**DELHI TECHNOLOGICAL UNIVERSITY**



**PROJECT REPORT**

Prepared By:

Rakshit Khullar (2K19/IT/105)

Sahil Yadav (2K19/IT/114)

Submitted To:🡪

Ms. Swati Sharda

**DELHI TECHNOLOGICAL UNIVERSITY**

(Formerly Delhi College of Engineering)

Bawana Road, Delhi, 110042

CANDIDATE’S DECLARATION

We, Sahil Yadav & Rakshit Khullar, Roll No(s). 2K19/IT/114 & 2K19/IT/105, students of B. Tech. in Information Technology, hereby declare that the project Dissertation titled ***Railway Booking System*** which is submitted by us to the Department of Information Technology, Delhi Technological University, Delhi in partial fullfillment of the requirement for the award of the mid-semester component evaluation, semester-3 of Bachelor of Technology is original and not copied from any source without proper citation. This work has not previously formed a basis for the award of any degree, Diploma Associateship, Fellowship, or any similar title or recognition

Place: Delhi Sahil Yadav

Date: 21/11/2020 Rakshit Khullar

**DELHI TECHNOLOGICAL UNIVERSITY**

(Formerly Delhi College of Engineering)

Bawana Road, Delhi, 110042

CERTIFICATE

We hereby declare that the project Dissertation titled “ ***Railway Booking System***” which is submitted by Sahil Yadav & Rakshit Khullar, Roll No(s). 2K19/IT/114 & 2K19/IT/105, Department of Information Technology, Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of the mid-semester component evaluation, semester-3 of Bachelor of Technology, is the record of the project work carried out by the students under my supervision.

Place: Delhi Ms. SwatiSharda

Date: 21/11/2020 Department of IT

Delhi Technological University

Bawana Road, Delhi,110042

ACKNOWLEDGEMENT

At the very outset of this report, we would like to extend our sincere and heartfelt obligation towards all the personages who have guided us with the project.

A special thanks to Ms. Swati Sharda for teaching us the subject “Data Structures”. She helped us visualize the subject and to find its applications in real life. She supervised us with the intricacies of this project. She also offered many relevant and productive recommendations for the project, for which we are very grateful.

Finally, a thank you to all our family and friends who helped us with the project during such difficult times and gave worthy ideas.

**Railway Booking System**



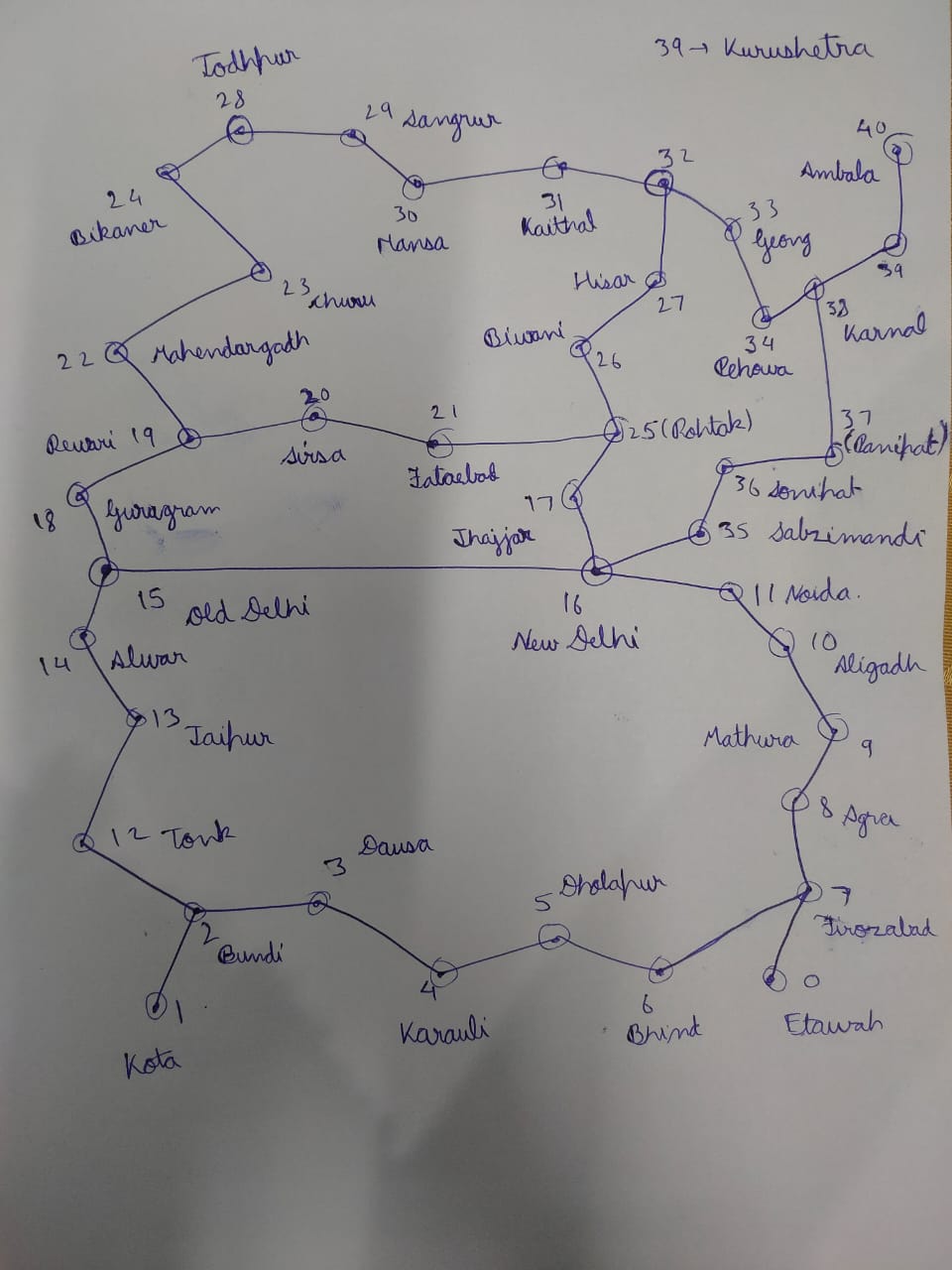
***Aim🡪***

Aim of our project is to create a railway booking system which works more efficiently and effectively to perform the operations.

We have designed our own graph and arranged the trains to go through. Our motive is to allow user to book seat, to check train status between stations, to see the path followed by the train and the amount to be paid for the travel and even to delete the selected seats if needed.

***Introduction🡪***

The first thing that we have done was to create the login system for our booking system which allows the user to login and even allow to enter his/her details to create their respective Id’s. The graph is shown below🡪



Following options will be displayed after login 🡪

**A : Check Trains Between stations**

**B : Check Travelling Amount And Path Of The Train**

**C : Book Train**

**D : Booked Seats/Remove Seats**

**E : Delete Account**

**F : Exit**

All the functions that are made will be explained below.

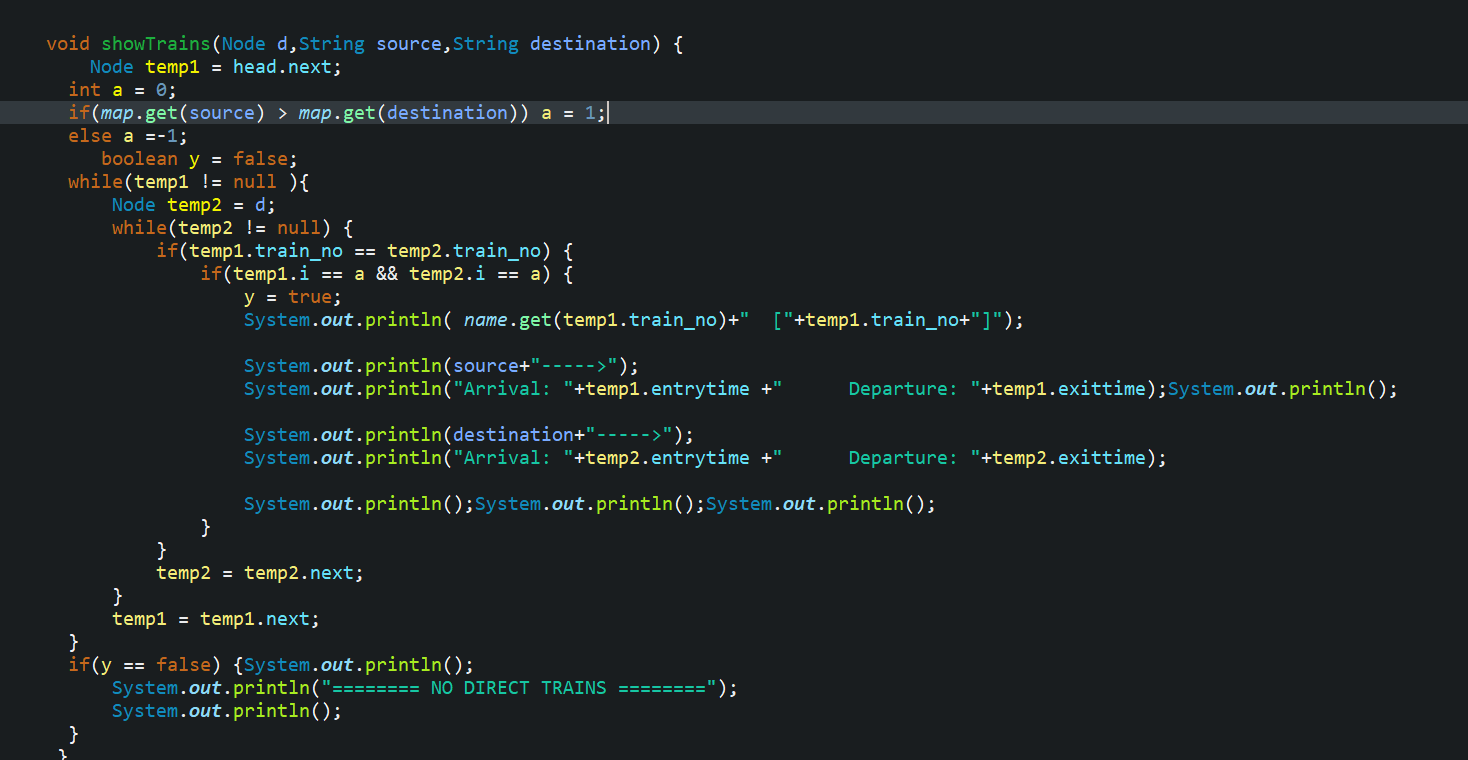
***Option ‘A’ 🡪***

To check the train between stations we have given a number to each station using hash map as examples shown below.

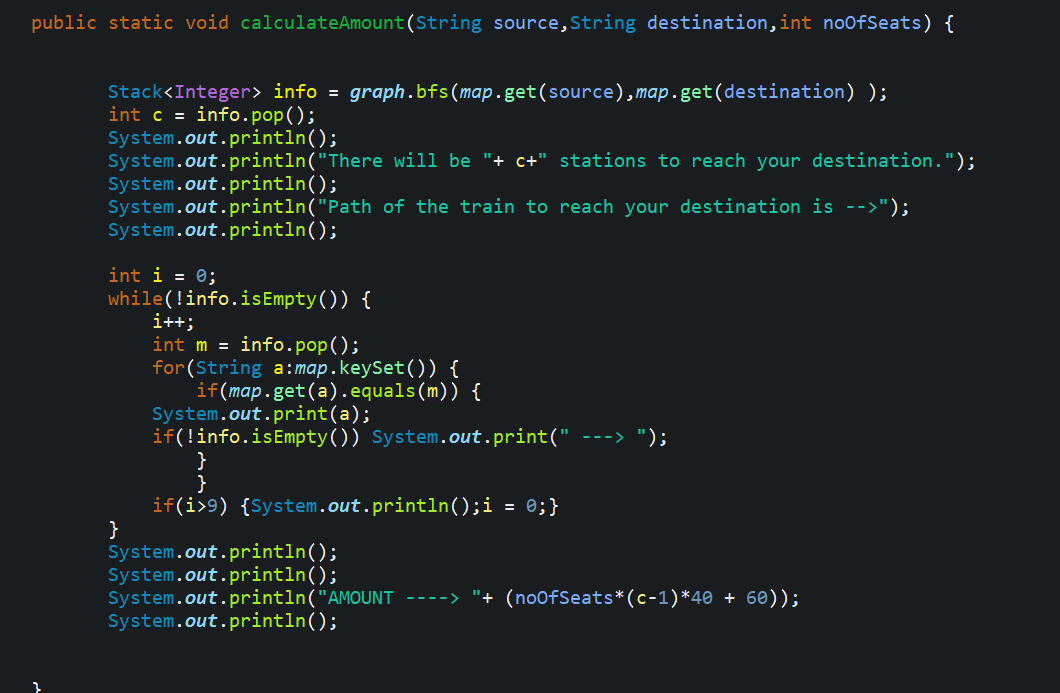
map.put(“NewDelhi”,16); map.put(“Jind”,32); map.put(“Ambala”,40); map.put(“Kota”,1);

All the data is kept in the Adjacency List. Each Node contains the train number, arrival time, departure time and direction of the train.

If the user needs to check train between NewDelhi and Jind then the Linked List from the corresponding nodes only will check and show the available trains if present so.



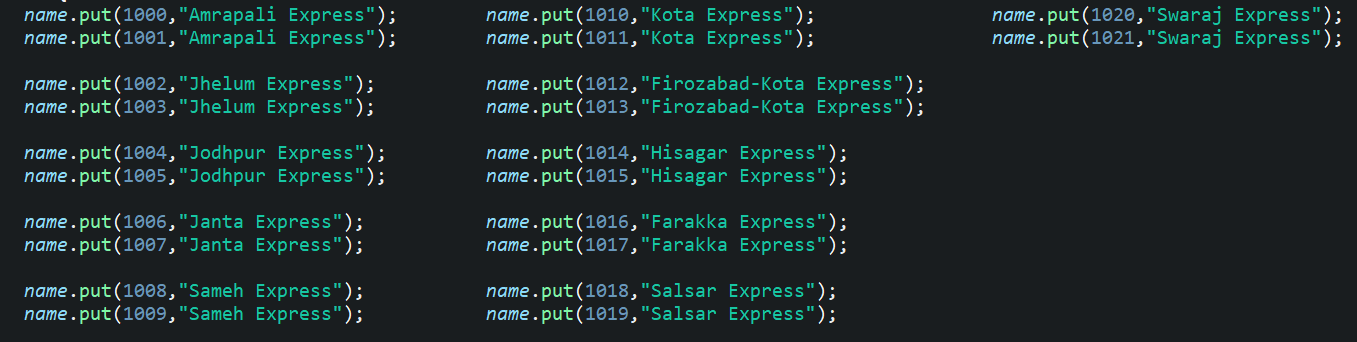
***Option ‘B’ 🡪***

The travelling amount and the path of the train will be shown by using Breadth First Search in the graph. The function used will be “calculateAmount” which will just take input of source, destination and the number of seats to travel with as shown.



***Option ‘C’ 🡪***

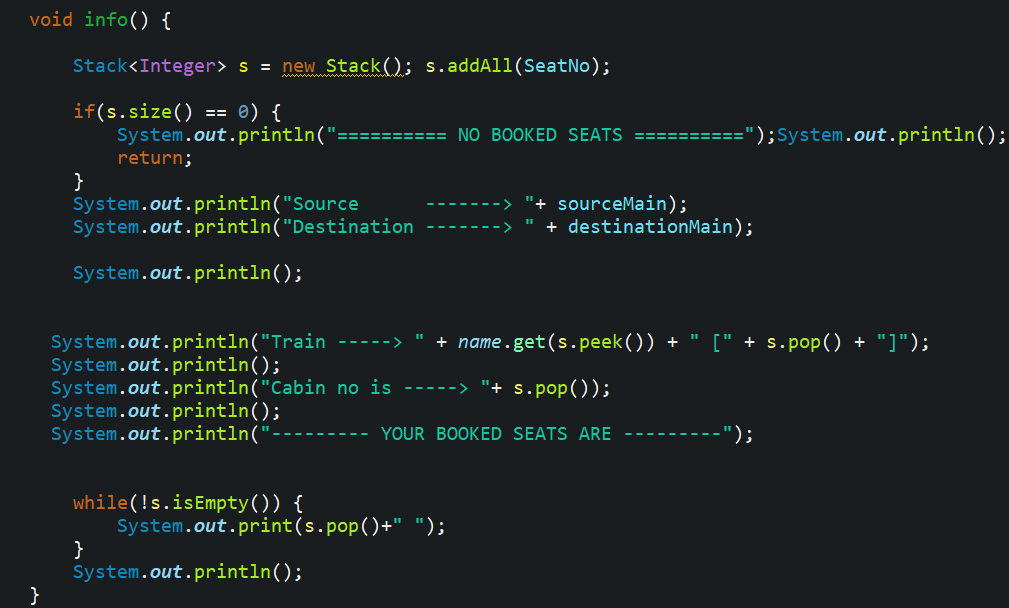
The user will first enter the source and destination of the journey. Trains will be displayed to the user if present. Then the user can book the train by entering the train number and cabin number of the train.





***Option ‘D’ 🡪***

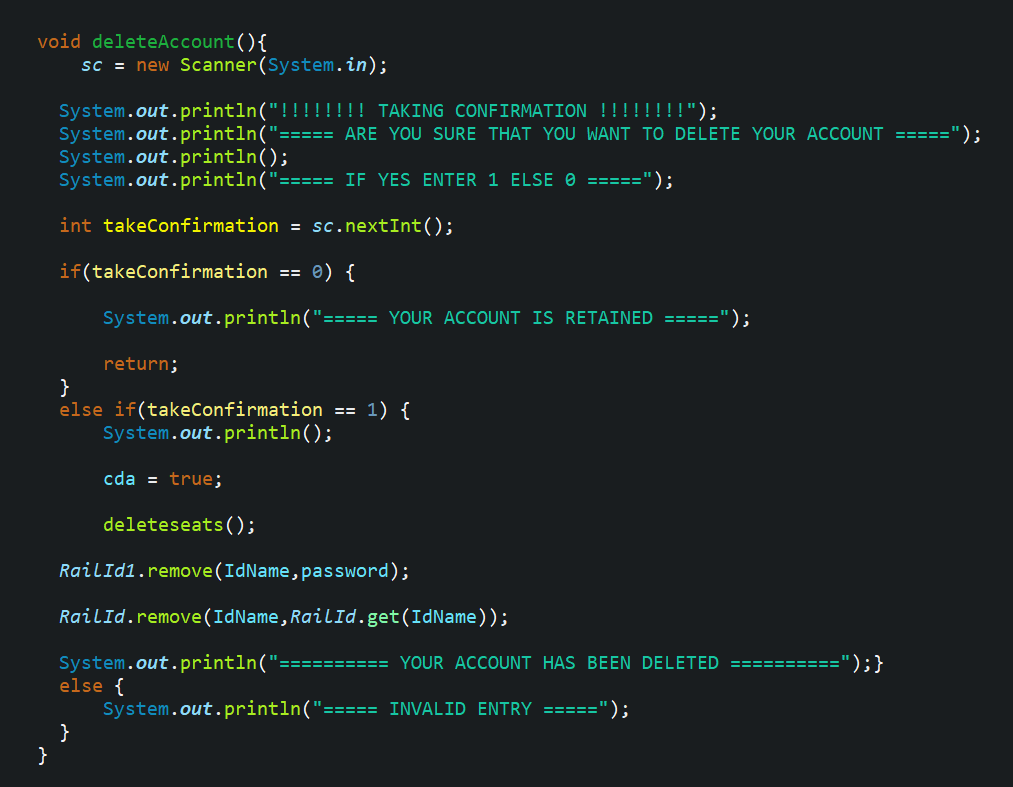
The information of the user will be stored which the user can check any time containing the train number, train name, cabin number and all the seats that the user has booked. User can also delete the seats that has been booked if needed in the similar way in which seats were booked.





***Option ‘E’ 🡪***

It will delete all the information i.e. id ,password and all other data of the object too.



*Classes used 🡪*

Main class🡪

It contains all the main information. All the Id’s, password, train numbers,

graph and the last id Login System also.

Seatbooking class🡪

It contains all the information of the user. The user is able to use all

functions because of this class.

Station Class🡪

It makes the graph as shown above and is used to calculate the and the

travelling amount of the path.

Train and Cabin class🡪

This class makes the actual train and the cabins. The seats are books and

deleted because of this.

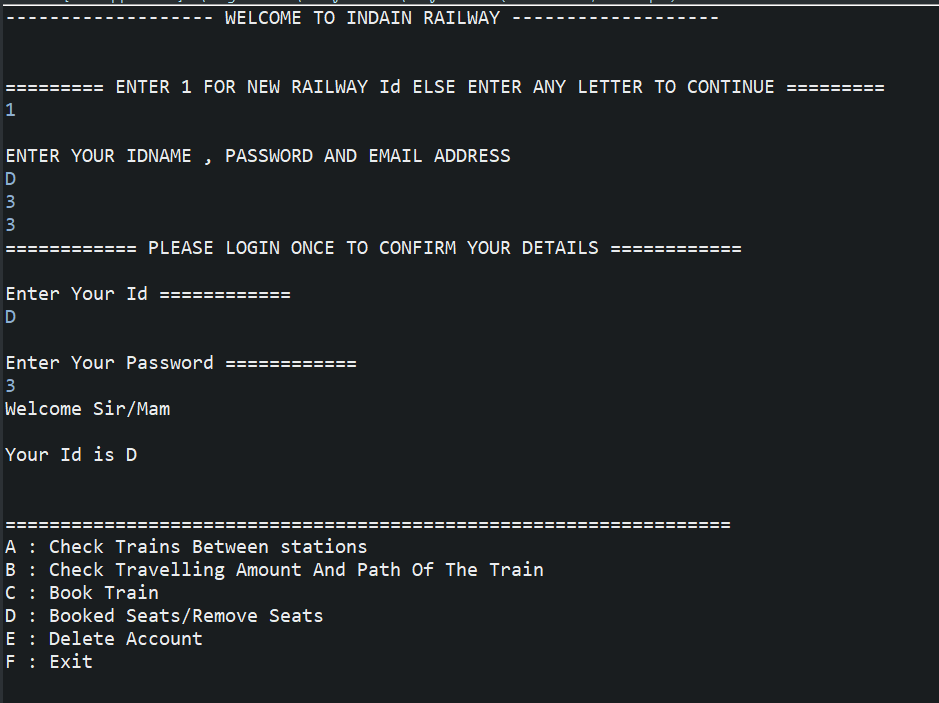
TrainData class🡪

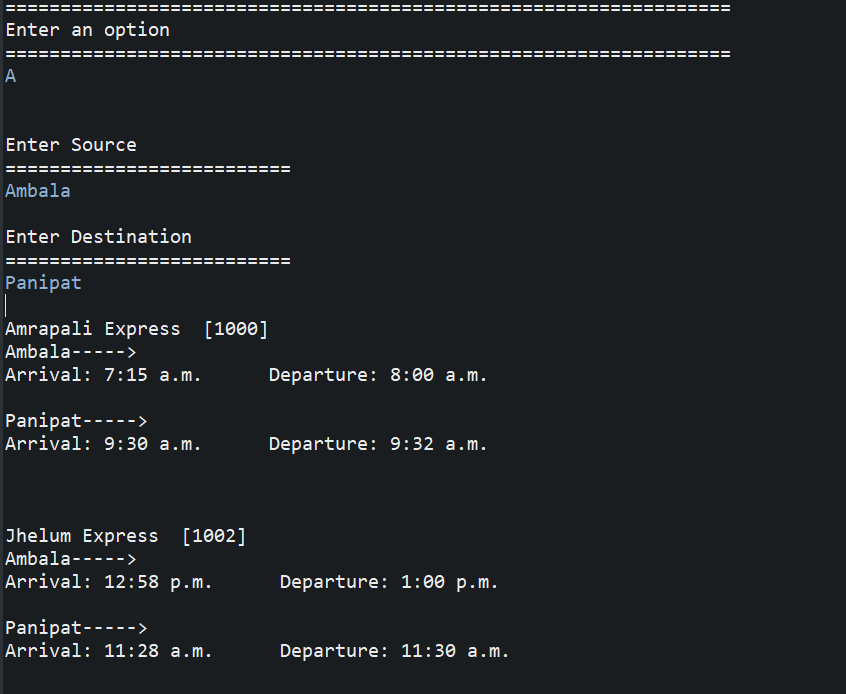
All the information about the train and the corresponding stations that they cross are kept in this class. It is done with help of Adjacency Linked List.

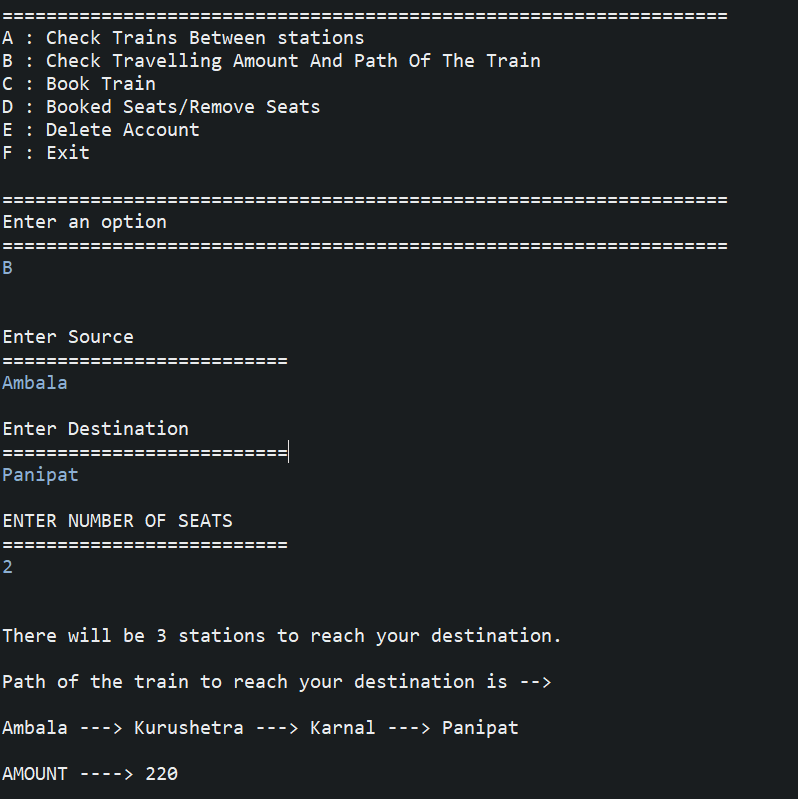
Graph class🡪

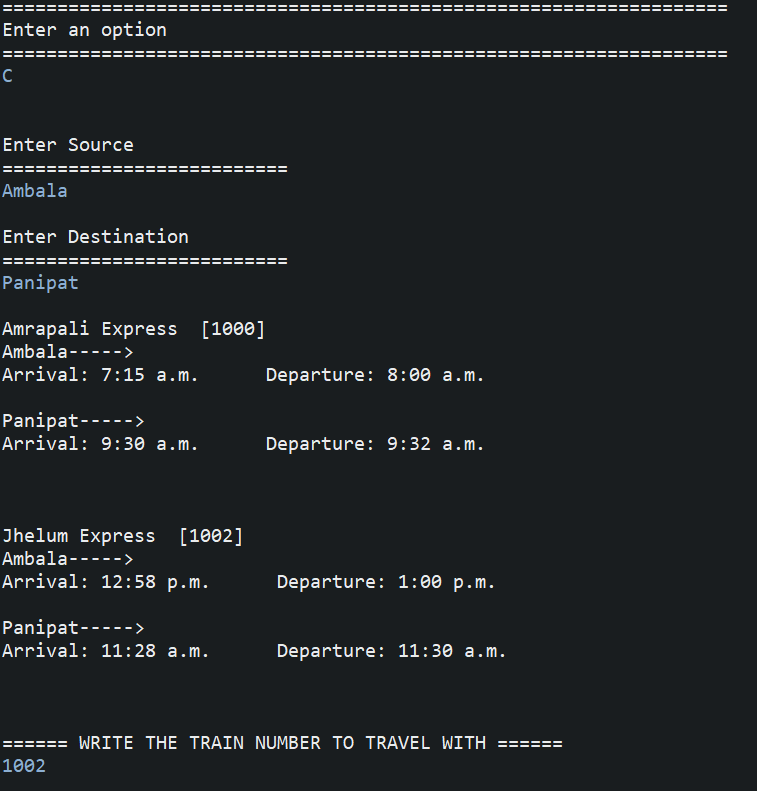
This class is used to make the graph with help of Adjacency Linked List and to calculate the minimum path between stations.

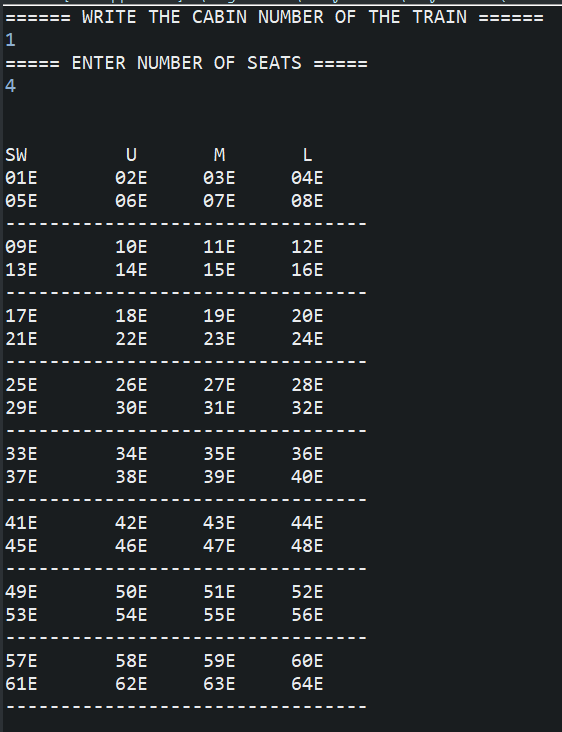
**Solved Example**🡪

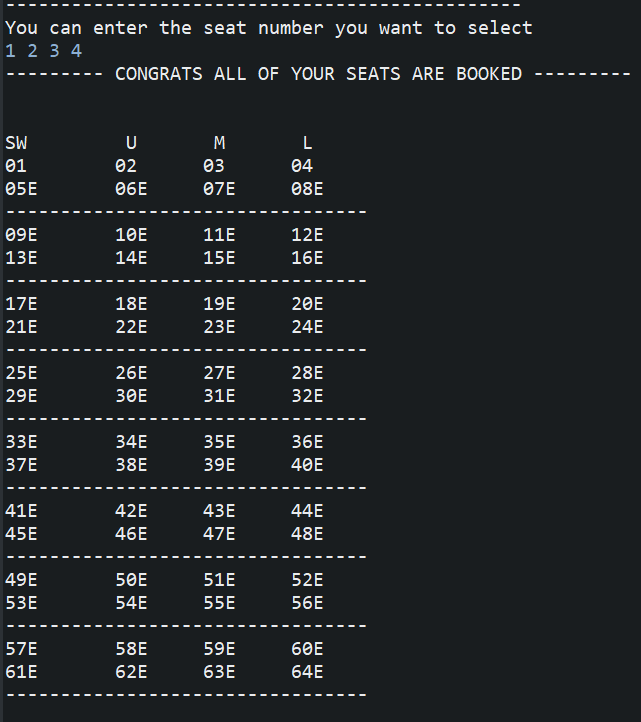


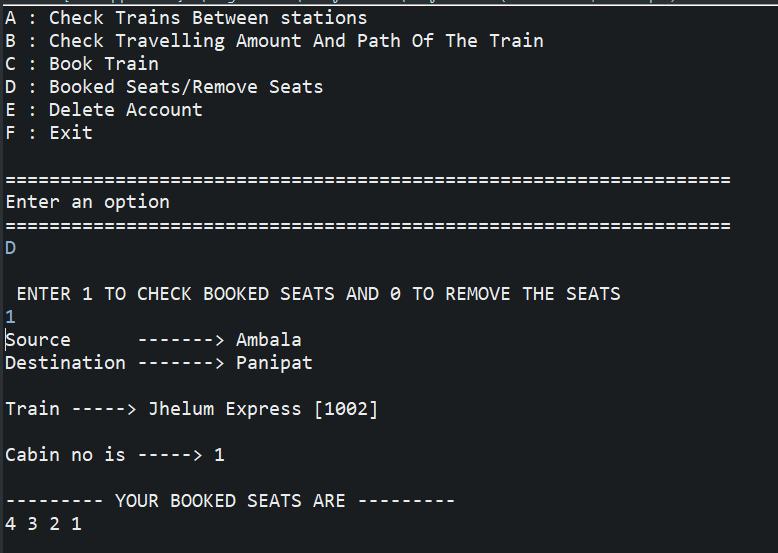


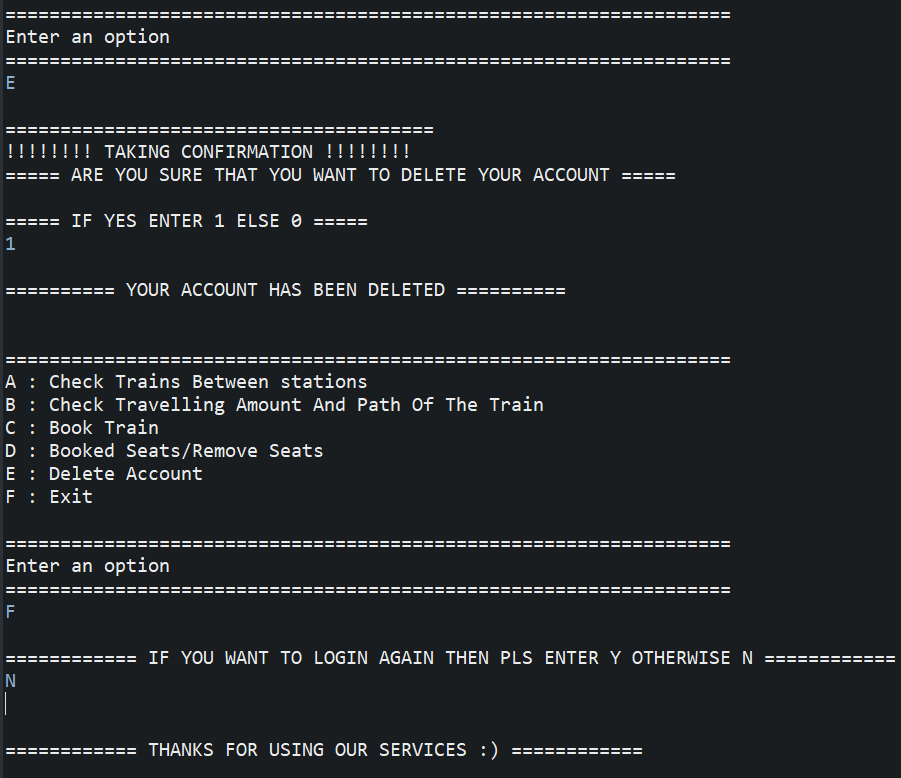












***Bibliography***🡪

Geeks For Geeks

Google

Youtube

***GitHub Link*** 🡪

https://github.com/RakshitKhullar/Railway-Booking-System.git