

Kendriya Vidyalaya No.3

AFS Nal, Bikaner



तत् त्वं पूषन् अपावृणु
केन्द्रीय विद्यालय संगठन

Session: 2022-23

Computer science project on
“Airport management system”

Submitted by:

Submitted to: Mr.Pawan Sihag

Name: Kavyansh

Class: 12 A

Roll no.:



तत् त्वं पूषन् अपावृणु
केन्द्रीय विद्यालय संगठन

Certificate

This is to certify that **Kavyansh**, a student of class 12A has completed the computer science project titled “**Airport Management System**” under the guidance of **Mr. Pawan Sihag**(PGT Computer Science) in partial fulfillment of the curriculum of **Central Board of Secondary Education (CBSE)** leading to the award of annual examination of the year **2022-23**.

Sign of internal examiner

Sign of external examiner

Sign of Principal

Acknowledgement

It is my privilege to express my most sincere regards to all who directly or indirectly helped me to complete my project work. I, sincerely express my sincere thanks to the Principal, **Mr. Narsi Lal** for their noble guidance & providing sufficient facilities which have contributed to the successful completion of this project.

I, hereby, acknowledge my heartiest gratitude to my computer science teacher **Mr. Pawan Sihag** for her valuable input, encouragement, and whole-hearted cooperation that led me all the way to accomplish this project.

I take this opportunity to thank all my teachers, parents & friends who have helped me with their valuable suggestions, encouragement & guidance have been very helpful in various phases of the completion of the project.

ABSTRACT

THE AIRPORT MANAGEMENT SYSTEM IS AN APPLICATION FOR USER'S DETAILS. IN THIS PROJECT WE TRIED TO SHOW THE WORKING OF AIRPORT AND COVER THE BASIC FUNCTIONALITY OF AN AIRCRAFT MANAGEMENT SYSTEM

TO DEVELOP A PROJECT FOR SOLVING PROBLEMS RELATED TO CUSTOMER AND AIRCRAFT DETAILS IN AIRPORT IN ORDER TO NUTURE THE NEEDS OF CUSTOMER BY PROVIDING VARIOUS WAYTS TO FULFILL THEIR NEEDS

THE AIRCRAFT MANAGEMENT SYSTEM UNDERTAKEN AS A PROJECT IS BASED ON RELEVANT TECHNOLOGIES, THIS ROJECT DESCRIBES TO CARRY OUT THE PROCESSES EASILY AND QUICKLY. WHICH IS NOT POSSIBLE WITH THE MANUALS SYSTEMS. ORGANIZATION NEEDS TO EFFECTIVELY DEFINE AND MANAGE REQUIREMENTS TO ENSURE THEY ARE NOT MEERTING NEEDS OF THE CUSTOMER

THE PROJECG ANALYZES THE SYSTEM REQUIRMENTS AND THE COMES UP WITH THE REUIREMENTS SPECIFICATIONS. THE MANAGEMENT SYSTEM DEALS WITH DATA ENTRY, VALIDATION CONFIRM AND UPDATING WHILES THE INTERACTIVE SYSTEM DEALS WITH SYSTEM INTERACTIONS THE AD MINS.

Objectives

We have created 4 tables in which we have Aircraft details, Booking details, Customer details, and Trips.

Now in Aircraft Detail, we can do These tasks

1. Add an Aircraft
2. Check Aircraft details
3. Update Aircraft details
4. Set an aircraft as not functioning

Now in Booking Detail, we can do These tasks

1. Add a booking
2. Check booking details
3. Update booking details
4. Cancel a booking

Now in Customer details, we can do These tasks

1. Add a customer
2. Check customer details
3. Update customer details
4. Deactivate a customer

Now on Trips, we can do These tasks

1. Add a trip
2. Check trip details
3. Update trips details
4. Cancel a trip

SOURCE CODE

```
import mysql.connector
import datetime
import ADD_AN_AIRCRAFT
import CHECK_AIRCRAFT_DETAILS
import ADD_TRIP
import CHECK_TRIP_DETAILS
import ADD_PASSENGER
import CHECK_PASSENGER_DETAILS
import ADD_BOOKING
import CHECK_BOOKING_DETAILS
from prettytable import from_db_cursor
```

```
cnx=mysql.connector.connect(user='root', password='kavyansh@cs',  
host='127.0.0.1')
```

```
cur=cnx.cursor()
```

```
cur.execute('use ams')
```

```
print('♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥')
```

```
print('*****')
```

```
print(' !!!WELCOME TO K.A.D. AIRPORT MANAGEMENT SYSTEM!!!')
```

```
print('*****')
```

```
print('♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥')
```

```
menu1=""
```

1.AIRCRAFT DETAILS

2. TRIPS

3.CUSTOMERS DETAILS

4. BOOKING DETAILS

5.EXIT

■■■

```
menu2=""
```

- 1.ADD AN AIRCRAFT
- 2.CHECK AIRCRAFT DETAILS
- 3.UPDATE AIRCRAFT DETAILS
- 4.SET AN AIRCRAFT AS NOT FUNCTIONING
- 5.EXIT

'''

menu3=""

- 1.ADD A TRIP
- 2.CHECK TRIP DETAILS
- 3.UPDATE TRIP DETAILS
- 4.CANCEL A TRIP
- 5.EXIT

'''

menu4=""

- 1.ADD A CUSTOMER
- 2.CHECK CUSTOMER DETAILS
- 3.UPDATE CUSTOMER DETAILS
- 4.DEACTIVATE A CUSTOMER
- 5.EXIT

'''

menu5=""

- 1.ADD A BOOKING
- 2.CHECK BOOKING DETAILS
- 3.UPDATE BOOKING DETAILS
- 4.CANCEL A BOOKING
- 5.EXIT

'''

creaters=""

CODERS

- 1.Kavyansh
- 2.Akshat Swami
- 3.Deepika

[CLASS 12TH SCIENCE]

INSTRUCTOR: MR.Pawan Sihag

'''

while True:

print(menu1)

ch1=input('Enter your choice: ')

if ch1=='1':

while True:

print(menu2)

ch2=input('Enter your choice: ')

if ch2=='1':

l=ADD_AN_AIRCRAFT.addAC()

cur.execute('insert into aircraft_details values(%s,%s,%s,%s,%s)',l)

cnx.commit()

elif ch2=='2':

s=input('Press "1" for all data and "2" for specific data search with Acode: ')

if s=='2':

l=CHECK_AIRCRAFT_DETAILS.checkAD()

cur.execute('select * from aircraft_details where Acode=%s',l)

data=cur.fetchall()

if len(data)!=0:

cur.execute('select * from aircraft_details where Acode=%s',l)

mytable = from_db_cursor(cur)

print(mytable)

else:

print('NO SUCH DATA FOUND!')

elif s=='1':

cur.execute('select * from aircraft_details')

mytable = from_db_cursor(cur)


```
    print(mytable)
else:
    print('INVALID CHOICE')
elif ch2=='3':
    cur.execute('select * from aircraft_details')
    mytable = from_db_cursor(cur)
    print(mytable)

while True:
    print()
    print('1.Acode\n2.Airlines\n3.Capacity\n4.Model\n5.Exit')
    print()
    try:
        A=input('Enter your choice: ')
        if A=='1':
            try:
                while True:
                    B=int(input('Enter Acode: '))
                    cur.execute('select * from aircraft_details where Acode=%s',[B])
                    check=cur.fetchall()
                    if len(check)==0:
                        print("ACODE DOESN'T EXIST!")
                        break
                    C=int(input('Enter new Acode: '))
                    l=[C,B]
                    cur.execute('update aircraft_details set Acode=%s where Acode=%s',l)
                    cnx.commit()
                    print('DONE')
                    break
            except:
                print('VALUE SHOULD BE INTEGER!')
        elif A=='2':
```

try:

while True:

B=int(input('Enter Acode: '))

cur.execute('select * from aircraft_details where Acode=%s',[B])

check=cur.fetchall()

if len(check)==0:

print("ACODE DOESN'T EXIST!")

break

C=input('Enter new Airline Name: ')

I=[C,B]

cur.execute('update aircraft_details set Airlines=%s where Acode=%s',I)

cnx.commit()

break

except:

print('Acode SHOULD BE INTEGER!')

elif A=='3':

try:

while True:

B=int(input('Enter Acode: '))

cur.execute('select * from aircraft_details where Acode=%s',[B])

check=cur.fetchall()

if len(check)==0:

print("ACODE DOESN'T EXIST!")

break

C=int(input('Enter new Capacity: '))

I=[C,B]

cur.execute('update aircraft_details set capacity=%s where Acode=%s',I)

cnx.commit()

break

except:

```

    print('VALUE SHOULD BE INTEGER!')
elif A=='4':
    try:
        while True:
            B=int(input('Enter Acode: '))
            cur.execute('select * from aircraft_details where Acode=%s',[B])
            check=cur.fetchall()
            if len(check)==0:
                print("ACODE DOESN'T EXIST!")
                break

            C=input('Enter new Model: ')
            l=[C,B]
            cur.execute('update aircraft_details set model=%s where Acode=%s',l)
            cnx.commit()

            break
        except:
            print('Acode SHOULD BE INTEGER!')
elif A=='5':
    break
else:
    print('INVALID INPUT!')
except:
    print('Enter input in correct datatype')
elif ch2=='4':
    cur.execute('select * from aircraft_details')
    mytable = from_db_cursor(cur)
    print(mytable)
    try:
        print()
        A=int(input('Enter Acode of the aircraft u want to set as not functioning(NF): '))
        l=[A]

```

```

        cur.execute('select * from aircraft_details where Acode=%s',l)
        data=cur.fetchall()
        if len(data)!=0:
            cur.execute('update aircraft_details set status="NF" where Acode=%s',l)
            cnx.commit()
            print('STATUS CHANGED OF AIRCRAFT WITH AIRCODE',A)
        else:
            print('NO SUCH DATA FOUND!')
    except:
        print()
        print('Enter integer only')
elif ch2=='5':
    break
else:
    print('INVALID INPUT')
elif ch1=='2':
    while True:
        print(menu3)
        ch3=input('Enter your choice: ')
        if ch3=='1':
            l=ADD_TRIP.add_Trips()
            cur.execute('insert into trips values(%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s)',l)
            cnx.commit()
            print('DONE')
        elif ch3=='2':
            s=input('Press "1" for all data and "2" for specific data search with Flight_code:
')
            if s=='2':
                l=CHECK_TRIP_DETAILS.checkT()
                cur.execute('select * from trips where flight_code=%s',l)
                data=cur.fetchall()
                if len(data)!=0:

```

```

        cur.execute('select * from trips where flight_code=%s',l)
        mytable = from_db_cursor(cur)
        print(mytable)
    else:
        print('NO SUCH DATA FOUND!')
elif s=='1':
    cur.execute('select * from trips')
    mytable = from_db_cursor(cur)
    print(mytable)
else:
    print('INVALID CHOICE')
elif ch3=='3':
    cur.execute('select * from trips')
    mytable = from_db_cursor(cur)
    print(mytable)
    print()

l=[] #[old ,new]
while True:
    print("1.Flight_code ")
    print("2.Departure date and time")
    print("3.Arrival date and time")
    print("4.Destination")
    print("5.Distance")
    print("6.Price of efare")
    print("7.Price of bfare")
    print("8.A_code")
    print("9.Exit\n")
    ch=input("Enter your choice which you want to update : ")

    if ch=='1':
        while True:

```

```

A=input("Enter old flight_code : ")
if len(A)<=30:
    A=A
    cur.execute('select * from trips where flight_code=%s',[C])
    check=cur.fetchall()
    if len(check)==0:
        print("FLIGHT_CODE DOESN'T EXIST!\n")
        break
    else:
        print("Length of flight code should not be greater than 30")
B=input("Enter new flight code : ")
if len(B)<=30:
    B=B
    break
else:
    print("Length of flight code should not be greater than 30")
l=[B,A]
cur.execute('update trips set flight_code=%s where flight_code=%s',l)
cnx.commit()

elif ch=='2':
    while True:
        C = input("Enter flight_code : ")
        cur.execute('select * from trips where flight_code=%s',[C])
        check=cur.fetchall()
        if len(check)==0:
            print("FLIGHT_CODE DOESN'T EXIST!\n")
            break
        try:
            D = input("Enter new Departure Date(YYYY-MM-DD) : ")
            datetime.datetime.strptime(D, '%Y-%m-%d')
            D = D

```

except:

print("Incorrect date format,Date should be like (YYYY-MM-DD)")

break

try:

D1 = input("Enter new Departure Time(HH:MM:SS) : ")

datetime.datetime.strptime(D1, '%H:%M:%S')

D1 = D1

except:

print("Enter valid Time, enter in correct format(HH:MM:SS)")

break

D2 = D + ' ' + D1 #concatenating date and time

l=[D2,C]

cur.execute('update trips set departure=%s where flight_code=%s',l)

cnx.commit()

print('DONE')

elif ch=='3':

while True:

E=input('Enter Flight_code: ')

cur.execute('select * from trips where flight_code=%s',[E])

check=cur.fetchall()

if len(check)==0:

print("FLIGHT_CODE DOESN'T EXIST!\n")

break

try:

F = input("Enter new Arrival Date(YYYY-MM-DD) : ")

datetime.datetime.strptime(F, '%Y-%m-%d')

F = F

except:

print("Incorrect date format,Date should be like (YYYY-MM-DD)")

break

try:

```
F1 = input("Enter new Arrival Time(HH:MM:SS) : ")
datetime.datetime.strptime(F1, '%H:%M:%S')
F1 = F1
except:
    print("Enter valid Time, enter in correct format(HH:MM:SS)")
    break
F2 = F + ' ' + F1 #concatenating date and time
l=[F2,E]
cur.execute('update trips set arival=%s where flight_code=%s',l)
cnx.commit()
print('DONE')
```

```
elif ch=='4':
```

```
while True:
```

```
G=input('Enter Flight_code: ')
cur.execute('select * from trips where flight_code=%s',[G])
check=cur.fetchall()
if len(check)==0:
    print("FLIGHT_CODE DOESN'T EXIST!\n")
    break
H=input("Enter new destination : ")
if len(H)<=25:
    H=H
else:
    print("Length of destination sholud not be greater than 25 ")
    break
l=[H,G]
cur.execute('update trips set destination=%s where flight_code=%s',l)
cnx.commit()
print('DONE')
break
```



```
elif ch=='5':  
    while True:  
        I=input('Enter Flight_code: ')  
        cur.execute('select * from trips where flight_code=%s',[I])  
        check=cur.fetchall()  
        if len(check)==0:  
            print("FLIGHT_CODE DOESN'T EXIST!\n")  
            break  
        try:  
            J=int(input("Enter new distance : "))  
            J=J  
        except:  
            print("Distance should numeric")  
            break  
        I=[J,I]  
        cur.execute('update trips set distance=%s where flight_code=%s',I)  
        cnx.commit()  
        print('DONE')  
        break
```

```
elif ch=='6':  
    while True:  
        K= input('Enter Flight_code: ')  
        cur.execute('select * from trips where flight_code=%s',[K])  
        check=cur.fetchall()  
        if len(check)==0:  
            print("FLIGHT_CODE DOESN'T EXIST!\n")  
            break  
        try:  
            L=int(input("Enter new efare : "))  
            L=L
```

except:

print("Price of efare should numeric")

break

l=[L,K]

cur.execute('update trips set efare=%s where flight_code=%s',l)

cnx.commit()

print('DONE')

break

elif ch=='7':

while True:

M=input('Enter Flight_code: ')

cur.execute('select * from trips where flight_code=%s',[M])

check=cur.fetchall()

if len(check)==0:

print("FLIGHT_CODE DOESN'T EXIST!\n")

break

try:

N = int(input("Enter new bfare : "))

N=N

except:

print("Price of efare should numeric")

break

l=[N,M]

cur.execute('update trips set bfare=%s where flight_code=%s',l)

cnx.commit()

print('DONE')

break

elif ch=='8':

while True:

```

try:
    O=input('Enter flight_code: ')
    cur.execute('select * from trips where flight_code=%s',[O])
    check=cur.fetchall()
    if len(check)==0:
        print("FLIGHT_CODE DOESN'T EXIST!\n")
        break
    P = int(input("Enter new Acode : "))
    P=P
    cur.execute('select * from aircraft_details where Acode=%s',[P])
    check1=cur.fetchall()
    if len(check1)==0:
        print("ACODE DOESN'T EXIST!\n")
        break
except:
    print("Acode should numeric")
    break
l=[P,O]
cur.execute('update trips set Acode=%s where flight_code=%s',l)
cnx.commit()
print('DONE')
break
elif ch=='9':
    break
elif ch3=='4':
    cur.execute('select * from trips')
    mytable = from_db_cursor(cur)
    print(mytable)
    print()
    A=input('Enter Flight_code of the trip u want to cancel: ')
    l=[A]
    cur.execute('select * from trips where Flight_code=%s',l)

```

```
data=cur.fetchall()
if len(data)!=0:
    cur.execute('update trips set status="Cancelled" where Flight_code=%s',l)
    cnx.commit()
    print('TRIP CANCELLED WITH FLIGHT_CODE',A)
else:
    print('NO SUCH DATA FOUND!')
elif ch3=='5':
    break
else:
    print('INVALID INPUT')
```

```
elif ch1=='3':
    while True:
        print(menu4)
        ch2=input('Enter your choice: ')
        if ch2=='1':
            l=ADD_PASSENGER.add_passenger()
            cur.execute('insert into passenger_details values(%s,%s,%s,%s,%s,%s)',l)
            cnx.commit()
        elif ch2=='2':
            s=input('Press "1" for all data and "2" for specific data search with CId: ')
            if s=='2':
                l=CHECK_PASSENGER_DETAILS.checkPD()
                cur.execute('select * from passenger_details where Cid=%s',l)
                data=cur.fetchall()
                if len(data)!=0:
                    cur.execute('select * from passenger_details where Cid=%s',l)
                    mytable = from_db_cursor(cur)
                    print(mytable)
                else:
                    print('NO SUCH DATA FOUND!')
```

```
elif s=='1':
    cur.execute('select * from passenger_details')
    mytable = from_db_cursor(cur)
    print(mytable)
else:
    print('INVALID CHOICE')
elif ch2=='3':
    # update customers detail

while True:
    print("1.Name")
    print("2.Age")
    print("3.Phone number")
    print("4.Gender")
    print("5.Exit\n")
    ch=input("Enter your choice : ")

if ch=='1':
    while True:
        try:
            A=int(input("Enter your customer id : "))
            A=A
            cur.execute('select * from passenger_details where cid=%s',[A])
            check=cur.fetchall()
            if len(check)==0:
                print("CId DOESN'T EXIST!\n")
                break
        except:
            print("Customer id should be numeric")
            break

    D=input("Enter new Name : ")
```

```
if D.isalpha():
    if len(D)<=13:
        D.capitalize()
    else:
        print("First Name sholud not be greater then 13")
        break
else:
    print("Name should be alphabet")
    break
```

```
l=[D,A]
cur.execute('update passenger_details set Name=%s where cid=%s',l)
cnx.commit()
print('DONE')
break
```

```
elif ch=='2':
    while True:
        try:
            A=int(input("Enter your customer id : "))
            A=A
            cur.execute('select * from passenger_details where cid=%s',[A])
            check=cur.fetchall()
            if len(check)==0:
                print("CId DOESN'T EXIST!\n")
                break
        except:
            print("Customer id should be numeric")
            break
    try:
        E=int(input("New Age : "))
        E=E
```

```
except:
```

```
    print("Age should be numeric")
```

```
    break
```

```
l=[E,A]
```

```
cur.execute('update passenger_details set Age=%s where cid=%s',l)
```

```
cnx.commit()
```

```
print('DONE')
```

```
break
```

```
elif ch=='3':
```

```
while True:
```

```
    try:
```

```
        A=int(input("Enter your customer id : "))
```

```
        A=A
```

```
        cur.execute('select * from passenger_details where cid=%s',[A])
```

```
        check=cur.fetchall()
```

```
        if len(check)==0:
```

```
            print("CId DOESN'T EXIST!\n")
```

```
            break
```

```
    except:
```

```
        print("Customer id should be numeric")
```

```
        break
```

```
    try:
```

```
        F= int(input("Please enter your new 10 digit Phone Number : "))
```

```
        G=str(F)
```

```
        if len(G)==10:
```

```
            G=G
```

```
        else:
```

```
            print("Phone number should be of 10 digits")
```

```
            break
```

```
except:
    print("Phone number should be numeric")
    break
l=[G,A]
cur.execute('update passenger_details set Phone_number=%s where
cid=%s',l)
cnx.commit()
print('DONE')
break
```

```
elif ch=='4':
```

```
while True:
```

```
try:
```

```
    A=int(input("Enter your customer id : "))
```

```
    A=A
```

```
    cur.execute('select * from passenger_details where cid=%s',[A])
```

```
    check=cur.fetchall()
```

```
    if len(check)==0:
```

```
        print("CId DOESN'T EXIST!\n")
```

```
        break
```

```
except:
```

```
    print("Customer id should be numeric")
```

```
    break
```

```
H=input("Enter Gender(M/F/O) : ")
```

```
if H.isalpha():
```

```
    l=H.capitalize()
```

```
    if l in ['M','F','O']:
```

```
        l=l
```

```
    else:
```

```
        print("Gender should be from given option")
```

```
        break
```

```
else:
```



```
print("Gender should be alphabet")
```

```
break
```

```
l=[l,A]
```

```
cur.execute('update passenger_details set gender=%s where cid=%s',l)
```

```
cnx.commit()
```

```
print('DONE')
```

```
break
```

```
elif ch=='5':
```

```
break
```

```
elif ch2=='4':
```

```
cur.execute('select * from passenger_details')
```

```
mytable = from_db_cursor(cur)
```

```
print(mytable)
```

```
print()
```

```
A=input('Enter CId of the customer u want to deactivate: ')
```

```
l=[A]
```

```
cur.execute('select * from passenger_details where cid=%s',l)
```

```
data=cur.fetchall()
```

```
if len(data)!=0:
```

```
cur.execute('update passenger_details set status="Deactivated" where  
cid=%s',l)
```

```
cnx.commit()
```

```
print('User deactivated with CId ',A)
```

```
else:
```

```
print('NO SUCH DATA FOUND!')
```

```
elif ch2=='5':
```

```
break
```

```
elif ch1=='4':
```

```
while True:
```

```
print(menu5)
```

```

ch2=input('Enter your choice: ')
if ch2=='1':
    l=ADD_BOOKING.add_booking_details()
    cur.execute('insert into booking_details values(%s,%s,%s,%s,%s)',l)
    cnx.commit()
elif ch2=='2':
    s=input('Press "1" for all data and "2" for specific data search with Flight_code
and CId: ')
    if s=='2':
        l=CHECK_BOOKING_DETAILS.checkBD()
        cur.execute('select * from booking_details where Flight_code=%s and
cid=%s',l)
        data=cur.fetchall()
        if len(data)!=0:
            cur.execute('select * from booking_details where Flight_code=%s and
cid=%s',l)
            mytable = from_db_cursor(cur)
            print(mytable)
        else:
            print('NO SUCH DATA FOUND!')
    elif s=='1':
        cur.execute('select * from booking_details')
        mytable = from_db_cursor(cur)
        print(mytable)
    else:
        print('INVALID CHOICE')
elif ch2=='3':
    l=[]

    while True :
        print("1. Flight code ")
        print("2. Fare")
        print("3. Exit")

```

```
ch=input("Enter your choice : ")
```

```
if ch=='1':
```

```
    while True:
```

```
        try:
```

```
            A=int(input("Enter customer id : "))
```

```
            A=A
```

```
            cur.execute('select * from passenger_details where cid=%s',[A])
```

```
            check=cur.fetchall()
```

```
            if len(check)==0:
```

```
                print("CId DOESN'T EXIST!\n")
```

```
                break
```

```
        except :
```

```
            print("Customer id should be numeric")
```

```
            break
```

```
    B=input("Enter new flight code : ")
```

```
    if len(B)<=30:
```

```
        B=B
```

```
    else:
```

```
        print("Length of flight code should not be greater than 30")
```

```
        break
```

```
    l=[B,A]
```

```
    cur.execute('update booking_details set flight_code=%s where cid=%s',l)
```

```
    cnx.commit()
```

```
    print('DONE')
```

```
    break
```

```
elif ch=='2':
```

```
    while True:
```

```
        try:
```

```
            A=int(input("Enter customer id : "))
```

```
            A=A
```

```

cur.execute('select * from passenger_details where cid=%s',[A])
check=cur.fetchall()
if len(check)==0:
    print("CId DOESN'T EXIST!\n")
    break
except :
    print("Customer id should be numeric")
    break
try:
    D=int(input("Enter new fare : "))
    D=D
except:
    print("Price of efare should numeric")
    break
l=[D,A]
cur.execute('update booking_details set fare=%s where cid=%s',l)
cnx.commit()
print('DONE')
break
elif ch=='3':
    break
elif ch2=='4':
    cur.execute('select * from booking_details')
    mytable = from_db_cursor(cur)
    print(mytable)
    print()
    A=input('Enter CId of the customer u want to deactivate: ')
    B=input('Enter Flight_code of the customer u want to deactivate: ')
    l=[B,A]
    cur.execute('select * from booking_details where flight_code=%s and
cid=%s',l)
    data=cur.fetchall()

```

```
if len(data)!=0:
    cur.execute('update booking_details set status="Cancelled" where
flight_code=%s and cid=%s',l)
    cnx.commit()
    print('BOOKING CANCELLED')
else:
    print('NO SUCH DATA FOUND!')
elif ch2=='5':
    break
if ch1=='5':
    print("\n*****THANKS FOR VISITING!!*****\n')
    print(creaters)
    break
```

FUNCTIONS

1.ADD_AN_AIRCRAFT

```
def addAC():
    l=[]
    while True:
        try:
            A=int(input('Enter Acode: '))
            l.append(A)
            break
        except:
            print('Acode should be numeric')
    B=input('Enter Airline: ')
    while True:
        try:
            C=int(input('Enter Capacity: '))
            l.append(C)
            break
        except:
            print('Capacity should be numeric')
    D=input('Enter Model: ')

    l=[A,B,C,D,'F']
    return l
```

2.CHECK_AIRCRAFT_DETAILS

```
def checkAD():
    while True:
        try:
            A=int(input('Enter Acode: '))
            l=[A]
            return l
            break
        except:
            print('Acode must be in numeric')
```

3.ADD_TRIP

```
def add_Trips():

    l = []

    while True:
        A = input("Enter Flight_code : ")
        if len(A) <= 30:
            l.append(A)
            break
        else:
            print("Length of flight_code should not be greater than 30")

    import datetime
    while True:
        try:
            B = input("Enter Departure Date(YYYY-MM-DD) : ")
            datetime.datetime.strptime(B, "%Y-%m-%d")
            B = B
            break
        except:
            print("Incorrect date format,Date should be like (YYYY-MM-DD)")

    while True:
        try:
            B1 = input("Enter Departure Time(HH:MM:SS) : ")
            datetime.datetime.strptime(B1, "%H:%M:%S")
            B1 = B1
            break
        except:
            print("Enter valid Time, enter in correct format(HH:MM:SS)")

    B2 = B + ' ' + B1 #concatenating date and time
    l.append(B2)

    while True:
        try:
            C = input("Enter Arrival Date(YYYY-MM-DD) : ")
            datetime.datetime.strptime(C, "%Y-%m-%d")
            C = C
            break
        except:
            print("Incorrect date format,Date should be like (YYYY-MM-DD)")

    while True:
        try:
            C1 = input("Enter Arrival Time(HH:MM:SS) : ")
```

```

        datetime.datetime.strptime(C1, '%H:%M:%S')
        C1 = C1
        break
    except:
        print("Enter valid Time, enter in correct format(HH:MM:SS)")

C2 = C + ' ' + C1
l.append(C2)

l.append('Delhi')

while True:
    E = input("Enter Destination : ")
    if E.isalpha():
        if len(E) <= 30:
            l.append(E)
            break
        else:
            print("Length of destination should be less than 30")
    else:
        print("Destination should be in alphabet")

while True:
    try:
        F = int(input("Enter Distance(Km): "))
        l.append(F)
        break
    except:
        print("Distance should numeric")

t1 = datetime.datetime.strptime(B1, "%H:%M:%S")

t2 = datetime.datetime.strptime(C1, "%H:%M:%S")

s = t2 - t1
l.append(str(s))

while True:
    try:
        H = int(input("Price of efare : "))
        l.append(H)
        break
    except:
        print("Price of efare should numeric")

while True:
    try:
        I = int(input("Price of bfare : "))
        l.append(I)

```



```

        break
    except:
        print("Price of efare should numeric")

while True:
    try:
        J = int(input("A_code : "))
        l.append(J)
        break
    except:
        print("A_code should numeric")

status="Confirmed"
l.append(status)

return l

```

4. CHECK_TRIP_DETAILS

```

def checkT():
    l=[]
    while True:
        A = input("Enter Flight_code : ")
        if len(A) <= 30:
            l.append(A)
            break
        else:
            print("Length of flight_code should not be greater than 30")
    return l

```

5. ADD_PASSENGER

```

def add_passenger():

    l=[]

    while True:
        try:
            A=int(input("Customer_id : "))
            l.append(A)
            break
        except:
            print("Customer_id should be numeric")

    while True:
        B=input("First Name : ")
        if B.isalpha():
            if len(B)<=13:
                C=B.capitalize()

```

```

        break
    else:
        print("Name sholud not be greater then 13")
    else:
        print("Name should be alphabet")

while True:
    D=input("Last Name : ")
    if D.isalpha():
        if len(D)<=12:
            E=D.capitalize()
            break
        else:
            print("Name sholud not be greater then 12")
    else:
        print("Name should be alphabet")

F= C +' '+ E    #concatenating first and last name
l.append(F)

while True:
    try:
        G=int(input("Age : "))
        l.append(G)
        break
    except:
        print("Age should be numeric")

while True:
    try:
        H= int(input("Please enter your 10 digit Phone Number : "))
        l=str(H)
        if len(l)==10:
            l.append(H)
            break
        else:
            print("Phone number should be of 10 digits")
    except:
        print("Phone number should be numeric")

while True:
    J=input("Gender(M/F/O) : ")
    if J.isalpha():
        K=J.capitalize()
        if K in ['M','F','O']:
            l.append(K)
            break
        else:

```

```

        print("Gender should be from given option")
    else:
        print("Gender should be alphabet")
    l.append('Activated')
    return l

```

6.CHECK_PASSENGER_DETAILS

```

def checkPD():
    while True:
        try:
            A=int(input('Enter Cld: '))
            l=[A]
            return l
            break
        except:
            print('Cld must be in numeric')

```

7.ADD_BOOKING

```

def add_booking_details():

    l=[]

    while True:
        try:
            A=int(input("Customer_id : "))
            l.append(A)
            break
        except:
            print("Customer id should be numeric")

    while True:
        B=input("Flight_code : ")
        if len(B)<=30:
            l.append(B)
            break
        else:
            print("Length of flight_code should not be greater than 30")

    import datetime
    while True:
        try:
            C=input("Date of booking('YYYY-MM-DD') : ")
            datetime.datetime.strptime(C,'%Y-%m-%d')
            l.append(C)
            break
        except:

```

```
        print("Incorrect data format, should be YYYY-MM-DD")

while True:
    try:
        D=int(input("Fare : "))
        l.append(D)
        break
    except:
        print("Fare should be numeric ")

l.append('Confirmed')
return l
```

8. CHECK_BOOKING_DETAILS

```
def checkBD():
    l=[]
    while True:
        A = input("Enter Flight_code : ")
        if len(A) <= 30:
            l.append(A)
            break
        else:
            print("Length of flight_code should not be greater than 30")
    while True:
        try:
            B=int(input("Customer_id : "))
            l.append(B)
            break
        except:
            print("Customer id should be numeric")
    return l
```

STRUCTURE OF TABLES

```
mysql> use ams;  
Database changed  
mysql> show tables;
```

Tables_in_ams
aircraft_details
booking_details
passenger_details
trips

4 rows in set (0.01 sec)

```
mysql> desc aircraft_details;
```

Field	Type	Null	Key	Default	Extra
ACode	int	NO	PRI	NULL	
Airlines	varchar(25)	NO		NULL	
Capacity	int	YES		NULL	
Model	varchar(10)	YES		NULL	
Status	varchar(2)	YES		NULL	

5 rows in set (0.00 sec)

```
mysql> desc booking_details;
```

Field	Type	Null	Key	Default	Extra
cid	int	YES	MUL	NULL	
flight_code	varchar(30)	YES		NULL	
date_of_booking	date	YES		NULL	
fare	int	YES		NULL	
Status	varchar(20)	YES		NULL	

5 rows in set (0.00 sec)

```
mysql> desc passenger_details;
```

Field	Type	Null	Key	Default	Extra
cid	int	NO	PRI	NULL	
Name	varchar(30)	NO		NULL	
Age	varchar(3)	NO		NULL	
Phone_number	char(10)	NO		NULL	
gender	varchar(1)	YES		NULL	
status	varchar(20)	YES		NULL	

6 rows in set (0.00 sec)

```
mysql> desc trips;
```

Field	Type	Null	Key	Default	Extra
flight_code	varchar(30)	NO	PRI	NULL	
departure	datetime	YES		NULL	
arival	datetime	YES		NULL	
source	varchar(25)	YES		NULL	
destination	varchar(25)	YES		NULL	
distance	int	YES		NULL	
flight_duration	time	YES		NULL	
efare	int	YES	MUL	NULL	
bfare	int	YES		NULL	
Acode	int	YES		NULL	
status	varchar(20)	YES		NULL	

11 rows in set (0.00 sec)

EXISTING DATAS IN THE TABLES

Acode	Airlines	Capacity	Model	Status
105	EMIRATES	150	Airbus	F
108	Lufthansa	175	Boeing	F
109	IndiGo	120	AirbusA321	F
110	Indigo	200	Airbus	NF

4 rows in set (0.04 sec)

```
mysql> select * from booking_details;
```

cid	flight_code	date_of_booking	fare	Status
2712	101W22	2022-09-15	6000	Confirmed
1805	102T23	2022-09-15	5000	Confirmed
610	102T24	2022-09-02	0	Confirmed

3 rows in set (0.02 sec)

```
mysql> select * from passenger_details;
```

cid	Name	Age	Phone_number	gender	status
102	Sudhansh Soni	18	6549873210	O	Activated
203	Asdf Asddf	12	1234567890	M	Deactivated
610	Chatiyaa	89	1125532553	O	Activated
1805	Deepika	18	9418348015	F	Activated
2712	Kavyansh	17	8851120943	M	Activated

5 rows in set (0.02 sec)

```
mysql> select * from trips;
```

flight_code	departure	arival	source	destination	distance	flight_duration	efare	bfare	Acode	status
101W22	2022-09-22 18:25:00	2022-09-22 20:53:21	Delhi	Kolkata	1500	02:28:21	3000	6000	101	Cancelled
102	2018-01-01 12:12:12	2018-02-01 12:12:12	Delhi	wer	120	00:00:00	12000	15000	1099	Confirmed
102T23	2022-09-23 05:10:00	2022-09-23 07:20:00	Delhi	NYC	10000	02:10:00	9999	9999	110	Confirmed
102T24	2022-09-24 00:30:00	2022-09-24 02:50:00	Delhi	Bengluru	2500	02:20:00	1500	3000	103	Confirmed

4 rows in set (0.02 sec)

OUTPUTS

```
===== RESTART: C:\Users\Annu\Desktop\CS PROJECT\MAIN CODE.py
```

[illegible]

!!!WELCOME TO K.A.D. AIRPORT MANAGEMENT SYSTEM!!!

[illegible]

- ```
1.AIRCRAFT DETAILS
2.TRIPS
3.CUSTOMERS DETAILS
4.BOOKING DETAILS
5.EXIT
```

Enter your choice: 1

- 1.ADD AN AIRCRAFT
- 2.CHECK AIRCRAFT DETAILS
- 3.UPDATE AIRCRAFT DETAILS
- 4.SET AN AIRCRAFT AS NOT FUNCTIONING
- 5.EXIT

Enter your choice: 1

Enter Acode: 112

Enter Airline: Air Asia

Enter Capacity: 120

Enter Model: A320

Enter your choice: 2

Press "1" for all data and "2" for specific data search with Acode: 1

| ACode | Airlines  | Capacity | Model      | Status |
|-------|-----------|----------|------------|--------|
| 105   | EMIRATES  | 150      | Airbus     | F      |
| 108   | Lufthansa | 175      | Boeing     | F      |
| 109   | IndiGo    | 120      | AirbusA321 | F      |
| 110   | Indigo    | 200      | Airbus     | NF     |
| 112   | Air Asia  | 120      | A320       | F      |

- 1.ADD AN AIRCRAFT
- 2.CHECK AIRCRAFT DETAILS
- 3.UPDATE AIRCRAFT DETAILS
- 4.SET AN AIRCRAFT AS NOT FUNCTIONING
- 5.EXIT

Enter your choice: 2

```
Press "1" for all data and "2" for specific data search with Acode: 2
```

Enter Acode: 434

NO SUCH DATA FOUND!

Enter your choice: 3

| Acode | Airlines  | Capacity | Model      | Status |
|-------|-----------|----------|------------|--------|
| 105   | EMIRATES  | 150      | Airbus     | F      |
| 108   | Lufthansa | 175      | Boeing     | F      |
| 109   | IndiGo    | 120      | AirbusA321 | F      |
| 110   | Indigo    | 200      | Airbus     | NF     |
| 112   | Air Asia  | 120      | A320       | F      |

- 1.Acode
- 2.Airlines
- 3.Capacity
- 4.Model
- 5.Exit

Enter your choice: 3

Enter Acode: 105

Enter new Capacity: 130

- 1.Acode
- 2.Airlines
- 3.Capacity
- 4.Model
- 5.Exit

Enter your choice: 3

Enter Acode: 105

Enter new Capacity: 130

- 1.Acode
- 2.Airlines
- 3.Capacity
- 4.Model
- 5.Exit

Enter your choice: 4

Enter Acode: 544

ACODE DOESN'T EXIST!



- 1.ADD A TRIP
- 2.CHECK TRIP DETAILS
- 3.UPDATE TRIP DETAILS
- 4.CANCEL A TRIP
- 5.EXIT

Enter your choice: 2  
Press "1" for all data and "2" for specific data search with Flight code: 1

| flight_code |        | departure           |                     | arrival | source   | destination | distance | flight_duration | fare |
|-------------|--------|---------------------|---------------------|---------|----------|-------------|----------|-----------------|------|
| id          | code   | Acode               | status              |         |          |             |          |                 |      |
| 000         | 101    | 2022-12-12 19:19:19 | 2022-12-13 19:19:19 | Delhi   | DEL      | 1200        | 0:00:00  | 1200000         |      |
| 000         | 101W22 | 2022-09-22 18:25:00 | 2022-09-22 20:53:21 | Delhi   | Kolkata  | 1500        | 2:28:21  | 3000            |      |
| 000         | 6000   | 101                 | Cancelled           |         |          |             |          |                 |      |
| 000         | 102    | 2018-01-01 12:12:12 | 2018-02-01 12:12:12 | Delhi   | wer      | 120         | 0:00:00  | 12000           |      |
| 999         | 9999   | 1099                | Confirmed           |         |          |             |          |                 |      |
| 999         | 102T23 | 2022-09-23 05:10:00 | 2022-09-23 07:20:00 | Delhi   | NYC      | 10000       | 2:10:00  | 9000            |      |
| 500         | 102T24 | 2022-09-24 00:30:00 | 2022-09-24 02:50:00 | Delhi   | Bengluru | 2500        | 2:20:00  | 1500            |      |
| 500         | 3000   | 103                 | Confirmed           |         |          |             |          |                 |      |

```
Enter your choice: 2
Press "1" for all data and "2" for specific data search with Flight_code: 2
Enter Flight_code : 999
NO SUCH DATA FOUND!
```

Enter your choice: 3

| -----+-----+-----+-----+-----+-----+-----+-----+-----+----- |                     |                     |        |             |          |                 |    |     |         |
|-------------------------------------------------------------|---------------------|---------------------|--------|-------------|----------|-----------------|----|-----|---------|
| flight_code                                                 | departure           | arival              | source | destination | distance | flight_duration | ef | are | bfare   |
| -----+-----+-----+-----+-----+-----+-----+-----+-----+----- |                     |                     |        |             |          |                 |    |     |         |
| 101                                                         | 2022-12-12 19:19:19 | 2022-12-13 19:19:19 | Delhi  | DEL         | 1200     | 0:00:00         | 12 | 000 | 1200000 |
| 101W22                                                      | 2022-09-22 18:25:00 | 2022-09-22 20:53:21 | Delhi  | Kolkata     | 1500     | 2:28:21         | 3  | 000 | 6000    |
| 102                                                         | 2018-01-01 12:12:12 | 2018-02-01 12:12:12 | Delhi  | wer         | 120      | 0:00:00         | 12 | 000 | 15000   |
| 102T23                                                      | 2022-09-23 05:10:00 | 2022-09-23 07:20:00 | Delhi  | NYC         | 10000    | 2:10:00         | 9  | 999 | 9999    |
| 102T24                                                      | 2022-09-24 00:30:00 | 2022-09-24 02:50:00 | Delhi  | Bengluru    | 2500     | 2:20:00         | 1  | 500 | 3000    |
| -----+-----+-----+-----+-----+-----+-----+-----+-----+----- |                     |                     |        |             |          |                 |    |     |         |

- 1.Flight\_code
- 2.Departure date and time
- 3.Arrival date and time
- 4.Destination
- 5.Distance
- 6.Price of efare
- 7.Price of bfare
- 8.A\_code
- 9.Exit

Enter your choice which you want to update : 5

Enter Flight\_code: 101W22

Enter new distance : 2000

DONE

- 1.Flight\_code
- 2.Departure date and time
- 3.Arrival date and time
- 4.Destination
- 5.Distance
- 6.Price of efare
- 7.Price of bfare
- 8.A\_code
- 9.Exit

Enter your choice which you want to update : 6

Enter Flight\_code: 2222

FLIGHT\_CODE DOESN'T EXIST!

- 1.Flight\_code
- 2.Departure date and time
- 3.Arrival date and time
- 4.Destination
- 5.Distance
- 6.Price of efare
- 7.Price of bfare
- 8.A\_code
- 9.Exit

Enter your choice which you want to update : 7

Enter Flight\_code: 101W22

Enter new bfare : 340000

Enter your choice: 3

- 1.ADD A CUSTOMER
- 2.CHECK CUSTOMER DETAILS
- 3.UPDATE CUSTOMER DETAILS
- 4.DEACTIVATE A CUSTOMER
- 5.EXIT

Enter your choice: 1

Customer\_id : 1310

First Name : AKSHAT

Last Name : SWAMI

Age : SEVENTEEN

Age should be numeric

Age : 17

Please enter your 10 digit Phone Number : 1800-120-2323

Phone number should be numeric

Please enter your 10 digit Phone Number : 8949873278

Gender(M/F/O) : j

Gender should be from given option

Gender(M/F/O) : m

- 1.ADD A CUSTOMER
- 2.CHECK CUSTOMER DETAILS
- 3.UPDATE CUSTOMER DETAILS
- 4.DEACTIVATE A CUSTOMER
- 5.EXIT

Enter your choice: 2

Press "1" for all data and "2" for specific data search with CId: 1

| cid  | Name          | Age | Phone_number | gender | status      |
|------|---------------|-----|--------------|--------|-------------|
| 102  | Sudhansh Soni | 18  | 6549873210   | O      | Activated   |
| 203  | Asdf Asddf    | 12  | 1234567890   | M      | Deactivated |
| 610  | Chatiyaa      | 89  | 1125532553   | O      | Activated   |
| 1310 | Akshat Swami  | 17  | 8949873278   | M      | Activated   |
| 1805 | Deepika       | 18  | 9418348015   | F      | Activated   |
| 2712 | Kavyansh      | 17  | 8851120943   | M      | Activated   |

- 1.ADD A CUSTOMER
- 2.CHECK CUSTOMER DETAILS
- 3.UPDATE CUSTOMER DETAILS
- 4.DEACTIVATE A CUSTOMER
- 5.EXIT

Enter your choice: 2

Press "1" for all data and "2" for specific data search with CId: 2

Enter CId: 1805

| cid  | Name    | Age | Phone_number | gender | status    |
|------|---------|-----|--------------|--------|-----------|
| 1805 | Deepika | 18  | 9418348015   | F      | Activated |

Enter your choice: 3

- 1.Name
- 2.Age
- 3.Phone number
- 4.Gender
- 5.Exit

Enter your choice : 3

Enter your customer id : 1805

Please enter your new 10 digit Phone Number : 9876543210  
DONE

- 1.Name
- 2.Age
- 3.Phone number
- 4.Gender
- 5.Exit

Enter your choice : 4

Enter your customer id : 1805

Enter Gender(M/F/O) : F  
DONE

Enter your choice : 4

Enter your customer id : 2712

Enter Gender(M/F/O) : M  
DONE

- 1.Name
- 2.Age
- 3.Phone number
- 4.Gender
- 5.Exit

Enter your choice : 2

Enter your customer id : 2712

New Age : 18  
DONE

- 1.AIRCRAFT DETAILS
- 2.TRIPS
- 3.CUSTOMERS DETAILS
- 4.BOOKING DETAILS
- 5.EXIT

Enter your choice: 4

- 1.ADD A BOOKING
- 2.CHECK BOOKING DETAILS
- 3.UPDATE BOOKING DETAILS
- 4.CANCEL A BOOKING
- 5.EXIT

Enter your choice: 1

Customer\_id : 1805

Flight\_code : 101W22

Date of booking('YYYY-MM-DD') : QWERTY

Incorrect data format, should be YYYY-MM-DD

Date of booking('YYYY-MM-DD') : 2022-12-12

Fare : 1200

Enter your choice: 2

Press "1" for all data and "2" for specific data search with Flight\_code and CId: 3  
INVALID CHOICE

- 1.ADD A BOOKING
- 2.CHECK BOOKING DETAILS
- 3.UPDATE BOOKING DETAILS
- 4.CANCEL A BOOKING
- 5.EXIT

Enter your choice: 2

Press "1" for all data and "2" for specific data search with Flight\_code and CId: 1

| cid  | flight_code | date_of_booking | fare | Status    |
|------|-------------|-----------------|------|-----------|
| 2712 | 101W22      | 2022-09-15      | 6000 | Confirmed |
| 1805 | 102T23      | 2022-09-15      | 5000 | Confirmed |
| 610  | 102T24      | 2022-09-02      | 0    | Confirmed |
| 1805 | 101W22      | 2022-12-12      | 1200 | Confirmed |

- 1.ADD A BOOKING
- 2.CHECK BOOKING DETAILS
- 3.UPDATE BOOKING DETAILS
- 4.CANCEL A BOOKING
- 5.EXIT

Enter your choice: 2  
Press "1" for all data and "2" for specific data search with Flight\_code and CId: 2  
Enter Flight\_code : 101w22  
Customer\_id : 2712

|      |             |                 |      |           |
|------|-------------|-----------------|------|-----------|
| cid  | flight_code | date_of_booking | fare | Status    |
| 2712 | 101W22      | 2022-09-15      | 6000 | Confirmed |

- 1.ADD A BOOKING
- 2.CHECK BOOKING DETAILS
- 3.UPDATE BOOKING DETAILS
- 4.CANCEL A BOOKING
- 5.EXIT

Enter your choice: 3

1. Flight code
2. Fare
3. Exit

Enter your choice : 2  
Enter customer id : 1234  
CId DOESN'T EXIST!

1. Flight code
2. Fare
3. Exit

Enter your choice : 2  
Enter customer id : 1310  
Enter new fare : 1000000  
DONE

Enter your choice: 4

- 1.ADD A BOOKING
- 2.CHECK BOOKING DETAILS
- 3.UPDATE BOOKING DETAILS
- 4.CANCEL A BOOKING
- 5.EXIT

Enter your choice: 4

|      |             |                 |      |           |
|------|-------------|-----------------|------|-----------|
| cid  | flight_code | date_of_booking | fare | Status    |
| 2712 | 101W22      | 2022-09-15      | 6000 | Confirmed |
| 1805 | 102T23      | 2022-09-15      | 5000 | Confirmed |
| 610  | 102T24      | 2022-09-02      | 0    | Confirmed |
| 1805 | 101W22      | 2022-12-12      | 1200 | Confirmed |

Enter CId of the customer u want to deactivate: 420  
Enter Flight\_code of the customer u want to deactivate: 102W23  
NO SUCH DATA FOUND!

- 1.ADD A BOOKING
- 2.CHECK BOOKING DETAILS
- 3.UPDATE BOOKING DETAILS
- 4.CANCEL A BOOKING
- 5.EXIT

Enter your choice: 4

| cid  | flight_code | date_of_booking | fare | Status    |
|------|-------------|-----------------|------|-----------|
| 2712 | 101W22      | 2022-09-15      | 6000 | Confirmed |
| 1805 | 102T23      | 2022-09-15      | 5000 | Confirmed |
| 610  | 102T24      | 2022-09-02      | 0    | Confirmed |
| 1805 | 101W22      | 2022-12-12      | 1200 | Confirmed |

Enter CId of the customer u want to deactivate: 610

Enter Flight\_code of the customer u want to deactivate: 102t24

BOOKING CANCELLED

- 1.ADD A BOOKING
- 2.CHECK BOOKING DETAILS
- 3.UPDATE BOOKING DETAILS
- 4.CANCEL A BOOKING
- 5.EXIT

Enter your choice: 4

| cid  | flight_code | date_of_booking | fare | Status    |
|------|-------------|-----------------|------|-----------|
| 2712 | 101W22      | 2022-09-15      | 6000 | Confirmed |
| 1805 | 102T23      | 2022-09-15      | 5000 | Confirmed |
| 610  | 102T24      | 2022-09-02      | 0    | Cancelled |
| 1805 | 101W22      | 2022-12-12      | 1200 | Confirmed |

- 1.AIRCRAFT DETAILS
- 2.TRIPS
- 3.CUSTOMERS DETAILS
- 4.BOOKING DETAILS
- 5.EXIT

Enter your choice: 5

\*\*\*\*\*THANKS FOR VISITING!!\*\*\*\*\*

\*\*\*CODERS\*\*\*

- 1.Kavyansh
- 2.Akshat Swami
- 3.Deepika

[CLASS 12TH SCIENCE]

INSTRUCTOR: MR.Pawan Sihag

## DATAS IN TABLES AFTER EXECUTION

```
mysql> select * from aircraft_details;
```

| ACode | Airlines  | Capacity | Model      | Status |
|-------|-----------|----------|------------|--------|
| 105   | EMIRATES  | 130      | Airbus     | F      |
| 108   | Lufthansa | 175      | Boeing     | NF     |
| 109   | IndiGo    | 120      | AirbusA321 | F      |
| 110   | Indigo    | 200      | Airbus     | NF     |
| 112   | Air Asia  | 120      | A320       | F      |

5 rows in set (0.00 sec)

```
mysql> select * from booking_details;
```

| cid  | flight_code | date_of_booking | fare | Status    |
|------|-------------|-----------------|------|-----------|
| 2712 | 101W22      | 2022-09-15      | 6000 | Confirmed |
| 1805 | 102T23      | 2022-09-15      | 5000 | Confirmed |
| 610  | 102T24      | 2022-09-02      | 0    | Cancelled |
| 1805 | 101W22      | 2022-12-12      | 1200 | Confirmed |

4 rows in set (0.00 sec)

```
mysql> select * from passenger_details;
```

| cid  | Name          | Age | Phone_number | gender | status      |
|------|---------------|-----|--------------|--------|-------------|
| 102  | Sudhansh Soni | 18  | 6549873210   | O      | Activated   |
| 203  | Asdf Asddf    | 12  | 1234567890   | M      | Deactivated |
| 610  | Chatiyaa      | 89  | 1125532553   | O      | Activated   |
| 1310 | Akshat Swami  | 17  | 8949873278   | M      | Activated   |
| 1805 | Deepika       | 18  | 9876543210   | F      | Activated   |
| 2712 | Kavyansh      | 18  | 8851120943   | M      | Activated   |

6 rows in set (0.00 sec)

```
mysql> select * from trips;
```

| flight_code | departure           | arival              | source | destination | distance | flight_duration | efare | bfare   | Acode | status    |
|-------------|---------------------|---------------------|--------|-------------|----------|-----------------|-------|---------|-------|-----------|
| 101         | 2022-12-12 19:19:19 | 2022-12-13 19:19:19 | Delhi  | DEL         | 1200     | 00:00:00        | 12000 | 1200000 | 105   | Confirmed |
| 101W22      | 2022-09-22 18:25:00 | 2022-09-22 20:53:21 | Delhi  | Kolkata     | 2000     | 02:28:21        | 3000  | 340000  | 101   | Cancelled |
| 102         | 2018-01-01 12:12:12 | 2018-02-01 12:12:12 | Delhi  | wer         | 120      | 00:00:00        | 12000 | 15000   | 1099  | Cancelled |
| 102T23      | 2022-09-23 05:10:00 | 2022-09-23 07:20:00 | Delhi  | NYC         | 10000    | 02:10:00        | 9999  | 9999    | 110   | Confirmed |
| 102T24      | 2022-09-24 00:30:00 | 2022-09-24 02:50:00 | Delhi  | Bengluru    | 2500     | 02:20:00        | 1500  | 3000    | 103   | Confirmed |

5 rows in set (0.00 sec)



# Bibliography

1. Computer Science with python - Sumita arora
2. <https://www.mysql.com/>
3. <https://replit.com/>
4. <https://www.geeksforgeeks.org/>
5. <https://www.qrcode-monkey.com/>
6. [https://www.altexsoft.com/blog/travel/airport-technology-management-operations-software-solutions-and-vendors/#:~:text=Airport%20management%20systems%2C%20as%20airport,air%20traffic%20control%20\(ATC\).](https://www.altexsoft.com/blog/travel/airport-technology-management-operations-software-solutions-and-vendors/#:~:text=Airport%20management%20systems%2C%20as%20airport,air%20traffic%20control%20(ATC).)
7. [https://www.altexsoft.com/blog/travel/airport-technology-management-operations-software-solutions-and-vendors/#:~:text=Airport%20management%20systems%2C%20as%20airport,air%20traffic%20control%20\(ATC\).](https://www.altexsoft.com/blog/travel/airport-technology-management-operations-software-solutions-and-vendors/#:~:text=Airport%20management%20systems%2C%20as%20airport,air%20traffic%20control%20(ATC).)
8. <https://www.python.org/downloads/>

# QR code

SCAN TO GET ALL FUNCTIONS, MAIN CODE AND FILE



```
print("THANK YOU!!!")
```