

VELAMMAL VIDHYASHRAM MAMBAKKAM



COMPUTER SCIENCE

Topic: Railway ticket booking

Academic year: 2020-2021

Done By:

Harsavardhanan K.V.

XII - 'D'

BONAFIDE CERTIFICATE

This is to certify that this **COMPUTER SCIENCE** Project on the topic

Railway ticket booking has been successfully completed by

HARSAVARDHANAN K.V. of class **XII D**, Roll.no..... at Velammal

Vidhyashram, Mambakkam, for the partial fulfilment of this project as a

part of **All India Senior School Certificate Examination**-CBSE, New Delhi

for the academic Year **2020 - 2021**

Date:

Signature of Principal

Signature of the Guide

Signature of the
Internal Examiner

Signature of the
External Examiner

ACKNOWLEDGEMENT:

I wish to express my deep gratitude and sincere thanks to the Senior Principal Mr.R.Arul Kumar, VelammalVidhyashram,Mambakkam for his encouragement given to me and for all the facilities that he provided for this project work. I sincerely appreciate this magnanimity by taking me into his fold for which I shall remain indebted to him.

I extend my hearty thanks to Miss.Dhivya Rajendran,Computer Science Teacher, who guided me to the successful completion of this project. I take this opportunity to express my deep sense of gratitude for her valuable guidance, constant encouragement, immense motivation, which sustained my efforts at all the stage of this project work.

I would like to thank my lab assistant, lab in charge and technical staffs for providing correct information which makes the completion of this project work in an excellent way.

I can't forget to offer my sincere thanks to parents, and also to my classmates who helped me to carry out this project work successful and for their valuable advice and support, which received from them time to time.

TABLE OF CONTENT:

- a. ABSTRACT
- b. SYSTEM CONFIGURATION
- c. INTRODUCTION
 - a. PYTHON
 - b. MYSQL
- d. RELATION BETWEEN PYTHON AND MYSQL
- e. MYSQL QUERIES USED IN THIS PROJECT
- f. DATABASE CONNECTIVITY
- g. LIBRARIES AND MODULES USED
IN THIS PROJECT
- h. SOURCE CODE
- i. OUTPUT (SCREENSHOT)
- j. CONCLUSION
- k. BIBLIOGRAPHY

ABSTRACT:

Railway ticket booking project can be done using python and mysql applications. Railways which is the cheapest mean of transport compared other mean of transport. Nearly more than 20 lakh poor peoples are benifited through the railways and the ticket booking system , therefore in India more than 50 lakhs booking are done within a day and many new schemes and facilities are now created by Indian railways . Moreover Indian railways has also started launching private trains. Indian railways has also introduced India's fastest train named Gatimaan Express from Hazarat Nizamuddin to Jhansi which covers distance of 403 km in 255min(4.15 hrs) with a average speed of 122km/hr and a max speed of 160km/hr.Many lhb trains like rajdhani,shadabthi,jan shadabthi,tejas,duronto etc. are introduced by Indian railways. Due to railways in India the central government of India earns more than 10 Cr profit.

SYSTEM CONFIGURATION:

HARDWARE CONFIGURATION

Microsoft windows 7 professional/windows 8/windows 8.2:

- **Processor** : Intel Core i5 or equivalent
- **Memory** : 2 GB (32-bit), 4 GB (64-bit)
- **Disk space** : 1.5 GB of free disk space

SOFTWARE REQUIREMENTS

- 1 GB RAM (2 GB+ recommended)
- 9-58 GB free hard disk space depending on edition and configuration, including space required for temporary files
- DVD-ROM drive (if installing from a Media Kit DVD)
- Basic GPU – Any vendor DirectX 9.0 class or better (Pixel Shader Level 2)
- Intel® Pentium® or compatible, 1.6 GHz minimum (2GHz+ recommended)
- 1024x768 or higher-resolution monitor
- MOUSE OR OTHER POINTING DEVICE

INTRODUCTION:

PYTHON:

Python is a high-level, interpreted scripting language developed in the late 1980s by Guido van Rossum at the National Research Institute for Mathematics and Computer Science in the Netherlands. The initial version was published at the alt.sources newsgroup in 1991, and version 1.0 was released in 1994.

Python 2.0 was released in 2000, and the 2.x versions were the prevalent releases until December 2008. At that time, the development team made the decision to release version 3.0, which contained a few relatively small

but significant changes that were not backward compatible with the 2.x versions. Python 2 and 3 are very similar, and some features of Python 3 have been backported to Python 2. But in general, they remain not quite compatible.

Both Python 2 and 3 have continued to be maintained and developed, with periodic release updates for both. As of this writing, the most recent versions available are 2.7.15 and 3.6.5. However, an official End of life of January 1,2020 has been established for Python 2, after which time it will no longer be maintained. If you are a newcomer to Python, it is recommended that you focus on Python 3, as this tutorial will do.

Python is still maintained by a core development team at the Institute, and Guido is still in charge, having been given the title of BDFL (Benevolent Dictator For Life) by the Python community. The name Python, by the way, derives not from the snake, but from the British comedy troupe Monty Python's Flying Circus, of which Guido was, and presumably still is, a fan. It is common to find references to Monty Python sketches and movies scattered throughout the Python documentation.

Python is Popular

Python has been growing in popularity over the last few years. The 2018 Stack Overflow Developer Survey ranked Python as the 7th most popular and the number one most wanted technology of the year. World-class software development countries around the globe use Python every single day. According to research by Dice Python is also one of the hottest skills to have and the most popular programming language in the world based on the Popularity of programming Language Index.

Python is Interpreted

Many languages are compiled, meaning the source code you create needs to be translated into machine code, the language of your computer's processor, before it can be run. Programs written in an interpreted language are passed straight to an interpreter that runs them directly.

This makes for a quicker development cycle because you just type in your code and run it, without the intermediate compilation step.

One potential downside to interpreted languages is execution speed. Programs that are compiled into the native language of the computer

processor tend to run more quickly than interpreted programs. For some applications that are particularly computationally intensive, like graphics processing or intense number crunching, this can be limiting.

In practice, however, for most programs, the difference in execution speed is measured in milliseconds, or seconds at most, and not appreciably noticeable to a human user. The expediency of coding in an interpreted language is typically worth it for most applications.

Python is Free

The Python interpreter is developed under an OSI-approved open-source license, making it free to install, use, and distribute, even for commercial purposes.

A version of the interpreter is available for virtually any platform there is, including all flavors of Unix, Windows, macOS, smartphones and tablets, and probably anything else you ever heard of. A version even exists for the half dozen people remaining who use OS/2. **Python is Portable**

Because Python code is interpreted and not compiled into native machine instructions, code written for one platform will work on any other platform that has the Python interpreter installed. (This is true of any interpreted language, not just Python.)

Python is Simple

As programming languages go, Python is relatively uncluttered, and the developers have deliberately kept it that way. A rough estimate of the complexity of a language can be gleaned from the number of keywords or

reserved words in the language. These are words that are reserved for special meaning by the compiler or interpreter because they designate specific built-in functionality of the language.

Python 3 has 33 keywords, and Python 2 has 31. By contrast, C++ has 62, Java has 53, and Visual Basic has more than 120, though these latter examples probably vary somewhat by implementation or dialect.

MYSQL:

Database Management System & Types of DBMS:

A **Database Management System (DBMS)** is a software application that interacts with the user, applications and the database itself to capture and analyze data. The data stored in the database can be modified, retrieved and deleted, and can be of any type like strings, numbers, images etc.

Types of DBMS:

There are mainly 4 types of DBMS, which are Hierarchical, Relational, Network, and Object-Oriented DBMS.

Hierarchical DBMS: As the name suggests, this type of DBMS has a style of predecessor-successor type of relationship. So, it has a structure similar to that of a tree, wherein the nodes represent records and the branches of the tree represent fields.

Relational DBMS (RDBMS): This type of DBMS, uses a structure that allows the users to identify and access data *in relation* to another piece of data in the database.

Network DBMS: This type of DBMS supports many to many relations wherein multiple member records can be linked.

Object-oriented DBMS: This type of DBMS uses small individual software called objects. Each object contains a piece of data, and the instructions for the actions to be done with the data.

Structured Query Language (SQL)

SQL is the core of a relational database which is used for accessing and managing the database. By using SQL, you can add, update or delete rows of data, retrieve subsets of information, modify databases and perform many actions. The different subsets of SQL are as follows:

- **DDL** (*Data Definition Language*) –It allows you to perform various operations on the database such as CREATE, ALTER and DELETE objects.
- **DML** (*Data Manipulation Language*)– It allows you to access and manipulate data. It helps you to insert, update, delete and retrieve data from the database.
- **DCL** (*Data Control Language*)– It allows you to control access to the database. Example – Grant or Revoke access permissions.
- **TCL**(*Transaction Control Language*) – It allows you to deal with the transaction of the database. Example – Commit, Rollback, Savepoint, Set Transaction.

MySQL & its Features

MySQL is an open-source relational database management system that works on many platforms. It provides multi-user access to support many storage engines and is backed by Oracle. So, you can buy a commercial license version from Oracle to get premium support services.

The features of MySQL are as follows:

Ease of Management –The software very easily gets downloaded and also uses an event scheduler to schedule the tasks automatically.

- **Robust Transactional Support** –Holds the ACID (Atomicity, Consistency, Isolation, Durability) property, and also allows distributed multi-version support.
- **Comprehensive Application Development** –MySQL has plugin libraries to embed the database into any application. It also supports stored procedures, triggers, functions, views and many more for application development. You can refer to the **RDS Tutorial**, to understand Amazon's RDBMS.
- **High Performance** –Provides fast load utilities with distinct memory caches and table index partitioning.
- **Low Total Cost Of Ownership** –This reduces licensing costs and hardware expenditures.
- **Open Source & 24 * 7 Support** –This RDBMS can be used on any platform and offers 24*7 support for open source and enterprise edition.
- **Secure Data Protection** –MySQL supports powerful mechanisms to ensure that only authorized users have access to the databases.
- **High Availability** –MySQL can run high-speed master/slave replication configurations and it offers cluster servers.
- **Scalability & Flexibility** –With MySQL you can run deeply embedded applications and create data warehouses holding a humongous amount of data.

MySQL Data Types

- **Numeric** – This data type includes integers of various sizes, floating-point(real) of various precisions and formatted numbers.
- **Character-string** – These data types either have a fixed, or a varying number of characters. This data type also has a variable-length string called *CHARACTER*

LARGE OBJECT(CLOB) which is used to specify columns that have large text values.

- **Bit-string** – These data types are either of a fixed length or varying length of bits. There is also a variable-length bit string data type called *BINARY LARGE OBJECT(BLOB)*, which is available to specify columns that have large binary values, such as images.
- **Boolean** –This data type has TRUE or FALSE values. Since SQL, has NULL values, a three-valued logic is used, which is UNKNOWN.
- **Date & Time** –The DATE data type has: YEAR, MONTH, and DAY in the form YYYY-MM-DD. Similarly, the TIME data type has the components HOUR, MINUTE, and SECOND in the form HH:MM: SS. These formats can change based on the requirement.
- **Timestamp & Interval** –The TIMESTAMP data type includes a minimum of six positions, for decimal fractions of seconds and an optional WITH TIME ZONE qualifier in addition to the DATE and TIME fields. The INTERVAL data type mentions a relative value that can be used to increment or decrement an absolute value of a date, time, or timestamp.

Python MySQL Database Connection:

Arguments required to connect MySQL from Python

You need to know the following detail of the MySQL server to perform the connection from Python.

- **Username** – i.e., the username that you use to work with MySQL Server. The default username for the MySQL database is a **root**

Password – Password is given by the user at the time of installing the MySQL database. If you are using root then you won't need the password.

- **Host Name** – is the server name or Ip address on which MySQL is running. if you are running on localhost, then you can use localhost, or it's IP, i.e. 127.0.0.0
- **Database Name** – Database name to which you want to connect.

READ Operation

READ Operation on any database means to fetch some useful information from the database.

Once our database connection is established, you are ready to make a query into this database. You can use either **fetchone()** method to fetch single record or **fetchall()** method to fetch multiple values from a database table.

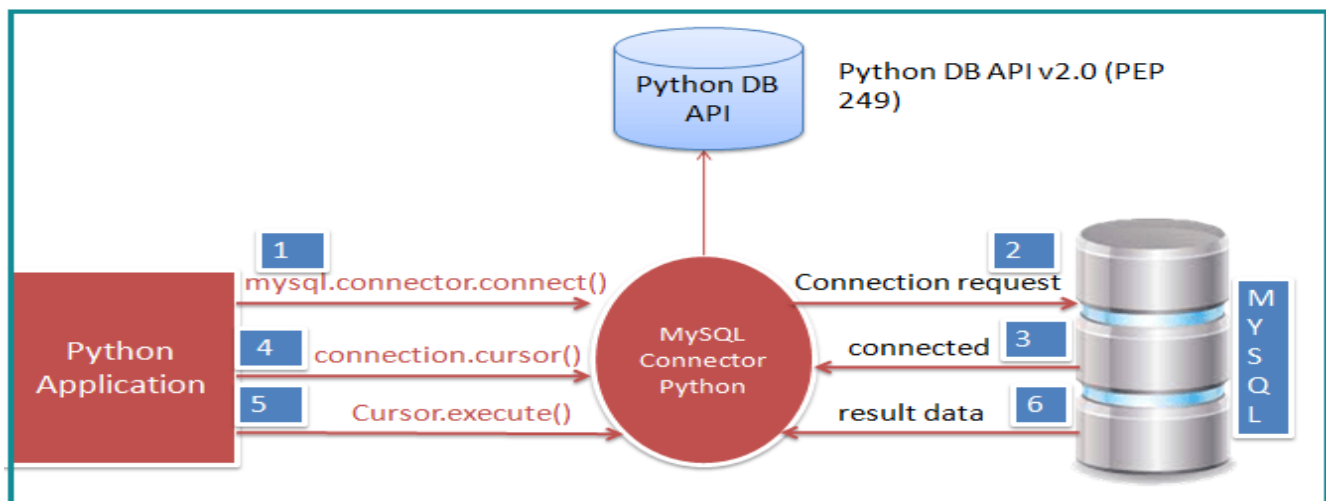
- **fetchone()** – It fetches the next row of a query result set. A result set is an object that is returned when a cursor object is used to query a table.

- **fetchall()** – It fetches all the rows in a result set. If some rows have already been extracted from the result set, then it retrieves the remaining rows from the result set.

DATABASE CONNECTIVITY:

Steps to connect MySQL database in Python using MySQL Connector Python

1. Install MySQL Connector Python using pip.
2. Use the **mysql.connector.connect()** method of MySQL Connector Python with required parameters to connect MySQL.
3. Use the connection object returned by a **connect()** method to create a **cursor** object to perform Database Operations.



4. The **cursor.execute()** to execute SQL queries from Python.
5. Close the Cursor object using a **cursor.close()** and MySQL database connection using **connection.close()** after your work completes.
6. Catch Exception if any that may occur during this process.

MYSQL QUERIES USED IN THIS PROJECT:

1. create database railwayticketproject;
2. use railwayticketproject;
3. create table ticket(name varchar(20), age integer, gender varchar(20), tr_name varchar(40), destination varchar(20), fare integer, date varchar(20), class varchar(5), seat_no varchar(10), booking_date varchar(15), ticket_id varchar(10), phone_no varchar(15), travel_ins varchar(5));
4. create table regis(name varchar(30), mail_id varchar(40), passwd varchar(30), pin int(5));

5. `select count(destination)from ticket where tr_name="%s" and destination="%s" and class="%s" and date="%s"`
6. `select count(destination),class,seat_no from ticket where tr_name="%s" and name="%s"`
7. `select * from ticket where name="%s" and class="%s"`
8. `select * from ticket where date="%s" and tr_name="%s"`
9. `select sum(fare) from ticket where date="%s"`
10. `select sum(fare) from ticket where date="%s"`
11. `delete from ticket where name="%s"`
12. `insert into ticket(name,age,gender,tr_name,destination,fare,date,class,seat_no)values("%s",%s,"%s", "%s", "%s",%s,"%s", "%s", "%s", "%s")`
13. `insert into regis(name,mail_id,passwd,pin)values("%s",%s,"%s", "%s",%s)`
14. `select pin,name from regis where pin in ("%s")`

LIBRARIES AND MODULES USED IN THIS PROJECT:

1.PILLOW LIBRARY:

Pillow package enables opening , manipulating and saving many different file image formats.

2.DATE AND TIME LIBRARY:

Date and time is a python built in package enables in playing with system time,date and converting between representations.

3.MATH LIBRARY:

Math is a python built in module enables user to carry mathematical operations.

4.MY SQL CONNECTOR:

My sql module enables python and my sql database to connect each other and collects data from python and stores data permanently in the form of tables.

5.USER DEFINED MODULE:

seat_arrangement():

User defined module used for seat layout.

check_ticket():

User defined module used for checking seat number for reserving ticket.

status():

User defined module used for checking pnr status.

view_ticket():

User defined module used for viewing our copy of ticket.

date_book():

User defined module used for checking booking date.

date_budget():

User defined module used for checking budget on the given ticket.

ph_one_no():

User defined module used for looking or updating phone no.

delete_ticket():

User defined module used for cancelling ticket.

ticket_reserve():

User defined module used for reserving ticket.

ticket_book ():

User defined module which is also used for reserving ticket.

railway_ticket():

User defined module used for displaying menu of ticket reservation.

today_budget():

User defined module used for checking todays ticket booking.

registration():

User defined module used for registration in railway E ticket booking and catering services.

SOURCE CODE:

```
from PIL import Image,ImageFont,ImageDraw
import datetime
import math,random
import mysql.connector
mydb=mysql.connector.connect(host='localhost',user='root',password='Harish2003',
,database='railwayticketproject')
mycursor=mydb.cursor()
# SOURCE CODE FOR RAILWAY TICKET BOOKING
def ph_one_no():
    zy=input('enter the ticket id for which the no to be changed')
    zw=input('enter the phone no to updated')
    zx='update ticket set phone_no="%s "where ticket_id="%s" '
    ai=zw,zy
    mycursor.execute(zx,ai)
    print('TICKET PHONE NUMBER UPDATED SUCESSFULLY')
    mydb.commit()
    railway_ticket()
def registration():
    Name=input('enter user name')
```

```
email=input('enter user mail id')
    passwd=input('enter password')
    pin=int(input('enter your pin in numbers for security purpose'))
    swl='insert into regis (name,mail_id,passwd,pin) values ("%s","%s","%s",%s)'
    ap=Name,email,passwd,pin
    mycursor.execute(swl,ap)
    print('REGISTRATION DONE SUCESSFULLY')
    mydb.commit()
    railway_ticket()
def today_budget():
    now=datetime.datetime.now()
    abc=now.strftime('%Y-%m-%d %H:%M:%S')
    abcd=abc[0:11:]
    efg='select sum(fare) from ticket where booking_date="%s"'
    mycursor.execute(efg,(abcd, ))
    ghi=mycursor.fetchall()
    for fg in ghi:
        print('TODAY TOTAL BUDGET OF BOOKING',fg[0])
    railway_ticket()
def seat_arrangement():
    aa=1
    bb=2
    cc=3
    dd=4
    ee=5
    ff=6
    gg=7
    hh=8
    seatclass=input('enter SL OR 2A OR 2S OR CC COACH')
```

if seatclass=='SL' or '2S' or 'CC':

print('each coach has 72 seats and total 5 SL coaches')

for mn in range(1,28):

print(aa,bb,cc,' ',gg,)

print(dd,ee,ff,' ',hh,)

print('***')**

aa+=8

bb+=8

cc+=8

dd+=8

ee+=8

ff+=8

gg+=8

hh+=8

else:

aa=1

bb=2

cc=3

dd=4

ee=5

ff=6

print('Each coach has 48 seats and 5 2A coaches')

for nm in range(1,37):

print(aa,bb,cc)

print(dd,ee,ff)

print('***')**

aa+=6

bb+=6

cc+=6

dd+=6

ee+=6

ff+=6

print('BERTH ARRANGEMENT FOR SL AND 2A')

print('LOWER MIDDLE UPPER SIDE UPPER')

print('LOWER MIDDLE UPPER SIDE LOWER')

print(20*'*')

railway_ticket()

def check_ticket(t_r,f_r,d_s,c_h):

phone_no=input('enter your phone no')

if len(phone_no)==10:

pass

else:

print('INVALID PHONE NO')

railway_ticket()

d_ate=input('enter the date which you want to check no of tickets present in your date of journey (yyyy-mm-dd)')

year=d_ate[0:4:]

y_=int(year)

month=d_ate[5:7:]

m_=int(month)

day=d_ate[8:10]

d_=int(day)

if d_<=31 and m_<=12 and y_==2020:

pass

else:

print('Invalid date')

railway_ticket()

asw='select count(destination),seat_no from ticket where tr_name="%s" and destination="%s" and class="%s" and date="%s"'

mycursor.execute(asw,(t_r,d_s,c_h,d_ate))

```

uv=mycursor.fetchall()
for sq in uv:
    pass
yy=sq[0]
if sq[0]==0:
    w=216
    print(20*'*')
    print('AVAILABLE#',w)
    print(20*'*')
    wd=c_h+'001'
    ticket_reserve(t_r,f_r,d_s,c_h,wd,phone_no,d_ate)
elif sq[0]<=216:
    print(20*'*')
    print('AVAILABLE',216-sq[0])
    print(20*'*')
    Asw='select seat_no from ticket where tr_name="%s" and destination="%s"
and class="%s" and date="%s"'
    mycursor.execute(Asw,(t_r,d_s,c_h,d_ate))
    Uv=mycursor.fetchall()
    for ay in Uv:
        pass
        hv=ay[0]
        az=hv[1:6:]
        hz=az[2:5:]
        rz=int(hz)+1
        rv=str(rz)
        if rz<=9:
            hz=az[0:4:]+rv
        elif rz<=99:
            hz=az[0:3:]+rv

```

else:

hz=az[0:2:]+rv

ticket_reserve(t_r,f_r,d_s,c_h,hz,phone_no,d_ate)

elif sq[0]==217:

print(20')**

print('AVAILABLE WL')

print(20')**

c_h='WL'

hy='WL001'

ticket_reserve(t_r,f_r,d_s,c_h,hy,phone_no,d_ate)

elif sq[0]>217:

print(20')**

print('AVAILABLE WL')

print(20')**

c_h='WL'

Asw='select seat_no from ticket where tr_name="%s" and destination="%s" and class="%s" and date="%s"'

mycursor.execute(Asw,(t_r,d_s,c_h,d_ate))

Uv=mycursor.fetchall()

for ay in Uv:

pass

az=ay[1:6:]

hz=az[2:5:]

rz=int(hz)+1

rv=str(rz)

if rz<=9:

hz=az[0:4:]+rv

elif rz<=99:

hz=az[0:3:]+rv

else:

```

        hz=az[0:2:]+rv
        ticket_reserve(t_r,f_r,d_s,c_h,hz,phone_no,d_ate)
    else:
        exit(0)
    mydb.commit()
def status():
    nam_e=input('enter the name to view status')
    _trname=input('enter the train name')
    tickt=input('enter the ticket id')
    ws='select count(destination),class,seat_no from ticket where tr_name="%s" and
name="%s" and ticket_id="%s" '
    mycursor.execute(ws,(_trname,nam_e,tickt))
    wv=mycursor.fetchall()
    for we in wv:
        if we[1] in ("SL","2A","2S","CC"):
            print('THIS TICKET WAS ALREADY CONFIRMED', we[2])
            railway_ticket()
        else:
            print('YOUR TICKET IS NOT YET CONFIRMED AND IT WILL BE
CONFIRMED ONLY AFTER THE FINAL PREPARATION OF CHART')
            railway_ticket()
    mydb.commit()
def year_budget():
    az='select sum(fare) from ticket'
    mycursor.execute(az)
    xi=mycursor.fetchall()
    for ay in xi:
        print('TOTAL BUDGET OF ENTIRE BOOKING',ay[0])
        mydb.commit()
    railway_ticket()

```

```
def view_ticket():  
    no_=int(input('enter the the no of tickets to be viewed'))  
    for o in range(no_):  
        n_ame=input('enter passenger name which you want to view your ticket')  
        c_lass=input('enter your 8 code ticket id')  
        Sql='select * from ticket where name="%s" and ticket_id="%s" '  
        mycursor.execute(Sql,(n_ame,c_lass))  
        l=mycursor.fetchall()  
        for v in l:  
            print(50*'*')  
            print('NAME:',v[0])  
            print('AGE:',v[1])  
            print('GENDER:',v[2])  
            print('TRAIN NAME:',v[3])  
            print('DESTINATION:',v[4])  
            print('FARE:',v[5])  
            print('DATE:',v[6])  
            print('CLASS:',v[7])  
            print('SEAT NO:',v[8])  
            print('BOOKING DATE:',v[9])  
            print('TICKET ID:',v[10])  
            print('PHONE NUMBER:',v[11])  
            print('TRAVEL INSURANCE OPTED:',v[12])  
            print(50*'*')  
            zz='NAME:'+str(v[0])  
            yy='AGE:'+str(v[1])  
            xx='GENDER:'+str(v[2])  
            ww='TRAIN NAME:'+str(v[3])  
            vv='DESTINATION:'+str(v[4])  
            uu='FARE:'+str(v[5])
```

```

tt='DATE:'+str(v[6])
ss='CLASS:'+str(v[7])
qq='SEAT NO:'+str(v[8])
pp='BOOKING DATE:'+str(v[9])
oo='TICKET ID:'+str(v[10])
nn='PHONE NUMBER:'+str(v[11])
mm='TRAVEL INSURANCE OPTED:'+str(v[12])
dfile=input('enter the file name with .txt extension to save your copy
permanently')
dfile='D:\\files\\'+dfile
file=open(dfile,'w')
file.write(50*'*'+'\n'+ 'E-
TICKET'+'\n'+zz+'\n'+yy+'\n'+xx+'\n'+vv+'\n'+uu+'\n'+tt+'\n'+ss+'\n'+qq+'\n'+pp+'
\n'+oo+'\n'+nn+'\n'+mm+'\n'+ 'THANKS FOR BOOKING'+'\n'+50*'*')
file.close()
img = Image.open('D:\\train2.jpg')
font=ImageFont.truetype('arial.ttf',50)
font2=ImageFont.truetype('arial.ttf',30)
font3=ImageFont.truetype('arial.ttf',32)
draw=ImageDraw.Draw(img)
r='TRAIN TICKET'
draw.text(xy=(400,0),text=r,font=font,fill='white')
draw.text(xy=(0,60),text='SOURCE:CHENNAI
EGMORE',font=font2,fill='white')
draw.text(xy=(700,60),text=vv,font=font2,fill='white')
draw.text(xy=(0,140),text=zz,font=font2,fill='white')
draw.text(xy=(0,180),text=yy,font=font2,fill='white')
draw.text(xy=(0,220),text=ww,font=font2,fill='white')
draw.text(xy=(0,260),text=xx,font=font2,fill='white')
draw.text(xy=(0,300),text=uu,font=font2,fill='white')
draw.text(xy=(0,340),text=tt,font=font2,fill='white')

```

```

draw.text(xy=(0,380),text=ss,font=font2,fill='white')
draw.text(xy=(0,420),text=qq,font=font2,fill='white')
draw.text(xy=(0,460),text=pp,font=font2,fill='white')
draw.text(xy=(0,500),text=oo,font=font2,fill='white')
draw.text(xy=(0,540),text=nn,font=font2,fill='white')
draw.text(xy=(0,580),text=mm,font=font2,fill='white')
draw.text(xy=(60,630),text='A JOURNEY OF THOUSAND MILES STARTS
WITH A SINGLE STEP,',font=font3,fill='white')
draw.text(xy=(30,700),text='MAY YOUR JOURNEY BE
ADVENTUROUS,DELIGHTFUL AND ENJOYABLE',font=font3,fill='white')
img.show()
print('YOUR COPY SAVED SUCCESSFULLY')
mydb.commit()
railway_ticket()
def date_book():
    d_ate=input('enter the date which you want to check(yyyy-mm-dd)')
    year=d_ate[0:4:]
    y_=int(year)
    month=d_ate[5:7:]
    m_=int(month)
    day=d_ate[8:10]
    d_=int(day)
    if d_<=31 and m_<=12 and y_==2020:
        pass
    else:
        print('Invalid date')
        date_book()
    t_name=input('enter the train name')
    sql='select * from ticket where date="%s" and tr_name="%s"'
    mycursor.execute(sql,(d_ate,t_name))

```

```
x=mycursor.fetchall()
for q in x:
    print(q)
    mydb.commit()
railway_ticket()
def date_budget():
    d_ate=input('enter the date which you want to check the budget (yyyy-mm-dd)')
    year=d_ate[0:4:]
    y_=int(year)
    month=d_ate[5:7:]
    m_=int(month)
    day=d_ate[8:10]
    d_=int(day)
    if d_<=31 and m_<=12 and y_==2020:
        pass
    else:
        print('Invalid date')
        date_budget()
    sQl='select sum(fare) from ticket where date="%s"'
    mycursor.execute(sQl,(d_ate, ))
    qu=mycursor.fetchall()
    for y in qu:
        print( d_ate,"budget",y[0])
        mydb.commit()
    railway_ticket()
def delete_ticket():
    re=int(input('enter no of tickets to be deleted'))
    for kl in range(re):
        t_name=input('enter passenger name for deleting ticket')
        ticketid=input('enter your 8 digit code ticket id')
```



```

    dele='delete from ticket where name="%s" and ticket_id="%s"'
    mycursor.execute(dele,(t_name,ticketid))
    print('record deleted successfully')
    print('YOUR AMOUNT WILL BE REFUNDED SOON')
    mydb.commit()

railway_ticket()

def ticket_reserve(TRNAME,FARE,DEST,CLASS,SEAT_NO,phoneno,DATE):
    print('WELCOME TO E-BOOKING SYSTEM')
    r=('Y')
    while r!='n':
        print('1.WANT TO GO BACK')
        print('2.TICKET BOOKING RESERVATION')
        option=int(input('enter your choice'))
        if option==1:
            railway_ticket()
        elif option==2:
            people=int(input('enter no of tickets required'))
            arri='CHENNAI EGMORE'
            name_1=[]
            age_1=[]
            gender_1=[]
            tr_name1=[]
            fare_1=[]
            seat1=0
            now=datetime.datetime.now()
            abc=now.strftime('%Y-%m-%d %H:%M:%S')
            ol=abc[0:11:]
            for p in range(people):
                DEST=DEST.upper()
                TRNAME=TRNAME.upper()

```

```
name=input('enter the passenger name')
```

```
name_1.append(name)
```

```
age=int(input('enter your age'))
```

```
age_1.append(age)
```

```
gender=input('enter male or female')
```

```
Gender=gender.upper()
```

```
gender_1.append(gender)
```

```
tr_name1.append(TRNAME)
```

```
fare_1.append(FARE)
```

```
Seat_=SEAT_NO[2:5:]
```

```
seat_=int(Seat_)
```

```
if seat_<=9:
```

```
    po=seat_+seat1
```

```
    pi=str(po)
```

```
    SEAT_NO=SEAT_NO[0:4:]+pi
```

```
    seat1+=1
```

```
    if seat1>1:
```

```
        seat1=1
```

```
    else:
```

```
        pass
```

```
elif seat_<=99:
```

```
    po=seat_+seat1
```

```
    pi=str(po)
```

```
    SEAT_NO=SEAT_NO[0:3:]+pi
```

```
    seat1+=1
```

```
elif seat_<=216:
```

```
    po=seat_+seat1
```

```
    pi=str(po)
```

```
    SEAT_NO=SEAT_NO[0:2:]+pi
```

```
    seat1+=1
```

```

else:
    po=Seat_+seat1
    pi=str(po)
    SEAT_NO=SEAT_NO[0:2:]+pi
    seat1+=1
digits = "0123456789"
_ti= ""
for i in range(8) :
    _ti += digits[math.floor(random.random() * 10)]
    ticketid=_ti
travel=input('WANT TO OPT TRAVEL INSURANCE GIVE y OR ELSE n
')
if travel=='y':
    print('TRAVEL INSURANCE OF 10 RUPEES WILL BE CHARGED')
    FARE+=10
    travel='YES'
else:
    travel='NO'

sqlf='insert into
ticket(name,age,gender,tr_name,destination,fare,date,class,seat_no,booking_date,ticket_id,phone_no,travel_ins)values("%s",%s,"%s", "%s", "%s", %s, "%s", "%s", "%s", "%s", "%s", "%s", "%s")'

s=name,age,Gender,TRNAME,DEST,FARE,DATE,CLASS,SEAT_NO,ol,ticketid,phono,travel

mycursor.execute(sqlf,s)
print('Record inserted sucesfully')
mydb.commit()
x=0
print(50*'#','TICKET',50*'#')
print('NAME:',name)

```

```
print('AGE:',age)
print('GENDER:',Gender)
print('TRAIN NAME:',TRNAME)
print('FARE:',FARE)
print('BOOKED DATE:',ol)
print('DATE OF JOURNEY:',DATE)
print('FROM',arri,'TO',DEST)
print('CLASS:',CLASS)
print('SEAT NO:',SEAT_NO)
print('TICKET ID:',ticketid)
print('PHONE NO:',phoneno)
print('TRAVEL INSURANCE OPTED:',travel)
print(108**')
print('KEEP THIS COPY SAFE FOR FUTURE USE')
x+=1
```

```
railway_ticket()
```

```
def ticket_book():
```

```
    res='Y'
```

```
    while res!='n':
```

```
        print('WELCOME TO RAILWAY TICKET BOOKING')
```

```
        print('SOURCE: CHENNAI EGMORE')
```

```
        print('your destinations are 1.MADURAI 2. TIRUNELVELI
3.KANYAKUMARI 4.TRICHY 5.PUDUCHERRY')
```

```
        print('6.VILLUPURAM 7.TUTICORIN 8.RAMESWARAM 9.SENGOTTAI
10.THANJAVUR')
```

```
        print('11.TIRUCHENDUR 12.KOLLAM 13.GURUVAYUR 14.KARIKAL
15.MANNARGUDI')
```

```
        print('16.KARAIKUDI 17.SALEM 18 MANGALURU CENTRAL 19 DADAR
20 GAYA')
```

```
        dest=input('enter your destination')
```

```
        if dest in ('MADURAI','Madurai','madurai'):
```

```
tr_no=12635
tr_name='Vaigai superfast express'
print('CLASS: 1.SL 2. CC')
Choice=int(input('enter your choice(1 or 2)'))
if Choice==1:
    fare=150
    Choice='SL'
    print("THE TRAIN NO",tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)
else:
    fare=500
    Choice='CC'
    print("THE TRAIN NO",tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)
    check_ticket(tr_name,fare,dest,Choice)
elif dest in ('TIRUNELVELI','Tirunelveli','tirunelveli'):
    tr_no=12631
    tr_name='Nellai superfast express'
    print('CLASS: 1.SL 2. 2A')
    Choice=int(input('enter your choice(1 or 2)'))
    if Choice==1:
        fare=385
        Choice='SL'
        print("THE TRAIN NO",tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)
    else:
        fare=1425
        Choice='2A'
        print("THE TRAIN NO",tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)
        check_ticket(tr_name,fare,dest,Choice)
```

```
elif dest in ('KANYAKUMARI','Kanyakumari','kanyakumari'):
    tr_no=12633
    tr_name='Kanyakumari superfast express'
    print('CLASS: 1.SL 2. 2A')
    Choice=int(input('enter your choice(1 or 2)'))
    if Choice==1:
        fare=415
        Choice='SL'
        print("THE TRAIN NO",tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)
    else:
        fare=1555
        Choice='2A'
        print("THE TRAIN NO",tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)
    check_ticket(tr_name,fare,dest,Choice)
elif dest in ('TRICHY','Trichy','trichy'):
    tr_no=16795
    tr_name='Cholan express'
    print('CLASS: 1.SL 2. 2A')
    Choice=int(input('enter your choice(1 or 2)'))
    if Choice==1:
        fare=250
        Choice='SL'
        print("THE TRAIN NO",tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)
    else:
        fare=965
        Choice='2A'
        print("THE TRAIN NO",tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)
```

```
    check_ticket(tr_name,fare,dest,Choice)
elif dest in ('PUDUCHERRY','Puducherry','puducherry'):
    tr_no=66051
    tr_name='Pondicherry MEMU passenger'
    print('CLASS: 1.2S 2.CC')
    Choice=int(input('enter your choice(1 or 2)'))
    if Choice==1:
        fare=80
        Choice='2S'
        print('THE TRAIN NO',tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)
    else:
        fare=100
        Choice='CC'
        print('THE TRAIN NO',tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)
    check_ticket(tr_name,fare,dest,Choice)
elif dest in ('VILLUPURAM','Villupuram','villupuram'):
    tr_no=66055
    tr_name='Villupuram MEMU passenger'
    print('CLASS: 1.2S 2.CC')
    Choice=int(input('enter your choice(1 or 2)'))
    if Choice==1:
        fare=60
        Choice='2S'
        print('THE TRAIN NO',tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)
    else:
        fare=90
        Choice='CC'
```

```
print("THE TRAIN NO",tr_no,'AND TRAIN NAME',tr_name,'OF  
CLASS',Choice,'WITH FARE OF',fare)  
check_ticket(tr_name,fare,dest,Choice)  
elif dest in ('TUTICORIN','Tuticorin','tuticorin'):  
tr_no=12693  
tr_name='Pearl city superfast express'  
print('CLASS: 1.SL 2. 2A')  
Choice=int(input('enter your choice(1 or 2)'))  
if Choice==1:  
fare=385  
Choice='SL'  
print("THE TRAIN NO",tr_no,'AND TRAIN NAME',tr_name,'OF  
CLASS',Choice,'WITH FARE OF',fare)  
else:  
fare=1425  
Choice='2A'  
print("THE TRAIN NO",tr_no,'AND TRAIN NAME',tr_name,'OF  
CLASS',Choice,'WITH FARE OF',fare)  
check_ticket(tr_name,fare,dest,Choice)  
elif dest in ('RAMESWARAM','Rameswaram','rameswaram'):  
tr_no=22661  
tr_name='Sethu superfast express'  
print('CLASS: 1.SL 2. 2A')  
Choice=int(input('enter your choice(1 or 2)'))  
if Choice==1:  
fare=360  
Choice='SL'  
print("THE TRAIN NO",tr_no,'AND TRAIN NAME',tr_name,'OF  
CLASS',Choice,'WITH FARE OF',fare)  
else:  
fare=1335
```



```
Choice='2A'

print("THE TRAIN NO",tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)

check_ticket(tr_name,fare,dest,Choice)

elif dest in ('SENGOTTAI','Sengottai','sengottai'):

tr_no=12661

tr_name='Pothigai superfast express'

print('CLASS: 1.SL 2. 2A')

Choice=int(input('enter your choice(1 or 2)'))

if Choice==1:

fare=385

Choice='SL'

print("THE TRAIN NO",tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)

else:

fare=1440

Choice='2A'

print("THE TRAIN NO",tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)

check_ticket(tr_name,fare,dest,Choice)

elif dest in ('THANJAVUR','Thanjavur','thanjavur'):

tr_no=22153

tr_name='Uzhavan express'

print('CLASS: 1.SL 2. 2A')

Choice=int(input('enter your choice(1 or 2)'))

if Choice==1:

fare=230

Choice='SL'

print("THE TRAIN NO",tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)

else:
```

```
fare=800
Choice='2A'
print('THE TRAIN NO',tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)
check_ticket(tr_name,fare,dest,Choice)
elif dest in ('TIRUCHENDUR','Tiruchendur','tiruchendur'):
tr_no=16105
tr_name='Chendur express'
print('CLASS: 1.SL 2. 2A')
Choice=int(input('enter your choice(1 or 2)'))
if Choice==1:
fare=395
Choice='SL'
print('THE TRAIN NO',tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)
else:
fare=1545
Choice='2A'
print('THE TRAIN NO',tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)
check_ticket(tr_name,fare,dest,Choice)
elif dest in ('KOLLAM','Kollam','kollam'):
tr_no=16723
tr_name='Anathapuri express'
print('CLASS: 1.SL 2. 2A')
Choice=int(input('enter your choice(1 or 2)'))
if Choice==1:
fare=440
Choice='SL'
print('THE TRAIN NO',tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)
```

```
else:
    fare=485
    Choice='2A'
    print('THE TRAIN NO',tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)
    check_ticket(tr_name,fare,dest,Choice)
elif dest in ('GURUVAYUR','Guruvayur','guruvayur'):
    tr_no=16127
    tr_name='Guruvayur express'
    print('CLASS: 1.SL 2. 2A')
    Choice=int(input('enter your choice(1 or 2)'))
    if Choice==1:
        fare=485
        Choice='SL'
        print('THE TRAIN NO',tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)
    else:
        fare=1545
        Choice='2A'
        print('THE TRAIN NO',tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)
        check_ticket(tr_name,fare,dest,Choice)
elif dest in ('KARAIKAL','Karikal','karikal'):
    tr_no=16175
    tr_name='Kamban express'
    print('CLASS: 1.SL 2. 2A')
    Choice=int(input('enter your choice(1 or 2)'))
    if Choice==1:
        fare=230
        Choice='SL'
```

```
print('THE TRAIN NO',tr_no,'AND TRAIN NAME',tr_name,'OF  
CLASS',Choice,'WITH FARE OF',fare)
```

```
else:
```

```
fare=880
```

```
Choice='2A'
```

```
print('THE TRAIN NO',tr_no,'AND TRAIN NAME',tr_name,'OF  
CLASS',Choice,'WITH FARE OF',fare)
```

```
check_ticket(tr_name,fare,dest,Choice)
```

```
elif dest in ('MANNARGUDI','Mannargudi','mannargudi'):
```

```
tr_no=16179
```

```
tr_name='Mannai express'
```

```
print('CLASS: 1.SL 2. 2A')
```

```
Choice=int(input('enter your choice(1 or 2)'))
```

```
if Choice==1:
```

```
fare=240
```

```
Choice='SL'
```

```
print('THE TRAIN NO',tr_no,'AND TRAIN NAME',tr_name,'OF  
CLASS',Choice,'WITH FARE OF',fare)
```

```
else:
```

```
fare=930
```

```
Choice='2A'
```

```
print('THE TRAIN NO',tr_no,'AND TRAIN NAME',tr_name,'OF  
CLASS',Choice,'WITH FARE OF',fare)
```

```
check_ticket(tr_name,fare,dest,Choice)
```

```
elif dest in ('KARAIKUDI','Karaikudi','karaikudi'):
```

```
tr_no=12605
```

```
tr_name='Pallavan superfast express'
```

```
print('CLASS: 1.2S 2. CC')
```

```
Choice=int(input('enter your choice(1 or 2)'))
```

```
if Choice==1:
```

```
fare=120
```

```
Choice='2S'

print("THE TRAIN NO",tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)

else:

    fare=300

    Choice='CC'

    print("THE TRAIN NO",tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)

    check_ticket(tr_name,fare,dest,Choice)

elif dest in ('SALEM','Salem','salem'):

    tr_no=22153

    tr_name='Salem superfast express'

    print('CLASS: 1.SL 2. 2A')

    Choice=int(input('enter your choice(1 or 2)'))

    if Choice==1:

        fare=260

        Choice='SL'

        print("THE TRAIN NO",tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)

    else:

        fare=930

        Choice='2A'

        print("THE TRAIN NO",tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)

        check_ticket(tr_name,fare,dest,Choice)

elif dest in ('MANGALURU CENTRAL','Mangaluru central','mangaluru
central'):

    tr_no=16159

    tr_name='Mangalore mail express'

    print('CLASS: 1.SL 2. 2A')

    Choice=int(input('enter your choice(1 or 2)'))
```

```
if Choice==1:
    fare=480
    Choice='SL'
    print('THE TRAIN NO',tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)
else:
    fare=1855
    Choice='2A'
    print('THE TRAIN NO',tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)
    check_ticket(tr_name,fare,dest,Choice)
elif dest in ('DADAR','Dadar','dadar'):
    tr_no=12164
    tr_name='Chennai express'
    print('CLASS: 1.SL 2. 2A')
    Choice=int(input('enter your choice(1 or 2)'))
    if Choice==1:
        fare=570
        Choice='SL'
        print('THE TRAIN NO',tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)
    else:
        fare=2160
        Choice='2A'
        print('THE TRAIN NO',tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)
        check_ticket(tr_name,fare,dest,Choice)
elif dest in ('GAYA','Gaya','gaya'):
    tr_no=12390
    tr_name='Gaya superfast express'
    print('CLASS: 1.SL 2. 2A')
```

```
Choice=int(input('enter your choice(1 or 2)'))
if Choice==1:
    fare=800
    Choice='SL'
    print('THE TRAIN NO',tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)
else:
    fare=3085
    Choice='2A'
    print('THE TRAIN NO',tr_no,'AND TRAIN NAME',tr_name,'OF
CLASS',Choice,'WITH FARE OF',fare)
    check_ticket(tr_name,fare,dest,Choice)
else:
    print('invalid station')
    res=input('want to book give y or else n')
    if res=='y':
        res=('Y')
    else:
        pass
def railway_ticket():
    print('WELCOME TO RAILWAY E TICKET BOOKING AND CATERING
SERVICES')
    print('COVID19 SPECIAL TRAINS ORIGINATING FROM CHENNAI
EGMORE')
    print('INSTRUCTIONS:')
    print('1.EACH CLASS (SL,2A) OF EACH TRAIN HAS 216 SEATS ONLY')
    print('2.IF SEAT EXCEEDS MORE THAN 216 SEATS WAITING LIST WILL
BE PROVIDED BASED ON CANCELATION OF OTHER TICKTES YOUR
TICKETS MAY CONFIRM OR NOT CONFIRM')
    print('3.IF TICKETS ARE NOT CONFIRMED YOUR AMOUNT WILL BE
REFUNDED WITHIN 2 WEEKS')
    print('4.PANTRY CARS ARE AVAILABLE')
```

```
print('5.AFTER BOOKING YOUR TICKETS KEEP YOUR TICKET COPY  
(ESPECIALLY TICKET_ID ) FOR FUTURE USE')
```

```
print()
```

```
print('OPTIONS')
```

```
print('1.BOOK TICKETS')
```

```
print('2.DELETE TICKETS')
```

```
print('3.VIEW TICKET')
```

```
print('4.DETAILS OF PASSENGER OF PARTICULAR TRAINS')
```

```
print('5.BUDGET OF THE ANY DAY')
```

```
print('6.OVERALL TICKET BUDGET')
```

```
print('7. PNR STATUS')
```

```
print('8.SEAT LAYOUT')
```

```
print('9.TODAYS BUDGET OF BOOKING')
```

```
print('10.REGISTRATION')
```

```
print('11.UPDATE PHONE NUMBER FOR BOOKED TICKET')
```

```
choice=int(input("enter your choice"))
```

```
if choice==1:
```

```
    ticket_book()
```

```
elif choice==2:
```

```
    delete_ticket()
```

```
elif choice==3:
```

```
    view_ticket()
```

```
elif choice==4:
```

```
    date_book()
```

```
elif choice==5:
```

```
    date_budget()
```

```
elif choice==6:
```

```
    year_budget()
```

```
elif choice==7:
```

```
    status()
```



```
elif choice==8:
    seat_arrangement()
elif choice==9:
    today_budget()
elif choice==10:
    registration()
elif choice==11:
    ph_one_no()
else:
    print('INVALID CHOICE')
    railway_ticket()
we=input('want to register give y or else n')
if we in 'y ':
    registration()
else:
    pini=int(input('enter your 4 code pin'))
    srl='select pin,name from regis where pin in ("%s") '
    mycursor.execute(srl,(pini, ))
    ag=mycursor.fetchall( )
    for li in ag:
        if pini==li[0]:
            print(' Welcome  to my dear passenger',li[1],sep='')
            railway_ticket()
        else:
            print('invalid pin')
            exit(0)
```

```

Python 3.8.3 Shell
File Edit Shell Debug Options Window Help
Python 3.8.3 (tags/v3.8.3:6f8c832, May 13 2020, 22:37:02) [MSC v.1924 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\Kumar\Desktop\Railwayticketreservation.py =====
want to register give y or else ny
enter user nameHarish
enter user mail idharish123@gmail.com
enter passwordHarish123
enter your pin in numbers for security purpose1234
REGISTRATION DONE SUCESSFULLY
WELCOME TO RAILWAY E TICKET BOOKING AND CATERING SERVICES
COVID19 SPECIAL TRAINS ORIGINATING FROM CHENNAI EGMORE
INSTRUCTIONS:
1.EACH CLASS (SL,2A) OF EACH TRAIN HAS 216 SEATS ONLY
2.IF SEAT EXCEEDS MORE THAN 216 SEATS WAITING LIST WILL BE PROVIDED BASED ON CANCELATION OF OTHER TICKTES YOUR TICKETS MAY CONFIRM OR NOT CONFIRM
3.IF TICKETS ARE NOT CONFIRMED YOUR AMOUNT WILL BE REFUNDED WITHIN 2 WEEKS
4.FOOD ORDER SERVICES ARE AVAILABLE
5.AFTER BOOKING YOUR TICKETS KEEP YOUR TICKET COPY (ESPECIALLY TICKET_ID ) FOR FUTURE USE

OPTIONS
1.BOOK TICKETS
2.DELETE TICKETS
3.VIEW TICKET
4.DETAILS OF PASSENGER OF PARTICULAR TRAINS
5.BUDGET OF THE ANY DAY
6.OVERALL TICKET BUDGET
7. PNR STATUS
8.SEAT LAYOUT
9.TODAYS BUDGET OF BOOKING
10.REGISTRATION
11.UPDATE PHONE NUMBER FOR BOOKED TICKET
enter your choice

```

```

5 rows in set (0.00 sec)

mysql> select * from regis;
+-----+-----+-----+-----+
| name      | mail_id                | passwd      | pin      |
+-----+-----+-----+-----+
| 'Harish'   | 'harish123@gmail.com'  | 'Harish123' | 1234     |
| 'Frank'    | 'frank987@gmail.com'   | 'Frank987'  | 9876     |
+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql>

```

```

Python 3.8.3 Shell
File Edit Shell Debug Options Window Help
Python 3.8.3 (tags/v3.8.3:6f8c832, May 13 2020, 22:37:02) [MSC v.1924 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\Kumar\Desktop\Railwayticketreservation.py =====
want to register give y or else nn
enter your 4 code pin1234
Welcome to my dear passenger'Harish'
WELCOME TO RAILWAY E TICKET BOOKING AND CATERING SERVICES
COVID19 SPECIAL TRAINS ORIGINATING FROM CHENNAI EGMORE
INSTRUCTIONS:
1.EACH CLASS (SL,2A) OF EACH TRAIN HAS 216 SEATS ONLY
2.IF SEAT EXCEEDS MORE THAN 216 SEATS WAITING LIST WILL BE PROVIDED BASED ON CANCELATION OF OTHER TICKTES YOUR TICKETS MAY CONFIRM OR NOT CONFIRM
3.IF TICKETS ARE NOT CONFIRMED YOUR AMOUNT WILL BE REFUNDED WITHIN 2 WEEKS
4.FOOD ORDER SERVICES ARE AVAILABLE
5.AFTER BOOKING YOUR TICKETS KEEP YOUR TICKET COPY (ESPECIALLY TICKET_ID ) FOR FUTURE USE

OPTIONS
1.BOOK TICKETS
2.DELETE TICKETS
3.VIEW TICKET
4.DETAILS OF PASSENGER OF PARTICULAR TRAINS
5.BUDGET OF THE ANY DAY
6.OVERALL TICKET BUDGET
7. PNR STATUS
8.SEAT LAYOUT
9.TODAYS BUDGET OF BOOKING
10.REGISTRATION
11.UPDATE PHONE NUMBER FOR BOOKED TICKET
enter your choice

```

```

Python 3.8.3 Shell
File Edit Shell Debug Options Window Help
CLASS: 1.SL 2. 2A
enter your choice(1 or 2)1
THE TRAIN NO 12390 AND TRAIN NAME Gaya superfast express OF CLASS SL WITH FARE OF 800
enter your phone no0123456789
enter the date which you want to check no of tickets present in your date of journey (yyyy-mm-dd)2020-08-15
*****
AVAILABLE 212
*****
WELCOME TO E-BOOKING SYSTEM
1.WANT TO GO BACK
2.TICKET BOOKING RESERVATION
enter your choice2
enter no of tickets required1
enter the passenger nameLawson
enter your age40
enter male or femalemale
WANT TO OPT TRAVEL INSURANCE GIVE y OR ELSE n n
Record inserted sucesfully
***** TICKET *****
NAME: Lawson
AGE: 40
GENDER: MALE
TRAIN NAME: GAYA SUPERFAST EXPRESS
FARE: 800
BOOKED DATE: 2020-07-25
DATE OF JOURNEY: 2020-08-15
FROM CHENNAI EGMORE TO GAYA
CLASS: SL
SEAT NO: SL005
TICKET ID: 40960603
PHONE NO: 0123456789
TRAVEL INSURANCE OPTED: NO
*****
KEEP THIS COPY SAFE FOR FUTURE USE

```

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

```
mysql> use railwayticketproject;
```

```
Database changed
```

```
mysql> select * from ticket;
```

name	age	gender	tr_name	destination	fare	date	class	seat_no	booking_date	ticket_id	phone_no	travel_ins
'Harish'	17	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	810	'2020-08-15'	'SL'	'SL001'	'2020-07-24 '	'19225717'	'9876543210'	'YES'
'Frank'	40	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	800	'2020-08-15'	'SL'	'SL002'	'2020-07-25 '	'44833185'	'1234567890'	'NO'
'Jack'	50	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	800	'2020-08-15'	'SL'	'SL003'	'2020-07-25 '	'73153398'	'1234567890'	'NO'
'Joe'	30	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	800	'2020-08-15'	'SL'	'SL004'	'2020-07-25 '	'80312558'	'1234567890'	'NO'

4 rows in set (0.00 sec)

```
mysql> select * from ticket;
```

name	age	gender	tr_name	destination	fare	date	class	seat_no	booking_date	ticket_id	phone_no	travel_ins
'Harish'	17	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	810	'2020-08-15'	'SL'	'SL001'	'2020-07-24 '	'19225717'	'9876543210'	'YES'
'Frank'	40	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	800	'2020-08-15'	'SL'	'SL002'	'2020-07-25 '	'44833185'	'1234567890'	'NO'
'Jack'	50	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	800	'2020-08-15'	'SL'	'SL003'	'2020-07-25 '	'73153398'	'1234567890'	'NO'
'Joe'	30	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	800	'2020-08-15'	'SL'	'SL004'	'2020-07-25 '	'80312558'	'1234567890'	'NO'
'Lawson'	40	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	800	'2020-08-15'	'SL'	'SL005'	'2020-07-25 '	'40960603'	'0123456789'	'NO'

5 rows in set (0.07 sec)

WELCOME TO RAILWAY E TICKET BOOKING AND CATERING SERVICES

COVID19 SPECIAL TRAINS ORIGINATING FROM CHENNAI EGMORE

INSTRUCTIONS:

1.EACH CLASS (SL,2A) OF EACH TRAIN HAS 216 SEATS ONLY

2.IF SEAT EXCEEDS MORE THAN 216 SEATS WAITING LIST WILL BE PROVIDED BASED ON CANCELATION OF OTHER TICKETS YOUR TICKETS MAY CONFIRM OR NOT CONFIRM

3.IF TICKETS ARE NOT CONFIRMED YOUR AMOUNT WILL BE REFUNDED WITHIN 2 WEEKS

4.FOOD ORDER SERVICES ARE AVAILABLE

5.AFTER BOOKING YOUR TICKETS KEEP YOUR TICKET COPY (ESPECIALLY TICKET_ID) FOR FUTURE USE

OPTIONS

1.BOOK TICKETS

2.DELETE TICKETS

3.VIEW TICKET

4.DETAILS OF PASSENGER OF PARTICULAR TRAINS

5.BUDGET OF THE ANY DAY

6.OVERALL TICKET BUDGET

7. PNR STATUS

8.SEAT LAYOUT

9.TODAYS BUDGET OF BOOKING

10.REGISTRATION

11.UPDATE PHONE NUMBER FOR BOOKED TICKET

enter your choice2

enter no of tickets to be deleted1

enter passenger name for deleting ticketLawson

enter your 8 digit code ticket id40960603

record deleted successfully

YOUR AMOUNT WILL BE REFUNDED SOON

```
mysql> select * from ticket;
```

name	age	gender	tr_name	destination	fare	date	class	seat_no	booking_date	ticket_id	phone_no	travel_ins
'Harish'	17	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	810	'2020-08-15'	'SL'	'SL001'	'2020-07-24'	'19225717'	'9876543210'	'YES'
'Frank'	40	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	800	'2020-08-15'	'SL'	'SL002'	'2020-07-25'	'44833185'	'1234567890'	'NO'
'Jack'	50	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	800	'2020-08-15'	'SL'	'SL003'	'2020-07-25'	'73153398'	'1234567890'	'NO'
'Joe'	30	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	800	'2020-08-15'	'SL'	'SL004'	'2020-07-25'	'80312558'	'1234567890'	'NO'
'Lawson'	40	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	800	'2020-08-15'	'SL'	'SL005'	'2020-07-25'	'40960603'	'0123456789'	'NO'

5 rows in set (0.07 sec)

```
mysql> select * from ticket;
```

name	age	gender	tr_name	destination	fare	date	class	seat_no	booking_date	ticket_id	phone_no	travel_ins
'Harish'	17	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	810	'2020-08-15'	'SL'	'SL001'	'2020-07-24'	'19225717'	'9876543210'	'YES'
'Frank'	40	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	800	'2020-08-15'	'SL'	'SL002'	'2020-07-25'	'44833185'	'1234567890'	'NO'
'Jack'	50	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	800	'2020-08-15'	'SL'	'SL003'	'2020-07-25'	'73153398'	'1234567890'	'NO'
'Joe'	30	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	800	'2020-08-15'	'SL'	'SL004'	'2020-07-25'	'80312558'	'1234567890'	'NO'

4 rows in set (0.00 sec)

```
mysql>
```

Python 3.8.3 Shell

File Edit Shell Debug Options Window Help

OPTIONS

1.BOOK TICKETS

2.DELETE TICKETS

3.VIEW TICKET

4.DETAILS OF PASSENGER OF PARTICULAR TRAINS

5.BUDGET OF THE ANY DAY

6.OVERALL TICKET BUDGET

7. PNR STATUS

8.SEAT LAYOUT

9.TODAYS BUDGET OF BOOKING

10.REGISTRATION

11.UPDATE PHONE NUMBER FOR BOOKED TICKET

enter your choice3

enter the no of tickets to be viewed1

enter passenger name which you want to view your ticketHarish

enter your 8 code ticket id19225717

NAME: 'Harish'

AGE: 17

GENDER: 'MALE'

TRAIN NAME: 'GAYA SUPERFAST EXPRESS'

DESTINATION: 'GAYA'

FARE: 810

DATE: '2020-08-15'

CLASS: 'SL'

SEAT NO: 'SL001'

BOOKING DATE: '2020-07-24'

TICKET ID: '19225717'

PHONE NUMBER: '9876543210'

TRAVEL INSURANCE OPTED: 'YES'

enter the file name with .txt extension to save your copy permanentlyhar.txt

YOUR COPY SAVED SUCCESSFULLY



```
har - Notepad
File Edit Format View Help
*****
E-TICKET
NAME:'Harish'
AGE:17
GENDER:'MALE'
DESTINATION:'GAYA'
FARE:810
DATE:'2020-08-15'
CLASS:'SL'
SEAT NO:'SL001'
BOOKING DATE:'2020-07-24'
TICKET ID:'19225717'
PHONE NUMBER:'9876543210'
TRAVEL INSURANCE OPTED:'YES'
THANKS FOR BOOKING
*****
```

```
Python 3.8.3 Shell
File Edit Shell Debug Options Window Help
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\Kumar\Desktop\Railwayticketreservation.py =====
want to register give y or else nn
enter your 4 code pin1234
>Welcome to my dear passenger'Harish'
WELCOME TO RAILWAY E TICKET BOOKING AND CATERING SERVICES
COVID19 SPECIAL TRAINS ORIGINATING FROM CHENNAI EGMORE
INSTRUCTIONS:
1.EACH CLASS (SL,2A) OF EACH TRAIN HAS 216 SEATS ONLY
2.IF SEAT EXCEEDS MORE THAN 216 SEATS WAITING LIST WILL BE PROVIDED BASED ON CANCELATION OF OTHER TICKETS YOUR TICKETS MAY CONFIRM OR NOT CONFIRM
3.IF TICKETS ARE NOT CONFIRMED YOUR AMOUNT WILL BE REFUNDED WITHIN 2 WEEKS
4.FOOD ORDER SERVICES ARE AVAILABLE
5.AFTER BOOKING YOUR TICKETS KEEP YOUR TICKET COPY (ESPECIALLY TICKET_ID) FOR FUTURE USE

OPTIONS
1.BOOK TICKETS
2.DELETE TICKETS
3.VIEW TICKET
4.DETAILS OF PASSENGER OF PARTICULAR TRAINS
5.BUDGET OF THE ANY DAY
6.OVERALL TICKET BUDGET
7.PNR STATUS
8.SEAT LAYOUT
9.TODAYS BUDGET OF BOOKING
10.REGISTRATION
11.UPDATE PHONE NUMBER FOR BOOKED TICKET
enter your choice4
enter the date which you want to check(yyyy-mm-dd)2020-08-15
enter the train namegaya superfast express
('Harish', 17, 'MALE', 'GAYA SUPERFAST EXPRESS', 'GAYA', 810, '2020-08-15', 'SL', 'SL001', '2020-07-24', '19225717', '9876543210', 'YES')
('Frank', 40, 'MALE', 'GAYA SUPERFAST EXPRESS', 'GAYA', 800, '2020-08-15', 'SL', 'SL002', '2020-07-25', '44833185', '1234567890', 'NO')
('Jack', 50, 'MALE', 'GAYA SUPERFAST EXPRESS', 'GAYA', 800, '2020-08-15', 'SL', 'SL003', '2020-07-25', '73153398', '1234567890', 'NO')
('Joe', 30, 'MALE', 'GAYA SUPERFAST EXPRESS', 'GAYA', 800, '2020-08-15', 'SL', 'SL004', '2020-07-25', '80312558', '1234567890', 'NO')
```

```
WELCOME TO RAILWAY E TICKET BOOKING AND CATERING SERVICES
COVID19 SPECIAL TRAINS ORIGINATING FROM CHENNAI EGMORE
INSTRUCTIONS:
1.EACH CLASS (SL,2A) OF EACH TRAIN HAS 216 SEATS ONLY
2.IF SEAT EXCEEDS MORE THAN 216 SEATS WAITING LIST WILL BE PROVIDED BASED ON CANCELATION OF OTHER TICKETS YOUR TICKETS MAY CONFIRM OR NOT CONFIRM
3.IF TICKETS ARE NOT CONFIRMED YOUR AMOUNT WILL BE REFUNDED WITHIN 2 WEEKS
4.FOOD ORDER SERVICES ARE AVAILABLE
5.AFTER BOOKING YOUR TICKETS KEEP YOUR TICKET COPY (ESPECIALLY TICKET_ID ) FOR FUTURE USE
```

OPTIONS

```
1.BOOK TICKETS
2.DELETE TICKETS
3.VIEW TICKET
4.DETAILS OF PASSENGER OF PARTICULAR TRAINS
5.BUDGET OF THE ANY DAY
6.OVERALL TICKET BUDGET
7. PNR STATUS
8.SEAT LAYOUT
9.TODAYS BUDGET OF BOOKING
10.REGISTRATION
11.UPDATE PHONE NUMBER FOR BOOKED TICKET
enter your choice5
```

```
enter the date which you want to check the budget (yyyy-mm-dd)2020-08-15
2020-08-15 budaet 3210
```

OPTIONS

```
1.BOOK TICKETS
2.DELETE TICKETS
3.VIEW TICKET
4.DETAILS OF PASSENGER OF PARTICULAR TRAINS
5.BUDGET OF THE ANY DAY
6.OVERALL TICKET BUDGET
7. PNR STATUS
8.SEAT LAYOUT
9.TODAYS BUDGET OF BOOKING
10.REGISTRATION
11.UPDATE PHONE NUMBER FOR BOOKED TICKET
enter your choice6
TOTAL BUDGET OF ENTIRE BOOKING 3210
```

```
Python 3.8.3 Shell
File Edit Shell Debug Options Window Help
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\Kumar\Desktop\Railwayticketreservation.py =====
want to register give y or else nn
enter your 4 code pin1234
'Welcome to my dear passenger'Harish'
WELCOME TO RAILWAY E TICKET BOOKING AND CATERING SERVICES
COVID19 SPECIAL TRAINS ORIGINATING FROM CHENNAI EGMORE
INSTRUCTIONS:
1.EACH CLASS (SL,2A) OF EACH TRAIN HAS 216 SEATS ONLY
2.IF SEAT EXCEEDS MORE THAN 216 SEATS WAITING LIST WILL BE PROVIDED BASED ON CANCELATION OF OTHER TICKETS YOUR TICKETS MAY CONFIRM OR NOT CONFIRM
3.IF TICKETS ARE NOT CONFIRMED YOUR AMOUNT WILL BE REFUNDED WITHIN 2 WEEKS
4.FOOD ORDER SERVICES ARE AVAILABLE
5.AFTER BOOKING YOUR TICKETS KEEP YOUR TICKET COPY (ESPECIALLY TICKET_ID ) FOR FUTURE USE

OPTIONS
1.BOOK TICKETS
2.DELETE TICKETS
3.VIEW TICKET
4.DETAILS OF PASSENGER OF PARTICULAR TRAINS
5.BUDGET OF THE ANY DAY
6.OVERALL TICKET BUDGET
7. PNR STATUS
8.SEAT LAYOUT
9.TODAYS BUDGET OF BOOKING
10.REGISTRATION
11.UPDATE PHONE NUMBER FOR BOOKED TICKET
enter your choice7
enter the name to view statusHarish
enter the train namegaya superfast express
enter the ticket id19225717
THIS TICKET WAS ALREADY CONFIRMED 'SL001'
```



```

1.BOOK TICKETS
2.DELETE TICKETS
3.VIEW TICKET
4.DETAILS OF PASSENGER OF PARTICULAR TRAINS
5.BUDGET OF THE ANY DAY
6.OVERALL TICKET BUDGET
7. PNR STATUS
8.SEAT LAYOUT
9.TODAYS BUDGET OF BOOKING
10.REGISTRATION
11.UPDATE PHONE NUMBER FOR BOOKED TICKET
enter your choice9
TODAY TOTAL BUDGET OF BOOKING 2400

```

WELCOME TO RAILWAY E TICKET BOOKING AND CATERING SERVICES

COVID19 SPECIAL TRAINS ORIGINATING FROM CHENNAI EGMORE

INSTRUCTIONS:

```

1.EACH CLASS (SL,2A) OF EACH TRAIN HAS 216 SEATS ONLY
2.IF SEAT EXCEEDS MORE THAN 216 SEATS WAITING LIST WILL BE PROVIDED BASED ON CANCELATION OF OTHER TICKETS YOUR TICKETS MAY CONFIRM OR NOT CONFIRM
3.IF TICKETS ARE NOT CONFIRMED YOUR AMOUNT WILL BE REFUNDED WITHIN 2 WEEKS
4.FOOD ORDER SERVICES ARE AVAILABLE
5.AFTER BOOKING YOUR TICKETS KEEP YOUR TICKET COPY (ESPECIALLY TICKET_ID ) FOR FUTURE USE

```

OPTIONS

```

1.BOOK TICKETS
2.DELETE TICKETS
3.VIEW TICKET
4.DETAILS OF PASSENGER OF PARTICULAR TRAINS
5.BUDGET OF THE ANY DAY
6.OVERALL TICKET BUDGET
7. PNR STATUS
8.SEAT LAYOUT
9.TODAYS BUDGET OF BOOKING
10.REGISTRATION
11.UPDATE PHONE NUMBER FOR BOOKED TICKET
enter your choice10
enter user nameFrank
enter user mail idFrank456@gmail.com
enter passwordFrank4567
enter your pin in numbers for security purpose4567
REGISTRATION DONE SUCESSFULLY

```

```

mysql> select * from regis;
+-----+-----+-----+-----+
| name   | mail_id                | passwd      | pin   |
+-----+-----+-----+-----+
| 'Harish' | 'harish123@gmail.com' | 'Harish123' | 1234 |
| 'Frank'  | 'frank987@gmail.com'  | 'Frank987'  | 9876 |
| 'Frank'  | 'Frank456@gmail.com'  | 'Frank4567' | 4567 |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)

```

WELCOME TO RAILWAY E TICKET BOOKING AND CATERING SERVICES

COVID19 SPECIAL TRAINS ORIGINATING FROM CHENNAI EGMORE

INSTRUCTIONS:

- 1.EACH CLASS (SL,2A) OF EACH TRAIN HAS 216 SEATS ONLY
- 2.IF SEAT EXCEEDS MORE THAN 216 SEATS WAITING LIST WILL BE PROVIDED BASED ON CANCELATION OF OTHER TICKETS YOUR TICKETS MAY CONFIRM OR NOT CONFIRM
- 3.IF TICKETS ARE NOT CONFIRMED YOUR AMOUNT WILL BE REFUNDED WITHIN 2 WEEKS
- 4.FOOD ORDER SERVICES ARE AVAILABLE
- 5.AFTER BOOKING YOUR TICKETS KEEP YOUR TICKET COPY (ESPECIALLY TICKET_ID) FOR FUTURE USE

OPTIONS

- 1.BOOK TICKETS
- 2.DELETE TICKETS
- 3.VIEW TICKET
- 4.DETAILS OF PASSENGER OF PARTICULAR TRAINS
- 5.BUDGET OF THE ANY DAY
- 6.OVERALL TICKET BUDGET
7. PNR STATUS
- 8.SEAT LAYOUT
- 9.TODAYS BUDGET OF BOOKING
- 10.REGISTRATION
- 11.UPDATE PHONE NUMBER FOR BOOKED TICKET

enter your choice11

enter the ticket id for which the no to be changed80312558

enter the phone no to updated987654012

TICKET PHONE NUMBER UPDATED SUCESSFULLY

```
mysql> select * from ticket;
```

name	age	gender	tr_name	destination	fare	date	class	seat_no	booking_date	ticket_id	phone_no	travel_ins
'Harish'	17	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	810	'2020-08-15'	'SL'	'SL001'	'2020-07-24 '	'19225717'	'9876543210'	'YES'
'Frank'	40	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	800	'2020-08-15'	'SL'	'SL002'	'2020-07-25 '	'44833185'	'1234567890'	'NO'
'Jack'	50	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	800	'2020-08-15'	'SL'	'SL003'	'2020-07-25 '	'73153398'	'1234567890'	'NO'
'Joe'	30	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	800	'2020-08-15'	'SL'	'SL004'	'2020-07-25 '	'80312558'	'1234567890'	'NO'

4 rows in set (0.00 sec)

```
mysql> select * from ticket;
```

name	age	gender	tr_name	destination	fare	date	class	seat_no	booking_date	ticket_id	phone_no	travel_ins
'Harish'	17	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	810	'2020-08-15'	'SL'	'SL001'	'2020-07-24 '	'19225717'	'9876543210'	'YES'
'Frank'	40	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	800	'2020-08-15'	'SL'	'SL002'	'2020-07-25 '	'44833185'	'1234567890'	'NO'
'Jack'	50	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	800	'2020-08-15'	'SL'	'SL003'	'2020-07-25 '	'73153398'	'1234567890'	'NO'
'Joe'	30	'MALE'	'GAYA SUPERFAST EXPRESS'	'GAYA'	800	'2020-08-15'	'SL'	'SL004'	'2020-07-25 '	'80312558'	'987654012'	'NO'

4 rows in set (0.00 sec)


```
*Python 3.8.3 Shell*
File Edit Shell Debug Options Window Help
OPTIONS
1.BOOK TICKETS
2.DELETE TICKETS
3.VIEW TICKET
4.DETAILS OF PASSENGER OF PARTICULAR TRAINS
5.BUDGET OF THE ANY DAY
6.OVERALL TICKET BUDGET
7. PNR STATUS
8.SEAT LAYOUT
9.TODAYS BUDGET OF BOOKING
10.REGISTRATION
11.UPDATE PHONE NUMBER FOR BOOKED TICKET
enter your choice8
enter SL OR 2A OR 2S OR CC COACHSL
each coach has 72 seats and total 5 SL coaches
1 2 3 7
4 5 6 8
*****
9 10 11 15
12 13 14 16
*****
17 18 19 23
20 21 22 24
*****
25 26 27 31
28 29 30 32
*****
33 34 35 39
36 37 38 40
*****
41 42 43 47
44 45 46 48
*****
49 50 51 55
```

```
*Python 3.8.3 Shell*
File Edit Shell Debug Options Window Help
*****
57 58 59 63
60 61 62 64
*****
65 66 67 71
68 69 70 72
*****
73 74 75 79
76 77 78 80
*****
81 82 83 87
84 85 86 88
*****
89 90 91 95
92 93 94 96
*****
97 98 99 103
100 101 102 104
*****
105 106 107 111
108 109 110 112
*****
113 114 115 119
116 117 118 120
*****
121 122 123 127
124 125 126 128
*****
129 130 131 135
132 133 134 136
*****
137 138 139 143
140 141 142 144
*****
```

```
*Python 3.8.3 Shell*
File Edit Shell Debug Options Window Help
137 138 139 143
140 141 142 144
*****
145 146 147 151
148 149 150 152
*****
153 154 155 159
156 157 158 160
*****
161 162 163 167
164 165 166 168
*****
169 170 171 175
172 173 174 176
*****
177 178 179 183
180 181 182 184
*****
185 186 187 191
188 189 190 192
*****
193 194 195 199
196 197 198 200
*****
201 202 203 207
204 205 206 208
*****
209 210 211 215
212 213 214 216
*****
BERTH ARRANGEMENT FOR SL AND 2A
LOWER MIDDLE UPPER SIDE UPPER
LOWER MIDDLE UPPER SIDE LOWER
*****
```

CONCLUSION:

Railway ticket booking system is an environment where all the process of the people who book tickets are fetched and saved . It is done through the automated computerized method. Conventionally this system is done using papers, file and binders.

This system saves the time of the administrator. It includes process like calculating total,individual budget of train tickets

This makes the system easy to handle and feasible for finding the omission with updating at the same time.

If filed a query to search or update in a manual system, it will take a lot of time to process the query and make a report which is a tedious job.

This computerized system store all the data in the database which makes it easy to fetch and update whenever needed.

BIBILOGRAPHY:

Computer Science with Python by Sumita Arora for Class 12

Computer Science with Python by Preeti Arora for Class 12

<https://code-projects.org/>

<https://www.w3schools.com/>