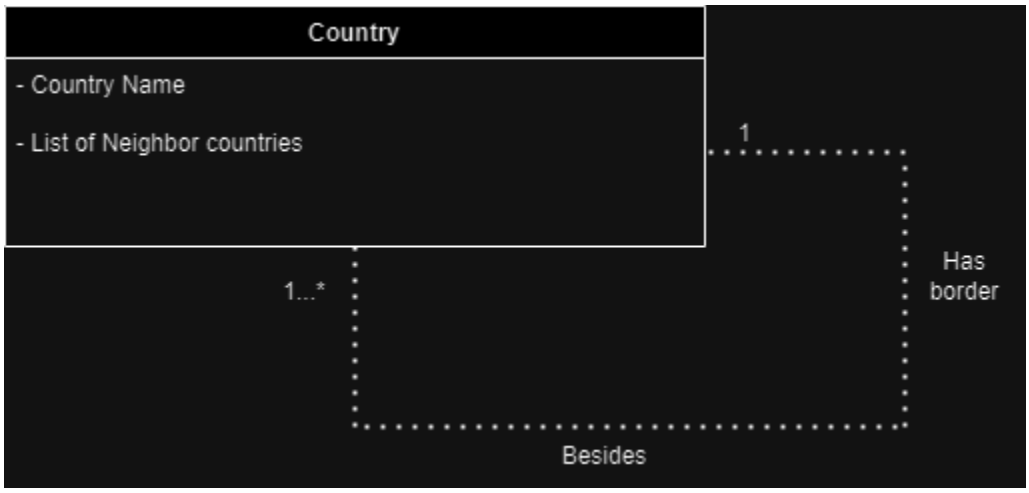
202201297 - Shail Patel
IT-314 Lab 4

Question - 1

Question:- Prepare a class diagram for the following object diagram that shows a portion of Europe.

Figure-1

Class Diagram



Question - 2

Question:- Prepare a class diagram for the object diagram given in Figure -2. Explain your multiplicity decisions. What is the smallest number of points required to construct a polygon? Does it make a difference whether or not point may be shared between polygons? Your answer should address the fact that points are ordered.

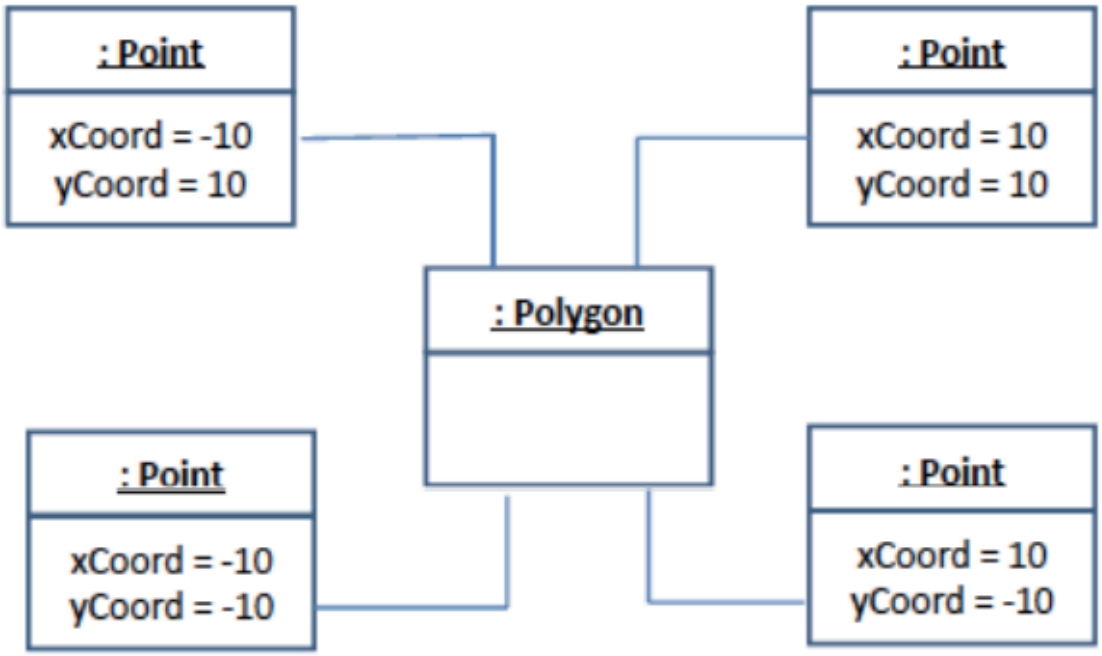
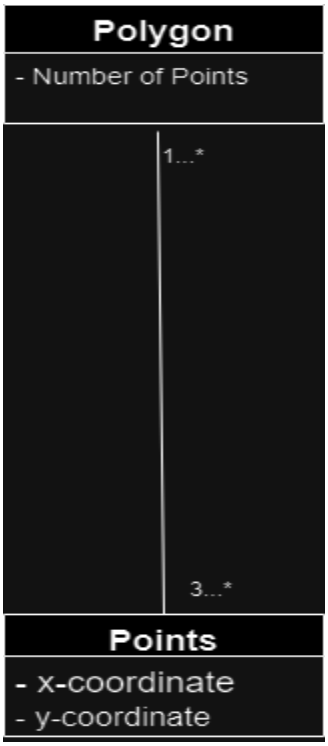


Figure - 2



Multiplicity Decisions:-
Polygon to Point:
Polygon to Point: 0..* (zero to many)
A polygon can have zero or many points. In practice, it should have at least 3 points.
These choices ensure flexibility in modeling while reflecting practical and theoretical use cases.

Does it make a difference whether or not point may be shared between polygons?

It does make a difference whether points may be shared between polygons, especially when considering ordered points in polygons. Here's a concise explanation:

Impact of Shared Points

1. **Shared Points Between Polygons**:
    - **Reusability**: Allowing points to be shared between polygons is practical and efficient. It means that the same set of points can be used to define multiple polygons, which is common in geometric modeling and real-world applications (e.g., mapping).
2. **Ordered Points**:

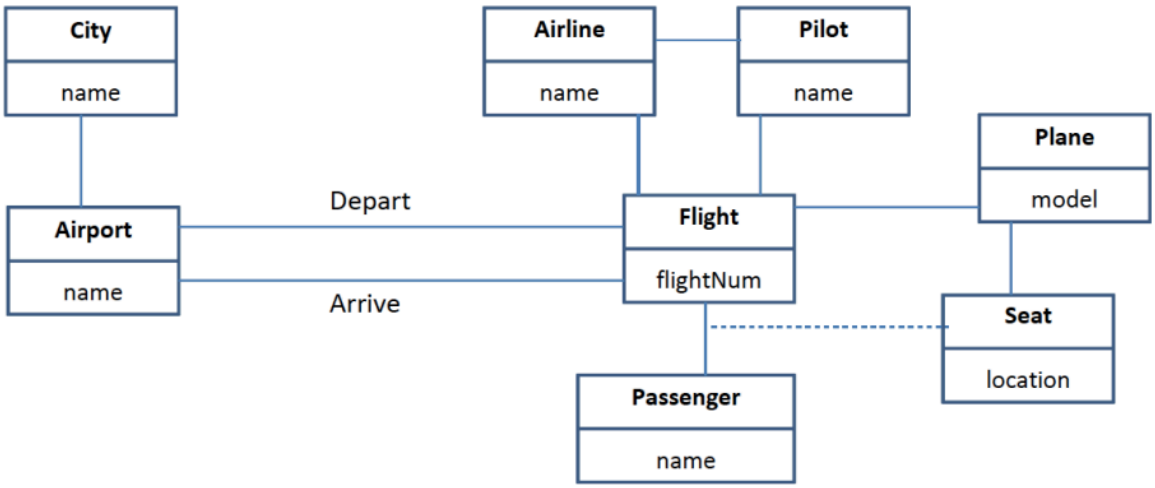
    - **Polygon Construction**: Points must be ordered to define the boundary of a polygon. The order determines the sequence of vertices and, therefore, the shape of the polygon.
    - **Sharing Points**: Even if points are shared between polygons, their order in each polygon is crucial. Each polygon must have its own specific order of points to accurately represent its shape. Shared points must be appropriately sequenced in each polygon to maintain correct geometry.

Summary

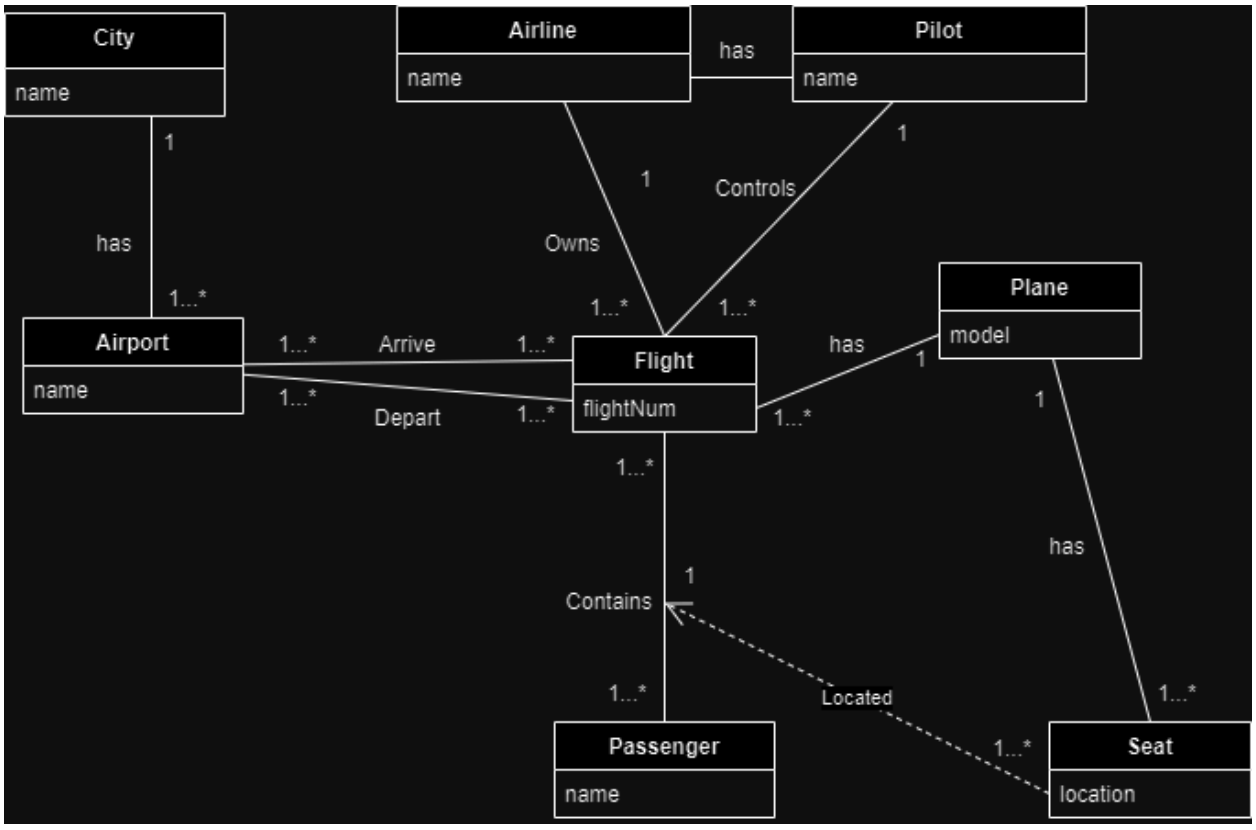
- **Shared Points**: Allowing points to be shared between polygons is beneficial for efficiency and flexibility.
- **Ordered Points**: Regardless of whether points are shared, the order of points is essential for correctly defining the shape of each polygon. Each polygon needs its own ordered sequence of points to be accurately represented.

Question - 3

Question:- Figure 3 is a partially completed class diagram of an air transportation system. Add multiplicities in the diagram. Also add association names to unlevelled associations.



Completed Class Diagram



Question - 4

Question:- We want to model a system for management of flights and pilots. An airline operates flights. Each airline has an ID. Each flight has an ID a departure airport and an arrival airport: an airport as a unique identifier. Each flight has a pilot and a co-pilot, and it uses an aircraft of a certain type; a flight has also a departure time and an arrival time. An airline owns a set of aircrafts of different types. An aircraft can be in a working state or it can be under repair. In a particular moment an aircraft can be landed or airborne. A company has a set of pilots: each pilot has an experience level: 1 is minimum, 3 is maximum. A type of airplane may need a particular number of pilots, with a different role (e.g.: captain, co-pilot, navigator): there must be at least one captain and one co-pilot, and a captain must have a level 3.

Answer:-

Entities and Attributes

1. Airline
    - ID (Primary Key)
2. Airport
    - ID (Primary Key)
    - Name (Optional)
3. Flight

    - ID (Primary Key)
    - Departure Airport ID (Foreign Key, references Airport)
    - Arrival Airport ID (Foreign Key, references Airport)
    - Pilot ID (Foreign Key, references Pilot)
    - Co-Pilot ID (Foreign Key, references Pilot)
    - Aircraft ID (Foreign Key, references Aircraft)
    - Departure Time
    - Arrival Time
    -
4. Aircraft
    - ID (Primary Key)
    - Type (e.g., Model, Make)
    - State (e.g., Working, Under Repair)
    - Status (e.g., Landed, Airborne)
5. Pilot
    - ID (Primary Key)
    - Experience Level (e.g., 1 to 3)
6. AircraftType
    - Type ID (Primary Key)
    - Required Pilots (e.g., Captain, Co-Pilot, Navigator)
    - Min Experience Level (e.g., Captain must be level 3)

Relationships

1. Airline and Aircraft
    - An airline owns multiple aircraft.
    - Each aircraft belongs to one airline.
2. Flight and Aircraft
    - A flight uses one aircraft.
    - Each aircraft can be used in multiple flights.
3. Flight and Pilot
    - A flight has one pilot and one co-pilot.
    - Pilots can be assigned to multiple flights.
4. Pilot and AircraftType
    - A pilot's role (e.g., Captain, Co-Pilot) is matched with the required roles for a specific aircraft type.
    - Each pilot has a specific role based on the aircraft type they are assigned to.
5. Aircraft and AircraftType
    - Each aircraft is of a specific type.
    - Each aircraft type has specific requirements for pilots.

Constraints

1. Pilot Experience Levels:
    - A captain must have an experience level of 3.
    - A co-pilot can have any experience level.
2. Aircraft Requirements:
    - Each aircraft type requires at least one captain and one co-pilot.
    - The number of pilots and their roles are specified for each aircraft type.