Assignment 4 – Oisin Mc Laughlin 22441106

Question 1 – Analysis

So for this assignment we have the freedom to create the main layout of the project. For this I have decided to create; TestTravel, Booking, Trip, BusTemplate, BusEireann, CityLink, GoBus. My idea for this assignment is to have the BusTemplate class be an abstract class where most of the work happens. BusEireann, CityLink and GoBus will be subclasses of BusTemplate and the only thing I will be writing in them is calling the super() reference variable with cName (company name), seats and price as parameters. The Trip class will just hold information about the actual trip such as origin, destination, departure date, departure time, arrival date, arrival time and the id of the trip. The id of the trip will be very important as that's how the customer will choose where they want to go. The only other thing I will have in this class is a toString to display all the information about the trip, this will be called in BusTemplate. BusTemplate is where the good stuff will happen, it will have an array to hold every trip, a trip object, cName, seats, price and total price. There will be a constructor to initialise instance variables, method to add trips to the array, set seats, getSeats, to check if the trip exists when customer is booking, another to check the avalibility (if seats on bus are greater than or equal to ones being booked then performing an action depending on the outcome of it) and finally a toString displaying all the information about the bus and the trips that they offer. The last class that I will need will be a test class, this class will test to see if the bus company can display all the bus information and the trips that each companies offer, to make a successful booking and reduce the amount of seats by the amount being booked and to also check what happens if someone tries to book more seats than the seats that are available on the bus.

Question 2 – Coding

Trip:

```
public class Trip
{
   String origin, dest, dDate, aDate, dTime, aTime;
   int id;

   public Trip(String origin, String dest, String dDate, String aDate, String dTime, String aTime,
int id)
   {
      this.origin = origin;
      this.dest = dest;
      this.dDate = dDate;
      this.aDate = aDate;
      this.dTime = dTime;
      this.aTime = aTime;
      this.id = id;
   }

   @Override
   public String toString() {
      //toString returning all information of the trip as a string.
      String out = "";

      out+= "\nOrigin:\n-";
      out+= origin;
      out+= "\nDestination:\n-";
      out+= dest;
      out+= "\nDeparture Date:\n-";
      out+= dDate;
      out+= "\nDeparture Time:\n-";
      out+= dTime;
      out+= "\nArrival Date:\n-";
      out+= aDate;
      out+= "\nArrival Time:\n-";
      out+= aTime;
      out+= "\nID:\n-";
      out+= id;

      return out;
   }
}
```

Booking:

```
import java.util.Random;

public class Booking
{
    //Trip trip;
    BusTemplate bus;
    int numSeats;
    int tripId;
    int bookingId;

    public Booking(int numSeats, int tripId, BusTemplate bus)
    {
        //this.trip = trip;
        this.numSeats = numSeats;
        this.tripId = tripId;
        this.bus = bus;
        this.bookingId = genId();
    }

    public int genId() {
        //Method to generate a random id between 1 and 999
        Random rand = new Random();
        int randomNum = rand.nextInt(999) + 1;

        return randomNum;
    }

    public boolean makeBook() {
        //Calls checkAval from BusTemplate with numSeats and tripId as parameters.
        return bus.checkAval(numSeats, tripId);
    }

    public int getBookId() {
        return bookingId;
    }
}
```

BusTemplate:

```
import java.util.ArrayList;

public abstract class BusTemplate
{
    public ArrayList<Trip> trips = new ArrayList<>();
    Trip trip;
    String cName;
    int seats;
    double price;
    double totalP;

    public BusTemplate(String cName, int seats, double price)
    {
        this.cName = cName;
        this.seats = seats;
        this.price = price;
    }

    public void addTrip(Trip trip) {
        //Adds trip object to trip array.
        trips.add(trip);
    }

    public void setSeats(int numS) {
        seats = numS;
    }
    public int getSeats() {
        return seats;
    }

    public Trip findTrip(int id) {
        for (Trip trip : trips) {
            //Loops through trip array and checks if the id entered matches any id's in the trip
array
            if (trip.id == id) {
                return trip;
            }
        }
        //otherwise return null
        return null;
    }

    public boolean checkAval(int numBook, int id) {
        Trip trip = findTrip(id);
        //Checks if trip exists.
        if (trip == null) {
            //If trip doesn't exist, print message and return false.
            System.out.println(id + " does not exist\n");
            return false;
        }

        //If seats on bus is greater than or equal to the number of seats being booked,
        if (this.seats >= numBook) {
            //Print message, reduce seats on bus by number of seats being booked
```

```
      System.out.println("Booking successful for " + numBook + " seats\n");
      this.seats -= numBook;
      //Print message and multiply number of seats booked by the price of bus, print total
      System.out.println("\nYour Journey:\n" + findTrip(id));
      totalP = numBook * price;
      System.out.println("\nTotal Price: €" + totalP);
      //Return true
      return true;
    } else {
      //Else print message and return false.
      System.out.println("Booking unsuccessful, not enough seats avalible");
      return false;
    }
  }

  @Override
  public String toString() {
    //Returns string with all information of bus and trips that bus company offers by looping
through trip array.
    String out = "";

    out+= "\n\n=Bus Information=\n\n";
    out+= "\nCompany\n-";
    out+= cName;
    out+= "\nAvalible Seats:\n-";
    out+= seats;
    out+= "\nPrice:\n-";
    out+= price;

    out+= "\n\n=Trip Information=\n";
    for (Trip trip : trips) {
      out+= trip.toString();
      out+= "\n";
    }

    out+= "\n\n";
    return out;
  }
}
```

BusEireann:

```
public class BusEireann extends BusTemplate
{
  public BusEireann(String cName, int seats, double price)
```

```
    {
        //Subclass of BusTemplate, super called with cName, seats and price
        super(cName, seats, price);
    }
}
```

CityLink:

```
public class CityLink extends BusTemplate
{
    public CityLink(String cName, int seats, double price)
    {
        //Subclass of BusTemplate, super called with cName, seats and price
        super(cName, seats, price);
    }
}
```

GoBus:

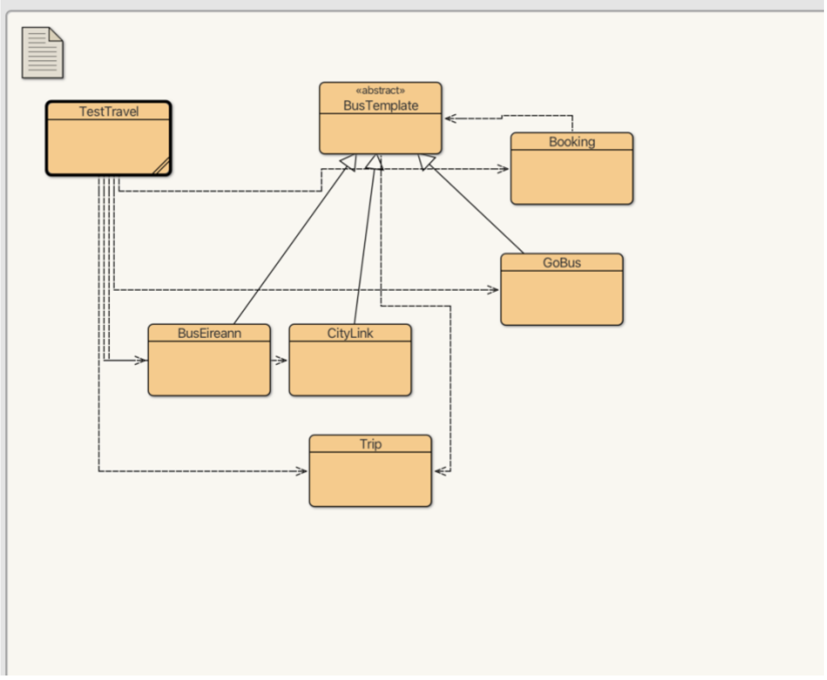
```
public class GoBus extends BusTemplate
{
    public GoBus(String cName, int seats, double price)
    {
        //Subclass of BusTemplate, super called with cName, seats and price
        super(cName, seats, price);
    }
}
```

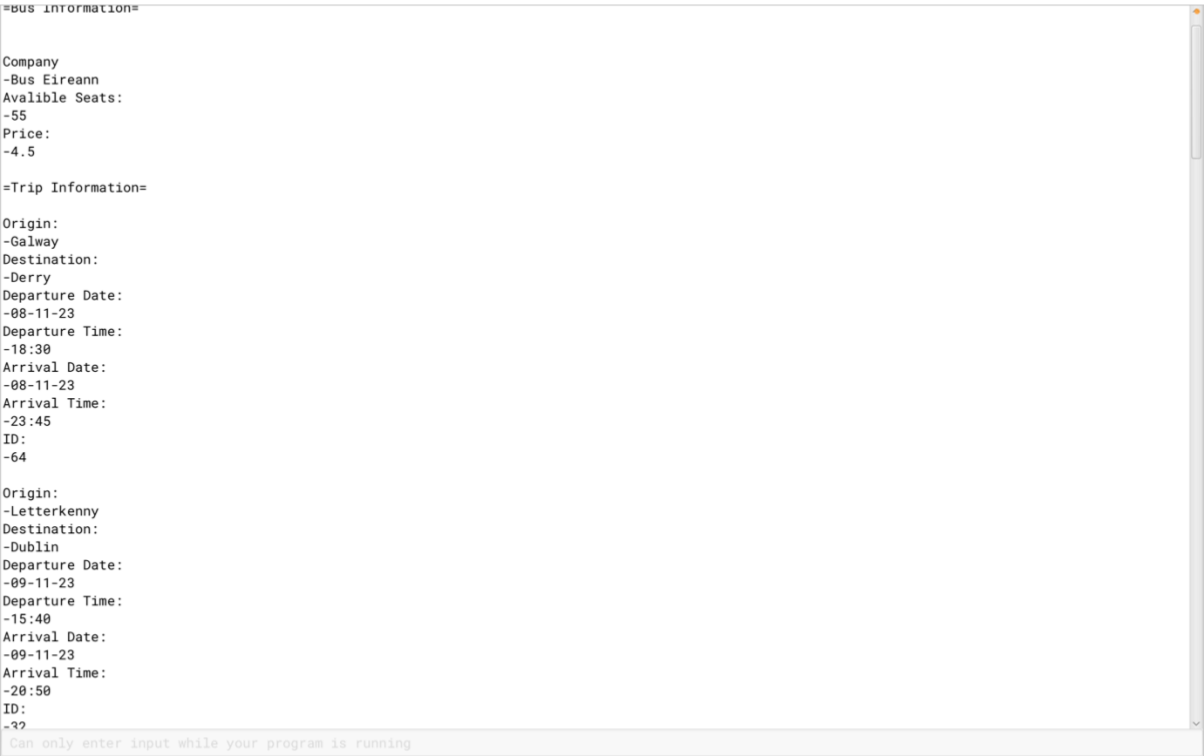
TestTravel:

```
public class TestTravel
{

    public TestTravel()
    {
    }

    public static void main(String args[]) {
        //Bus objects created for each company with company name, seats and price.
        BusEireann be = new BusEireann("Bus Eireann", 55, 4.50);
        CityLink cl = new CityLink("City Link", 45, 8.40);
        GoBus gb = new GoBus("GoBus", 60, 11.90);
        //Trips created with origin, destination, departure and arrival date, departure time and
arrival time, id.
        Trip trip1 = new Trip("Galway", "Derry", "08-11-23", "08-11-23", "18:30", "23:45", 64);
        Trip trip2 = new Trip("Letterkenny", "Dublin", "09-11-23", "09-11-23", "15:40", "20:50",
32);
        Trip trip3 = new Trip("Moville", "Letterkenny", "10-11-23", "10-11-23", "07:45", "08:50",
238);
        Trip trip4 = new Trip("Derry", "Moville", "11-11-23", "11-11-23", "17:10", "17:50", 957);
        Trip trip5 = new Trip("Eyre Square", "Seacrest", "12-11-23", "12-11-23", "16:50",
"17:25", 402);
        Trip trip6 = new Trip("Rahoon", "Eyre Square", "13-11-23", "13-11-23", "06:35", "07:10",
405);

        //Adding trips to each company.
        be.addTrip(trip1);
        be.addTrip(trip2);
        cl.addTrip(trip3);
        cl.addTrip(trip4);
        gb.addTrip(trip5);
        gb.addTrip(trip6);

        //Printing the trips of each company
        System.out.println(be);
        System.out.println(cl);
        System.out.println(gb);

        //Creating a booking with the amount of seats, id and company.
        Booking booking1 = new Booking(8, 64, be);
        //Calling makeBook() method from booking.
        boolean success = booking1.makeBook();

        if (success) {
          //If success is true, print message with booking id.
          System.out.println("\nBooking ID " + booking1.getBookId() + " was successful\n");
        }
        else {
          //If false, print booking id and message.
          System.out.println("\nBooking ID: " + booking1.getBookId() + " was unsuccessful\n");
        }

        //Print to show that seats was reduced for the bus.
        System.out.println(be);

        //Same thing as first booking but trying to book more seats than seats that are avalible.
        Booking booking2 = new Booking(50, 957, cl);
        boolean unsuccess = booking2.makeBook();
```

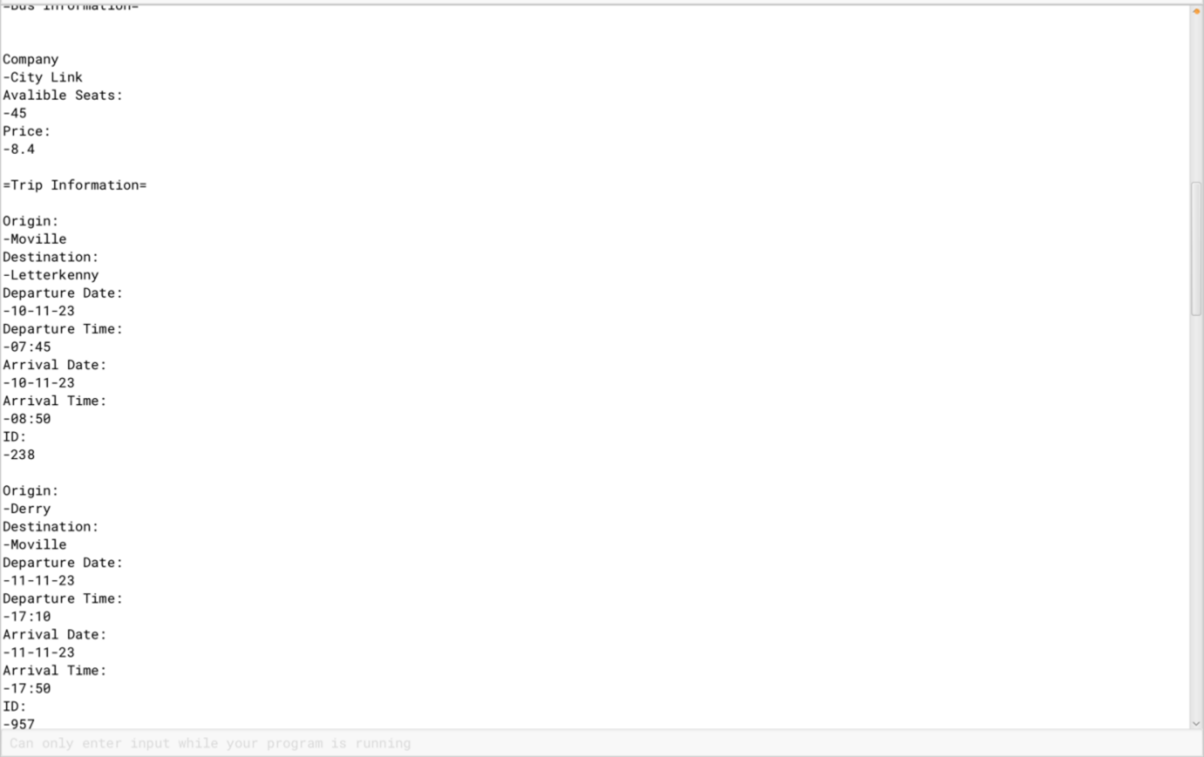
```
        if (unsuccess) {
            System.out.println("\nBooking ID " + booking2.getBookId() + " was successful\n");
        }
        else {
            System.out.println("\nBooking ID: " + booking2.getBookId() + " was unsuccessful\n");
        }
    }
}
```

Question 3 – Analysis

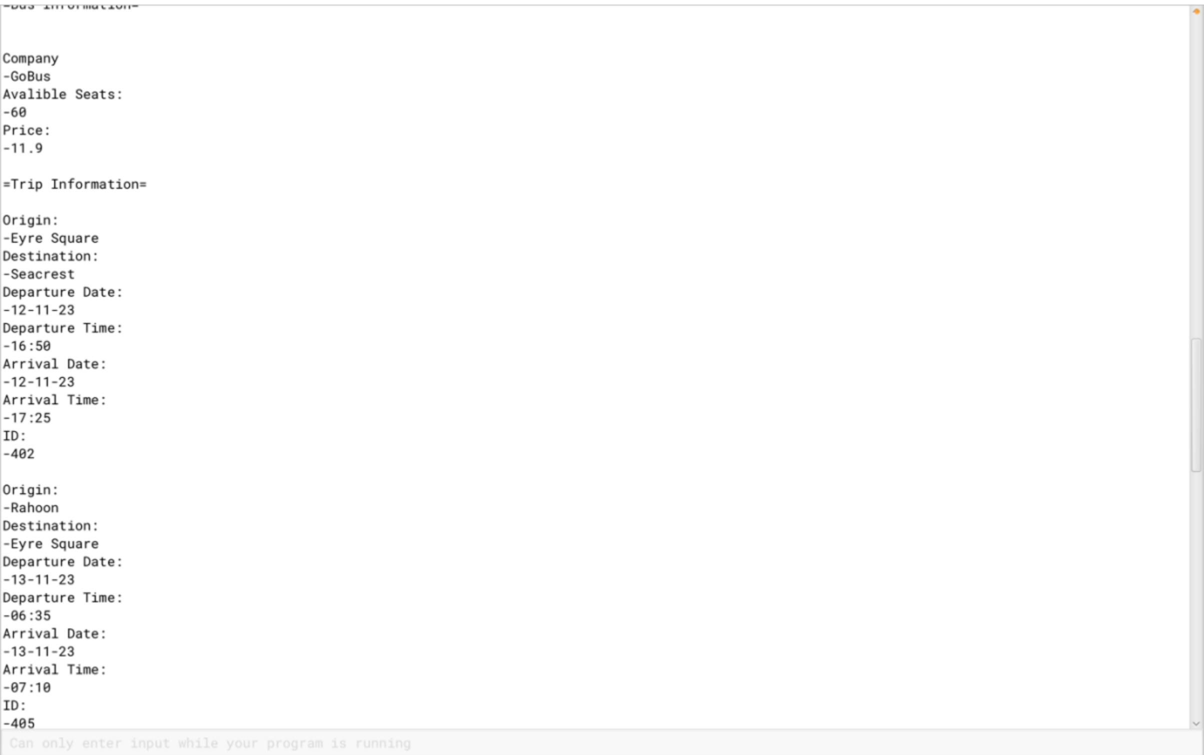




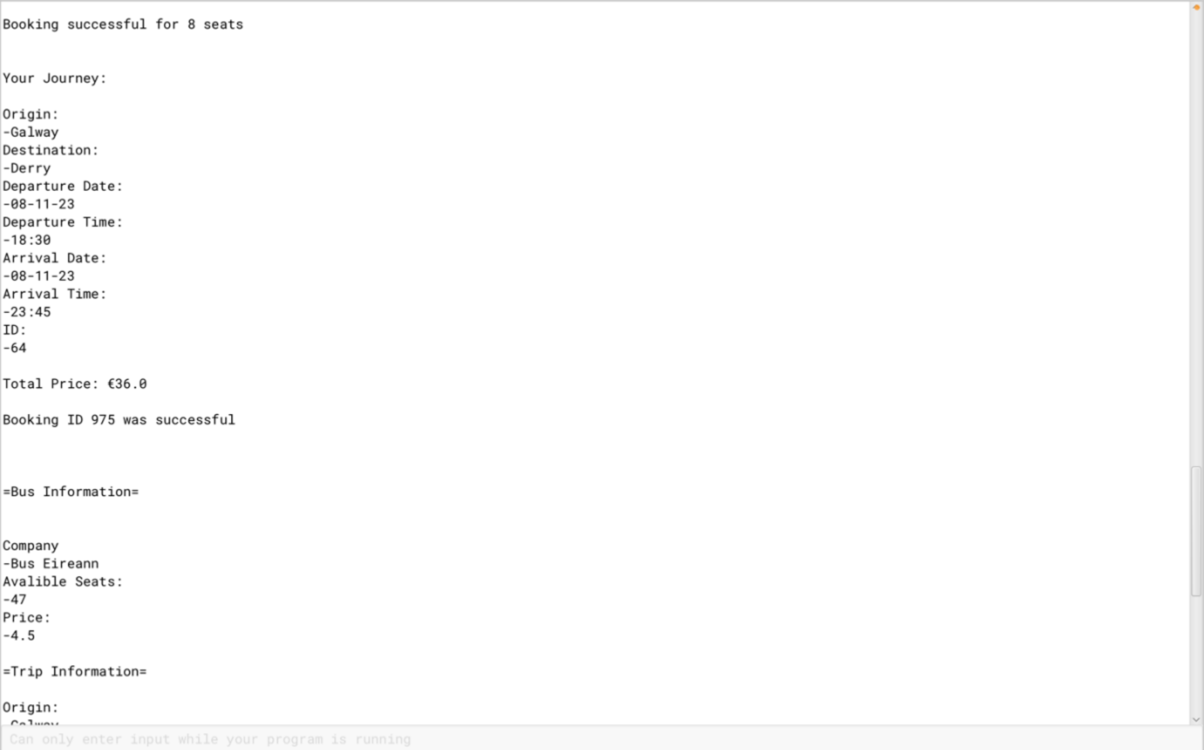
```
=Bus Information=


Company
-Bus Eireann
Avalible Seats:
-55
Price:
-4.5

=Trip Information=

Origin:
-Galway
Destination:
-Derry
Departure Date:
-08-11-23
Departure Time:
-18:30
Arrival Date:
-08-11-23
Arrival Time:
-23:45
ID:
-64

Origin:
-Letterkenny
Destination:
-Dublin
Departure Date:
-09-11-23
Departure Time:
-15:40
Arrival Date:
-09-11-23
Arrival Time:
-20:50
ID:
-32
```



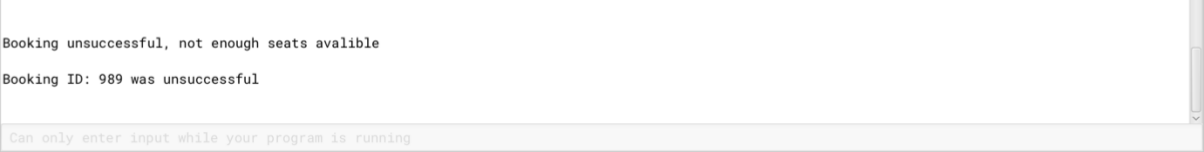
```
=Bus Information=


Company
-City Link
Avalible Seats:
-45
Price:
-8.4

=Trip Information=

Origin:
-Moville
Destination:
-Letterkenny
Departure Date:
-10-11-23
Departure Time:
-07:45
Arrival Date:
-10-11-23
Arrival Time:
-08:50
ID:
-238

Origin:
-Derry
Destination:
-Moville
Departure Date:
-11-11-23
Departure Time:
-17:10
Arrival Date:
-11-11-23
Arrival Time:
-17:50
ID:
-957
```



```
=Bus Information=


Company
-GoBus
Avalible Seats:
-60
Price:
-11.9

=Trip Information=

Origin:
-Eyre Square
Destination:
-Seacrest
Departure Date:
-12-11-23
Departure Time:
-16:50
Arrival Date:
-12-11-23
Arrival Time:
-17:25
ID:
-402

Origin:
-Rahoon
Destination:
-Eyre Square
Departure Date:
-13-11-23
Departure Time:
-06:35
Arrival Date:
-13-11-23
Arrival Time:
-07:10
ID:
-405
```



```
Booking successful for 8 seats


Your Journey:

Origin:
-Galway
Destination:
-Derry
Departure Date:
-08-11-23
Departure Time:
-18:30
Arrival Date:
-08-11-23
Arrival Time:
-23:45
ID:
-64

Total Price: €36.0

Booking ID 975 was successful



=Bus Information=


Company
-Bus Eireann
Avalible Seats:
-47
Price:
-4.5

=Trip Information=

Origin:
-Galway
```



```

Booking unsuccessful, not enough seats avalible

Booking ID: 989 was unsuccessful

```

For my testing of this, I tested by printing each of the companies trips that I added to each company which worked. I then tested to see if I could make a booking for 8 seats, when I did this I then printed that companies trips out and it showed that the bus seats was reduced by 8. The last testing I did was by trying to book more seats than the amount of seats on the bus, my code picked up on this and displayed the correct message, all of this is indicated by the screenshots. Overall my code worked as I intended it to.