# RAILWAY MANAGEMENT SYSTEM

**PRESENTED BY:**
**R.JENNIEs**

# OBJECTIVES

ALL THE MANUEL WORK SHOULD BE COMPUTERIZIED SO THAT LOAD OF EMPLOYESS SHOULD BE DECREASED

THE DATABASE IS SAVED IN COMPUTER RATHER THAN REGISTER MANUALLY

EASY TO STORE INFORMATION AND ACCESS THE INFORMATION

# PROBLEM STATEMENTS:

- Railway management system This is done to replace the manual entering and processing of booking tickets which are error prone and tedious . This system also maintains information about passengers.

# IN THIS PROJECT WE ARE GOING TO INCLUDE  THE ENTITIES RELATED TO RAILWAY MANAGEMENT SYSTEM

Login

Passenger_info

Train

Booking_detail

# LIST OF TABLES:

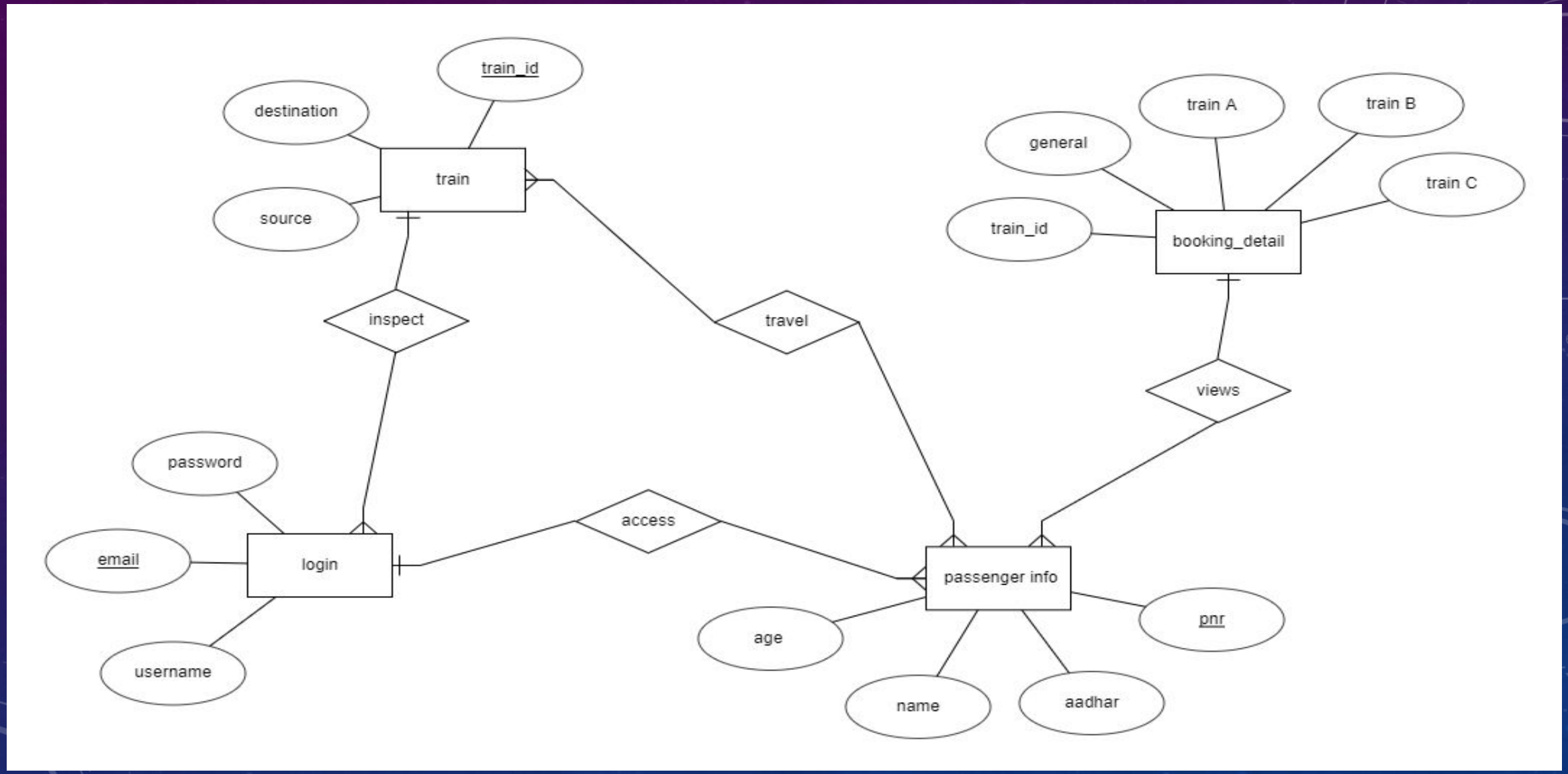
```
mysql> desc login;
+----------+-------------+------+-----+---------+-------+
| Field    | Type        | Null | Key | Default | Extra |
+----------+-------------+------+-----+---------+-------+
| username | varchar(15) | NO   | PRI | NULL    |       |
| password | varchar(15) | YES  |     | NULL    |       |
| email    | varchar(50) | YES  |     | NULL    |       |
+----------+-------------+------+-----+---------+-------+
3 rows in set (0.00 sec)
```

```
mysql> desc passenger_info;
+--------+-------------+------+-----+---------+-------+
| Field  | Type        | Null | Key | Default | Extra |
+--------+-------------+------+-----+---------+-------+
| name   | varchar(20) | YES  |     | NULL    |       |
| aadhar | varchar(15) | YES  |     | NULL    |       |
| pnr    | varchar(6)  | NO   | PRI | NULL    |       |
| age    | varchar(20) | YES  |     | NULL    |       |
+--------+-------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

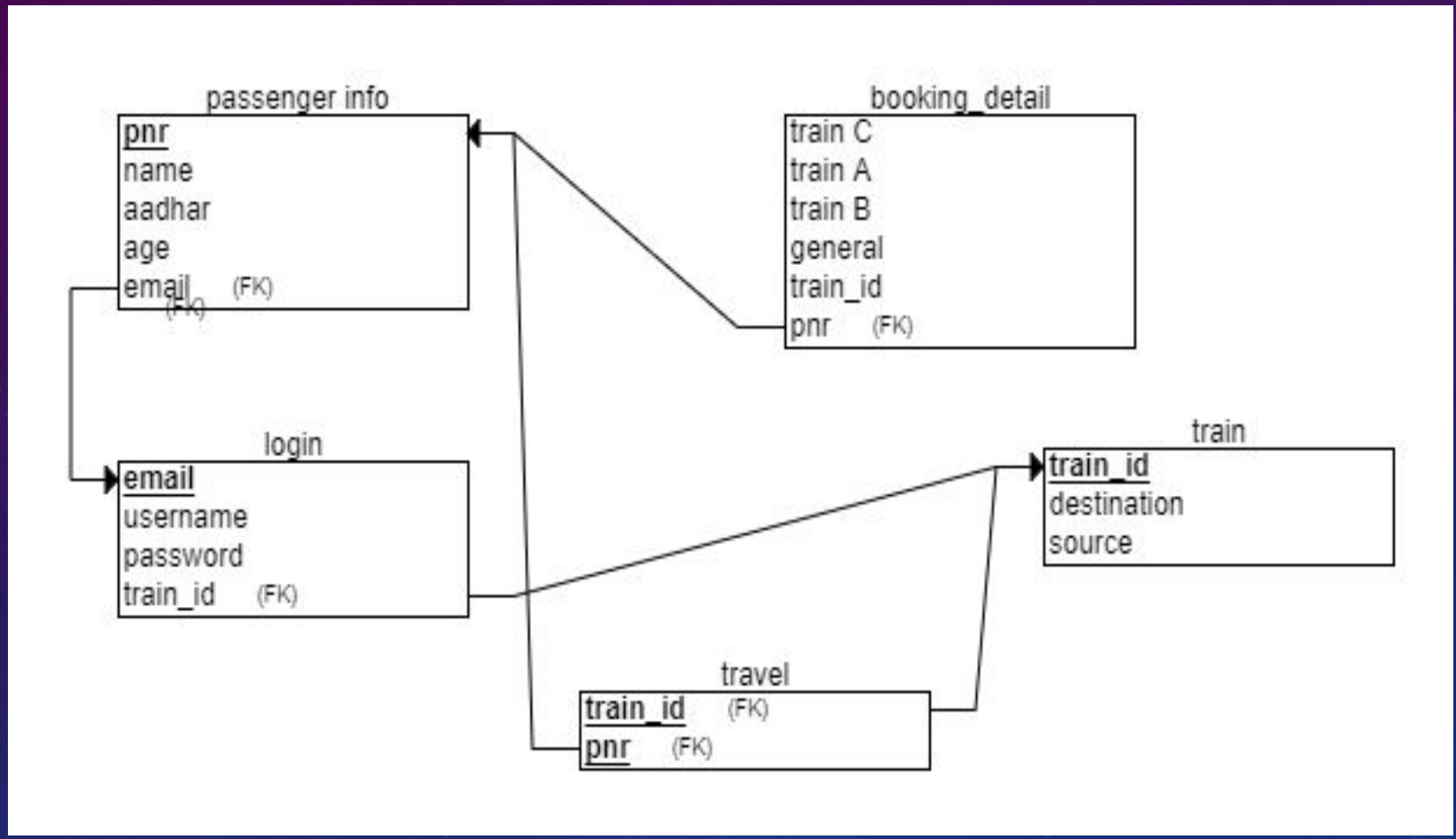
```
mysql> desc train;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| train_id    | int         | NO   | PRI | NULL    |       |
| source      | varchar(20) | NO   |     | NULL    |       |
| destination | varchar(20) | YES  |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
3 rows in set (0.00 sec)
```

```
mysql> desc booking_details;
+----------+-------------+------+-----+---------+-------+
| Field    | Type        | Null | Key | Default | Extra |
+----------+-------------+------+-----+---------+-------+
| train_id | int         | YES  | MUL | NULL    |       |
| train_a  | varchar(20) | YES  |     | NULL    |       |
| train_b  | varchar(20) | YES  |     | NULL    |       |
| train_c  | varchar(20) | YES  |     | NULL    |       |
| general  | varchar(20) | YES  |     | NULL    |       |
+----------+-------------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```

# E-R DIAGRAM:


train_id
destination
train
source
inspect
travel
train A
train B
general
train C
train_id
booking_detail
views
password
access
email
login
passenger info
pnr
age
username
name
aadhar

# SCHEMA DIAGRAM:



# NORMALISATION

1 NF:

since all the records are unique in the table its satisfies 1nf

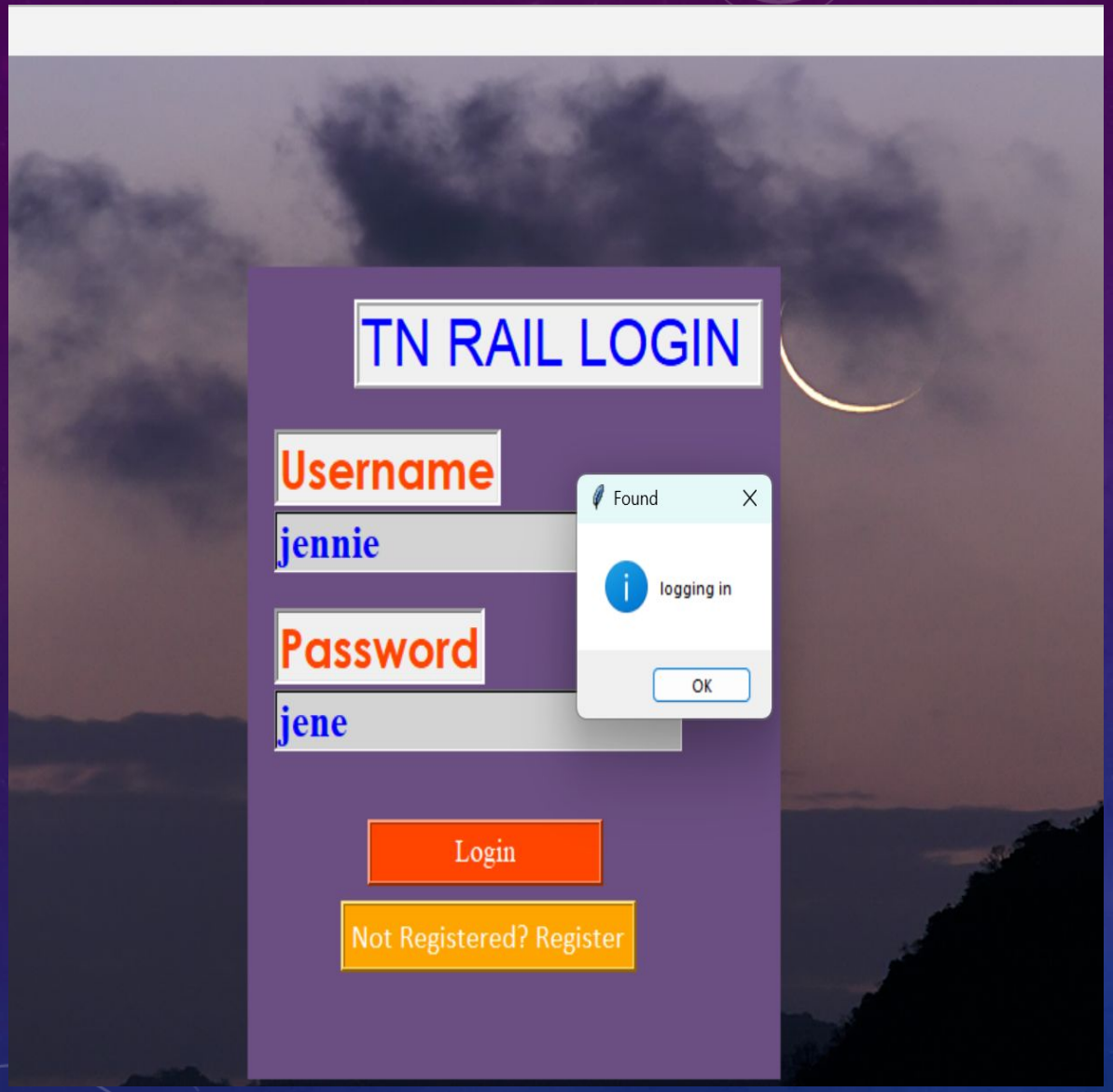
2nf:

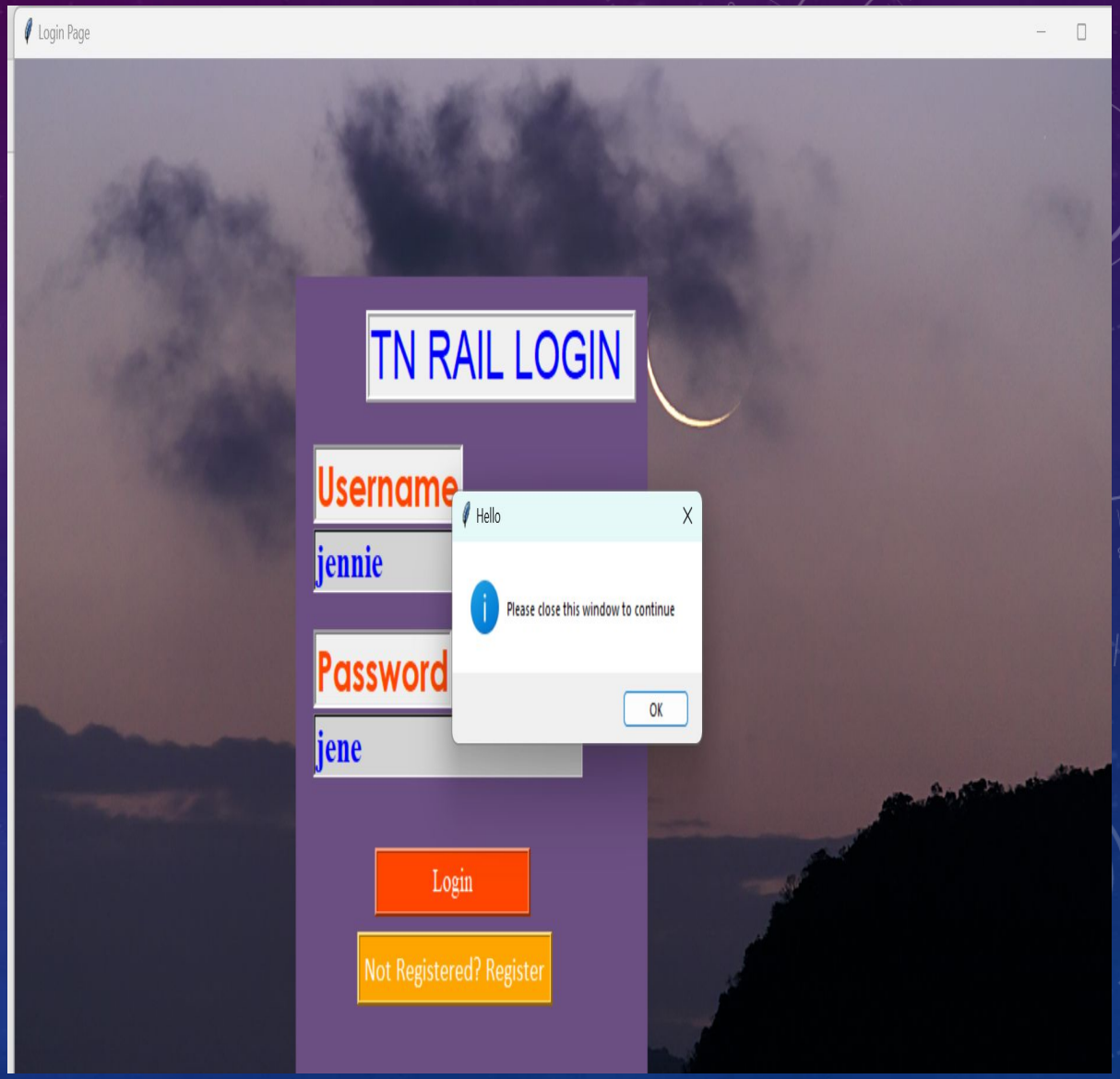
partial functional dependency (train_id,source),(train_id,destination)

```
import tkinter as tk
from tkinter import messagebox
import sys
import calendar

w1=tk.Tk()
w1.geometry('1366x720')
w1.title("Login Page")

# Variables to store entered values on login page
uservalue=tk.StringVar()
passwordvalue=tk.StringVar()
# Function to be called if Login btn is clicked

counter=0

def login():

    # connection
    import mysql.connector as con
    connection = con.connect(host='localhost',port='3306',user='root', password='jennie@123', database='rail1')

    cursor = connection.cursor()


    if uservalue.get() == "" or passwordvalue.get() == "":
        messagebox.showerror("Error", "All fields are required", parent=w1)
        sys.exit()

  #         calculating number of rows in the database
    cursor.execute("select count(*) from data")
    afetch=cursor.fetchone()
    bfetch=afetch[0]  #It has the number of rows

    username = uservalue.get()
    password = passwordvalue.get()

#     Checking if username and password are correct and match
    cursor.execute('select username,password from data where username=username and password=password')
    row=cursor.fetchall()
```

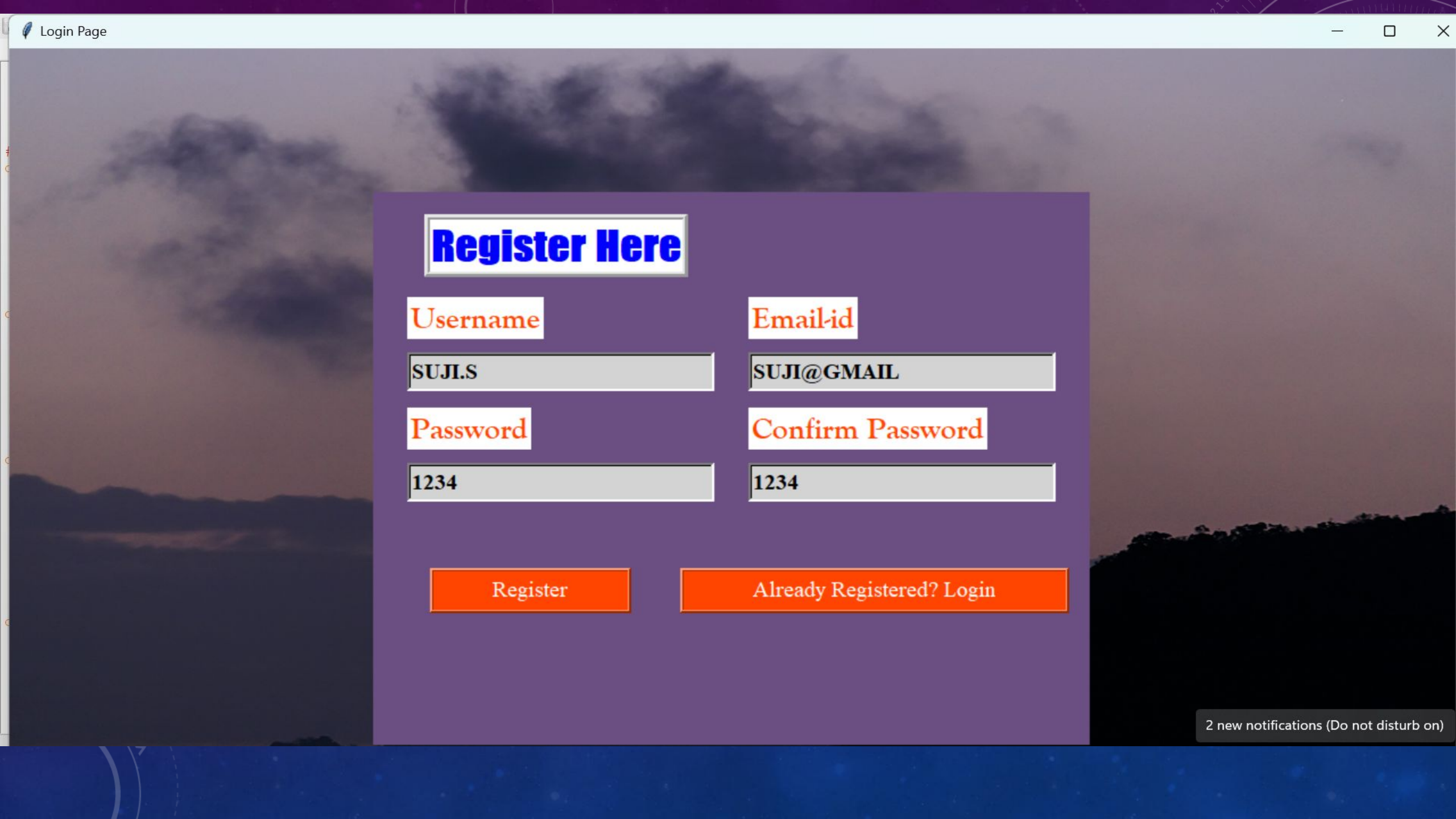
```
    global counter

    # as declared currently counter=0

    for i in range(0,bfetch):
        if(row[i][0]==username and row[i][1]==password):
            counter=1
    if(counter==1):
        messagebox.showinfo("Found","logging in")
        messagebox.showinfo("Hello", "Please close this window to continue")
        # print("found")


    else:
        messagebox.showinfo("Not found","password and uservalue  does not match")

    connection.close()

# Variables for storing entered sign up data
user2value=tk.StringVar()
pass2value=tk.StringVar()
emailvalue=tk.StringVar()
confpassvalue=tk.StringVar()
    # function  to be called if Register button clicked

# If Register btn is clicked
```

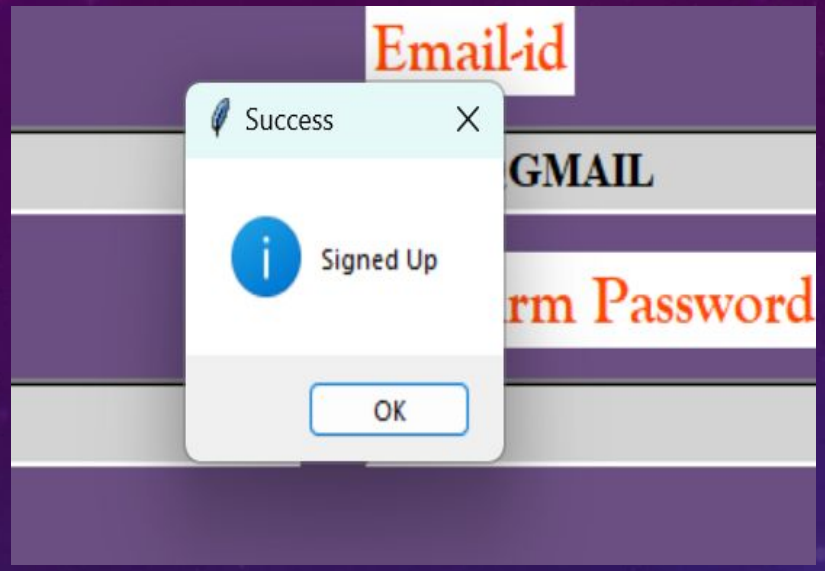

TN RAIL LOGIN
Username
jennie
Password
jene
Login
Not Registered? Register

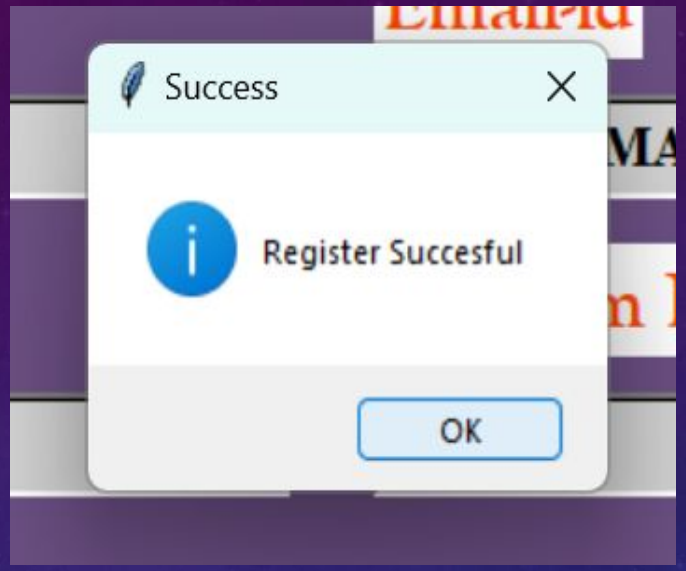

TN RAIL LOGIN
Username
jennie
Password
jene
Login
Not Registered? Register
Found
logging in
OK

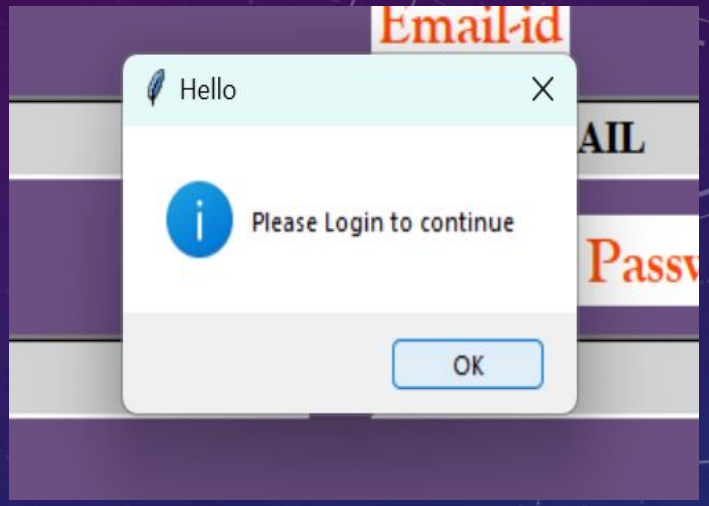

Login Page
TN RAIL LOGIN
Username
jennie
Password
jene
Login
Not Registered? Register
Hello
Please close this window to continue
OK

```
def register():
    if (user2value.get()==""or emailvalue.get()==""or pass2value.get()==""or confpassvalue.get()==""):

        messagebox.showerror("Error","All Fields Are Required",parent=w1)

    elif pass2value.get()!=confpassvalue.get():
        messagebox.showerror("Error","Password and Confirm Password Should Be Same",parent=w1)
    else:
        printuser=user2value.get()
        printemail=emailvalue.get()
        printpass=pass2value.get()

    # connection
    import mysql.connector as con
    connection = con.connect(host='localhost',port='3306', user='root', password='jennie@123', database='rail1')
    cursor = connection.cursor()


    cursor.execute("INSERT INTO data VALUES(%s,%s,%s)",(printuser,printpass,printemail))
    messagebox.showinfo("Success","Signed Up")
    print(printuser,printpass,printemail)

   #trial
    cursor.execute("INSERT INTO data(username,password,email) VALUES (%s, %s , %s)", (printuser, printpass, printemail))
    connection.commit()


    # connnection.commit()

    connection.close()

    messagebox.showinfo("Success", "Register Succesful", parent=w1)
    messagebox.showinfo("Hello","Please Login to continue" )
```


Login Page
Register Here
Username
SUJI.S
Email-id
SUJI@GMAIL
Password
1234
Confirm Password
1234
Register
Already Registered? Login
2 new notifications (Do not disturb on)



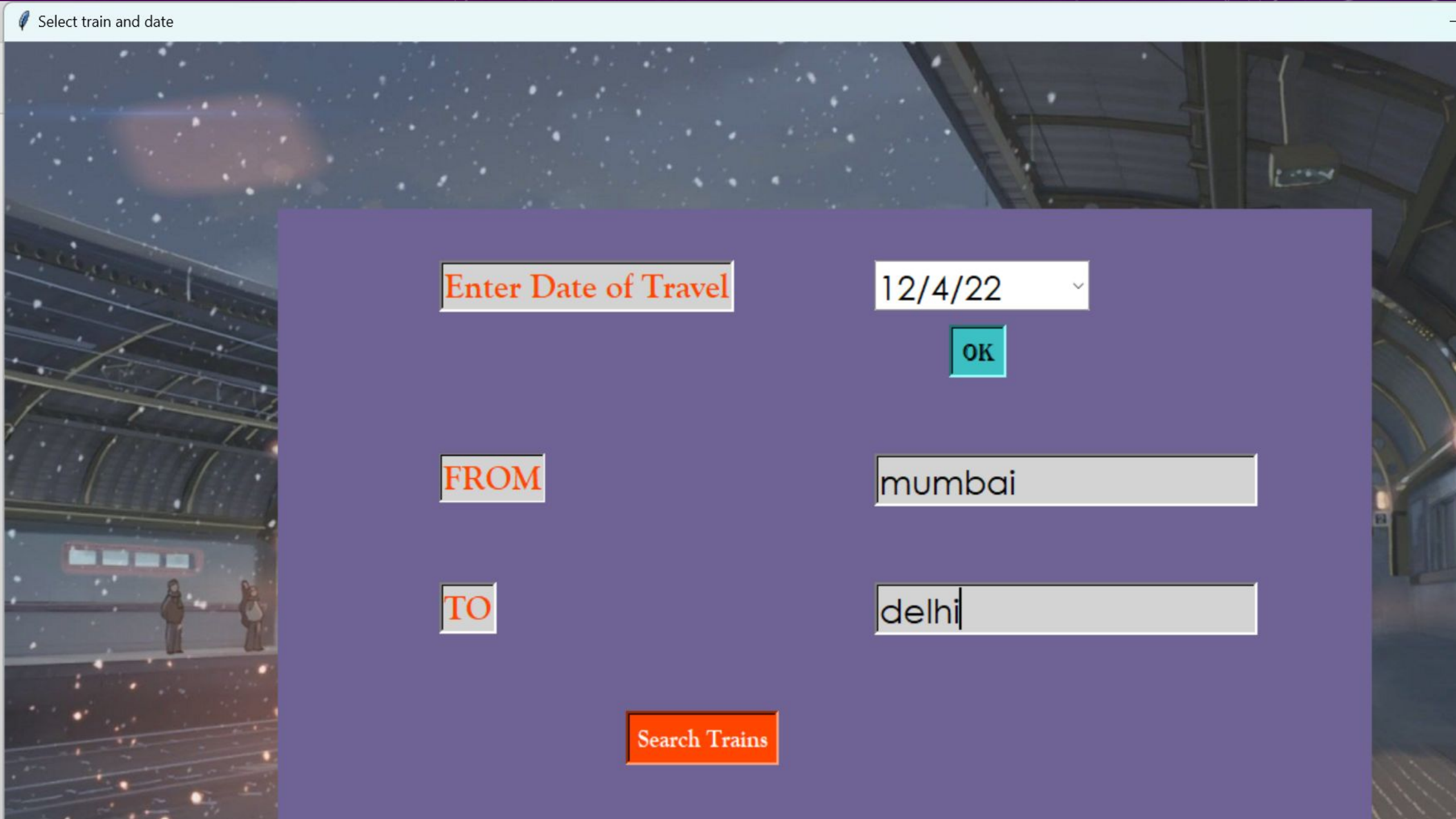



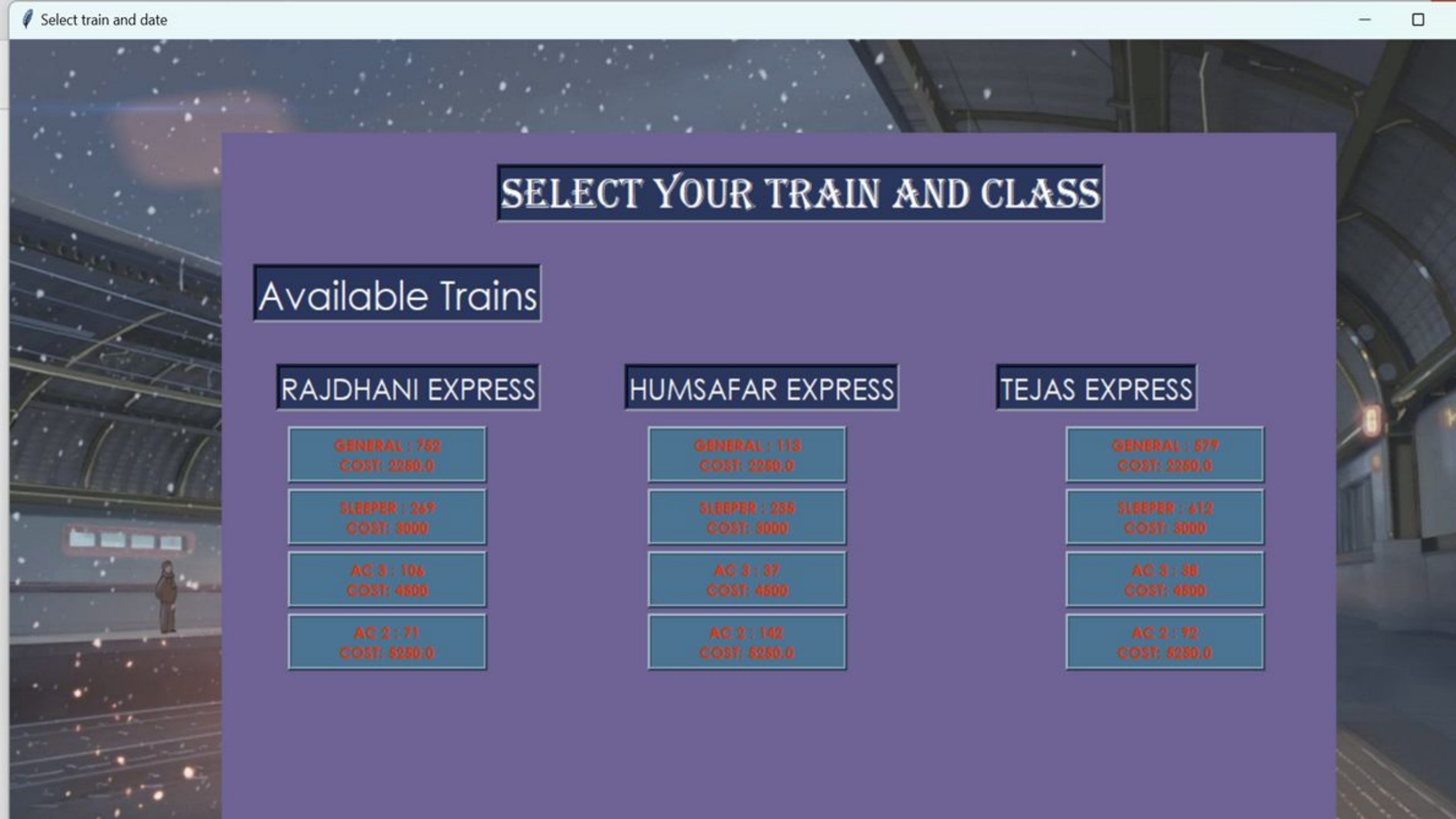

| | | |
|---|---|---|
| raj | koothrapalli | rajesh@gmail |
| root | root | root@gmail |
| sheldon1 | cooper | sheldon@nobel |
| shikha | 123 | shikha2@gmail |
| skd | jebi | fslfn@gmail |
| trial20 | 123 | trial@gmail |
| trial234 | 12 | qwe@gmail |
| world | 12 | world@gmail |
| xyz123@12 | gh | yzxdr@gmail |
| youtube | youtube | you@gmail |
| teju | 12345 | teju@gmail |
| teju | 12345 | teju@gmail |
| ramyagk | giri | ramya@gmail |
| ramyagk | giri | ramya@gmail |
| SUJI.S | 1234 | SUJI@GMAIL |

File Edit Format Run Options Window Help

```
t1,t2,t3,distance='','','',''
def train():
    source=fromvalue.get()
    destination=tovalue.get()
    source=source.upper()
    destination=destination.upper()
    import mysql.connector as connection
    con=connection.connect(host="localhost",user="root",password="jennie@123",db="rail1")
    cursor=con.cursor()
    cursor.execute("select count(*) from detail")
    afetch=cursor.fetchone()
    bfetch=afetch[0]  #It has the number of rows

    c=0
    global distance
    global t1
    global t2
    global t3

#     Checking if source and destination  match the database
    cursor.execute('select * from detail where source=source AND destination=destination')
    row=cursor.fetchall()
    # print(row[1])
# global counter

    # as declared currently counter=0

    for i in range(0,bfetch):
        if(row[i][0]==source and row[i][1]==destination):
                c=1
                distance=row[i][2]
                t1=row[i][3]
                t2=row[i][4]
                t3=row[i][5]


    if(c==1):

        print(distance,t1,t2,t3)
```


Select train and date
Enter Date of Travel
12/4/22
OK
FROM
mumbai
TO
delhi
Search Trains

```
mysql> select*from detail;
+-----------+-------------+----------+----------------------+--------------------+--------------------+
| source    | destination | distance | traina               | trainb             | trainc             |
+-----------+-------------+----------+----------------------+--------------------+--------------------+
| DELHI     | MUMBAI      | 3000     | RAJDHANI EXPRESS     | INTERCITY EXPRESS  | SHATABDI EXPRESS   |
| GORAKHPUR | LUCKNOW     | 1000     | VAISHALI EXPRESS     | INTERCITY EXPRESS  | GORAKHDHAM EXPRESS |
| a         | a           | 1000     | bullet               | train              | super              |
| GKP       | DEL         | 1500     | ABC                  | ABC                | ABC                |
| DELHI     | LUCKNOW     | 1000     | VAISHALI EXPRESS     | GORAKHDHAM EXPRESS | RAJDHANI EXPRESS   |
| GORAKHPUR | MUMBAI      | 3000     | KUSHINAGAR EXPRESS   | GKP LTT EXPRESS    | BANDRA EXPRESS     |
| MUMBAI    | GORAKHPUR   | 3000     | KUSHINAGAR EXPRESS   | GKP LTT EXPRESS    | BANDRA EXPRESS     |
| LUCKNOW   | DELHI       | 1000     | VAISHALI EXPRESS     | GORAKHDHAM EXPRESS | RAJDHANI EXPRESS   |
| GORAKHPUR | DELHI       | 2000     | VAISHALI EXPRESS     | GORAKHDHAM EXPRESS | SAMPARK EXPRESS    |
| DELHI     | GORAKPUR    | 2000     | VAISHALI EXPRESS     | GORAKHDHAM EXPRESS | SAMPARK EXPRESS    |
| DELHI     | MUMBAI      | 1500     | RAJDHANI EXPRESS     | HUMSAFAR EXPRESS   | TEJAS EXPRESS      |
| MUMBAI    | DELHI       | 1500     | RAJDHANI EXPRESS     | HUMSAFAR EXPRESS   | TEJAS EXPRESS      |
| GORAKHPUR | HYDERABAD   | 2500     | GKP YELHANKA EXPRESS | GKP SC EXPRESS     | GKP YPR EXPRESS    |
| HYDERABAD | GORAKHPUR   | 2500     | GKP YELHANKA EXPRESS | GKP SC EXPRESS     | GKP YPR EXPRESS    |
| GORAKHPUR | PATNA       | 700      | PATLIPUTRA EXPRESS   | VAISHALI EXPRESS   | SAMPARK EXPRESS    |
| PATNA     | GORAKHPUR   | 700      | PATLIPUTRA EXPRESS   | VAISHALI EXPRESS   | SAMPARK EXPRESS    |
+-----------+-------------+----------+----------------------+--------------------+--------------------+
16 rows in set (0.00 sec)
```


Select train and date
SELECT YOUR TRAIN AND CLASS
Available Trains
RAJDHANI EXPRESS
GENERAL : 752
COST: 2250.0
SLEEPER : 269
COST: 3000
AC 3 : 106
COST: 4500
AC 2 : 71
COST: 5250.0
HUMSAFAR EXPRESS
GENERAL : 113
COST: 2250.0
SLEEPER : 235
COST: 3000
AC 3 : 37
COST: 4500
AC 2 : 142
COST: 5250.0
TEJAS EXPRESS
GENERAL : 579
COST: 2250.0
SLEEPER : 612
COST: 3000
AC 3 : 38
COST: 4500
AC 2 : 92
COST: 5250.0

```
pnr=0
pnrvalue1,pnrvalue2='',''
number_of_passengers=0
# if book btn clicked
def booked():
    global number_of_passengers
    m=aadhar1.get()
    n=aadhar2.get()
    if m!='' and n!='':
        number_of_passengers=2
    elif m=='' and n!='':
        number_of_passengers=1
    elif m!='' and n=='':
        number_of_passengers=1
    else:
        number_of_passengers=0

    print(number_of_passengers)

    global pnr
    global pnrvalue1
    global pnrvalue2
    if number_of_passengers==2:
        pnr=2
    elif number_of_passengers==1:
        pnr=1
    else:
        pnr=0
    if(pnr==2):
        pnrvalue1="D"+str(random.randint(1,9))+str(random.randint(0,9))+str(random.randint(1,9))+str(random.randint(1,9))+str(random.randint(1,9))

        pnrvalue2='H'+str(random.randint(1,9))+str(random.randint(1,9))+"R"+str(random.randint(1,9))+str(random.randint(1,9))
    elif(pnr==1):
        pnrvalue1="D"+str(random.randint(1,9))+str(random.randint(0,9))+str(random.randint(1,9))+str(random.randint(1,9))+str(random.randint(1,9))

    print(pnrvalue1)
    print(pnrvalue2)

    n1=name1.get()
    n2=name2.get()
    a1=age1.get()
```

```
    a1=age1.get()
    a2=age2.get()
    if number_of_passengers==2:
        import mysql.connector as my
        con=my.connect(host="localhost",port='3306',user='root',password='jennie@123',db='rail1')
        cursor=con.cursor()

        cursor.execute("INSERT INTO passenger values(%s,%s,%s,%s)",(n1,m,a1,pnrvalue1))
        cursor.execute("INSERT INTO passenger values(%s,%s,%s,%s)",(n2,n,a2,pnrvalue2))
        con.commit()

        final()
        # print('here')
    elif number_of_passengers==1:
        import mysql.connector as my
        con=my.connect(host="localhost",user='root',password='jennie@123',db='rail1')
        cursor=con.cursor()

        cursor.execute("INSERT INTO passenger values(%s,%s,%s,%s)",(n1,m,a1,pnrvalue1))

        con.commit()

        final()
```

Select train and date

# PASSENGER DETAILS

NAME jennie.R

AGE 19

AADHAR NUMBER 546346767

BOOK YOUR TICKET

MORE PASSANGERS?

Select train and date

# BOOKING SUCCESSFUL

Passenger : jennie.R

pnr : D28587

YOUR BOOKING IS IN SLEEPER

# SAMPLE OUTPUT

1500 RAJDHANI EXPRESS HUMSAFAR EXPRESS TEJAS EXPRESS

1

D28587

Jennie .R

# CONCLUSION

From this project ,we like to conclude that this mini-project reduces the human intervention in booking tickets of railway and successfully we got outputs of passenger details.

# THANK YOU