

## Airport.java

**Purpose:** Represents an airport with international and domestic terminals. It manages planes and bookings.

### Attributes:

- **internationalTerminal**: A Terminal object representing the international terminal.
  - **domesticTerminal**: A Terminal object representing the domestic terminal.
  - **planes**: A list of Plane objects representing all the planes at the airport.
  - **bookings**: A list of Booking objects representing all the bookings at the airport.
- Methods:**
- **main(String[] args)**: The main method to run the airport operations.

## Terminal.java

**Purpose:** Represents a terminal in the airport. It manages planes and bookings.

### Attributes:

- **terminalName**: A String representing the name of the terminal.
- **planes**: A list of Plane objects representing all the planes at the terminal.
- **bookings**: A list of Booking objects representing all the bookings at the terminal.

### Methods:

- **Constructors:** Overloaded constructors to create a Terminal object with or without initial planes and bookings.
- **addPlane(Plane plane)**: Adds a plane to the terminal.
- **displayTerminalDetails()**: Displays details of the terminal, including planes and bookings.

## Plane.java (Abstract)

**Purpose:** Represents an abstract class for a plane. It manages passengers and capacity.

### Attributes:

- **planeType**: A String representing the type of the plane.
- **planeVersion**: A String representing the version of the plane.
- **capacity**: An integer representing the capacity of the plane.
- **passengers**: A list of Passenger objects representing the passengers on the plane.

### Methods:

- **Constructors:** An abstract constructor to initialize common attributes of all types of planes.
- **addPassenger(Passenger passenger)**: Adds a passenger to the plane.
- **removePassenger(Passenger passenger)**: Removes a passenger from the plane.
- **displayPlaneDetails()**: Abstract method to display details of the plane.

## PassengerPlane.java (Hierarchical)

**Purpose:** Represents a passenger plane, a type of Plane. It manages crew members and passengers.

### Attributes:

- **crew**: A list of CrewMember objects representing the crew members of the plane.
- Methods:**

- **Constructors:** Overloaded constructors to create a PassengerPlane object with or without initial crew members.
- **addCrewMember(CrewMember crewMember):** Adds a crew member to the plane.

### **CargoPlane.java (Multiple Inheritance)**

- **Purpose:** Represents a cargo plane, a type of Plane. It manages cargo and flight operations.
- **Attributes:**
  - **maxCargoWeight:** A double representing the maximum cargo weight of the plane.
  - **cargos:** An array of Cargo objects representing the cargo on the plane.
- **Methods:**
  - **Constructors:** Creates a CargoPlane object with the given parameters.
  - **loadCargo(Cargo cargo):** Loads cargo onto the plane.
  - **getMaxCargoWeight():** Returns the maximum cargo weight of the plane.

### **Booking.java (Interface)**

- **Purpose:** Represents a booking for a plane. It manages flight bookings.
- **Methods:**
  - **bookFlight(Plane plane, Passenger passenger, Cargo cargo):** Books a flight for a passenger with cargo.
  - **cancelBooking():** Cancels a flight booking.
  - **getBookingDetails():** Gets the details of the booking.

### **FlightOperations.java (Interface)**

- **Purpose:** Represents flight operations for a plane. It manages takeoff and landing.
- **Methods:**
  - **takeOff():** Performs takeoff operation for a flight.
  - **land():** Performs landing operation for a flight.

