abstract class Flight {

protected String flightNumber;

protected String origin;

protected String destination;

protected double baseFare;

public Flight(String flightNumber, String origin, String destination, double baseFare) {

this.flightNumber = flightNumber;

this.origin = origin;

this.destination = destination;

this.baseFare = baseFare;

}

public abstract double calculateFare();

public void displayDetails() {

System.out.println("Flight Number: " + flightNumber);

System.out.println("Origin: " + origin);

System.out.println("Destination: " + destination);

System.out.println("Base Fare: " + baseFare);

}

}

class DomesticFlight extends Flight {

private static final double domesticTax = 0.10;

public DomesticFlight(String flightNumber, String origin, String destination, double baseFare) {

super(flightNumber, origin, destination, baseFare);

}

@Override

public double calculateFare() {

return baseFare + (baseFare \* domesticTax);

}

@Override

public void displayDetails() {

super.displayDetails();

System.out.println("Domestic Tax: " + (domesticTax \* 100) + "%");

System.out.println("Total Fare: " + calculateFare());

}

}

class InternationalFlight extends Flight {

private static final double internationalTax = 0.20;

private double immigrationFee;

public InternationalFlight(String flightNumber, String origin, String destination, double baseFare, double immigrationFee) {

super(flightNumber, origin, destination, baseFare);

this.immigrationFee = immigrationFee;

}

@Override

public double calculateFare() {

return baseFare + (baseFare \* internationalTax) + immigrationFee;

}

@Override

public void displayDetails() {

super.displayDetails();

System.out.println("International Tax: " + (internationalTax \* 100) + "%");

System.out.println("Immigration Fee: " + immigrationFee);

System.out.println("Total Fare: " + calculateFare());

}

}

public class FlightManager {

public static void main(String[] args) {

Flight[] flights = new Flight[4];

flights[0] = new DomesticFlight("D101", "Delhi", "Mumbai", 5000);

flights[1] = new InternationalFlight("I201", "Mumbai", "Dubai", 15000, 3000);

flights[2] = new DomesticFlight("D102", "Bangalore", "Chennai", 4000);

flights[3] = new InternationalFlight("I202", "Delhi", "London", 25000, 5000);

for (Flight flight : flights) {

System.out.println("---------------");

flight.displayDetails();

}

}

}

OUTPUT

---------------

Flight Number: D101

Origin: Delhi

Destination: Mumbai

Base Fare: 5000.0

Domestic Tax: 10.0%

Total Fare: 5500.0

---------------

Flight Number: I201

Origin: Mumbai

Destination: Dubai

Base Fare: 15000.0

International Tax: 20.0%

Immigration Fee: 3000.0

Total Fare: 21000.0

---------------

Flight Number: D102

Origin: Bangalore

Destination: Chennai

Base Fare: 4000.0

Domestic Tax: 10.0%

Total Fare: 4400.0

---------------

Flight Number: I202

Origin: Delhi

Destination: London

Base Fare: 25000.0

International Tax: 20.0%

Immigration Fee: 5000.0

Total Fare: 35000.0