

A PROJECT REPORT ON CAFÉ SYSTEM

MODULES USED:

1. csv:

The csv module in Python provides functionality to read from and write to CSV (Comma-Separated Values) files. It simplifies the process of handling CSV data, allowing users to parse and write CSV files easily by providing functions like `csv.reader()` and `csv.writer()`.

2. Tabulate:

"Tabulate" module in Python, it's a third-party library that helps in formatting and printing tabular data in a readable and organized manner. It supports various output formats and provides a convenient way to create tables from data structures like lists or dictionaries.

BRIEF OVERVIEW OF PROJECT:

The main objective of the python project on café management system is to manage the details of menu, orders, customer and reviews digitally.

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

The purpose of the project is to build an application program to reduce the manual work for managing the menu, orders, customer information and reviews.

It tracks all the details about food, customer and orders placed; it also prints various reports as per input given by the user.

INPUT DATA AND VALIDATION OF PROJECT

1. All the fields such as ids, items and menu are validated and does not take invalid values.
2. Each form of item, payment and price cannot accept the blank values.
3. Avoiding errors in data.
4. Controlling amount of input.

FUNCTIONS USED IN THE PROJECT

- **createbackend():** As mysql is working at the backend so using this function tables are created in the existing database.
- **Show_menu():** To display the food and drinks menu of the café.
- **Add_items():** To add items in the menu.
- **delete_items():** To delete items and records.
- **modify_item():** To update the price of any food item.
- **Place_order():** To place an order and getting the bill.
- **cancel_order():** To cancel an order.
- **review():** To view and add thoughts of the customer.
- **Show_choices():** To show the main menu of the café.
- **start():** To ask the choice of user.
- **Commit():** It is use to make sure the changes made to databases are consistent.
- **Execute():** Executes the given database operation.
- **Fetchone():** It returns one row or a single record at a time.
- **Fetchall():** It returns all the records of the table.

SOURCE CODE

```
import mysql.connector
import tabulate
mydb = mysql.connector.connect(host="localhost", user="root", passwd="laxmi17",auth_plugin='mysql_native_password', database="cafe")
mycursor = mydb.cursor()

#CREATING TABLES
def createbackend():
    menu = "create table if not exists menu(ID int primary key,Item varchar(30), Price int, Dish_Type varchar(50))"
    mycursor.execute(menu)
    mydb.commit()
    orders = "create table if not exists orders(O_id int primary key, Customer_name varchar(50), item_name varchar(20), quantity int,Bill_Amount int)"
    mycursor.execute(orders)
    mydb.commit()
    review = "create table if not exists review(C_id int primary key, Customer_Name varchar(30), Stars int, Additional_Thoughts varchar(75))"
    mycursor.execute(review)
    mydb.commit()

#Fetching all data
def Show_menu():
    mycursor.execute("select * from menu")
    myresult = mycursor.fetchall()
    header=['Item name','Price']
    while True:
        print("\n\t\t\t\t\t**FOOD & DRINKS MENU**")
        print("\t\t\t\t\t1...Main Course")
        print("\t\t\t\t\t2...Appetiser")
        print("\t\t\t\t\t3...Dessert")
        print("\t\t\t\t\t4...Drinks")
        print("\t\t\t\t\t5...Back to Main Menu")

        ch=int(input("\t\t\t\t\tEnter Your Choice: "))
        if ch==1:
            table=[]
            table.append(header)
            for x in myresult:
                if (x[3].lower()=="main course"):
                    ele=[x[1],x[2]]
                    table.append(ele)
            print(tabulate.tabulate(table,tablefmt='grid'))
        elif ch==2:
            table=[]
            table.append(header)
            for x in myresult:
                if (x[3].lower()=="appetiser") :
                    # table=[]
                    #table.append(header)
                    ele=[x[1],x[2]]
                    table.append(ele)
            print(tabulate.tabulate(table,tablefmt='grid'))
```

```

elif ch==3:
    table=[]
    table.append(header)
    for x in myresult:
        if (x[3].lower()=="dessert"):
            #table=[]
            #table.append(header)
            ele=[x[1],x[2]]
            table.append(ele)
    print(tabulate.tabulate(table,tablefmt='grid'))

elif ch==4:
    table=[]
    table.append(header)
    for x in myresult:
        if (x[3].lower()=="drinks" or x[2]=="drink"):
            #table=[]
            #table.append(header)
            ele=[x[1],x[2]]
            table.append(ele)
    print(tabulate.tabulate(table,tablefmt='grid'))
else:
    break

#Insertion method
def Add_items():
    tItem= input("\t\t\tEnter the item name: ")
    mycursor.execute("select * from menu")
    myresult = mycursor.fetchall()
    for i in myresult:
        if i[1]==tItem:
            print('\t\t\tItem already exist in menu')
            break
    else:
        while True:
            tid=int(input("\t\t\tEnter the item ID: "))
            tPrice=int(input("\t\t\tEnter the item price: "))
            tDish = input("\t\t\tEnter the item type: ")
            query = "insert into menu values({},'{}',{},{})".format(tid,tItem,tPrice,tDish)
            mycursor.execute(query)
            mydb.commit()
            print("\t\t\trecord inserted")
            conti=input("\n\t\t\tWould you like to add more items?(Yes/No: ")
            if(conti.upper()!="YES" or conti.upper()!='Y'):
                break

```

#Deletion method

```
def delete_items():
    f=0
    mycursor.execute("select * from menu")
    myresult = mycursor.fetchall()
    table=[['Item No','Item Name','Price','Category']]
    for i in myresult:
        table.append(i)
    print(tabulate.tabulate(table,tablefmt='grid'))
    tItem = input("\t\t\tEnter the item name you would like to remove: ")
    for x in myresult:
        if(x[1]==tItem):
            query = "delete from menu where Item = '{}'.format(tItem)
            mycursor.execute(query)
            mydb.commit()
            print("\t\t\tItem has been deleted")
            f=1
            break
    if(f==0):
        print("\t\t\tItem does not exist")
```

#Update method

```
def modify_item():
    mycursor.execute("select * from menu")
    myresult = mycursor.fetchall()
    table=[['Item id','Item Name','Price','Category']]
    for i in myresult:
        table.append(i)
    print(tabulate.tabulate(table,tablefmt='grid'))
    tItem=input("\t\t\tEnter the item whose price you would like to update: ")
    tPrice = int(input("\t\t\tEnter the new price of the item: "))
    f=0

    for x in myresult:
        if (x[0]==tItem.lower()):
            query = "update menu set Price = {} where Item='{}'.format(tPrice,tItem)
            mycursor.execute(query)
            mydb.commit()
            print("\t\t\tItem price has been updated")
            f=1
            break
    if(f==0):
        print("\t\t\tItem does not exist")
```

#Inserting data and calculating the bill

```

def Place_order():
    mycursor.execute("select * from menu")
    myresult = mycursor.fetchall()
    table=[['Item id','Item Name','Price','Category']]
    for i in myresult:
        table.append(i)
    print(tabulate.tabulate(table,tablefmt='grid'))
    Name= input("\t\t\tEnter the customer name: ")
    t=0
    order =""
    oid=int(input("\t\t\tEnter the order ID:"))
    it = input("\t\t\tEnter the item ordered:")
    f=0
    #order= order + " " + it
    for x in myresult:
        if(x[1]==it):
            f=1
            tq = int(input("\t\t\tEnter the quantity of the item ordered: "))
            t = t+tq*x[2]
            tax=t*18/100
            bill =t+tax
            tpay = input("\t\t\tEnter payment method: ")
            query = "insert into orders values({},'{}', '{}', {},{})".format(oid,Name,it,tq, bill)
            mycursor.execute(query)
            mydb.commit()
            print("\t\t\tYour order has been placed. Total payable amount is", bill)
            break
    else:
        if f==0:
            print("\t\t\tItem does not exist")

#deletion_method
def cancel_order():
    mycursor.execute("select * from orders")
    myresult = mycursor.fetchall()
    table=[['O_id','C_Name','Item Name','Quantity','Bill Amount']]
    for i in myresult:
        table.append(i)
    print(tabulate.tabulate(table,tablefmt='psql'))
    tcustom=input("\t\tEnter the name of the customer who would like to cancel their order: ")
    f=0

    for x in myresult:
        if x[1].lower()==tcustom:
            query = "delete from orders where Customer_name = {}".format(tcustom)
            mycursor.execute(query)
            mydb.commit()
            print("\t\t\tOrder cancelled")

```



```

        f=1
        break
    if f==0:
        print("Customer does not exist")

def review():
    print("\t\t\t1. View all reviews")
    print("\t\t\t2. Add new review")
    print("\t\t\t3. Back to Main Menu")
    ch = int(input("\t\t\tEnter your choice: "))
    while True:
        if ch==1:
            mycursor.execute("select * from review")
            myresult = mycursor.fetchall()
            table=[['C_id', 'C_Name', 'Rating', 'Additional Review']]
            for i in myresult:
                table.append(i)
            print(tabulate.tabulate(table, tablefmt='grid'))
            break
        elif ch==2:
            c_id=int(input("\t\t\tEnter you customer id: "))
            tname =input("\t\t\tEnter customer name: ")
            tstars = float(input("Please enter your rating of number of stars(out of 5): "))
            feedback = input("\t\t\tPlease enter any additional thoughts: ")
            query = "insert into review values ({}, '{}', {}, '{}')".format(c_id, tname, tstars, feedback)
            mycursor.execute(query)
            mydb.commit()
            print("\t\t\tThank you for your feedback")
            break
        else:
            Show_choices()

#Display function
def Show_choices():
    print("\n\t\t\t*****^*****^*****")
    print("\t\t\t*****^*****^*****")
    print("\t\t\t*****^*****^*****")
    print("\t\t\t*****VINTAGE CAFE*****")
    print("\t\t\t*****^*****^*****")
    print("\t\t\t*****^*****^*****")
    print("\t\t\t*****^*****^*****")
    print("\n\t\t\t-----MAIN MENU-----")
    print("\t\t\t1. Show food and Beverage menu")
    print("\t\t\t2. Add Items to menu")
    print("\t\t\t3. Delete Items from menu")
    print("\t\t\t4. Modify Items in menu")
    print("\t\t\t5. Place order")
    print("\t\t\t6. Cancel order")
    print("\t\t\t7. Customer Review")
    print("\t\t\t8. Exit")

```

```
#starting function
def start():
    createbackend()
    while True:
        Show_choices()
        choice = int(input("\t\t\tEnter your choice: "))
        if choice == 1:
            Show_menu()
        elif choice == 2:
            Add_items()
        elif choice == 3:
            delete_items()
        elif choice == 4:
            modify_item()
        elif choice == 5:
            Place_order()
        elif choice == 6:
            cancel_order()
        elif choice == 7:
            review()
        elif choice == 8:
            print('Thankyou. You have been logged out')
            break
        else:
            print("\t\t\tPlease enter a valid choice")
start()
```

Databases and Tables made in MySQL for Project are as follows

DBMS:MySQL

Host: localhost

User:root

Pass: laxmi17

Database:cafe

Table Structure: (Images Below)

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

```
+-----+
| Tables_in_cafe |
+-----+
| menu           |
| orders         |
| review         |
+-----+
```

3 rows in set (0.01 sec)

```
mysql> desc menu;
```

Field	Type	Null	Key	Default	Extra
ID	int	NO	PRI	NULL	
Item	varchar(30)	YES		NULL	
Price	int	YES		NULL	
Dish_Type	varchar(50)	YES		NULL	

4 rows in set (0.01 sec)

```
mysql> desc orders;
```

Field	Type	Null	Key	Default	Extra
O_id	int	NO	PRI	NULL	
Customer_name	varchar(50)	YES		NULL	
item_name	varchar(20)	YES		NULL	
quantity	int	YES		NULL	
Bill_Amount	int	YES		NULL	

5 rows in set (0.00 sec)

```
mysql> desc review;
```

Field	Type	Null	Key	Default	Extra
C_id	int	NO	PRI	NULL	
Customer_Name	varchar(30)	YES		NULL	
Stars	int	YES		NULL	
Additional_Thoughts	varchar(75)	YES		NULL	

```
mysql> select * from menu;
```

ID	Item	Price	Dish_Type
101	egg wrap	350	main course
102	cheese sandwich	230	main course
104	paneer tikka	395	appetiser
105	chinese platter	595	appetiser
106	cheesecake	350	dessert
107	icecream cake	415	dessert
110	coke	65	drinks
111	sprite	50	drinks

```
8 rows in set (0.01 sec)
```

```
mysql> select * from orders;
```

O_id	Customer_name	item_name	quantity	Bill_Amount
1	Ria	egg wrap	1	413
2	Rohan	coke	2	153
3	Arshia	chinese platter	1	702
4	Mishthi	cheesecake	3	1239
6	Virat	cheese sandwich	2	543
7	Jogi	sprite	3	177

```
6 rows in set (0.00 sec)
```

```
mysql> select * from review;
```

C_id	Customer_Name	Stars	Additional_Thoughts
1	Ria	4	food was yum
2	Rohan	5	place was pretty
4	Mishthi	5	desserts were delicious
7	Jogi	4	drinks were very cool

```
4 rows in set (0.00 sec)
```

OUTPUT OF THE CODE:

*****^*****
*****^*****
*****^*****
*****^*****
*****VINTAGE CAFE*****
*****^*****
*****^*****
*****^*****

-----MAIN MENU-----

1. Show food and Beverage menu
2. Add Items to menu
3. Delete Items from menu
4. Modify Items in menu
5. Place order
6. Cancel order
7. Customer Review
8. Exit

Enter your choice: 1

FOOD & DRINKS MENU

- 1...Main Course
- 2...Appetiser
- 3...Dessert
- 4...Drinks

5...Back to Main Menu

Enter Your Choice: 1

Item name	Price
egg wrap	350
cheese sandwich	230

FOOD & DRINKS MENU

- 1...Main Course
- 2...Appetiser
- 3...Dessert
- 4...Drinks
- 5...Back to Main Menu

Enter Your Choice: 2

+-----+-----+	
Item name	Price
+-----+-----+	
paneer tikka	395
+-----+-----+	
chinese platter	595
+-----+-----+	

****FOOD & DRINKS MENU****

1...Main Course

2...Appetiser

3...Dessert

4...Drinks

5...Back to Main Menu

Enter Your Choice: 3

+-----+-----+	
Item name	Price
+-----+-----+	
cheesecake	350
+-----+-----+	
icecream cake	415
+-----+-----+	

****FOOD & DRINKS MENU****

1...Main Course

2...Appetiser

3...Dessert

4...Drinks

5...Back to Main Menu

Enter Your Choice: 4

Item name	Price
coke	65
sprite	50

FOOD & DRINKS MENU

- 1...Main Course
 - 2...Appetiser
 - 3...Dessert
 - 4...Drinks
 - 5...Back to Main Menu
- Enter Your Choice: 5

```

*****^*****
*****^*****
*****^*****
*****VINTAGE CAFE*****
*****^*****
*****^*****
*****^*****

```

-----MAIN MENU-----

1. Show food and Beverage menu
2. Add Items to menu
3. Delete Items from menu
4. Modify Items in menu
5. Place order
6. Cancel order
7. Customer Review
8. Exit

Enter your choice: 2
Enter the item name: pesto panini
Enter the item ID: 103
Enter the item price: 275
Enter the item type: main course
record inserted

Would you like to add more items?(Yes/No): no

```
*****^*****  
*****^*****  
*****^*****  
*****VINTAGE CAFE*****  
*****^*****  
*****^*****  
*****^*****
```

-----MAIN MENU-----

1. Show food and Beverage menu
2. Add Items to menu
3. Delete Items from menu
4. Modify Items in menu
5. Place order
6. Cancel order
7. Customer Review
8. Exit

Enter your choice: 3

```
+-----+-----+-----+-----+  
| Item No | Item Name   | Price | Category |  
+-----+-----+-----+-----+  
| 101    | egg wrap    | 350   | main course |  
+-----+-----+-----+-----+  
| 102    | cheese sandwich | 230   | main course |  
+-----+-----+-----+-----+  
| 103    | pesto panini | 275   | main course |  
+-----+-----+-----+-----+  
| 104    | paneer tikka | 395   | appetiser  |  
+-----+-----+-----+-----+  
| 105    | chinese platter | 595   | appetiser  |  
+-----+-----+-----+-----+  
| 106    | cheesecake  | 350   | dessert    |  
+-----+-----+-----+-----+  
| 107    | icecream cake | 415   | dessert    |  
+-----+-----+-----+-----+  
| 110    | coke        | 65    | drinks     |  
+-----+-----+-----+-----+  
| 111    | sprite      | 50    | drinks     |  
+-----+-----+-----+-----+
```

Enter the item name you would like to remove: pesto panini
Item has been deleted

106	cheesecake	350	dessert
107	icecream cake	415	dessert
110	coke	65	drinks
111	sprite	50	drinks

Enter the item name you would like to remove: water
Item does not exist

```

*****^*****
*****^*****
*****^*****
*****VINTAGE CAFE*****
*****^*****
*****^*****
*****^*****

```

-----MAIN MENU-----

1. Show food and Beverage menu
2. Add Items to menu
3. Delete Items from menu
4. Modify Items in menu
5. Place order

6. Cancel order
7. Customer Review
8. Exit

Enter your choice: 4

Item id	Item Name	Price	Category
101	egg wrap	350	main course
102	cheese sandwich	230	main course
104	paneer tikka	395	appetiser
105	chinese platter	595	appetiser
106	cheesecake	350	dessert
107	icecream cake	415	dessert
110	coke	65	drinks
111	sprite	50	drinks

Enter the item whose price you would like to update: cheese sandwich

Item price has been updated

-----MAIN MENU-----

- Enter your choice: 5

Your order has been placed. Total payable amount is 177.0

```

*****^~^~^~^*****
*****^~^~^~^*****
*****^~^~^~^*****
*****VINTAGE CAFE*****
*****^~^~^~^*****
*****^~^~^~^*****
*****^~^~^~^*****

```

-----MAIN MENU-----

1. Show food and Beverage menu
2. Add Items to menu
3. Delete Items from menu
4. Modify Items in menu
5. Place order
6. Cancel order
7. Customer Review
8. Exit

Enter your choice: 6

```

+-----+
| O_id | C_Name | Item Name | Quantity | Bill Amount |
| 1 | Ria | egg wrap | 1 | 413 |
| 2 | Rohan | coke | 2 | 153 |
| 3 | Arshia | chinese platter | 1 | 702 |
| 4 | Mishthi | cheesecake | 3 | 1239 |
| 5 | Parth | paneer tikka | 1 | 466 |
| 6 | Virat | cheese sandwich | 2 | 543 |
| 7 | Jogi | sprite | 3 | 177 |
+-----+

```

Enter the name of the customer who would like to cancel their order: Parth
Order cancelled

```

*****^~^~^~^*****
*****^~^~^~^*****
*****^~^~^~^*****
*****VINTAGE CAFE*****
*****^~^~^~^*****
*****^~^~^~^*****
*****^~^~^~^*****

```

-----MAIN MENU-----

1. Show food and Beverage menu
2. Add Items to menu
3. Delete Items from menu
4. Modify Items in menu
5. Place order
6. Cancel order
7. Customer Review
8. Exit

Enter your choice: 7

1. View all reviews
2. Add new review
3. Back to Main Menu

Enter your choice: 1

C_id	C_Name	Rating	Additional Review
1	Ria	4	food was yum
2	Rohan	5	place was pretty
4	Mishthi	5	desserts were delicious

```
*****^*****
*****^*****
*****^*****
*****VINTAGE CAFE*****
*****^*****
*****^*****
*****^*****
```

-----MAIN MENU-----

1. Show food and Beverage menu
2. Add Items to menu
3. Delete Items from menu
4. Modify Items in menu
5. Place order
6. Cancel order
7. Customer Review
8. Exit

Enter your choice: 7

1. View all reviews
2. Add new review
3. Back to Main Menu

Enter your choice: 2

Enter you customer id: 07

Enter customer name: Jogi

Please enter your rating of number of stars(out of 5): 4

Please enter any additional thoughts: drinks were very cool

Thank you for your feedback

```
*****^*****
*****^*****
*****^*****
*****VINTAGE CAFE*****
*****^*****
*****^*****
*****^*****
```

-----MAIN MENU-----

1. Show food and Beverage menu
2. Add Items to menu
3. Delete Items from menu
4. Modify Items in menu
5. Place order
6. Cancel order
7. Customer Review
8. Exit

Enter your choice: 8

Thankyou. You have been logged out