DELHI PUBLIC SCHOOL GHAZIABAD MEERUT ROAD

Computer Science Project File

Submitted By:

Nandita Krishnan

(XII- A5)

Submitted To:

Ruchi Sharma

CERTIFICATE

This is to certify that the 'Airline Ticket Reservation Management' Computer Science project was developed by Nandita Krishnan under my supervision in the computer lab in the session 2023-2024. The work done by him/her is original.

Ruchi Sharma

PGT Computer Science

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to my computer science teacher **Ms. Ruchi Sharma** for her vital support, guidance, and encouragement without which this project would not have forth from my side. She helped me complete the project by giving me ideas, and thoughts and made this project easy and accurate.

I wish to thank my parents for their undivided support and interest who inspired me and encouraged me to go my own way, without which I would be unable to complete my project.

INDEX

AIRLINE TICKET RESERVATION

BRIEF OVERVIEW OF PROJECT

The main objective of the Python project on Air ticket reservation is to manage the details of booking, payments, seats, and flights.

The project is built at the administrative end and only the administrator is guaranteed access.

The purpose of the project is to build an application program to reduce the manual work for managing bookings, discounts, seats, and payments.

It tracks all the details about seats, flights, and payments; it also prints various reports as per input given by the user.

INPUT DATA AND VALIDATION OF PROJECT

All the fields such as flight payment discounts are validated and do not take invalid values.

Each form of sales, discounts, and bookings cannot accept blank values.

Avoiding errors in data.

Controlling the amount of input.

SOFTWARE AND HARDWARE REQUIREMENTS

Data file handling has been effectively used in the program. The database is a collection of interrelated data to serve multiple applications. That is database programs create files of information. So, we see that files are worked with most, inside the program.

DBMS: The software required for the management of data is called as DBMS. It has3 models:

- Relation model
- Hierarchical model
- Network model

RELATIONAL MODEL: It's based on the concept on relation. Relation is the table that consists of rows and columns. The rows of the table are called tuple, and the columns of the table are called attribute. Numbers of rows in the table is called as cardinality. Number of columns in the table is called as degree.

HIERARCHICAL MODEL: In this type of model, we have multiple records for each record. A particular record has one parent record. No chide record can exist without parent record. In this, the records are organized in tree.

NETWORK MODEL: In this, the data is represented by

collection of records and relationship is represented by (ink or association.

CHARACTERISTICS OF DBMS:

- It reduces the redundancy
- Reduction of data in inconsistency
- Data sharing
- Data standardization

DIFFERENT TYPES OF FILES: -BASED ON ACCESS:

- Sequential file
- Serial file
- Random (direct access) file BASED ON STORAGE: -
- Text file
- Binary File

NEED OF COMPUTERISATION

Over the decades computers and air ticket bookings have developed gradually, changed with time. But nobody knew that a time would come when both these fields will complement each other so well. Today air ticket booking has reached new heights by computer aided methods of design. As a result of which, computer industry has got its new customer. Computer technology is making waves in the flight booking zone. Computers are a vital component of the ticket booking counters. Computer aided design (CAD) programs reduce the demand for manual sketches. New software programs continue to replace old manual skills. Those who lag in math can now breathe a little easier. Manually figuring of tickets insists that knowledge. Software programs constantly evolve. A program used today may be obsolete within several years. Being trained on today's software does not guarantee it will be used when you are ready to go out into the field. Understanding calculations is timeless, as is computer competency. Software, however, shifts rapidly.

ADVANTAGES

- It generates the report on sales, discounts and flights.
- Provides filter report on payments and flight booking.
- We can easily export PDF on sales, products and stocks.
- Applications can also provide excel export for bookings and discounts.
- It deals with monitoring the information and transaction of ticket bookings.
- It increases the efficiency of flight booking and offers discounts.
- It has higher efficiency for editing, adding and updating records.
- Provides the searching facilities on various factors.

LIMITS

- Excel export has not been developed for bookings.
- The transactions are executed in offline mode only.
- Online transactions for sales, bookings, or other data modifications are not possible.
- Offline reports of sales, bookings, and discounts cannot be generated due to batch mode execution.

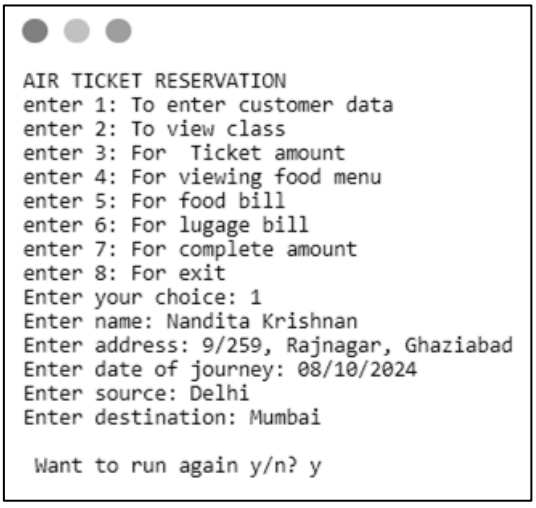
SOURCE CODE SCREENING

- DBMS: MySQL
- Host: local host
- User: root
- Pass: root
- Database: hotel
- Table Structure: (Images Below)

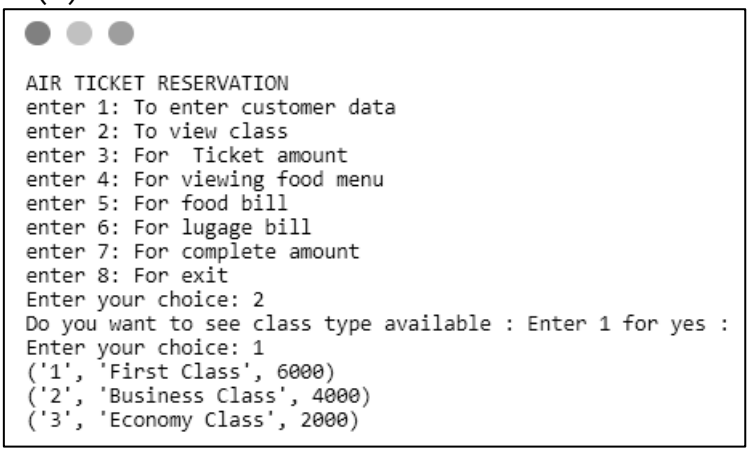
```
 MySQL  localhost:33060+ ssl  airindigo  SQL > DESC classtype;
+-----------+-------------+------+-----+---------+-------+
| Field     | Type        | Null | Key | Default | Extra |
+-----------+-------------+------+-----+---------+-------+
| sno       | varchar(10) | YES  |     | NULL    |       |
| classtype | varchar(30) | YES  |     | NULL    |       |
| price     | int         | YES  |     | NULL    |       |
+-----------+-------------+------+-----+---------+-------+
3 rows in set (0.0017 sec)
 MySQL  localhost:33060+ ssl  airindigo  SQL > DESC food;
+----------+-------------+------+-----+---------+-------+
| Field    | Type        | Null | Key | Default | Extra |
+----------+-------------+------+-----+---------+-------+
| sno      | int         | YES  |     | NULL    |       |
| itemname | varchar(10) | YES  |     | NULL    |       |
| rate     | int         | YES  |     | NULL    |       |
+----------+-------------+------+-----+---------+-------+
3 rows in set (0.0034 sec)
 MySQL  localhost:33060+ ssl  airindigo  SQL > DESC luggage;
+--------+-------------+------+-----+---------+-------+
| Field  | Type        | Null | Key | Default | Extra |
+--------+-------------+------+-----+---------+-------+
| sno    | int         | YES  |     | NULL    |       |
| weight | varchar(10) | YES  |     | NULL    |       |
| rate   | int         | YES  |     | NULL    |       |
+--------+-------------+------+-----+---------+-------+
3 rows in set (0.0017 sec)
 MySQL  localhost:33060+ ssl  airindigo  SQL > DESC pdata;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| custname    | varchar(30) | YES  |     | NULL    |       |
| addr        | varchar(30) | YES  |     | NULL    |       |
| jrdate      | varchar(10) | YES  |     | NULL    |       |
| source      | varchar(10) | YES  |     | NULL    |       |
| destination | varchar(10) | YES  |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
5 rows in set (0.0016 sec)
```

SOURCE CODE

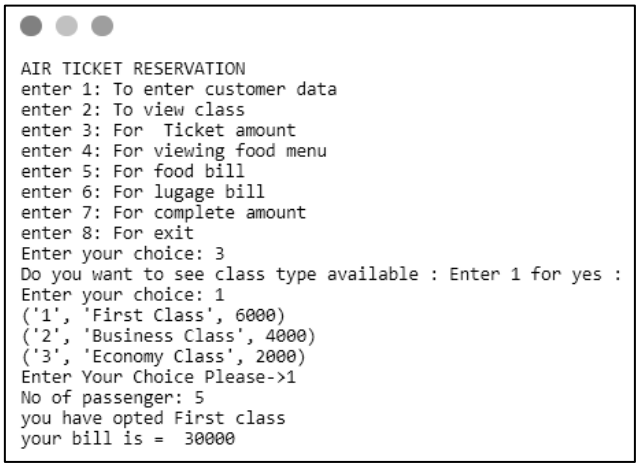
```
import os
import platform
import mysql.connector
import pandas as pd
import datetime

mydb = mysql.connector.connect(user='root', password='nebula_3F0',
host='localhost', database='airindigo')
mycursor = mydb.cursor()

def registercust():
    L = []
    name = input("Enter name: ")
    L.append(name)
    addr = input("Enter address: ")
    L.append(addr)
    jr_date = input("Enter date of journey: ")
    L.append(jr_date)
    source = input("Enter source: ")
    L.append(source)
    destination = input("Enter destination: ")
    L.append(destination)
    cust = tuple(L)
    sql = "insert into pdata (custname, addr, jrdate, source, destination)
values (%s, %s, %s, %s, %s)"
    mycursor.execute(sql, cust)
    mydb.commit()
```



```
def classtypeview():
    print ("Do you want to see class type available: Enter 1 for yes:")
    ch = int (input ("Enter your choice: "))
    if ch == 1:
        sql = "select * from classtype"
        mycursor.execute(sql)
        rows = mycursor.fetchall ()
        global x
        for x in rows:
            print(x)
```




```
def ticketprice():
    classtypeview()
    x=int (input("Enter Your Choice Please->"))
    n = int (input("No of passenger: "))
    if x == 1:
        print("you have opted for First class")
        s = 6000*n
    elif x == 2:
        print("you have opted for Business class")
        s = 4000*n
    elif x == 3:
        print("you have opted for Economy class")
        s = 2000*n
    else:
        print("Please choose a class type")
    print("your room rent is = ", s, "\n")
```



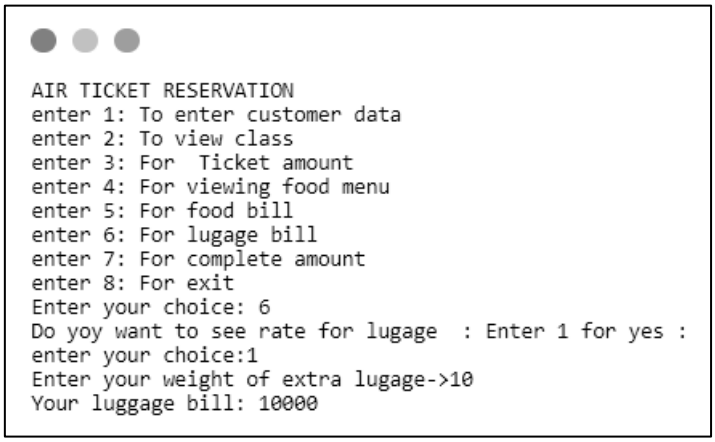
```
def menuview ():
    print("Do you want to see menu available: Enter 1 for yes:")
    ch = int(input ("Enter your choice: "))
    if ch == 1:
        sql = "select * from food"
        mycursor.execute(sql)
        rows = mycursor.fetchall ()
        for x in rows:
            print(x)
```

```
AIR TICKET RESERVATION
enter 1: To enter customer data
enter 2: To view class
enter 3: For  Ticket amount
enter 4: For viewing food menu
enter 5: For food bill
enter 6: For lugage bill
enter 7: For complete amount
enter 8: For exit
Enter your choice: 4
Do you want to see menu available : Enter 1 for yes :
Enter your choice: 1
(1, 'Tea', 10)
(2, 'Coffee', 10)
(3, 'Colddrink', 20)
(4, 'Samosa', 10)
(5, 'Sandwich', 50)
(6, 'Dhokla', 30)
(7, 'Kachori', 10)
(8, 'Milk', 20)
(9, 'Noodles', 50)
(10, 'Pasta', 50)

 Want to run again y/n? y
```

```
def orderitem ():
    global s
    print("Do you want to see menu available: Enter 1 for yes:")
    ch = int(input ("enter your choice:"))
    if ch == 1:
        sql = "select * from food"
        mycursor.execute(sql)
        rows = mycursor.fetchall ()
        for x in rows:
            print(x)

        d = int(input ("Enter your choice:"))
        if d == 1:
            print("You have ordered tea")
            a = int(input ("Enter quantity: "))
            s = 10*a
            print("Your amount for tea is:", s, "\n")
        elif d == 2:
            print("You have ordered coffee")
            a = int(input ("Enter quantity: "))
            s = 10*a
```

```
    print("Your amount for coffee is:", s, "\n")
elif d == 3:
    print("You have ordered coldrink")
    a = int(input ("Enter quantity: "))
    s = 20*a
    print("Your amount for coldrink is:", s, "\n")
elif d == 4:
    print("You have ordered samosa")
    a = int(input ("Enter quantity: "))
    s = 10*a
    print("Your amount for samosa is:", s, "\n")
elif d == 5:
    print("You have ordered sandwich")
    a = int(input ("Enter quantity: "))
    s = 50*a
    print("Your amount for sandwich is:", s, "\n")
elif d == 6:
    print("You have ordered dhokla")
    a = int(input ("Enter quantity: "))
    s = 30*a
    print("Your amount for dhokla is:", s, "\n")
elif d == 7:
    print("You have ordered kachori")
    a = int(input ("Enter quantity: "))
    s = 10*a
    print("Your amount for kachori is:", s, "\n")
elif d == 8:
    print("You have ordered milk")
    a = int(input ("Enter quantity: "))
    s = 20*a
    print("Your amount for kachori is:", s, "\n")
elif d == 9:
    print("You have ordered noodles")
    a = int(input ("Enter quantity: "))
    s = 50*a
    print("Your amount for noodles is:", s, "\n")
elif d == 10:
    print("You have ordered pasta")
    a = int(input ("Enter quantity: "))
    s = 50*a
    print("Your amount for pasta is:", s, "\n")
else:
    print("Please enter your choice from the menu")
```


```
def luggagebill():
    global z
    print("Do you want to see rate for luggage: Enter 1 for yes:")
    ch = int(input ("enter your choice:"))
    if ch == 1:
        sql = "select * from lugage"
        mycursor.execute(sql)
        rows = mycursor.fetchall ()
        for x in rows:
            print(x)
        y = int(input("Enter your weight of extra luggage->"))
        z = y*1000
        print("Your luggage bill:", z, "\n")
    return z
```



```
def lb ():
    print(z)

def res ():
    print(s)

def ticketamount():
    a = input("enter customer name:")
    print("customer name:", a, "\n")
    print("lugage bill:")
    lb()
    print("food bill:")
    res()
    print("total amount")
```

```
AIR TICKET RESERVATION
enter 1: To enter customer data
enter 2: To view class
enter 3: For  Ticket amount
enter 4: For viewing food menu
enter 5: For food bill
enter 6: For lugage bill
enter 7: For complete amount
enter 8: For exit
Enter your choice: 7
enter customer name:Nandita Krishnan
customer name : Nandita Krishnan

lugage bill:
10000
food bill:
250
total amount

 Want to run again y/n? y
```

```
def Menuset():
    print("AIR TICKET RESERVATION")
    print("enter 1: To enter customer data")
    print("enter 2: To view class")
    print("enter 3: For Ticket amount")
    print("enter 4: For viewing food menu")
    print("enter 5: For food bill")
    print("enter 6: For lugage bill")
    print("enter 7: For complete amount")
    print("enter 8: For exit")
    userinput = int(input("Enter your choice: "))
    if userinput == 1:
        registercust()
    elif userinput == 2:
        classtypeview()
    elif userinput == 3:
        ticketprice()
    elif userinput == 4:
        menuview()
```

```
    elif userinput == 5:
        orderitem()
    elif userinput == 6:
        lugagebill()
    elif userinput == 7:
        ticketamount()
    elif userinput == 8:
        quit()
    else:
        print("Enter correct choice")

Menuset()

def runagain():
    runagn = input("\n Want to run again y/n? ")
    while runagn.lower() == 'y':
        if platform.system() == "windows":
            print(os. system('cls'))
        else:
            print(os. system('clear'))
        Menuset()
        runagn = input("\n Want to run again y/n? ")

runagain ()
```

FUTURE ENHANCEMENTS

- The solutions are given as a proposal. The suggestion is revised on user request and optimal changes are made. This loop terminates as soon as the user is gratified with the proposal.

- So, on the whole, system analysis is done to improve the system performance by monitoring it and obtaining the best throughput possible from it. Therefore, system analysis plays a crucial role in designing any system.

- This is an interface of a global distribution system to carry out reservations on desired airlines from any place.

- Airline reservation systems make the life of passengers very easy as they don't need to stand in queues to get their seats reserved.

- They can easily make reservations for any airline just from a single system. On the other hand, it also removes an extra burden from the Airline Department as most of the passengers and travel agencies use this service instead of making reservations at the counters.

BIBLIOGRAPHY

- http://www.google.com/
- http://en.wikipedia.org
- Computer science with Python by Sumita Arora
- NCERT Computer Science Class XII and Class XI