

MOTHER'S GLOBAL SCHOOL

COMPUTER SCIENCE PROJECT FILE (2020-2021)

PROJECT : ANTEIKU CAFE

**SIDDHANT BALI
12TH A
ROLL NO.**

ACKNOWLEDGEMENT

In performing our assignment, we had to take the help and guideline of some respected persons, who deserve our greatest gratitude. The completion of this assignment gives us much Pleasure. I would like to express my special thanks of gratitude to my teacher Mrs. Nidhi Rani Goel as well as our principal Mrs. Archana Ma'am ,who gave me the golden opportunity to do this wonderful project on the topic ANTEIKU CAFÉ , which also helped me in doing a lot of Research and i came to know about so many new things I am really thankful to them.

Secondly i would also like to thank my team members and friends who helped me a lot in finalizing this project within the limited time frame.

Minimum system requirements :'

- Window 8\Window 10
- Intel i3 processor
- 4gb ram

Software to be used:

- Python 3.x installation
- MySQL for storing information

Technical Stack:

- **Front End**
 - Python as business layer
- **Back End**
 - Python
 - MySQL as database

SOURCE CODE :

#ANTEIKU_CAFE COUNTER INTERFACE PROGRAM SOURCE CODE

#MODULES USED IN THE PROGRAM

import mysql.connector

import datetime

import csv

from prettytable import PrettyTable

#CONNECTION ESTABLISHMENT

mycon=mysql.connector.connect(host="localhost",user="root",passwd="IIT-JEE",database="anteiku_cafe")

if mycon.is_connected():

 print("WELCOME TO ANTEIKU_CAFE COUNTER INTERFACE PROGRAM !!!!")

cursor=mycon.cursor()

#OTHER FUNCTIONS

def commit():

 print("\n")

 print("DATA EXECUTED SUCCESSFULLY !!!!")

 mycon.commit()

 print("\n")

def historia():

 a=str(input("ENTER YOUR NAME :"))

 noww=str(datetime.datetime.now())

 noww=noww.split(" ")

 b=str(noww[-2])

 c=str(noww[-1])

 st="insert into history(NAME,DATE,TIME) values('{}','{}','{}')".format(a,b,c)

```
cursor.execute(st)

mycon.commit()
```

#ENTER NEW CUSTOMER ORDER

def one():

```
fields=["ID","NAME","DATE","TIME","PHONE_NO","ITEMS","AMOUNT","PAYMENT_MODE","ORDER_STATUS"]
```

```
    a=input("ENTER ID :")
    b=input("ENTER NAME :")
    noww=str(datetime.datetime.now())
    noww=noww.split(" ")
    c=str(noww[-2])
    d=str(noww[-1])
    e=input("ENTER PHONE NO. :")
    f=str(input("ENTER ITEMS :"))
    cal=input("DO YOU WANT TO USE CALCULATOR (y/n):")
    if cal=="y":
        eight()
    i=input("ENTER NET AMOUNT :")
    j=input("ENTER PAYMENT MODE :")
    k="PENDING"
    st="insert into customer values({}, '{}', '{}', '{}', {}, '{}', {}, '{}', '{}')".format(a,b,c,d,e,f,i,j,k)
    cursor.execute(st)
    commit()
```

#COMPLETE ORDER

def two():

```
    a="COMPLETED"
```

```
b=int(input("ENTER ID OF CUSTOMER ORDER LIST :"))  
st="update customer set ORDER_STATUS='{}' where ID={}".format(a,b)  
cursor.execute(st)  
commit()
```

#CANCEL ORDER

def three():

```
a="CANCELLED"  
b=int(input("ENTER ID OF CUSTOMER ORDER LIST :"))  
st="update customer set ORDER_STATUS='{}' AMOUNT=0 where ID={}".format(a,b)  
cursor.execute(st)  
commit()
```

#SHOW MENU

def four():

```
x = PrettyTable()  
st="select * from menu"  
cursor.execute(st)  
data=cursor.fetchall()  
x.field_names = [ 'ID','ITEM_NAME','CATEGORY','TYPE','RATE' ]
```

for i in data:

```
    x.add_row(i)  
print(x.get_string())
```

```
commit()
```

#SHOW CUSTOMER DETAILS

def five():

```

st="select * from customer"

cursor.execute(st)

data=cursor.fetchall()

x = PrettyTable()

x.field_names
=['ID','NAME','DATE','TIME','PHONE_NO','ITEMS','AMOUNT','PAYMENT_MODE','ORDER_STATUS']

for i in data:

    x.add_row(i)

print(x.get_string())

commit()

```

#SHOW STAFF DETAILS

```

def six():

    st="select * from staff"

    cursor.execute(st)

    data=cursor.fetchall()

    x = PrettyTable()

    x.field_names =['ID','NAME','DESIGNATION','PHONE_NO ']

    for i in data:

        x.add_row(i)

    print(x.get_string())

    commit()

```

#SHOW USER ACCESS HISTORY

```

def seven():

    st="select * from history"

    cursor.execute(st)

    data=cursor.fetchall()

    x = PrettyTable()

```

```
x.field_names = ['NAME', 'DATE', 'TIME']
```

```
for i in data:
```

```
    x.add_row(i)
```

```
print(x.get_string())
```

```
commit()
```

```
#OPEN CALCULATOR
```

```
def add(x, y):
```

```
    return x + y
```

```
def subtract(x, y):
```

```
    return x - y
```

```
def multiply(x, y):
```

```
    return x * y
```

```
def divide(x, y):
```

```
    return x / y
```

```
def eight():
```

```
    print("Select operation.")
```

```
    print("1.Add")
```

```
    print("2.Subtract")
```

```
    print("3.Multiply")
```

```
    print("4.Divide")
```

```
    while True:
```

```
        choice = input("Enter choice(1/2/3/4): ")
```

```
        if choice in ('1', '2', '3', '4'):
```

```
            num1 = float(input("Enter first number: "))
```

```
            num2 = float(input("Enter second number: "))
```

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

```
                print(num1, "+", num2, "=", add(num1, num2))
```

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



```

        print(num1, "-", num2, "=", subtract(num1, num2))
    elif choice == '3':
        print(num1, "*", num2, "=", multiply(num1, num2))
    elif choice == '4':
        print(num1, "/", num2, "=", divide(num1, num2))
    break
else:
    print("Invalid Input")

```

#CONVERT CUSTOMER LIST INTO EXCEL FILE

```

def nine():
    filename=str(input("FILE NAME :"))
    filename+=" .csv"
    f=open(filename,'w',newline='')

    fields=["ID","NAME","DATE","TIME","PHONE_NO","ITEMS","AMOUNT","PAYMENT_MODE","ORDER_STATUS"]

    csv.writer(f).writerow(fields)

    cursor.execute("select * from customer")

    data=cursor.fetchall()

    list1=[]

    for k in data:
        a=[]
        for j in k:
            a.append(j)
        list1.append(a)

    csv.writer(f).writerows(list1)

    f.close()

    commit()

```

#SHOW AMOUNT COLLECTED TODAY

def ten():

```
    noww=str(datetime.datetime.now())
    noww=noww.split(" ")
    a=str(noww[-2])
    st="select SUM(AMOUNT) from customer where date='{0}'.format(a)
    cursor.execute(st)
    data=cursor.fetchall()
    print("[ AMOUNT COLLECTED TODAY]")
    print(data)
    commit()
```

#CONVERT HISTORY LIST INTO EXCEL FILE

def eleven():

```
    filename=str(input("FILE NAME :"))
    filename+=" ".csv"
    f=open(filename,'w',newline='')
    fields=["NAME","DATE","TIME"]
    csv.writer(f).writerow(fields)
    cursor.execute("select * from history ")
    data=cursor.fetchall()
    list1=[]
    for k in data:
        a=[]
        for j in k:
            a.append(j)
        list1.append(a)
    csv.writer(f).writerows(list1)
```

```
f.close()
commit()
```

```
#CLOSE PROGRAM
```

```
def closeproj():
```

```
    mycon.close()
```

```
    if not mycon.is_connected():
```

```
        print("THANK YOU FOR USING THIS PROGRAM !!!!")
```

```
#MENU
```

```
historia()
```

```
setup='on'
```

```
while setup=='on':
```

```
    print("QUERIES\t:\n\t1. ENTER NEW CUSTOMER ORDER\n\t2. COMPLETE ORDER\n\t3. CANCEL  
ORDER\n\t4. SHOW MENU\n\t5. SHOW CUSTOMER DETAILS\n\t6. SHOW STAFF DETAILS\n\t7. SHOW  
USER ACCESS HISTORY\n\t8. OPEN CALCULATOR\n\t9. CONVERT CUSTOMER LIST INTO EXCEL  
FILE\n\t10. SHOW AMOUNT COLLECTED TODAY\n\t11. CONVERT USER ACCESS HISTORY LIST INTO EXCEL  
FILE\n\t12. CLOSE PROGRAM\n\tNOTE\t:\tTO RUN ANY QUERY ,ENTER IT\'S SERIAL NUMBER AT RUN\n")
```

```
    run=eval(input("\tRUN\t:\t"))
```

```
    if run==1:
```

```
        one()
```

```
    elif run==2:
```

```
        two()
```

```
    elif run==3:
```

```
        three()
```

```
    elif run==4:
```

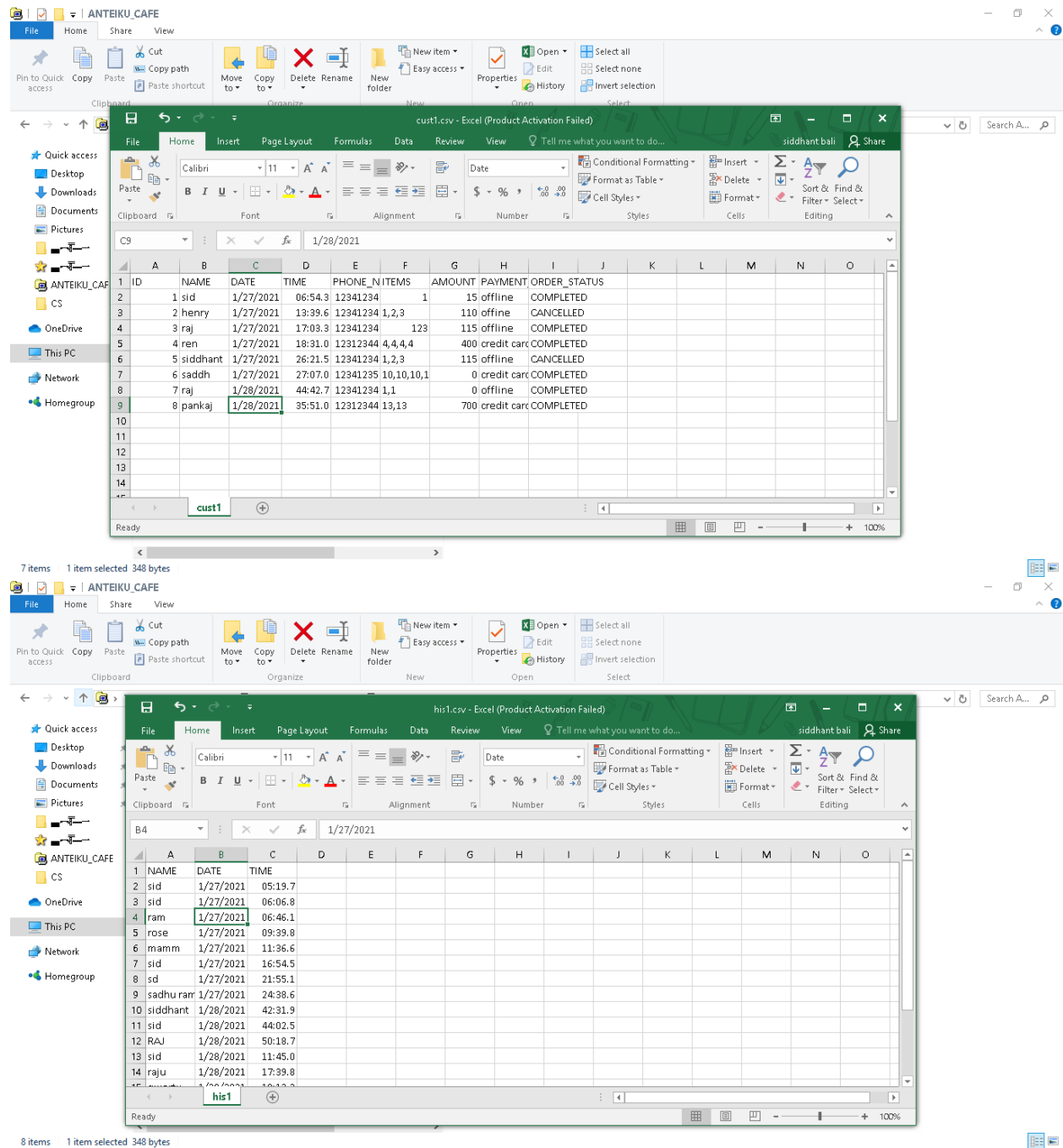
```
        four()
```

```
    elif run==5:
```

```
        five()
```

```
elif run==6:
    six()
elif run==7:
    seven()
elif run==8:
    eight()
elif run==9:
    nine()
elif run==10:
    ten()
elif run==11:
    eleven()
elif run==12:
    closeproj()
    setup="off"
else :
    print("ERROR !!!!\nTYPE CORRECTLY")
```

OUTPUT:



The top screenshot shows the 'cust1.csv' Excel spreadsheet. The data is as follows:

ID	NAME	DATE	TIME	PHONE_N	ITEMS	AMOUNT	PAYMENT	ORDER_STATUS
1	sid	1/27/2021	06:54.3	12341234	1	15	offline	COMPLETED
2	henry	1/27/2021	13:39.6	12341234	1,2,3	110	offline	CANCELLED
3	raj	1/27/2021	17:03.3	12341234	123	115	offline	COMPLETED
4	ren	1/27/2021	18:31.0	12312344	4,4,4,4	400	credit card	COMPLETED
5	siddhant	1/27/2021	26:21.5	12341234	1,2,3	115	offline	CANCELLED
6	saddh	1/27/2021	27:07.0	12341235	10,10,10,1	0	credit card	COMPLETED
7	raj	1/28/2021	44:42.7	12341234	1,1	0	offline	COMPLETED
8	pankaj	1/28/2021	35:51.0	12312344	13,13	700	credit card	COMPLETED

The bottom screenshot shows the 'his1.csv' Excel spreadsheet. The data is as follows:

NAME	DATE	TIME
sid	1/27/2021	05:19.7
sid	1/27/2021	06:06.8
ram	1/27/2021	06:46.1
rose	1/27/2021	09:39.8
mamm	1/27/2021	11:36.6
sid	1/27/2021	16:54.5
sd	1/27/2021	21:55.1
sadhu ram	1/27/2021	24:38.6
siddhant	1/28/2021	42:31.9
sid	1/28/2021	44:02.5
RAJ	1/28/2021	50:18.7
sid	1/28/2021	11:45.0
raju	1/28/2021	17:39.8

PYTHON WINDOW :

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/computer/Desktop/ /anteiku_cafe/ /ANTEIKU_CAFE/meme.py

WELCOME TO ANTEIKU_CAFE COUNTER INTERFACE PROGRAM !!!!

ENTER YOUR NAME :pankaj

QUERIES :

- 1.ENTER NEW CUSTOMER ORDER
- 2.COMPLETE ORDER
- 3.CANCEL ORDER
- 4.SHOW MENU
- 5.SHOW CUSTOMER DETAILS
- 6.SHOW STAFF DETAILS
- 7.SHOW USER ACCESS HISTORY
- 8.OPEN CALCULATOR
- 9.CONVERT CUSTOMER LIST INTO EXCEL FILE
- 10.SHOW AMOUNT COLLECTED TODAY
- 11.CONVERT USER ACCESS HISTORY LIST INTO EXCEL FILE
- 12.CLOSE PROGRAM

NOTE : TO RUN ANY QUERY ,ENTER IT'S SERIAL NUMBER AT RUN

RUN : 4

ID	ITEM_NAME	CATEGORY	TYPE	RATE
1	PLAIN TEA	BEVERAGES	VEG	15
2	GINGER TEA	BEVERAGES	VEG	30
3	PLAIN COFFEE	BEVERAGES	VEG	65
4	WHITE MOCHA COFFEE	BEVERAGES	VEG	100
5	DOUBLE CHOCOLATE CHIP COFFEE	BEVERAGES	VEG	100
6	BREWED COFFEE	BEVERAGES	VEG	100

7	CAPPUCCINO COFFEE	BEVERAGES VEG 150
8	CAFF'E LATTE COFFEE	BEVERAGES VEG 150
9	AFFOGATO COFFEE	BEVERAGES VEG 190
10	CHILLI CHEESE TOAST	BREAKFAST VEG 350
11	EGG TOAST	BREAKFAST NONVEG 350
12	MULTIGRAIN VEG WRAP	BREAKFAST VEG 350
13	MOSSE CAKE	DESSERT NONVEG 350

+---+-----+-----+-----+

DATA EXECUTED SUCCESSFULLY !!!!

QUERIES :

- 1.ENTER NEW CUSTOMER ORDER
- 2.COMPLETE ORDER
- 3.CANCEL ORDER
- 4.SHOW MENU
- 5.SHOW CUSTOMER DETAILS
- 6.SHOW STAFF DETAILS
- 7.SHOW USER ACCESS HISTORY
- 8.OPEN CALCULATOR
- 9.CONVERT CUSTOMER LIST INTO EXCEL FILE
- 10.SHOW AMOUNT COLLECTED TODAY
- 11.CONVERT USER ACCESS HISTORY LIST INTO EXCEL FILE
- 12.CLOSE PROGRAM

NOTE : TO RUN ANY QUERY ,ENTER IT'S SERIAL NUMBER AT RUN

RUN : 5

ID	NAME	DATE	TIME	PHONE_NO	ITEMS	AMOUNT	PAYMENT_MODE	ORDER_STATUS
1	sid	2021-01-27	20:06:54.264249	12341234	1	15	offline	COMPLETED
2	henry	2021-01-27	20:13:39.605427	12341234	1,2,3	110	offline	CANCELLED
3	raj	2021-01-27	20:17:03.307947	12341234	123	115	offline	COMPLETED
4	ren	2021-01-27	20:18:30.971667	12312344	4,4,4,4	400	credit card	COMPLETED
5	siddhant	2021-01-27	20:26:21.519174	12341234	1,2,3	115	offline	CANCELLED
6	saddh	2021-01-27	20:27:06.989290	12341235	10,10,10,10	0	credit card	COMPLETED
7	raj	2021-01-28	11:44:42.744105	12341234	1,1	0	offline	COMPLETED

DATA EXECUTED SUCCESSFULLY !!!!

QUERIES :

1. ENTER NEW CUSTOMER ORDER
2. COMPLETE ORDER
3. CANCEL ORDER
4. SHOW MENU
5. SHOW CUSTOMER DETAILS
6. SHOW STAFF DETAILS
7. SHOW USER ACCESS HISTORY
8. OPEN CALCULATOR

9.CONVERT CUSTOMER LIST INTO EXCEL FILE

10.SHOW AMOUNT COLLECTED TODAY

11.CONVERT USER ACCESS HISTORY LIST INTO EXCEL FILE

12.CLOSE PROGRAM

NOTE : TO RUN ANY QUERY ,ENTER IT'S SERIAL NUMBER AT RUN

RUN : 6

```
+---+-----+-----+-----+
| ID | NAME | DESIGNATION | PHONE_NO |
+---+-----+-----+-----+
| 1 | SIDDHANT | OWNER | 12341234 |
| 2 | DEEPANSHU | COUNTERMEN | 12342234 |
| 3 | PANKAJ | COUNTERMEN | 13342234 |
| 4 | ARJUN | CHEF | 13342234 |
| 5 | BNIDHI | CHEF | 18342234 |
| 6 | JYOTSANA | SWEEPER | 18302234 |
| 7 | KANIKA | SWEEPER | 18302234 |
| 8 | LARRY | CHEF | 8302234 |
| 9 | AMANDA | CHEF | 7302234 |
| 10 | MEGAN FOX | MANAGER | 12341234 |
+---+-----+-----+-----+
```

DATA EXECUTED SUCCESSFULLY !!!!

QUERIES :

1. ENTER NEW CUSTOMER ORDER

2. COMPLETE ORDER

3.CANCEL ORDER

4.SHOW MENU

5.SHOW CUSTOMER DETAILS

6.SHOW STAFF DETAILS

7.SHOW USER ACCESS HISTORY

8.OPEN CALCULATOR

9.CONVERT CUSTOMER LIST INTO EXCEL FILE

10.SHOW AMOUNT COLLECTED TODAY

11.CONVERT USER ACCESS HISTORY LIST INTO EXCEL FILE

12.CLOSE PROGRAM

NOTE : TO RUN ANY QUERY ,ENTER IT'S SERIAL NUMBER AT RUN

RUN : 7

```
+-----+-----+-----+
| NAME | DATE | TIME |
+-----+-----+-----+
| sid | 2021-01-27 | 20:05:19.703850 |
| sid | 2021-01-27 | 20:06:06.771317 |
| ram | 2021-01-27 | 20:06:46.052304 |
| rose | 2021-01-27 | 20:09:39.841033 |
| mamm | 2021-01-27 | 20:11:36.640368 |
| sid | 2021-01-27 | 20:16:54.533113 |
| sd | 2021-01-27 | 20:21:55.081159 |
| sadhu ram | 2021-01-27 | 20:24:38.610651 |
| siddhant | 2021-01-28 | 11:42:31.919458 |
| sid | 2021-01-28 | 11:44:02.470634 |
| RAJ | 2021-01-28 | 11:50:18.701642 |
| sid | 2021-01-28 | 12:11:44.981772 |
| raju | 2021-01-28 | 12:17:39.813473 |
```

| qwerty | 2021-01-28 | 12:18:12.241694 |

| gogo | 2021-01-28 | 12:19:43.224026 |

| pankaj | 2021-01-28 | 12:34:36.459247 |

+-----+-----+-----+

DATA EXECUTED SUCCESSFULLY !!!!

QUERIES :

1.ENTER NEW CUSTOMER ORDER

2.COMPLETE ORDER

3.CANCEL ORDER

4.SHOW MENU

5.SHOW CUSTOMER DETAILS

6.SHOW STAFF DETAILS

7.SHOW USER ACCESS HISTORY

8.OPEN CALCULATOR

9.CONVERT CUSTOMER LIST INTO EXCEL FILE

10.SHOW AMOUNT COLLECTED TODAY

11.CONVERT USER ACCESS HISTORY LIST INTO EXCEL FILE

12.CLOSE PROGRAM

NOTE : TO RUN ANY QUERY ,ENTER IT'S SERIAL NUMBER AT RUN

RUN : 8

Select operation.

1.Add

2.Subtract

3.Multiply

4.Divide

Enter choice(1/2/3/4): 2

Enter first number: 123

Enter second number: 1

123.0 - 1.0 = 122.0

QUERIES :

1. ENTER NEW CUSTOMER ORDER

2. COMPLETE ORDER

3. CANCEL ORDER

4. SHOW MENU

5. SHOW CUSTOMER DETAILS

6. SHOW STAFF DETAILS

7. SHOW USER ACCESS HISTORY

8. OPEN CALCULATOR

9. CONVERT CUSTOMER LIST INTO EXCEL FILE

10. SHOW AMOUNT COLLECTED TODAY

11. CONVERT USER ACCESS HISTORY LIST INTO EXCEL FILE

12. CLOSE PROGRAM

NOTE : TO RUN ANY QUERY ,ENTER IT'S SERIAL NUMBER AT RUN

RUN : 1

ENTER ID :8

ENTER NAME :pankaj

ENTER PHONE NO. :12312344

ENTER ITEMS :13,13

DO YOU WANT TO USE CALCULATOR (y/n):y

Select operation.

1.Add

2.Subtract

3.Multiply

4.Divide

Enter choice(1/2/3/4): 3

Enter first number: 350

Enter second number: 2

350.0 * 2.0 = 700.0

ENTER NET AMOUNT :700

ENTER PAYMENT MODE :credit card

DATA EXECUTED SUCCESSFULLY !!!!

QUERIES :

1. ENTER NEW CUSTOMER ORDER

2. COMPLETE ORDER

3. CANCEL ORDER

4. SHOW MENU

5. SHOW CUSTOMER DETAILS

6. SHOW STAFF DETAILS

7. SHOW USER ACCESS HISTORY

8. OPEN CALCULATOR

9. CONVERT CUSTOMER LIST INTO EXCEL FILE

10. SHOW AMOUNT COLLECTED TODAY

11. CONVERT USER ACCESS HISTORY LIST INTO EXCEL FILE

12. CLOSE PROGRAM

NOTE : TO RUN ANY QUERY ,ENTER IT'S SERIAL NUMBER AT RUN

RUN : 5

ID	NAME	DATE	TIME	PHONE_NO	ITEMS	AMOUNT	PAYMENT_MODE	ORDER_STATUS
1	sid	2021-01-27	20:06:54.264249	12341234	1	15	offline	COMPLETED
2	henry	2021-01-27	20:13:39.605427	12341234	1,2,3	110	offline	CANCELLED
3	raj	2021-01-27	20:17:03.307947	12341234	123	115	offline	COMPLETED
4	ren	2021-01-27	20:18:30.971667	12312344	4,4,4,4	400	credit card	COMPLETED
5	siddhant	2021-01-27	20:26:21.519174	12341234	1,2,3	115	offline	CANCELLED
6	saddh	2021-01-27	20:27:06.989290	12341235	10,10,10,10	0	credit card	COMPLETED
7	raj	2021-01-28	11:44:42.744105	12341234	1,1	0	offline	COMPLETED
8	pankaj	2021-01-28	12:35:50.990542	12312344	13,13	700	credit card	PENDING

DATA EXECUTED SUCCESSFULLY !!!!

QUERIES :

- 1.ENTER NEW CUSTOMER ORDER
- 2.COMPLETE ORDER
- 3.CANCEL ORDER
- 4.SHOW MENU
- 5.SHOW CUSTOMER DETAILS
- 6.SHOW STAFF DETAILS
- 7.SHOW USER ACCESS HISTORY

8.OPEN CALCULATOR

9.CONVERT CUSTOMER LIST INTO EXCEL FILE

10.SHOW AMOUNT COLLECTED TODAY

11.CONVERT USER ACCESS HISTORY LIST INTO EXCEL FILE

12.CLOSE PROGRAM

NOTE : TO RUN ANY QUERY ,ENTER IT'S SERIAL NUMBER AT RUN

RUN : 2

ENTER ID OF CUSTOMER ORDER LIST :8

DATA EXECUTED SUCCESSFULLY !!!!

QUERIES :

1. ENTER NEW CUSTOMER ORDER

2. COMPLETE ORDER

3. CANCEL ORDER

4. SHOW MENU

5. SHOW CUSTOMER DETAILS

6. SHOW STAFF DETAILS

7. SHOW USER ACCESS HISTORY

8. OPEN CALCULATOR

9. CONVERT CUSTOMER LIST INTO EXCEL FILE

10. SHOW AMOUNT COLLECTED TODAY

11. CONVERT USER ACCESS HISTORY LIST INTO EXCEL FILE

12. CLOSE PROGRAM

NOTE : TO RUN ANY QUERY ,ENTER IT'S SERIAL NUMBER AT RUN

RUN : 5

ID	NAME	DATE	TIME	PHONE_NO	ITEMS	AMOUNT	PAYMENT_MODE	ORDER_STATUS
1	sid	2021-01-27	20:06:54.264249	12341234	1	15	offline	COMPLETED
2	henry	2021-01-27	20:13:39.605427	12341234	1,2,3	110	offline	CANCELLED
3	raj	2021-01-27	20:17:03.307947	12341234	123	115	offline	COMPLETED
4	ren	2021-01-27	20:18:30.971667	12312344	4,4,4,4	400	credit card	COMPLETED
5	siddhant	2021-01-27	20:26:21.519174	12341234	1,2,3	115	offline	CANCELLED
6	saddh	2021-01-27	20:27:06.989290	12341235	10,10,10,10	0	credit card	COMPLETED
7	raj	2021-01-28	11:44:42.744105	12341234	1,1	0	offline	COMPLETED
8	pankaj	2021-01-28	12:35:50.990542	12312344	13,13	700	credit card	COMPLETED

DATA EXECUTED SUCCESSFULLY !!!!

QUERIES :

- 1.ENTER NEW CUSTOMER ORDER
- 2.COMPLETE ORDER
- 3.CANCEL ORDER
- 4.SHOW MENU
- 5.SHOW CUSTOMER DETAILS
- 6.SHOW STAFF DETAILS

7.SHOW USER ACCESS HISTORY

8.OPEN CALCULATOR

9.CONVERT CUSTOMER LIST INTO EXCEL FILE

10.SHOW AMOUNT COLLECTED TODAY

11.CONVERT USER ACCESS HISTORY LIST INTO EXCEL FILE

12.CLOSE PROGRAM

NOTE : TO RUN ANY QUERY ,ENTER IT'S SERIAL NUMBER AT RUN

RUN : 10

[AMOUNT COLLECTED TODAY]

[(Decimal('700'),,)]

DATA EXECUTED SUCCESSFULLY !!!!

QUERIES :

1. ENTER NEW CUSTOMER ORDER

2.COMplete ORDER

3.CANCEL ORDER

4.SHOW MENU

5.SHOW CUSTOMER DETAILS

6.SHOW STAFF DETAILS

7.SHOW USER ACCESS HISTORY

8.OPEN CALCULATOR

9.CONVERT CUSTOMER LIST INTO EXCEL FILE

10.SHOW AMOUNT COLLECTED TODAY

11.CONVERT USER ACCESS HISTORY LIST INTO EXCEL FILE

12.CLOSE PROGRAM

NOTE : TO RUN ANY QUERY ,ENTER IT'S SERIAL NUMBER AT RUN

RUN : 9

FILE NAME :cust1

DATA EXECUTED SUCCESSFULLY !!!!

QUERIES :

- 1.ENTER NEW CUSTOMER ORDER
- 2.COMPLETE ORDER
- 3.CANCEL ORDER
- 4.SHOW MENU
- 5.SHOW CUSTOMER DETAILS
- 6.SHOW STAFF DETAILS
- 7.SHOW USER ACCESS HISTORY
- 8.OPEN CALCULATOR
- 9.CONVERT CUSTOMER LIST INTO EXCEL FILE
- 10.SHOW AMOUNT COLLECTED TODAY
- 11.CONVERT USER ACCESS HISTORY LIST INTO EXCEL FILE
- 12.CLOSE PROGRAM

NOTE : TO RUN ANY QUERY ,ENTER IT'S SERIAL NUMBER AT RUN

RUN : 11

FILE NAME :his1

DATA EXECUTED SUCCESSFULLY !!!!

QUERIES :

1. ENTER NEW CUSTOMER ORDER
2. COMPLETE ORDER
3. CANCEL ORDER
4. SHOW MENU
5. SHOW CUSTOMER DETAILS
6. SHOW STAFF DETAILS
7. SHOW USER ACCESS HISTORY
8. OPEN CALCULATOR
9. CONVERT CUSTOMER LIST INTO EXCEL FILE
10. SHOW AMOUNT COLLECTED TODAY
11. CONVERT USER ACCESS HISTORY LIST INTO EXCEL FILE
12. CLOSE PROGRAM

NOTE : TO RUN ANY QUERY ,ENTER IT'S SERIAL NUMBER AT RUN

RUN : 1232

ERROR !!!!

TYPE CORRECTLY

QUERIES :

1. ENTER NEW CUSTOMER ORDER
2. COMPLETE ORDER
3. CANCEL ORDER
4. SHOW MENU
5. SHOW CUSTOMER DETAILS
6. SHOW STAFF DETAILS
7. SHOW USER ACCESS HISTORY
8. OPEN CALCULATOR

9.CONVERT CUSTOMER LIST INTO EXCEL FILE

10.SHOW AMOUNT COLLECTED TODAY

11.CONVERT USER ACCESS HISTORY LIST INTO EXCEL FILE

12.CLOSE PROGRAM

NOTE : TO RUN ANY QUERY ,ENTER IT'S SERIAL NUMBER AT RUN

RUN : 12

THANK YOU FOR USING THIS PROGRAM !!!!

>>>

SQL WINDOW :

Enter password: *****

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 24

Server version: 8.0.21 MySQL Community Server - GPL

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

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

```
mysql> SHOW DATABASES;
```

```
+-----+
| Database      |
+-----+
| anteiku_cafe  |
| entertainment |
| information_schema |
| mgs_cs_12a    |
| mysql         |
| office        |
| performance_schema |
| sakila        |
| school        |
| sys           |
```

```
| world      |
```

```
+-----+
```

11 rows in set (0.19 sec)

```
mysql> USE ANTEIKU_CAFE;
```

Database changed

```
mysql> SHOW TABLES;
```

```
+-----+
```

```
| Tables_in_anteiku_cafe |
```

```
+-----+
```

```
| customer      |
```

```
| history       |
```

```
| menu          |
```

```
| staff         |
```

```
+-----+
```

4 rows in set (0.06 sec)

```
mysql> SELECT * FROM CUSTOMER;
```

```
+-----+-----+-----+-----+-----+-----+-----+-----+
```

```
| ID | NAME  | DATE   | TIME          | PHONE_NO | ITEMS  | AMOUNT | PAYMENT_MODE |  
ORDER_STATUS |
```

```
+-----+-----+-----+-----+-----+-----+-----+-----+
```

```
| 1 | sid   | 2021-01-27 | 20:06:54.264249 | 12341234 | 1      | 15 | offline | COMPLETED |
```

```
| 2 | henry | 2021-01-27 | 20:13:39.605427 | 12341234 | 1,2,3  | 110 | offline | CANCELLED |
```

```
| 3 | raj   | 2021-01-27 | 20:17:03.307947 | 12341234 | 123    | 115 | offline | COMPLETED |
```

```
| 4 | ren   | 2021-01-27 | 20:18:30.971667 | 12312344 | 4,4,4,4 | 400 | credit card | COMPLETED |
```

5	siddhant	2021-01-27	20:26:21.519174	12341234	1,2,3	115	offline	CANCELLED
6	saddh	2021-01-27	20:27:06.989290	12341235	10,10,10,10	0	credit card	COMPLETED
7	raj	2021-01-28	11:44:42.744105	12341234	1,1	0	offline	COMPLETED
8	pankaj	2021-01-28	12:35:50.990542	12312344	13,13	700	credit card	COMPLETED

```
+-----+-----+-----+-----+-----+-----+-----+-----+
```

8 rows in set (0.00 sec)

```
mysql> SELECT * FROM HISTORY;
```

NAME	DATE	TIME
sid	2021-01-27	20:05:19.703850
sid	2021-01-27	20:06:06.771317
ram	2021-01-27	20:06:46.052304
rose	2021-01-27	20:09:39.841033
mamm	2021-01-27	20:11:36.640368
sid	2021-01-27	20:16:54.533113
sd	2021-01-27	20:21:55.081159
sadhu ram	2021-01-27	20:24:38.610651
siddhant	2021-01-28	11:42:31.919458
sid	2021-01-28	11:44:02.470634
RAJ	2021-01-28	11:50:18.701642
sid	2021-01-28	12:11:44.981772
raju	2021-01-28	12:17:39.813473
qwerty	2021-01-28	12:18:12.241694
gogo	2021-01-28	12:19:43.224026
pankaj	2021-01-28	12:34:36.459247

```
+-----+-----+-----+
```

16 rows in set (0.00 sec)

mysql> SELECT * FROM STAFF;

```
+-----+-----+-----+-----+
```

```
| ID | NAME   | DESIGNATION | PHONE_NO |
```

```
+-----+-----+-----+-----+
```

```
| 1 | SIDDHANT | OWNER      | 12341234 |
```

```
| 2 | DEEPANSHU | COUNTERMAN | 12342234 |
```

```
| 3 | PANKAJ   | COUNTERMAN | 13342234 |
```

```
| 4 | ARJUN    | CHEF       | 13342234 |
```

```
| 5 | BNIDHI   | CHEF       | 18342234 |
```

```
| 6 | JYOTSANA | SWEEPER    | 18302234 |
```

```
| 7 | KANIKA   | SWEEPER    | 18302234 |
```

```
| 8 | LARRY    | CHEF       | 8302234  |
```

```
| 9 | AMANDA   | CHEF       | 7302234  |
```

```
| 10 | MEGAN FOX | MANAGER    | 12341234 |
```

```
+-----+-----+-----+-----+
```

10 rows in set (0.00 sec)

mysql> SELECT * FROM MENU;

```
+-----+-----+-----+-----+-----+
```

```
| ID | ITEM_NAME           | CATEGORY | TYPE | RATE |
```

```
+-----+-----+-----+-----+-----+
```

```
| 1 | PLAIN TEA           | BEVERAGES | VEG  | 15 |
```

```
| 2 | GINGER TEA           | BEVERAGES | VEG  | 30 |
```

```
| 3 | PLAIN COFFEE         | BEVERAGES | VEG  | 65 |
```

```
| 4 | WHITE MOCHA COFFEE   | BEVERAGES | VEG  | 100 |
```

```
| 5 | DOUBLE CHOCOLATE CHIP COFFEE | BEVERAGES | VEG  | 100 |
```


6	BREWED COFFEE	BEVERAGES VEG	100
7	CAPPUCCINO COFFEE	BEVERAGES VEG	150
8	CAFF`E LATTE COFFEE	BEVERAGES VEG	150
9	AFFOGATO COFFEE	BEVERAGES VEG	190
10	CHILLI CHEESE TOAST	BREAKFAST VEG	350
11	EGG TOAST	BREAKFAST NONVEG	350
12	MULTIGRAIN VEG WRAP	BREAKFAST VEG	350
13	MOSSE CAKE	DESSERT NONVEG	350

```
+-----+-----+-----+-----+-----+
```

13 rows in set (0.00 sec)

mysql> DESC CUSTOMER;

```
+-----+-----+-----+-----+-----+
```

Field	Type	Null	Key	Default	Extra
-------	------	------	-----	---------	-------

```
+-----+-----+-----+-----+-----+
```

ID	int	YES		NULL	
NAME	varchar(100)	YES		NULL	
DATE	varchar(100)	YES		NULL	
TIME	varchar(100)	YES		NULL	
PHONE_NO	int	YES		NULL	
ITEMS	char(30)	YES		NULL	
AMOUNT	int	YES		NULL	
PAYMENT_MODE	varchar(30)	YES		NULL	
ORDER_STATUS	varchar(30)	YES		NULL	

```
+-----+-----+-----+-----+-----+
```

9 rows in set (0.12 sec)

mysql> DESC MENU;

```
+-----+-----+-----+-----+-----+
```

Field	Type	Null	Key	Default	Extra
ID	int	YES		NULL	
ITEM_NAME	varchar(100)	YES		NULL	
CATEGORY	varchar(100)	YES		NULL	
TYPE	varchar(100)	YES		NULL	
RATE	int	YES		NULL	

5 rows in set (0.00 sec)

mysql> DESC STAFF;

Field	Type	Null	Key	Default	Extra
ID	int	YES		NULL	
NAME	varchar(100)	YES		NULL	
DESIGNATION	varchar(100)	YES		NULL	
PHONE_NO	int	YES		NULL	

4 rows in set (0.00 sec)

mysql> DESC HISTORY;

Field	Type	Null	Key	Default	Extra
NAME	varchar(100)	YES		NULL	
DATE	varchar(100)	YES		NULL	
TIME	varchar(100)	YES		NULL	

3 rows in set (0.00 sec)

mysql>