

PYTHON PROJECT FILE

BY GURVEEN KAUR SAHNI AND
PALAK BHARDWAJ

Certificate

This is to certify that

of class XII has successfully completed his/her

Computer Science Project:

“FLIGHT MANAGEMENT SYSTEM”

This report is the result of their efforts. It is completed
under my guidance and supervision in partial fulfillment of the
requirements for

appearing in A.I.S.S.C.E. 2022

PROMILA PAGORE

PGT COMP. SC.

ACKNOWLEDGEMENT

I take this opportunity to express my gratitude to the people who have been instrumental in the successful completion of this project. I express my heartfelt gratitude to my parents for constant encouragement while carrying out this project. I gratefully acknowledge the contribution of the individuals who contributed in bringing this project up to this level and guided me through out.

I express my deep sense of gratitude to the luminary The Principal, K R MANGALAM WORLD SCHOOL, who has been continuously motivating and extending her helping hand to us. My sincere thanks to Ms. Promila, master In-Charge, a guide, a mentor, who critically reviewed the project and helped in solving each and every problem. She was always available for any guidance that I needed during the making of this project.

The support I received from each one of above was vital for success of this project. I am thankful to God for blessing me with such parents and mentors.

INDEX

S. NO.	DESCRIPTION	PAGE NO.
1.	SOFTWARE & HARDWARE REQUIREMENT	4
2.	PROJECT SYNOPSIS	5
3.	HOW TO RUN INSTRUCTIONS	6
4.	SOURCE CODE (PYTHON)	7
5.	SOURCE CODE (SQL)	9
6.	OUTPUT	10

HARDWARE AND SOFTWARE REQUIREMENTS

Operating System Name : Microsoft Windows 11

Operating System : 10.0.22000 build 22000

Processor : Intel core i5 10210U CPU 1.6 GHz , 2112 MHz

Dual core processor

RAM : 8 GB

Hard Disk : 238 GB

Monitor : As required (14 inch)

Software : Python version 3.8.5

PIP Installed SQL CONNECTIVITY, PRETTY TABLE

PROJECT SYNOPSIS

This programme is a basic demonstration of data management of passengers. The project "FLIGHT MANAGEMENT SYSTEM" has been developed keeping in mind the requirement of a flight database in maintaining the records of the clients. The user- friendly system is designed to be used by staff members in order to store data of the passengers as well as keep track of flights in the airport , by accessing all the details of the booked flights and to edit the flight records by adding, updating and deleting specific flight records. The system is designed with an intent to automate the processes of airport, so as to minimize the manual effort.

We have utilized various modules in the development of this programme. To elucidate, the PrettyTable module and the SQL Connectivity Module have played a colossal role to help us build such a programme.

Efforts by:

Gurveen Kaur sahani

Palak Bhardwaj

HOW TO RUN ?

The Flight Management System is designed to cater to management of flight records in an airport , to be used only by admins.

Kindly follow the Instructions :

- 1) Open python (version 3.8 or above)
- 2) Copy the code onto python file and run it.
- 3) Use the following admin username and passord to access the features :-
username : qrdrr88
password: abc4445

mysqlconnect.py - C:\Users\hp\AppData\Local\Programs\Python\Python39\mysqlconnect.py (3.8.5)
File Edit Format Run Options Window Help

```
import mysql.connector
from prettytable import PrettyTable
data=mysql.connector.connect(host="localhost",user="root",password="connectsql*664",charset="utf8")
mycursor=data.cursor()
#mycursor.execute("CREATE DATABASE ticket")
#print("database created")
mycursor.execute("use ticket")
def create_table():
    print("...CREATING TABLE...")
    mycursor.execute("drop table")
    qry="create table INDIGO(FlightNo varchar(10) primary key, FromCity varchar(10),ToCity varchar(10), Distance int(5), Fare int(6))"
    mycursor.execute(qry)
    print("table created")
    mycursor.execute("show tables")
    x=mycursor.fetchall()
    print(type(x))
    for i in x:
        print(i)
def create_admin():
    qry="create table admin(username varchar(43), password varchar(56))"
    mycursor.execute(qry)
    print("table created")
    y=mycursor.fetchall()
    for i in y:
        print(i)
def login():
    print("~~~*50")
    print("\tWELCOME TO FLIGHT MANAGEMENT LOGIN")
    print("~~~*50")
    print("PLEASE ENTER VALID USERNAME AND PASSWORD")
    print("~~~*50")
    us=str(input("enter username:"))
    ps=str(input("enter password:"))
    qry="select * from admin where username='{}' and password='{}'.format(us,ps)"
    mycursor.execute(qry)
    rec=mycursor.fetchall()
    rec=mycursor.fetchall()
    for i in rec:
        if i[0]==us and i[1]==ps:
            print("\tACCESS GRANTED")
            return True
        else:
            print("~~~*50")
            print("\tINVALID USERNAME OR PASSWORD")
            print("~~~*50")
            return False
def check():
    flightno=str(input("flight no:"))
    fromcity=str(input("enter city from:"))
    mycursor.execute("select * from indigo")
    rec=mycursor.fetchall()
    w=PrettyTable()
    w = PrettyTable(["flightno", "fromcity", "tocity", "distance","fare","name"])
    for i in range(len(rec)):
        if rec[i][0]==flightno and rec[i][1]==fromcity:
            print("~~~*50")
            print("\tWELCOME TO FLIGHT INDIGO")
            print("~~~*50")
            print("\tHERE ARE YOUR DETAILS")
            w.add_row([rec[i][0],rec[i][1],rec[i][2],rec[i][3],rec[i][4],rec[i][5]])
            print(w)
            break
        else:
            print("~~~*50")
            print("\tINVALID DETAILS")
            print("~~~*50")
def describe_table():
    qry="desc indigo"
    mycursor.execute(qry)
    r=mycursor.fetchall()
    for i in r:
        print(i)
def insert():
    print("...inserting values...")
    while True:
        flightno=str(input("flight no:"))
        fromcity=str(input("enter city from:"))
        tocity=str(input("enter city to:"))
        distance=int(input("distance:"))
        fare=int(input("enter fare:"))
        name=str(input("enter passenger name:"))
        qry="insert into INDIGO values('"+flightno+"','"+fromcity+"','"+tocity+"','"+str(distance)+"','"+str(fare)+"','"+name+"'"
        mycursor.execute(qry)
        data.commit()
        print("values added")
        ch=input("do you want to enter more data:")
        if ch=="n" or ch=="N":
            break
        data.commit()
    print("TICKET ADDED SUCCESSFULLY")
    show()
def show():
    mycursor.execute("select * from indigo")
    rec=mycursor.fetchall()
    w=PrettyTable()
    w = PrettyTable(["flightno", "fromcity", "tocity", "distance","fare","name"])
    for i in range(len(rec)):
        w.add_row([rec[i][0],rec[i][1],rec[i][2],rec[i][3],rec[i][4],rec[i][5]])
    print(w)
def add():
    qry="alter table indigo add (name varchar(20))"
    mycursor.execute(qry)
    data.commit()
def update():
    ans="y"
    while True:
```



```

while True:
    print(".....UPDATING RECORDS.....")
    flightno=str(input("enter flightno for which fare to be updated:"))
    fare=int(input("enter fare to be updated"))
    qry="update indigo set fare={} where flightno='{}' ".format(fare,flightno)
    mycursor.execute(qry)
    data.commit()
    print("row updated")
    ch=input("do you want to enter more data:")
    if ch=="n" or ch=="N":
        break
    mycursor.execute("select * from indigo")
    rec=mycursor.fetchall()
    print("FARE UPDATED SUCCESSFULLY")
    show()

def delete():
    ans="y"
    while True:
        print(".....DELETING VALUE.....")
        flightno=str(input("enter flightno for which details to be deleted:"))
        qry="delete from indigo where flightno='{}' ".format(flightno)
        mycursor.execute(qry)
        data.commit()
        ch=input("do you want to delete more :")
        if ch=="n" or ch=="N":
            break
    mycursor.execute("select * from indigo")
    rec=mycursor.fetchall()
    print("TICKET DELETED SUCCESSFULLY")
    show()

if login():
    while True:
        print("***50")
        print("\tWELCOME TO MENU DRIVEN PROGRAM")
        print("***50")
        w= PrettyTable()
        w=PrettyTable(["What would you like to do","Press"])
        w.add_row(["CHECK DETAILS OF A SPECIFIC FLIGHTNO.", "1"])
        w.add_row(["TO ADD A NEW FLIGHT TICKET", "2"])
        w.add_row(["TO VIEW BOOKED TICKETS", "3"])
        w.add_row(["UPDATE FARE", "4"])
        w.add_row(["CANCEL TICKET", "5"])
        print(w)
        ch=int(input("enter choice:"))
        if ch==1:
            check()
        elif ch==2:
            insert()
        elif ch==3:
            show()
        elif ch==4:
            update()
        elif ch==5:
            delete()
        else:
            break

```

SQL CODE :-

```
mysql> use ticket;
Database changed
mysql> select * from indigo;
```

FlightNo	FromCity	ToCity	Distance	Fare	name
A231	HYDERABAD	MUMBAI	1000	11000	LUCY
D345	MUMBAI	CHENNAI	2000	13000	BRIAN
T124	DELHI	MUMBAI	1500	17000	EMILY
C312	CHENNAI	DELHI	2500	14000	bella
J777	GOA	DELHI	2433	12000	SASHA
R366	DELHI	NEW YORK	23333	54666	TOM
CA999	DELHI	PUNJAB	1200	8700	DRAKE

```
7 rows in set (0.04 sec)

mysql> create table admin( username varchar(43), password varchar(56));
Query OK, 0 rows affected (0.26 sec)

mysql> insert into admin values ('qrdr88', 'abc4445');
Query OK, 1 row affected (0.06 sec)

mysql> insert into admin values ('gurveen', 'sql665');
Query OK, 1 row affected (0.04 sec)

mysql> insert into admin values ('palak','pqr193');
Query OK, 1 row affected (0.06 sec)

mysql> select * from admin;
```

username	password
qrdr88	abc4445
gurveen	sql665
palak	pqr193

```
3 rows in set (0.00 sec)

mysql>
```

OUTPUT

```
Python 3.8.5 Shell
File Edit Shell Debug Options Window Help
Python 3.8.5 (tags/v3.8.5:580fbb0, Jul 20 2020, 15:43:08) [MSC v.1926 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
== RESTART: C:\Users\hp\AppData\Local\Programs\Python\Python39\mysqlconnect.py =
*****
WELCOME TO FLIGHT MANAGEMENT LOGIN
*****
PLEASE ENTER VALID USERNAME AND PASSWORD
*****
enter username:qrdr88
enter password:abc4445
ACCESS GRANTED
*****
WELCOME TO MENU DRIVEN PROGRAM
*****
+-----+
| What would you like to do | Press |
+-----+
| CHECK DETAILS OF A SPECIFIC FLIGHTNO. | 1 |
| TO ADD A NEW FLIGHT TICKET | 2 |
| TO VIEW BOOKED TICKETS | 3 |
| UPDATE FARE | 4 |
| CANCEL TICKET | 5 |
+-----+
enter choice:1
flight no:A231
enter city from:HYDERABAD
*****
WELCOME TO FLIGHT INDIGO
*****
HERE ARE YOUR DETAILS
+-----+
| flightno | fromcity | tocity | distance | fare | name |
+-----+
| A231 | HYDERABAD | MUMBAI | 1000 | 98777 | LUCY |
+-----+
*****
WELCOME TO MENU DRIVEN PROGRAM
*****
+-----+
| What would you like to do | Press |
+-----+
| CHECK DETAILS OF A SPECIFIC FLIGHTNO. | 1 |
| TO ADD A NEW FLIGHT TICKET | 2 |
| TO VIEW BOOKED TICKETS | 3 |
| UPDATE FARE | 4 |
| CANCEL TICKET | 5 |
+-----+
enter choice:2
...inserting values...
flight no:BQ233
enter city from:PARIS
enter city to:MUNICH
distance:8289
enter fare:13999
enter passenger name:RACHEL
values added
do you want to enter more data:N
TICKET ADDED SUCCESSFULLY
+-----+
| flightno | fromcity | tocity | distance | fare | name |
+-----+
| A231 | HYDERABAD | MUMBAI | 1000 | 11000 | LUCY |
| D345 | MUMBAI | CHENNAI | 2000 | 13000 | BRIAN |
| T124 | DELHI | MUMBAI | 1500 | 17000 | EMILY |
| C312 | CHENNAI | DELHI | 2500 | 14000 | bella |
| J777 | GOA | DELHI | 2433 | 12000 | SASHA |
| R366 | DELHI | NEW YORK | 23333 | 54666 | TOM |
| CA999 | DELHI | PUNJAB | 1200 | 10000 | DRAKE |
| AU945 | SYDNEY | MUMBAI | 43222 | 78944 | WES |
| BQ233 | PARIS | MUNICH | 8289 | 13999 | RACHEL |
+-----+
```

What would you like to do	Press
CHECK DETAILS OF A SPECIFIC FLIGHTNO.	1
TO ADD A NEW FLIGHT TICKET	2
TO VIEW BOOKED TICKETS	3
UPDATE FARE	4
CANCEL TICKET	5

enter choice:3

flightno	fromcity	tocty	distance	fare	name
A231	HYDERABAD	MUMBAI	1000	11000	LUCY
D345	MUMBAI	CHENNAI	2000	13000	BRIAN
T124	DELHI	MUMBAI	1500	17000	EMILY
C312	CHENNAI	DELHI	2500	14000	bella
J777	GOA	DELHI	2433	12000	SASHA
R366	DELHI	NEW YORK	23333	54666	TOM
CA999	DELHI	PUNJAB	1200	10000	DRAKE
AU945	SYDNEY	MUMBAI	43222	78944	WES
BQ233	PARIS	MUNICH	8289	13999	RACHEL

WELCOME TO MENU DRVEN PROGRAM

What would you like to do	Press
CHECK DETAILS OF A SPECIFIC FLIGHTNO.	1
TO ADD A NEW FLIGHT TICKET	2
TO VIEW BOOKED TICKETS	3
UPDATE FARE	4
CANCEL TICKET	5

enter choice:4

.....UPDATING RECORDS.....

enter flightno for which fare to be updated:T124

enter fare to be updated14555

row updated

do you want to enter more data:N

FARE UPDATED SUCCESSFULLY

flightno	fromcity	tocty	distance	fare	name
A231	HYDERABAD	MUMBAI	1000	11000	LUCY
D345	MUMBAI	CHENNAI	2000	13000	BRIAN
T124	DELHI	MUMBAI	1500	14555	EMILY
C312	CHENNAI	DELHI	2500	14000	bella
J777	GOA	DELHI	2433	12000	SASHA
R366	DELHI	NEW YORK	23333	54666	TOM
CA999	DELHI	PUNJAB	1200	10000	DRAKE
AU945	SYDNEY	MUMBAI	43222	78944	WES
BQ233	PARIS	MUNICH	8289	13999	RACHEL

WELCOME TO MENU DRVEN PROGRAM

What would you like to do	Press
CHECK DETAILS OF A SPECIFIC FLIGHTNO.	1
TO ADD A NEW FLIGHT TICKET	2
TO VIEW BOOKED TICKETS	3
UPDATE FARE	4
CANCEL TICKET	5

enter choice:5

.....DELETING VALUE.....

enter flightno for which details to be deleted:R366

do you want to enter more data:n

TICKET DELETED SUCCESSFULLY

flightno	fromcity	to city	distance	fare	name
A231	HYDERABAD	MUMBAI	1000	11000	LUCY
D345	MUMBAI	CHENNAI	2000	13000	BRIAN
T124	DELHI	MUMBAI	1500	14555	EMILY
C312	CHENNAI	DELHI	2500	14000	bella
J777	GOA	DELHI	2433	12000	SASHA
CA999	DELHI	PUNJAB	1200	10000	DRAKE
AU945	SYDNEY	MUMBAI	43222	78944	WES
BQ233	PARIS	MUNICH	8289	13999	RACHEL

WELCOME TO MENU DRVEN PROGRAM

What would you like to do	Press
CHECK DETAILS OF A SPECIFIC FLIGHTNO.	1
TO ADD A NEW FLIGHT TICKET	2
TO VIEW BOOKED TICKETS	3
UPDATE FARE	4
CANCEL TICKET	5