

**PROJECT REPORT:**

**EXPLORER’S TREAT**

***INTRODUCTION***

“Travel makes one modest. You see what a tiny place you occupy in the world”

---Gustave Flauber

Travelling is something everyone is fond of. This program focuses on collecting and managing the data of various flights, trains and hotels, this is done by the implementation of various data structures. This program will house all details about train and flight schedules, and the availability of rooms in various hotels and will enable the user to book tickets for their preferred mode of travel and even make reservations in their chosen place of stay subject to availability.

# DATA STRUCTURES USED

## Graph: To check connectivity of two places

## Map: To implement the graph

## Vector: To store list of destinations connected to source

## Queue: To implement BFS of graph

MAIN FUNCTIONS

* All get and set functions in every class for value of private data members
* View\_(flight/train/hotel)\_details-to view details of all the registered entities
* Enter\_(flight/train/hotel)\_details-to enter the details of any entity.
* Train/flight/hotel reservation()-to make reservations
* Cancellation(); to cancel reservation
* Menu(): to display choice
* Mapping():Creates graph to connect source to all connected destinations
* Searchs() and bfs(): BFS traversal for graph to check for path between two places
* Updatebankbalance(): to update bank balance
* flight/train/hotel::delete1():to delete a specific entry based on the respective flight/hotel/train no
* Void user::deactivate()-this is to delete the data of user from the file
* void Deposit(void) – it is to add money to wallet from bank account
* Checkemail() and checkmobile()-there are to confirm whether the email and mobile entered by user is correct or not
* tickets::display()-Displays all reservations made from the user’s account

Name of classes:

* Admin
* Pb:passbook
* User
* Flight
* Travel
* Train
* Hotel
* Tickets
* Bank