Selected files

9 printable files

Main.java
BuyTicketFCFS.java
Scheduling/ShortestJobNext.java
Scheduling/Priority.java
Scheduling/Fcfs.java
Entities/TicketType.java
Entities/Ticket.java
Entities/Passenger.java
Entities/Luggage.java

Main.java

```java
import java.io.File;
import java.util.ArrayList;
import java.util.Scanner;

import Entities.Luggage;
import Entities.Passenger;
import Entities.TicketType;
import Scheduling.*;


public class Main {
    public static void main(String[] args) {
        ArrayList<Passenger> passengers = new ArrayList<>();
        ArrayList<Luggage> luggages = new ArrayList<>();
        loadPassenger(passengers);
        loadLuggages(luggages);

        System.out.print("Enter [1] FCFS [2] Priority [3] SJN [4] Buy Ticket FCFS:");
        Scanner input = new Scanner(System.in);
        switch (Integer.parseInt(input.nextLine())) {
            case 1:
                Fcfs.sort(passengers);
                break;
            case 2:
                Priority.sort(passengers);
                break;
            case 3:
                ShortestJobNext.lightestLuggageFirst(luggages);;
                break;
            case 4:
                BuyTicketFCFS.buyticket();
                passengers.clear();
                loadPassenger(passengers);
                print(passengers);
                break;
            default:
                System.out.println("UwU daddy piwck somwwnethuing bewter next time 0w0");
                break;
        }

```

```
41      }
42
43      public static void print(ArrayList<Passenger> passengers){
44          //we will print out arrival order
45          System.out.println("ARIAVAL ORDER");
46          System.out.printf("%-4s %-25s %-20s %-4s\n","NO", "NAME","TICKET TYPE",
"SEAT" );
47          for (Passenger passenger : passengers)
48          {
49              System.out.printf("%-4s %-25s %-20s %-4s\n",
passengers.indexOf(passenger)+1 ,passenger.getName(),
passenger.getTicketType().toString(), passenger.getSeat());
50          }
51      }
52
53      public static void loadPassenger(ArrayList<Passenger> passengers){
54          try{
55              //DO REPLACE THIS PATH IN YOUR OS
56              Scanner sc = new Scanner(new File("Ticket.csv"));
57              sc.nextLine(); //SKIPS COLUMN HEADER
58              while(sc.hasNextLine())
59              {
60                  String line = sc.nextLine();
61                  String[] columns = line.split(",");
62                  TicketType ticketType = TicketType.BUSINESS_CLASS; // FOR
INITIALIZATION
63                  switch(columns[2])
64                  {
65                      case "FIRST_CLASS":
66                      ticketType = TicketType.FIRST_CLASS;
67                      break;
68                      case "BUSINESS_CLASS":
69                      ticketType = TicketType.BUSINESS_CLASS;
70                      break;
71                      case "PREMIUM_ECONOMY":
72                      ticketType = TicketType.PREMIUM_ECONOMY;
73                      break;
74                      case "ECONOMY":
75                      ticketType = TicketType.ECONOMY;
76                      break;
77
78                  }
79                  Passenger p = new
Passenger(Integer.parseInt(columns[0]),columns[1], ticketType, columns[3],
Integer.parseInt(columns[4]) );
80                  passengers.add(p);
81              }
82          }catch (Exception e){
83              System.out.println(e.toString());
84          }
85      }
86      public static void loadLuggages(ArrayList<Luggage> luggages)
87      {
88          try{
89              //DO REPLACE THIS PATH IN YOUR OS
90              Scanner sc = new Scanner(new File("Luggage.csv"));
91              sc.nextLine(); //SKIPS COLUMN HEADER
92              while(sc.hasNextLine())
93              {
94
95                  String line = sc.nextLine();
```

```
                String[] columns = line.split(",");
                //Passenger ID,Luggage ID,Luggage Weight (kg),Luggage Color
                luggages.add(new Luggage(Integer.parseInt(columns[0]),
(columns[1]), Float.parseFloat(columns[2]), columns[3]));
            }
        }catch (Exception e){
            System.out.println(e.toString());
        }
    }
}
```

BuyTicketFCFS.java

```
import java.io.File;
import java.io.FileWriter;
import java.util.ArrayList;
import java.util.Random;
import java.util.Scanner;
import Entities.Passenger;
import Entities.TicketType;

public class BuyTicketFCFS {
    public static String getrandseat() {
        Random rand = new Random();
        int seatnum = rand.nextInt(38 - 1) + 1;
        char seatchar = (char) (rand.nextInt(6) + 'A');
        String seat = seatchar + Integer.toString(seatnum);
        return seat;
    }

    public static void buyticket() {
        String[] names = {
            "Alice", "Bob", "Charlie", "David", "Emma", "Frank", "Grace", "Henry", "I
"Jack",
            "Kate", "Liam", "Mia", "Noah", "Olivia", "Peter", "Quinn", "Ruby", "Samue
"Taylor",
            "Uma", "Victor", "Wendy", "Xavier", "Yara", "Zoe", "Andrew", "Bella", "Ca
"Daisy",
            "Ethan", "Faith", "George", "Hannah", "Isaac", "Julia", "Kevin", "Lily",
"Nora",
            "Oscar", "Penelope", "Quentin", "Rachel", "Simon", "Sophia", "Tristan", "
"Wyatt",
            "Xander", "Yasmine", "Zachary", "Abigail", "Benjamin", "Charlotte", "Dani
"Evelyn",
            "Fiona", "Gabriel", "Harper", "Ian", "Jasmine", "Kai", "Luna", "Matthew",
"Oliver",
            "Piper", "Quincy", "Rose", "Sebastian", "Thea", "Vincent", "Willow", "Xav
"Yara",
            "Zara", "Adam", "Beth", "Cameron", "Diana", "Elijah", "Felicity", "Gavin"
"Isaiah",
            "Jade", "Kyle", "Lauren", "Michael", "Natalie", "Owen", "Paige", "Quinn",
            "Samuel", "Tara", "Tyler", "Victoria", "Wesley", "Xena", "Yvette", "Zacha
"Alexa",
            "Brandon", "Chloe", "Dylan", "Emily", "Finn", "Grace", "Hayden", "Ivy", "
"Jessica",
            "Kevin", "Kylie", "Logan", "Madison", "Nathan", "Nora", "Oliver", "Olivia
            "Penelope", "Quentin", "Quinn", "Ryan", "Ruby", "Samantha", "Samuel", "Tr
"Uma",
            "Victoria", "Wyatt", "Willow", "Xavier", "Xena", "Yara", "Yasmine", "Zach
```

```
            "Aaron", "Brianna", "Caleb", "Danielle", "Ethan", "Emma", "Gavin", "Gabri
"Henry",
            "Hailey", "Isaac", "Isabella", "Jack", "Jasmine", "Kai", "Katherine", "Li
            "Mason", "Maya", "Nathan", "Natalie", "Owen", "Olivia", "Parker", "Peyton
            "Rebecca", "Samuel", "Sophia", "Tristan", "Taylor", "Victor", "Violet", "
"Willow",
            "Xavier", "Ximena", "Yara", "Yasmine", "Zachary", "Zoe"
        };
        Random random = new Random();
        TicketType[] types = TicketType.values();
        int lastid = 0;
        ArrayList<String> seats = new ArrayList<>();
        ArrayList<Passenger> queue = new ArrayList<Passenger>(10);
        try {
            Scanner reader = new Scanner(new File("Ticket.csv"));
            reader.nextLine();
            while (reader.hasNextLine()) {
                String line = reader.nextLine();
                String[] columns = line.split(",");
                lastid = Integer.parseInt(columns[0]);
                seats.add(columns[3]);
            }
            reader.close();
            for (int i = 0; i < 10; i++) {
                String seat = getrandseat();
                while (seats.contains(seat)) {
                    seat = getrandseat();
                }
                int num = Integer.parseInt(seat.substring(1));
                int index = 0;
                if(num >= 5 && num <= 10){
                    index = 1;
                }else if(num >= 11 && num <= 21){
                    index=2;
                }else{
                    index = 3;
                }
                queue.add(new Passenger(lastid+1+ i, names[random.nextInt(183)],
types[index],seat,num));

            }
        } catch (Exception e) {
            System.out.println(e.toString());
        }
        try {
        FileWriter writer = new FileWriter("Ticket.csv",true);
        writer.write("\n");
        for(Passenger x: queue){

writer.write(x.getID()+","+x.getName()+","+x.getTicketType()+","+x.getSeat()+","+x.ge
        if(queue.indexOf(x) < 9){
            writer.write("\n");
        }
        }
        writer.close();
        } catch (Exception e) {
        System.out.println("An error occurred.");
        e.printStackTrace();
        }
    }
}
```

```
92
```

Scheduling/ShortestJobNext.java

```
1  package Scheduling;
2
3  import java.util.ArrayList;
4
5  import Entities.Luggage;
6  import Entities.Passenger;
7
8  public class ShortestJobNext {
9      public static void sort(ArrayList<Passenger> passengers){
10
11         // sort by distance to the seat
12         passengers.sort((o1, o2) ->
   ((Integer)o1.getDistance()).compareTo((Integer)o2.getDistance()));
13         System.out.println("\n\n");
14
15         //print out the Boarding order
16         System.out.println("BOARDING ORDER");
17         System.out.printf("%-4s %-25s %-20s %-4s\n","NO", "NAME","TICKET TYPE",
   "SEAT" );
18         for (Passenger passenger : passengers)
19         {
20             System.out.printf("%-4s %-25s %-20s %-4s\n",
   passengers.indexOf(passenger)+1 ,passenger.getName(),
   passenger.getTicketType().toString(), passenger.getSeat());
21         }
22     }
23
24     //sort based on weight
25     public static void lightestLuggageFirst(ArrayList<Luggage> luggages)
26     {
27         luggages.sort((l1, l2) -> Float.compare(l1.getLuggageWeight(),
   l2.getLuggageWeight()));
28         System.out.println("\n\n");
29
30         //print out the Boarding order
31         System.out.println("LUGGAGE BOARDING ORDER");
32         //Passenger ID,Luggage ID,Luggage Weight (kg),Luggage Color
33         System.out.printf("%-15s %-15s %-15s %-9s\n","PASSENGER ID", "LUGGAGE
   ID","LUGGAGE WEIGHT (KG)", "COLOR" );
34         for (Luggage luggage : luggages)
35         {
36             System.out.printf("%-15d %-15s %-15.2f %-9s\n",
   luggage.getPassengerID(), luggage.getLuggageId(), luggage.getLuggageWeight(),
   luggage.getLuggageColor());
37         }
38     }
39 }
40
```

Scheduling/Priority.java

```
1  package Scheduling;
```

```
2
3  import java.util.ArrayList;
4
5  import Entities.Passenger;
6
7  public class Priority {
8      public static void sort(ArrayList<Passenger> passengers){
9
10         //start to sort passengers by priority for the boarding order
11         passengers.sort((o1, o2) ->
   o1.getTicketType().compareTo(o2.getTicketType()));
12         System.out.println("\n\n");
13
14         //print out the Boarding order
15         System.out.println("BOARDING ORDER");
16         System.out.printf("%-4s %-25s %-20s %-4s\n","NO", "NAME","TICKET TYPE",
   "SEAT" );
17         for (Passenger passenger : passengers)
18         {
19             System.out.printf("%-4d %-25s %-20s %-4s\n", passenger.getID()
   ,passenger.getName(), passenger.getTicketType().toString(), passenger.getSeat());
20         }
21     }
22 }
23
```

Scheduling/Fcfs.java

```
1  package Scheduling;
2
3  import java.util.ArrayList;
4  import java.util.Collections;
5
6  import Entities.Passenger;
7
8  public class Fcfs {
9
10     public static void sort(ArrayList<Passenger> passengers) {
11         System.out.println("\n\n");
12
13         Collections.shuffle(passengers);
14         System.out.println("ARRIVAL ORDER");
15         System.out.printf("%-4s %-25s %-20s %-4s\n", "NO", "NAME", "TICKET TYPE",
   "SEAT");
16         for (Passenger passenger : passengers) {
17             System.out.printf("%-4s %-25s %-20s %-4s\n",
   passengers.indexOf(passenger) + 1, passenger.getName(),
18                     passenger.getTicketType().toString(), passenger.getSeat());
19         }
20         // print out the oder the tickets were bought
21         System.out.println("BOARDING ORDER");
22         System.out.printf("%-4s %-25s %-20s %-4s\n", "NO", "NAME", "TICKET TYPE",
   "SEAT");
23         for (Passenger passenger : passengers) {
24             System.out.printf("%-4s %-25s %-20s %-4s\n",
   passengers.indexOf(passenger) + 1, passenger.getName(),
25                     passenger.getTicketType().toString(), passenger.getSeat());
26         }
```

```
    }
}

```

Entities/TicketType.java

```
package Entities;

public enum TicketType {
    FIRST_CLASS,
    BUSINESS_CLASS,
    PREMIUM_ECONOMY,
    ECONOMY
}

```

Entities/Ticket.java

```
package Entities;

public class Ticket {
    private TicketType ticketType;
    private String seat;
    private Integer distance;
    public Ticket(TicketType ticketType, String seat, Integer distance) {
        this.ticketType = ticketType;
        this.seat = seat;
        this.distance = distance;
    }
    public TicketType getTicketType() {
        return ticketType;
    }

    public String getSeat() {
        return seat;
    }

    public int getDistance() {
        return distance;
    }

}

```

Entities/Passenger.java

```
package Entities;

public class Passenger {
    private int ID;
    private String name;
```

```
    private Ticket ticket;
    public Passenger(int ID, String name, TicketType type,String seat,int distance)
{
        this.name = name;
        this.ID = ID;
        this.ticket = new Ticket(type, seat, distance);
    }

    public int getID() { return ID; }
    public String getName() {
        return name;
    }
    public TicketType getTicketType() {
        return ticket.getTicketType();
    }
    public int getDistance() {
        return ticket.getDistance();
    }
    public String getSeat(){
        return ticket.getSeat();
    }
}

```

Entities/Luggage.java

```
package Entities;

public class Luggage {
    private int passengerID;
    private float luggageWeight;
    private String luggageColor, luggageId;

    public Luggage(int pid, String lid, float weight, String color)
    {
        passengerID = pid;
        luggageId = lid;
        luggageWeight = weight;
        luggageColor = color;
    }

    public String getLuggageColor() {
        return luggageColor;
    }
    public String getLuggageId() {
        return luggageId;
    }
    public float getLuggageWeight() {
        return luggageWeight;
    }
    public int getPassengerID() {
        return passengerID;
    }

}

```