Course : CS 1083

Assignement 1

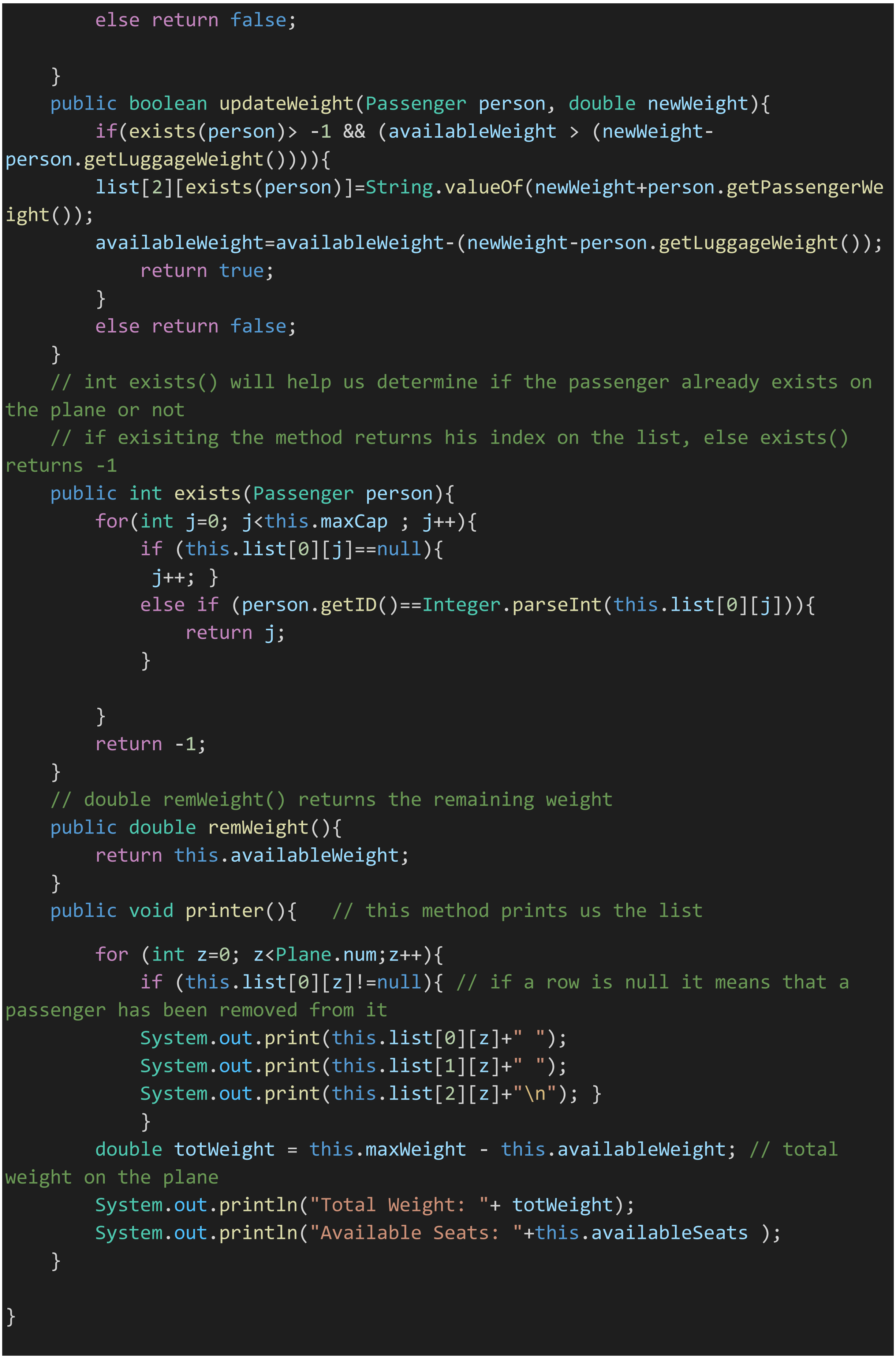
Name : Omar Sebri

Student ID : 3722350

Source Code :

```
public class Passenger{
    public static int i_d=0; // static int i_d helps us generate automatically
the ID
    public int id;
    public String name;
    double passengerWeight;
    double luggageWeight;
    public Passenger(String name, double passengerWeight, double
luggageWeight){
        this.name=name;
        this.passengerWeight =passengerWeight;
        this.luggageWeight=luggageWeight;
        i_d++;
        this.id=i_d;
    }
    public String getName(){
        return this.name;
    }
    public double getPassengerWeight(){
        return this.passengerWeight;
    }
    public double getLuggageWeight(){
        return this.luggageWeight;
    }
    public int getID(){
        return (this.id+999);
    }
    public void setName(String name){
        this.name=name;
    }
    public void setPassengerWeight(double passengerWeight){
        this.passengerWeight=passengerWeight;
    }
    public void setLuggageWeight(double luggageWeight){
        this.luggageWeight=luggageWeight;
    }
}
```

```
public class Plane{
    public double maxWeight;
    public int maxCap; // maxCap maximum capacity
    public double availableWeight;
    public int availableSeats;
    String [][] list ;
    public Plane(int maxCap, double maxWeight){
        this.maxCap=maxCap;
        this.maxWeight=maxWeight;
        this.availableWeight=maxWeight;
        this.availableSeats=maxCap;
        this.list= new String[3][maxCap+1];
        // our list contains 3 columns: one for id, one for name and one for
total weight
    }
    //public double availableWeight=maxWeight;
    //public int availableSeats=maxCap;
    public static int num=0; // int static num represents the total number of
people that have been on the list of passengers
    // static int num will be used to print the current passengers on the
plane efficiently
    public boolean addPass(Passenger person){
       double
personWeight=person.getPassengerWeight()+person.getLuggageWeight();
        if((personWeight < availableWeight) && (availableSeats>0) &&
exists(person)==-1){
            availableSeats--;
            availableWeight-=personWeight;
            list[0][num]=String.valueOf(person.getID());
            list[1][num]= person.getName();
            list[2][num]=String.valueOf(personWeight);
            num++;
            return true;
        }
        else return false;
    }
    public boolean removePass(Passenger person){
        if(this.exists(person)>-1){
            this.availableSeats++;
            this.availableWeight=person.getLuggageWeight()+person.getPassenger
Weight()+availableWeight;
            int k = exists(person);
            list[0][k]=null;
            list[1][k]=null;
            list[2][k]=null;
            return true;
        }
```



```
        else return false;

    }
    public boolean updateWeight(Passenger person, double newWeight){
        if(exists(person)> -1 && (availableWeight > (newWeight-
person.getLuggageWeight()))){
        list[2][exists(person)]=String.valueOf(newWeight+person.getPassengerWe
ight());
        availableWeight=availableWeight-(newWeight-person.getLuggageWeight());
            return true;
        }
        else return false;
    }
    // int exists() will help us determine if the passenger already exists on
the plane or not
    // if exisiting the method returns his index on the list, else exists()
returns -1
    public int exists(Passenger person){
        for(int j=0; j<this.maxCap ; j++){
            if (this.list[0][j]==null){
             j++; }
            else if (person.getID()==Integer.parseInt(this.list[0][j])){
                return j;
            }

        }
        return -1;
    }
    // double remWeight() returns the remaining weight
    public double remWeight(){
        return this.availableWeight;
    }
    public void printer(){    // this method prints us the list

        for (int z=0; z<Plane.num;z++){
            if (this.list[0][z]!=null){ // if a row is null it means that a
passenger has been removed from it
            System.out.print(this.list[0][z]+" ");
            System.out.print(this.list[1][z]+" ");
            System.out.print(this.list[2][z]+"\n"); }
            }
        double totWeight = this.maxWeight - this.availableWeight; // total
weight on the plane
        System.out.println("Total Weight: "+ totWeight);
        System.out.println("Available Seats: "+this.availableSeats );
    }

}
```

```java
public class Driver{
    public static void main(String[] args) {
        Passenger Fred = new Passenger("Fred", 120.0, 50.0);
        Passenger Ahmed = new Passenger("Ahmed", 70.0, 50.0);
        Passenger Caleb = new Passenger("Caleb", 90.0, 50.0);
        Passenger Lisa = new Passenger("Lisa", 60.0, 300.0);
        Plane p = new Plane(10, 600.0);
        System.out.println(p.addPass(Fred));
        System.out.println(p.addPass(Caleb));
        System.out.println(p.addPass(Ahmed));
        System.out.println(p.addPass(Lisa));
        System.out.println(p.updateWeight(Caleb, 30));
        System.out.println(p.updateWeight(Caleb, 300.0));
        System.out.println(p.removePass(Fred));
        System.out.println(p.removePass(Lisa));
        p.printer();
    }
}
```

Sample output :

```
true

true

true

false

true

false

true

false

1002 Caleb 120.0

1001 Ahmed 120.0

Total Weight: 240.0

Available Seats: 8
```

Output Discussion:

- The first 3 outputs were « true » because we added 3 passengers that obeyed the rules of the plane successfully.
- Lisa’s luggage would exceed the total capacity of the plane so the 4th output was “false” because Lisa wasn’t added to the plane

- The 5th output was “true” because Caleb’s new luggage’s weight obeys the rule of the plane and so its value was updated successfully
- The 6th output was false because Caleb’s new luggage’s weight would exceed the plane’s capacity and so it wasn’t updated with the new value
- Fred has been added before on the plane and so we were able to remove him from the list and that’s why our output was true
- Lisa wasn’t added to the plane in the first place that’s why we couldn’t remove her and so out 8th output has been false
- Our 9th output consisted of the remaining passengers’ names, ID and their weights combined with their luggage
- Our 10th output was the total weight on the plane
- Out 11th output was the available number of seats on the plane (it wasn’t asked from us to print but I just wanted to signal it)