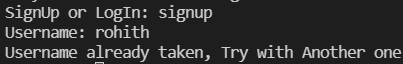
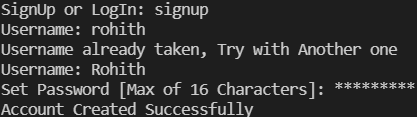
**SIGN UP:**

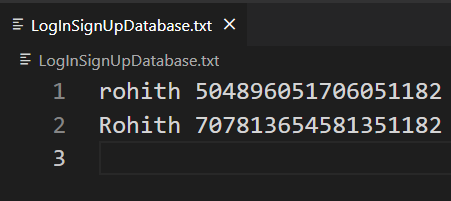
1) If the Username is taken already



2) If Username is accepted, Password is asked

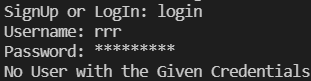


LogInSignUpDatabase.txt file after successful creation of “Rohith” account

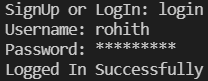


**LOGIN:**

1) If the Username or Password are incorrect or not in our database



2) If the Username or Password are correct



Rules of Password are:

a) Password size must be between 8 and maxPasswordLength (in this case 16)

b) At least One Lower Case Character must be Used

c) At least One Upper Case Character must be Used

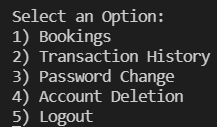
d) At least One Number must be Used

e) At least One Special Character must be Used

If all these are satisfied by the password entered, we create an account.

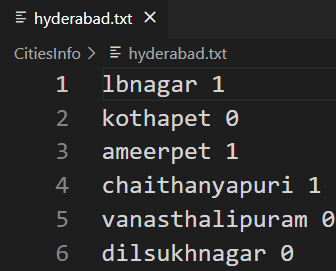
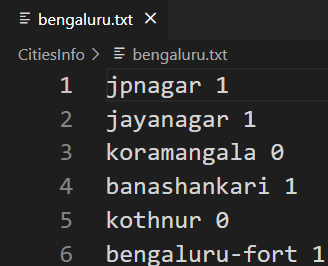
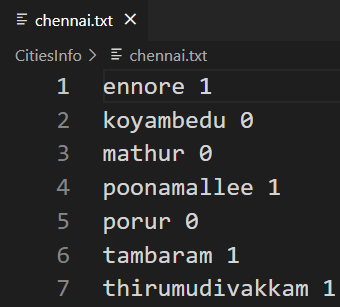
**After Successful Login/Signup:**

**Our Menu**:

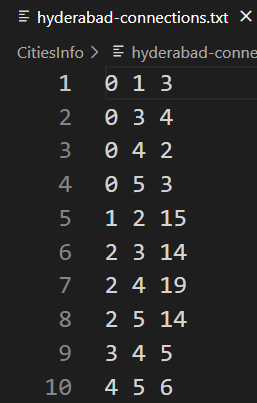
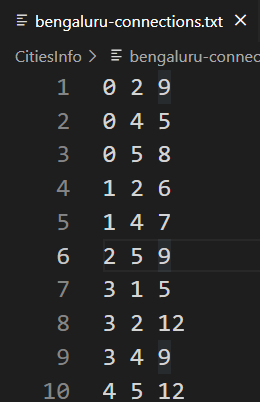
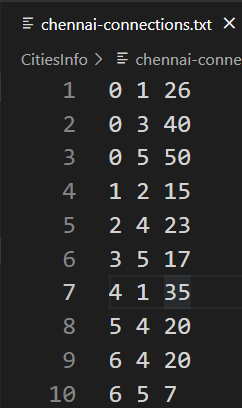


1) If Selected Option is 1:

* All the places that are in the city are stored in (city name).txt file.

* To complete this project, we have taken some places .
* The 0’s and 1’s in these files represent the availability of metro at that place.
* All the Connections between these places are stored in the (city name)-connections.txt file.

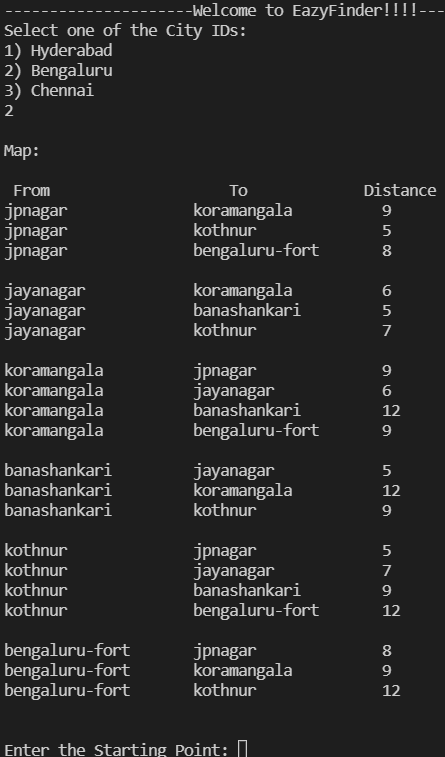
  

* The first number is the source as per the (city name).txt file. Zero-indexing is followed.
* The second number is the destination as per the (city name).txt file. Zero-indexing is followed.
* The third number is the distance between the source and destination.

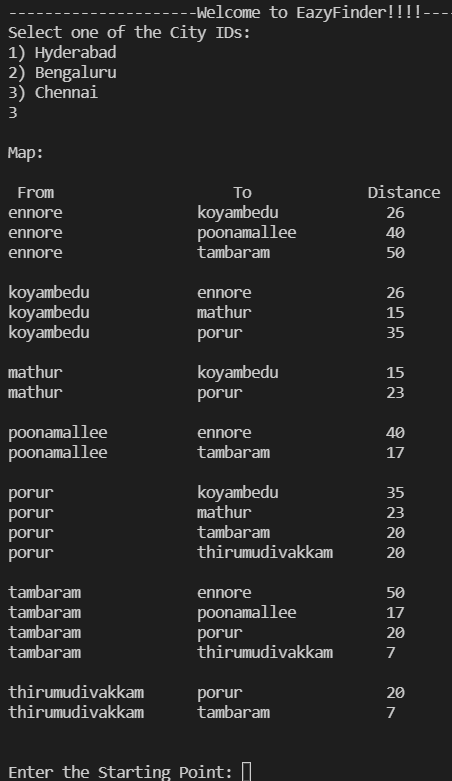
These are the 3 Cities where our services are available:



If Hyderabad is Selected If Bengaluru is Selected

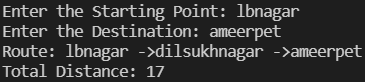
 

If Chennai is Selected



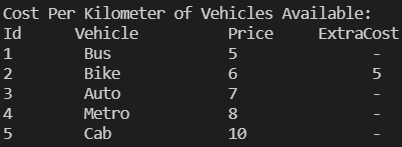
* The Map is displayed based on the selected city as shown above.

Source and Destination Locations Input:

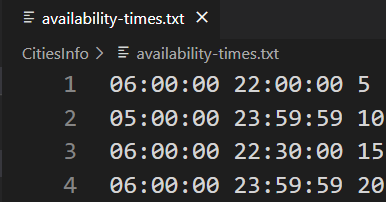


* This displays the route from source to destination that takes the shortest distance, and prints that distance. To get this shortest distance we have used the **Dijkstra’s Algorithm** (or) the **Single Source Shortest Path Algorithm.**

This is the Cost Per Kilometer of different Vehicles available.



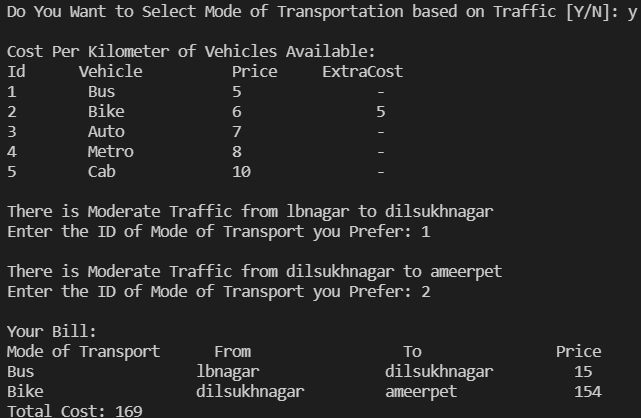
* The Extra Cost is applied based on the times that particular vehicle is available. The start and end times of a particular vehicle are stored in availability-times.txt file in the CitiesInfo folder.



* The Order is starting time, end time, and the extra cost that should be applied if the service is booked before start time or after end time. This cost is added to the actual cost of the vehicle and the total cost is calculated.
* The times are stored in 24 hr format because while calculating the current time the corresponding inbuilt function returns the time in 24 hr format.
* The 1st row is dedicated to Bike, 2nd to Auto, 3rd to Metro and 4th to Cab. As bus will be available at any time it is not given any row.

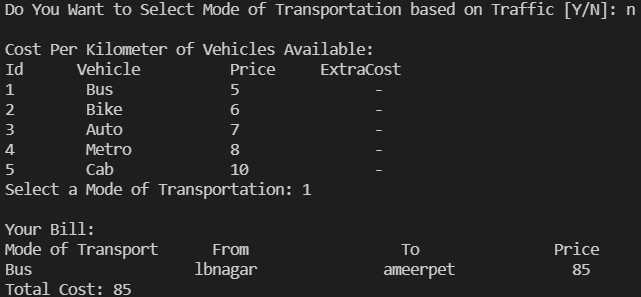
**User is given an option to select the mode of transportation based on Traffic**:

**a) If yes:**



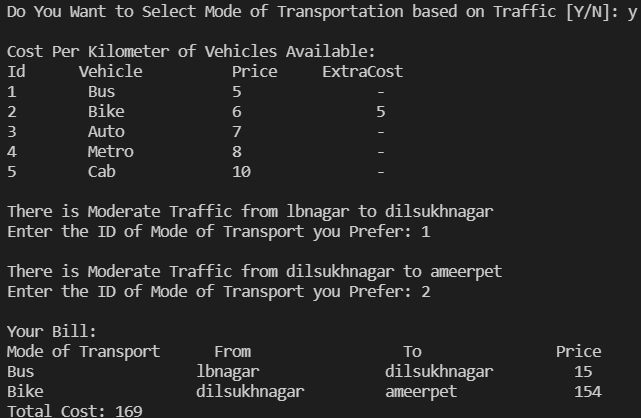
* The traffic between the places is calculated using the rand() inbuilt function. If the random number is 0, then there is no traffic between those places, else if the random number is between 0(Exclusive) and 0.5(Inclusive), then the traffic is Moderate, else if the random number is between 0.5(Exclusive) and 1(Inclusive), then the traffic is Heavy.

**b) If No:**

****

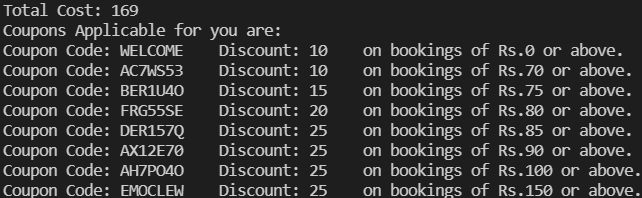
* If no, only one mode of transport is given.

**We Also Display the Bill to the User:**



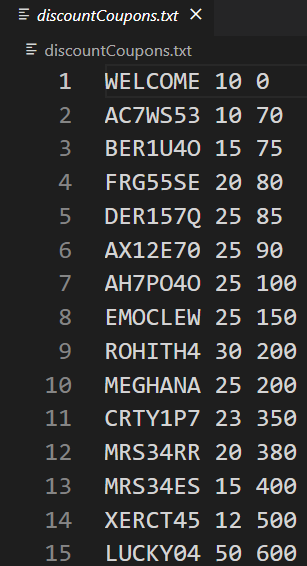
* We display bill to the user that contains the Mode of Transportation, places from which the transport is used and the cost that is calculated using the distance (the shortest) between those places and the cost of that vehicle. And at last the Total Cost is shown.

**Based on the Cost we also give the Coupons Applicable for the User:**

****

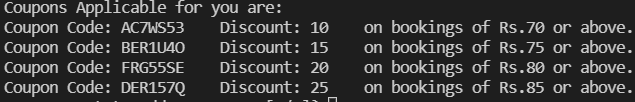
* The Coupons are stored in discountCoupons.txt file.

discountCoupons.txt file:



* The first is the Discount Coupon Code, second is the Discount Percentage and the third is the cost after which the discount coupon can be applied.

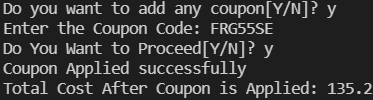
**Coupon Codes that get displayed for old users:**



* The WELCOME coupon is only applicable for the users who are booking for the first time.

**The User can apply the coupon code or just proceed without applying the coupon:**

**a) If yes:**

****

We also display the cost after applying the discount coupon.

Formula: float total\_cost = cost - ((float)couponDiscount[i]/100)\*cost;

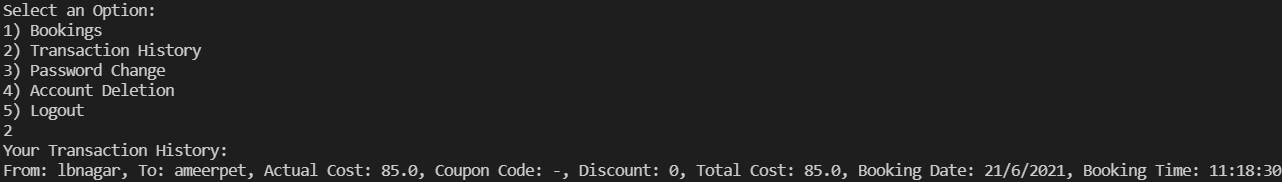
**b) If no:**

****

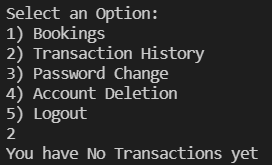
* Total Cost will be same if no Coupon is applied.

2) If Selected Option is 2:

a) If any Transactions are there, those are printed on the screen

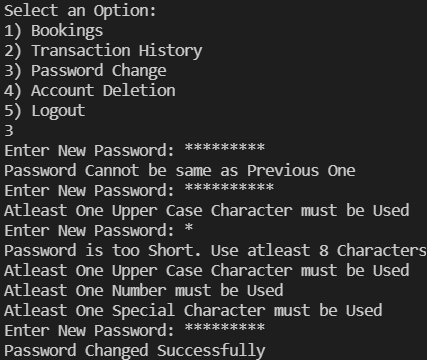


b) If there are no Transactions , the following message is shown



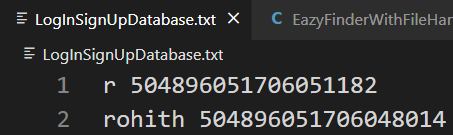
3) If Selected Option is 3:

We prompt for a new password.

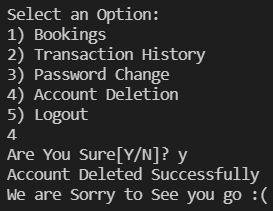


* Password Changing also depends on rules.
* The password must follow all the rules.
* The new password cannot be the same as old password.
* If all the rules are satisfied, the password gets changed.

LogInSignUpDatabase.txt file after password change:

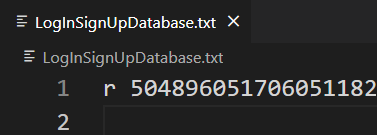


4) If Selected Option is 4:



* First we ask for confirmation. If yes the details of that user gets deleted from the LogInSignUpDatabase.txt file and also the (username).txt file in TransactionHistories get deleted automatically and the program exits.

LogInSignUpDatabase.txt file after account deletion:



* The rohith’s details got deleted.

TransactionHistories Folder after account deletion:



* rohith.txt folder is deleted automatically.