CMPE 202 Sec 48 – Individual Assignment Report

Submitted By: Sakshi Jain (sakshi.jain01@sjsu.edu)

Student Id – 015931526

GitHub Link - https://github.com/gopinathsjsu/individual-project-jainsakshi2395

Problem Statement

We will develop a flight booking application having flights information and users are allowed to book flights by booking them. There are a limited number of seats in each category for flights. There will be a validation for card, available seats and even if the flight user trying to be booking is even present or not. If a booking satisfies all the requirements, the total price for each booking based on the number of seats booked in each category will be calculated and the available seats would be updated.

Instructions to run the application:

Needs Java 8 installed in the system that is running the program.

Instructions to run the JAR file:

- Open Command Prompt in the location of JAR File.
  (Location of JAR File is: /Users/sakshijain/Desktop/projects/cmpe202_individual_project/cmpe202-individual-project/target/cmpe202-individual-project-1.0-SNAPSHOT.jar).
- Run the following command in the command prompt: java -cp <jar_name> "<Path to Flights.csv>" "<Path to Sample.csv>" "<Output.csv>" "<Output.txt>"

For Example:
java -cp cmpe202-individual-project-1.0-SNAPSHOT.jar test.RunClient
/Users/sakshijain/Desktop/projects/cmpe202_individual_project/cmpe202-individual-project/files/Sample.csv
/Users/sakshijain/Desktop/projects/cmpe202_individual_project/cmpe202-individual-project/files/flights.csv
/Users/sakshijain/Desktop/projects/cmpe202_individual_project/cmpe202-individual-project/files/Output.csv
/Users/sakshijain/Desktop/projects/cmpe202_individual_project/cmpe202-individual-project/files/Output.txt

- Output files will be created in the directory as the files provided as arguments to run the JAR, if not please check your target folder

Design Patterns used are:

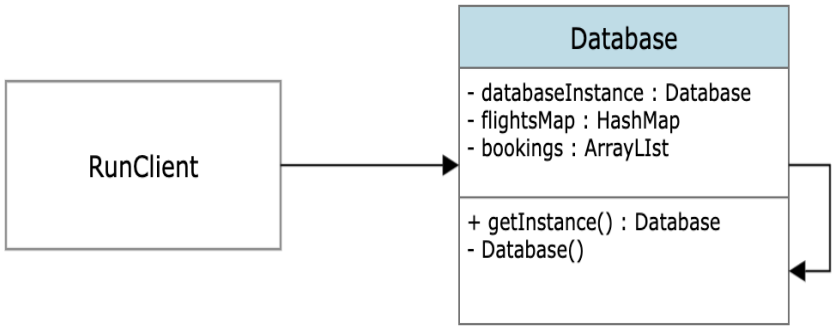
I decided to use these design patterns in my project to solve multiple problems speficied along with the design pattern.

- Singleton
- Factory
- Chains of Responsibility

Singleton:

- It is a creational design pattern that lets you ensure that a class has only one instance, while providing the global access point to this instance.
- This pattern is used to build the database for the application. Only the Database class is used as part of this pattern.
- Whenever an instance of the database is needed, the getInstance() method is called so that only one instance of the object can be reused.

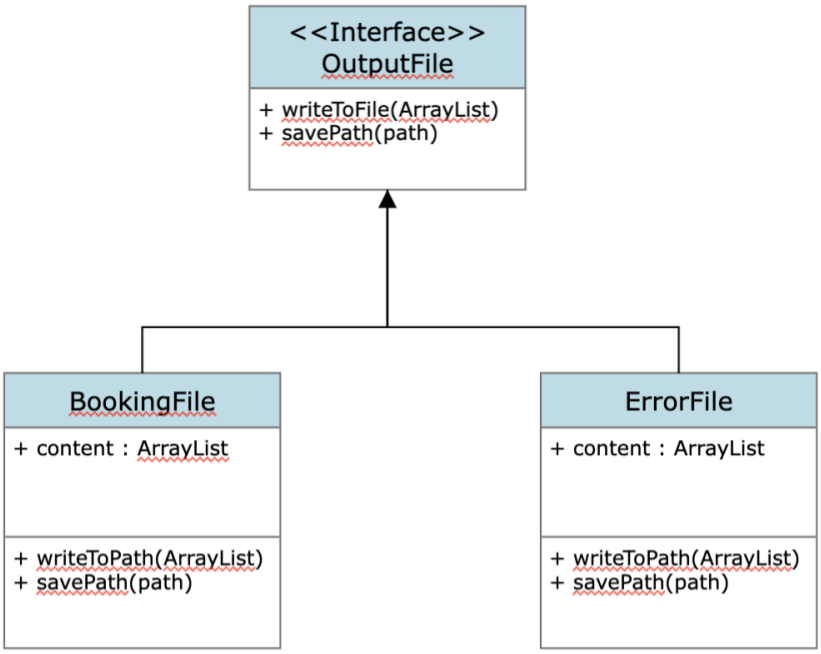
Class Diagram:



Factory:

- It is a creational design pattern that provides an interface for creating objects in a superclass but allows subcasses to alter the type of objects that will be created.
- We have used this pattern to output when a booking is being made. The following files are as part of this pattern implementation:
    - Interface OutputFile
    - Class BookingFile
    - Class ErrorFile
- When the respective concrete handler figures out the output should be a booking file or error file, we will create an object of OutputFile. The FileReader class is used to read and create files in the application.

Class Diagram:



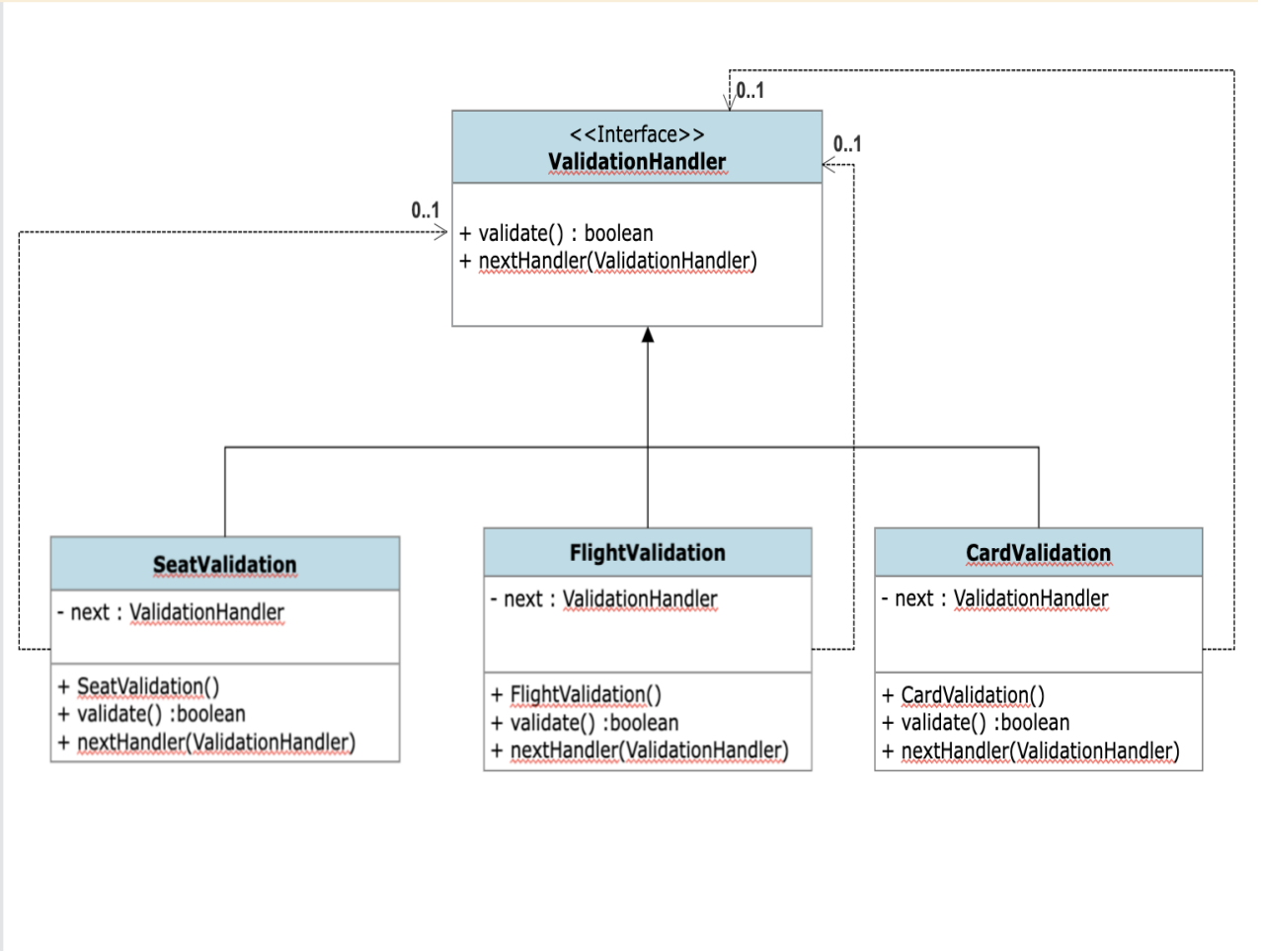
Chains of Responsibility:

- It is a behavioral design pattern that lets you pass requests along a chain of handlers. Upon receiving a request, each handler decided either to process the request or to pass it to the next handler in the chain.
- We have used this design pattern to determine for the validation parts like if the booking can be processed correctly or not. The following files are used in this pattern:

- Interface ValidationHandler
- Class CardValidation (to check if the card entered is valid or not)
- Class FlightValidation (to validate if the flight exists or not)
- Class SeatValidation (to validate if seats are available or not)

- Before we verify the booking can be processed or not, ValidationHandler object is created and will be passed on from one Handler to another using the same.

Class Diagram:



Test cases:

Test Case 1- Input: Sample set given in problem statement

flights

Category(Economy	PremiumEconomy	Business)	FlightNumber	AvailableSeats	Price	Arrival	Departure
Economy	SJ456	5	250	Seattle	San Jose		
Premium Economy	BY110	5	500	San Francisco	New York		
Business	BY110	5	2000	San Francisco	New York		
Economy	CA453	5	300	Seattle	San Jose		
Business	CA453	5	1500	Seattle	San Jose		

Sample

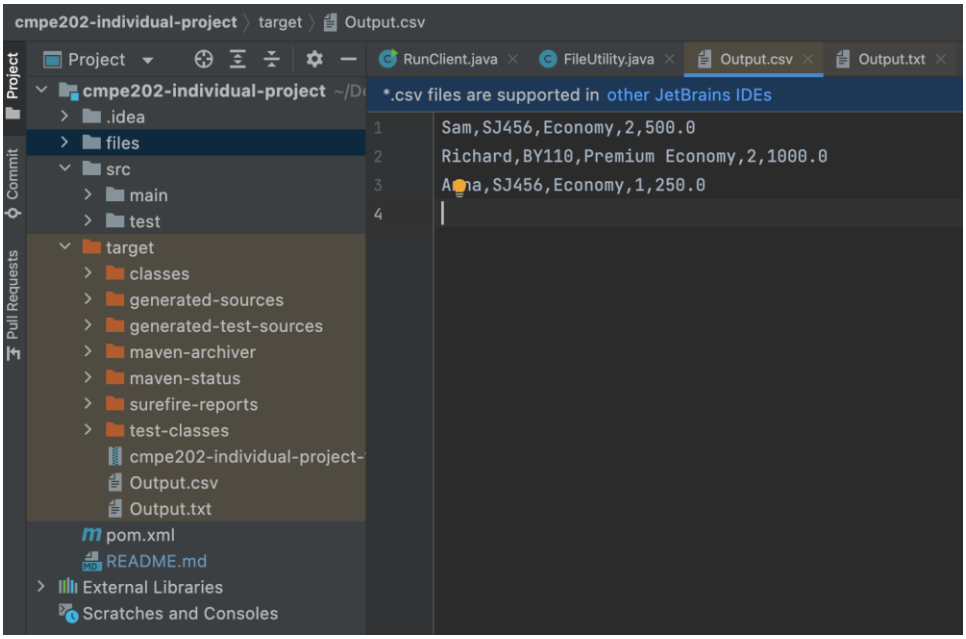
BookingName	flightNumber	seatCategory	numberOfSeats	paymentCardNumber
Sam	SJ456	Economy	2	5410000000000000
Richard	BY110	Premium Economy	2	341000000000000
Anna	SJ456	Economy	1	4120000000000
John	KL908	Economy	1	6011000000000000
Sierra	BY110	Business	1	1234561323130

Output:

```
[INFO] skip non existing resourceDirectory /Users/sakshijain/Desktop/projects/cmpe202_individual_project/cmpe202-individual-project/src/test/resources
[INFO]
[INFO] --- maven-compiler-plugin:3.8.0:testCompile (default-testCompile) @ cmpe202-individual-project ---
[INFO] Nothing to compile - all classes are up to date
[INFO]
[INFO] --- maven-surefire-plugin:2.22.1:test (default-test) @ cmpe202-individual-project ---
[INFO]
[INFO] -------------------------------------------------------
[INFO]  T E S T S
[INFO] -------------------------------------------------------
[INFO] Running test.RunClientTest
[INFO] Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.056 s - in test.RunClientTest
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 2, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO]
[INFO] --- maven-jar-plugin:3.0.2:jar (default-jar) @ cmpe202-individual-project ---
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  2.718 s
[INFO] Finished at: 2022-04-22T13:37:27-07:00
[INFO] ------------------------------------------------------------------------
sakshijain@Sakshis-MacBook-Pro-2 cmpe202-individual-project % cd target
sakshijain@Sakshis-MacBook-Pro-2 target % java -cp cmpe202-individual-project-1.0-SNAPSHOT.jar test.RunClient /Users/sakshijain/Desktop/projects/cmpe202_individual_project/cmpe202-individual-project/files
/Sample.csv /Users/sakshijain/Desktop/projects/cmpe202_individual_project/cmpe202-individual-project/files/flights.csv /Users/sakshijain/Desktop/projects/cmpe202_individual_project/cmpe202-individual-proj
ect/files/Output.csv /Users/sakshijain/Desktop/projects/cmpe202_individual_project/cmpe202-individual-project/files/Output.txt
```

```
sakshijain@Sakshis-MacBook-Pro-2 cmpe202-individual-project % cd target
sakshijain@Sakshis-MacBook-Pro-2 target % java -cp cmpe202-individual-project-1.0-SNAPSHOT.jar test.RunClient /Users/sakshijain/Desktop/projects/cmpe202_individual_project/cmpe202-individual-project/files
/Sample.csv /Users/sakshijain/Desktop/projects/cmpe202_individual_project/cmpe202-individual-project/files/flights.csv /Users/sakshijain/Desktop/projects/cmpe202_individual_project/cmpe202-individual-proj
ect/files/Output.csv /Users/sakshijain/Desktop/projects/cmpe202_individual_project/cmpe202-individual-project/files/Output.txt
Loading datasets for : /Users/sakshijain/Desktop/projects/cmpe202_individual_project/cmpe202-individual-project/files/flights.csv
Flight datasets : [Flights [arrivalCity=Seattle, availableSeats=5, category=Economy, departureCity=San Jose, flightNumber=SJ456, price=250.0], Flights [arrivalCity=San Francisco, availableSeats=5, categor
y=Premium Economy, departureCity=New York, flightNumber=BY110, price=500.0], Flights [arrivalCity=San Francisco, availableSeats=5, category=Business, departureCity=New York, flightNumber=BY110, price=2000
.0], Flights [arrivalCity=Seattle, availableSeats=5, category=Economy, departureCity=San Jose, flightNumber=CA453, price=300.0], Flights [arrivalCity=Seattle, availableSeats=5, category=Business, departur
eCity=San Jose, flightNumber=CA453, price=1500.0]]
Loading datasets for : /Users/sakshijain/Desktop/projects/cmpe202_individual_project/cmpe202-individual-project/files/Sample.csv
Booking datasets : [Bookings [bookingName=Sam, cardNumber=5410000000000000, flightNumber=SJ456, numberOfSeats=2, seatCategory=Economy], Bookings [bookingName=Richard, cardNumber=341000000000000, flightNum
ber=BY110, numberOfSeats=2, seatCategory=Premium Economy], Bookings [bookingName=Anna, cardNumber=4120000000000, flightNumber=SJ456, numberOfSeats=1, seatCategory=Economy], Bookings [bookingName=John, car
dNumber=6011000000000000, flightNumber=KL908, numberOfSeats=1, seatCategory=Economy], Bookings [bookingName=Sierra, cardNumber=1234561323130, flightNumber=BY110, numberOfSeats=1, seatCategory=Business]]
>> Processing input...
Calculating for booking : Sam
Available Seats updated for : SJ456 : 3
Inside updateAmount method for Sam
Economy >> 500.0
Calculation for booking ended : Sam
Calculating for booking : Richard
Available Seats updated for : BY110 : 3
Inside updateAmount method for Richard
Premium Economy >> 1000.0
Calculation for booking ended : Richard
Calculating for booking : Anna
Available Seats updated for : SJ456 : 2
Inside updateAmount method for Anna
Economy >> 250.0
Calculation for booking ended : Anna
Calculating for booking : John
>> Incorrect flight number.. writing to Output.txt
error log content : Please enter correct booking details for John : invalid flight number
Calculation for booking ended : John
Calculating for booking : Sierra
>> Incorrect card details.. writing to Output.txt
error log content : Please enter correct booking details for Sierra : incorrect card details
Calculation for booking ended : Sierra
sakshijain@Sakshis-MacBook-Pro-2 target %
```

Two files generated:

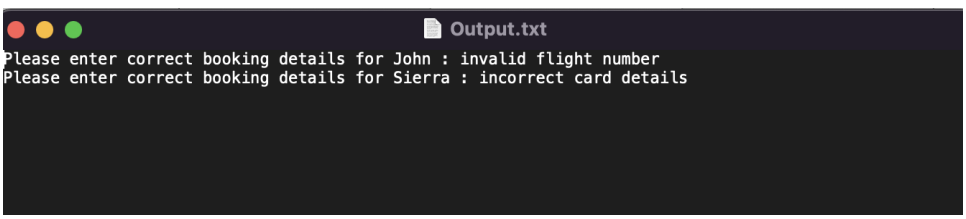


Output.csv

Output

Sam	**SJ456**	**Economy**	**2**	**500.0**
Richard	BY110	Premium Economy	2	1000.0
Anna	SJ456	Economy	1	250.0

Output.txt



Test Case 2: All Valid bookings (only output.csv file generated)

Input:

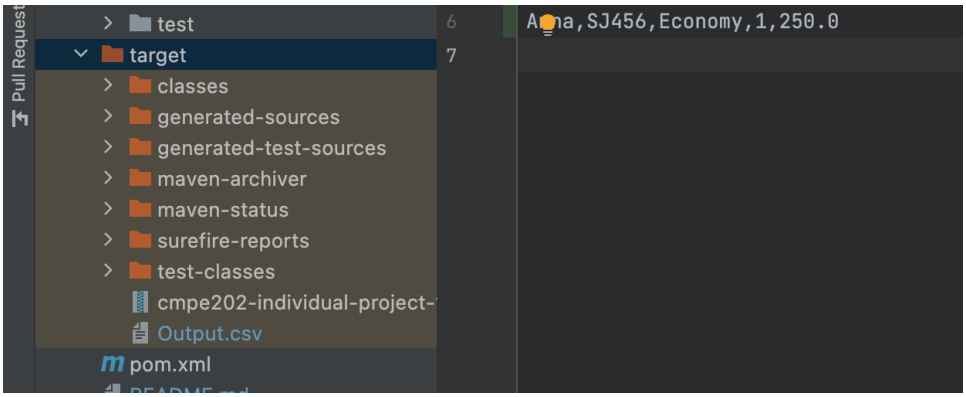
Sample

BookingName	flightNumber	seatCategory	numberOfSeats	paymentCardNumber
Sam	SJ456	Economy	2	5410000000000000
Richard	BY110	Premium Economy	2	341000000000000
Anna	SJ456	Economy	1	4120000000000

Output:

Output

Sam	**SJ456**	**Economy**	**2**	**500.0**
Richard	BY110	Premium Economy	2	1000.0
Anna	SJ456	Economy	1	250.0



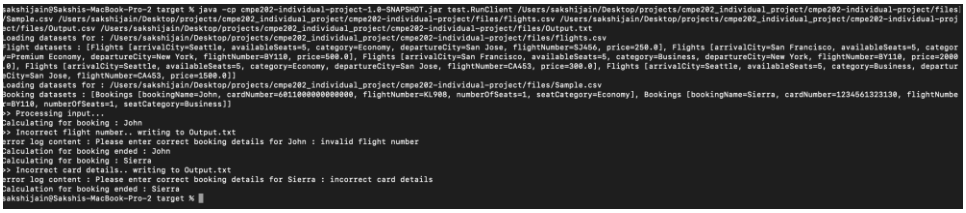
Test Case 3: All failed bookings

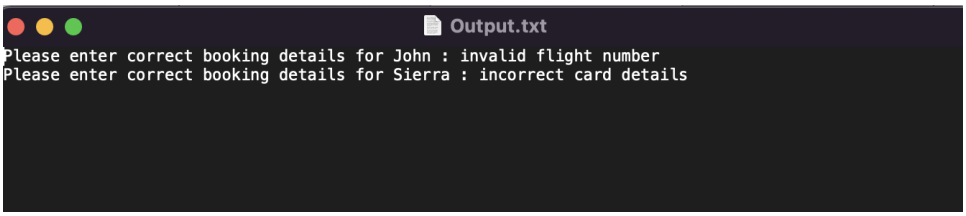
Input:

Sample

BookingName	flightNumber	seatCategory	numberOfSeats	paymentCardNumber
John	KL908	Economy	1	6011000000000000
Sierra	BY110	Business	1	1234561323130

Output: (only output.txt file generated)





Test Case 4: File path not given

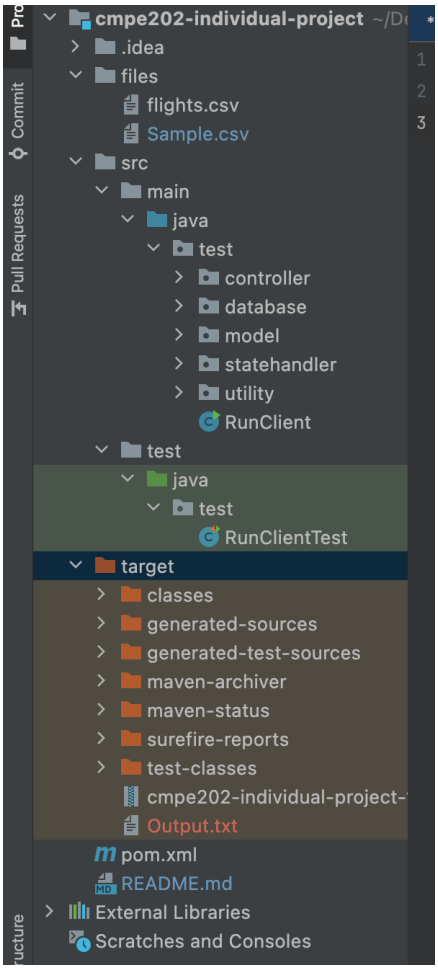
Input:

```
sakshijain@Sakshis-MacBook-Pro-2 target % java -cp cmpe202-individual-project-1.0-SNAPSHOT.jar test.RunClient
```

Output:

```
sakshijain@Sakshis-MacBook-Pro-2 target % java -cp cmpe202-individual-project-1.0-SNAPSHOT.jar test.RunClient
File path to Flights Dataset not provided. Please enter one.
sakshijain@Sakshis-MacBook-Pro-2 target %
```

Project Structure:


cmpe202-individual-project
.idea
files
flights.csv
Sample.csv
src
main
java
test
controller
database
model
statehandler
utility
RunClient
test
java
test
RunClientTest
target
classes
generated-sources
generated-test-sources
maven-archiver
maven-status
surefire-reports
test-classes
Output.txt
pom.xml
README.md
External Libraries
Scratches and Consoles
Commit
Pull Requests