

HOTEL MANAGEMENTS SYSTEM

A PROJECT REPORT

Submitted

By

**MADHAN K R
SARAN RAJ S
DEEPAK S**

Reg. No.



NARAYANA E-TECHNO SCHOOL

NO.180, ZAMIN PALLAVARAM, 200 FEET ROAD,

CHENNAI-600043

2022-2023



BONA-FIDE CERTIFICATE

SUBJECT : COMPUTER SCIENCE

REGISTRATION NUMBER :

Certified to be the bona-fide record of work done by

Of Class XII as PROJECT in topic

HOTEL MANAGEMENT SYSTEM

at Narayana E-Techno School, Chennai-600043 during the
academic year 2022-2023. Submitted for
AISSE-Practical Examination held on 2022-23.

Seal

Principal

Date:

Internal examiner

External Examiner

ACKNOWLEDGEMENT

The satisfaction and euphoria of the successful completion of any task would be incomplete without the mention of the people who made it possible. The constant guidance of these people and encouragement provided by them crowned my efforts with success and glory. I consider it as a privilege to express my gratitude to all those who led and guided me in the course of this project.

First and foremost I would like to express my gratitude to the MANAGEMENT of my school for making it possible for me to be a part of this project.

I would like to convey my sincere regards to Mr.Venkat Kohit, my internal guide for his valuable suggestions and necessary guidance during the course of the project.

I also take esteem privilege to thank Mr. Venkat Kohit who has been the very backbone and a catalyst for the success behind the completion of this project.

Last but not the least I thank my PARENTS AND FRIENDS who have been a constant source of inspiration for the completion of this project.

Above all, I thank the ALMIGHTY AND MY MASTER for the blessings to complete this project successfully.

CONTENTS

S.No.	Topic name	Page number
1.	Introduction of the project	1
1.1	About python	1.1
2.	Objective of the project	2
3.	Scope of the project	3
4.	Input and output requirements	4
5.	Hardware and software requirements	5
6.	System design	6
7.	Source code	8
8.	The Database	23
9.	Database table structure	25
10.	Output	27

11.	Advantages and disadvantages	33
12.	Further development areas	34
13.	Conclusion	35
14.	Bibliography	36

INTRODUCTION

The project is titled "Hotel management System". The program manages different varieties of services offered by the hotel for customers and different costs of each service.

The initial data required was first entered by creating a MySQL database and entering different services offered by the hotel and their respective costs.

The summing up of total cost of the service offered and the interpretation of the initial data was done by the python MySQL interface

The user identifies the Services he/she requires and gives the service code as input, the program selects the items and calculates a total bill the user is supposed to pay.

ABOUT PYTHON

- Python is a widely-used, interpreted, object-oriented, and high-level programming language with dynamic semantics, used for general-purpose programming. It was created by Guido van Rossum, and first released on February 20, 1991.
- Python programming language comes from an old BBC television comedy sketch series called Monty Python's Flying Circus.
- One of the amazing features of Python is the fact that it is actually one person's work. Usually, new programming languages are developed and published by large companies employing lots of professionals, and due to copyright rules, it is very hard to name any of the people involved in the project. Python is an exception.
- Of course, van Rossum did not develop and evolve all the Python components himself. The speed with which Python has spread around the world is a result of the continuous work of thousands (very often anonymous) programmers, testers, users (many of them aren't IT specialists), and enthusiasts, but it must be said that the very first idea (the seed from which Python sprouted) came to one head – Guido's.

Python goals:

In 1999, Guido van Rossum defined his goals for Python:

1. An easy and intuitive language just as powerful as those of the major competitors;
2. Open-source, so anyone can contribute to its development;
3. Code that is as understandable as plain English
4. Suitable for everyday tasks, allowing for short development times.

Why Python?

1. It's easy to learn – the time needed to learn Python is shorter than for many other languages; this means that it is possible to start the actual programming faster
2. it's easy to teach – the teaching workload is smaller than that needed by other languages. This means that the teacher can put more emphasis on general techniques, not wasting energy on exotic tricks, strange exceptions and incomprehensible rules.
3. it's easy to use for writing new software – it's often Possible to write code faster when using python.
4. it's easy to understand – it's also often easier to understand someone else's code faster when written In python

OBJECTIVE OF PROJECT

The primary objective of the program is to simplify the complex system of billing in hotels.

The program calculates the total and subtotal for the customer with ease, instead of making the customer go through the tedious process of calculating the total amount he/she has to pay.

SCOPE OF PROJECT

This project has further scope of improvement in the future to make it economical and useful for commercial use.

The program currently contains primarily only a fixed list of services with fixed prices of services without calculating taxes, the program can be advanced by:

- Updating tax rates
- Varying the sale price with algorithms so as to profit from each service
- Making an integrated database system by creating a user database
- Introduce online booking systems with relative ease.
- Introducing Financials of the hotel into the database so as to adjust rates.

And much more innovations can be made.

INPUT AND OUTPUT

- The Input expected from the user is only the codes of the services chosen by him/her.
- The Output expected would simply be the total cost of the services offered.

HARDWARE AND SOFTWARE REQUIREMENTS

SOFTWARE:

Python Idle

Python 3.9.0 shell

Mozilla firefox

Google chrome

Windows 10 pro OS

Adobe Reader XI

MySQL 8.0.30.0

HARDWARE:

- HP laptop
- Wi-fi Modem

SYSTEM DESIGN

□

DECLARATION OF GLOBAL VARIABLES

ESTABLISHING PYTHON SQL CONNECTION

STORING OF CUSTOMER DETAILS

BOOKING ROOM FOR CUSTOMER

CALCULATING ROOM RENT

CALCULATING RESTAURANT BILL

CALCULATING GAMING BILL

CALCULATING FASHION STORE BILL

DISPLAYING CUSTOMER DETAILS

GENERATING TOTAL BILL AMOUNT

GENERATING OLD BILL

EXITING THE PROGRAM

SOURCE CODE

```
import mysql.connector

# GLOBAL VARIABLES DECLARATION

myConnnection=""

cursor=""

userName=""

password=""

roomrent=0

restaurentbill=0

gamingbill=0

fashionbill=0

totalAmount=0

cid=" "

#MODULE TO CHECK MYSQL CONNECTIVITY

def

    MySQLconnectionCheck():

        global myConnnection

        global username global

        password

        userName = input("\n ENTER MYSQL SERVER'S USERNAME : ")

        password = input("\n ENTER MYSQL SERVER'S PASSWORD : ")

        myConnnection=mysql.connector.connect(host="localhost",user=userName,passwd=password ,

        auth_plugin='mysql_native_password' )
```



```
if myConnection:

    print("\n CONGRATULATIONS ! YOUR MYSQL CONNECTION HAS BEEN ESTABLISHED !")

    cursor=myConnection.cursor()

    cursor.execute("CREATE DATABASE IF NOT EXISTS HMS")

    cursor.execute("COMMIT")

    cursor.close()

    return myConnection

else:

    print("\nERROR ESTABLISHING MYSQL CONNECTION CHECK USERNAME AND PASSWORD !")

#MODULE TO ESTABLISHED MYSQL CONNECTION

def MySQLconnection ()

    global userName

    global password

    global

    global cid

    myConnection=mysql.connector.connect(host="localhost",user=userName,passwd=password,
    database="HMS" , auth_plugin='mysql_native_password' )

    if myConnection:

        else:

            print("\nERROR ESTABLISHING MYSQL CONNECTION !")

            myConnection.close()

def userEntry()

    global cid

    if myConnection:

        cursor=myConnection.cursor()

        createTable = """CREATE TABLE IF NOT EXISTS C_DETAILS(CID
        VARCHAR(20),C_NAME VARCHAR(30),C_ADDRESS VARCHAR(30),C_AGE
        VARCHAR(30),
```



```
C_COUNTRY VARCHAR(30),P_NO VARCHAR(30),C_EMAIL VARCHAR(30))"""
```

```
cursor.execute(createTable)
```

```
cid = input("Enter Customer Identification Number : ")
```

```
name = input("Enter Customer Name : ")
```

```
address = input("Enter Customer
```

```
Address : ")
```

```
age= input("Enter Customer Age : ")
```

```
nationality = input("Enter Customer Country : ")
```

```
phoneno= input("Enter Customer Contact Number : ")
```

```
email = input("Enter Customer Email : ")
```

```
sql = "INSERT INTO C_Details VALUES(%s,%s,%s,%s,%s,%s,%s,%s)"
```

```
values=
```

```
(cid,name,address,age,nationality,phoneno,email)
```

```
cursor.execute(sql,values)
```

```
cursor.execute("COMMIT")
```

```
print("\nNew Customer Entered In The System Successfully !")
```

```
cursor.close()
```

```
else:
```

```
print("\nERROR ESTABLISHING MYSQL CONNECTION !")
```

```
def bookingRecord():
```

```
global cid
```

```
customer=searchCustomer()
```

```
if customer:
```

```
if myConnection:
```

```
cursor=myConnection.cursor()
```

```
createTable ="CREATE TABLE IF NOT EXISTS BOOKING_RECORD(CID  
VARCHAR(20),CHECK_INDATE ,CHECK_OUT DATE)"
```

```
cursor.execute(createTable)
```

```
checkin=input("\n Enter Customer CheckIN Date [ YYYY-MM-DD ] : ")
```

```
checkout=input("\n Enter Customer CheckOUT Date [ YYYY-MM-DD ] : ")
```

```
sql= "INSERT INTO BOOKING_RECORD VALUES(%s,%s,%s)"
```



```
values=(cid,checkin,checkout)

cursor.execute(sql,values)

cursor.

execute("COMMIT")

print("\nCHECK-IN AND CHECK-OUT ENTRY ADDED SUCCESSFULLY !")

cursor.close()

else:

    print("\nERROR ESTABLISHING MYSQL CONNECTION !")


def roomRent():

    global cid

    customer=searchCustomer(

    ifcustomer:

    global roomrent

if my Connection:

    cursor=myConnection.cursor()

    createTable = """CREATE TABLE IF NOT EXISTS ROOM_RENT(CID
VARCHAR(20),ROOM_CHOICEINT,NO_OF_DAYS INT,ROOMNO INT ,ROOM RENT INT)"""

    cursor.execute(createTable)

    print ("\n ##### We have The Following Rooms For You #####")

    print (" 1. Ultra Royal      > 10000

Rs.") print (" 2. Royal      > 5000 Rs. ")

    print (" 3. Elite          > 3500 Rs. ")

    print (" 4. Budget          > 2500 USD ")

    roomchoice =int(input("Enter Your Option : "))

    roomno=int(input("Enter Customer Room No : "))

    noofdays=int(input("Enter No. Of Days      : "))
```



```
if roomchoice==1:
    roomrent = noofdays * 10000
    print("\nUltra Royal Room Rent : ",roomrent)
elif roomchoice==2:
    roomrent = noofdays * 5000
    print("\nRoyal Room Rent : ",roomrent)
elif roomchoice==3:
    roomrent = noofdays * 3500
    print("\nElite Royal Room Rent : ",roomrent)
elif roomchoice==4:
    roomrent = noofdays * 2500
    print("\nBudget Room Rent : ",roomrent)
else:
    print("Sorry ,May Be You Are Giving Me Wrong Input, Please Try Again !!! ")

    return

sql= "INSERT INTO ROOM_RENT VALUES(%s,%s,%s,%s,%s)"
values=(cid,roomchoice,noofdays,roomno,roomrent,)
cursor.execute(sql,values)
cursor.execute("COMMIT")
print("Thank You , Your Room Has Been Booked For : ",noofdays , "Days" )
print("Your Total Room Rent is : Rs. ",roomrent)
cursor.close()
else:
    print("\nERROR ESTABLISHING MYSQL CONNECTION !")

def Restaurent():
    global cid
    customer=searchCustomer()
```



```
if customer:
    global
    restaurentbill

    if myConnection:
        cursor=myConnection.cursor(
createTable = """CREATE TABLE IF NOT EXISTS RESTAURENT(CID
VARCHAR(20),CUISINEVARCHAR(30),QUANTITY VARCHAR(30),BILL VARCHAR(30))
        """

    cursor.execute(createTable)

    print("1. Vegetarian Combo          > 300 Rs.")
    print("2. Non-Vegetarian Combo      > 500 Rs.")
    print("3. Vegetarian & Non-Vegetarian Combo    > 750 Rs.")
    choice_dish = int(input("Enter Your Cusine : "))

    quantity=int(input("Enter Quantity : "))

    if choice_dish==1:
        print("\nSO YOU HAVE ORDER: Vegetarian Combo ")

        restaurentbill = quantity * 300

    elif choice_dish==2:
        print("\nSO YOU HAVE ORDER: Non-Vegetarian Combo ")

        restaurentbill = quantity * 500

    elif choice_dish==3:
        print("\nSO YOU HAVE ORDER: Vegetarian & Non-Vegetarian Combo ")

        restaurentbill= quantity * 750
```



```
else:

    print("Sorry ,May Be You Are Giving Me Wrong Input, Please Try Again !!! ")return

sql= "INSERT INTO RESTAURENT VALUES(%s,%s,%s,%s)"

values=(cid,choice_dish,quantity,restaurentbill)

cursor.execute(sql,values)

cursor.execute("COMMIT")

print("Your Total Bill Amount Is : Rs. ",restaurentbill)

print("\n\n**** WE HOPE YOU WILL ENJOY YOUR MEAL ***\n\n" )

cursor.close()

else:

    print("\nERROR ESTABLISHING MYSQL CONNECTION !")

def Gaming():

    global cid

    customer=searchCustomer()

    if customer:

        global gamingbill

        if myConnection:

            cursor=myConnection.cursor()

            createTable ="""CREATE TABLE IF NOT EXISTS GAMING(CID
            VARCHAR(20),GAMESVARCHAR(30),HOURS VARCHAR(30),GAMING_BILL
            VARCHAR(30))

            """

            cursor.execute(createTable)

            print("""

            1. Table Tennis          > 150 Rs./HR

            2. Bowling                > 100 Rs./HR
```

3. Snooker > 250 Rs./HR
4. VR World Gaming > 400 Rs./HR
5. Video Games > 300 Rs./HR
6. Swimming Pool Games -----> 350 Rs./HR
7. Exit

""")

game=int(input("Enter What Game You Want To Play : "))

hour=int(input("Enter No Of Hours You Want To Play : "))

print("\n\n#####")

if game==1:

print("YOU HAVE SELECTED TO PLAY : Table Tennis")

gamingbill = hour * 150

elif game==2:

print("YOU HAVE SELECTED TO PLAY : Bowling")

gamingbill = hour * 100

elif game==3:

print("YOU HAVE SELECTED TO PLAY : Snooker")

gamingbill = hour * 250

elif game==4:

print("YOU HAVE SELECTED TO PLAY : VR World Gaming")

gamingbill = hour * 400

elif game==5:

print("YOU HAVE SELECTED TO PLAY : Video

Games")gamingbill = hour *

300 elif game ==6:

print("YOU HAVE SELECTED TO PLAY : Swimming Pool Games")

gamingbill = hour * 350



```
else:

    print("Sorry ,May Be You Are Giving Me Wrong Input, Please Try Again !!! ")

    return

sql= "INSERT INTO GAMING VALUES(%s,%s,%s,%s)"
values=(cid,game,hour,gamingbill)
cursor.execute(sql,values)
cursor.execute("COMMIT")
print("Your Total Gaming Bill Is : Rs. ",gamingbill)
print("FOR : ",hour," HOURS","\n *** WE HOPE YOU WILL ENJOY YOUR GAME ***")
print("\n\n#####")
cursor.close()

else:

    print("ERROR ESTABLISHING MYSQL CONNECTION !")

def Fashion():
    global cid
    customer=searchCustomer()if
    customer:
        global fashionbill
        if myConnection:
            cursor=myConnection.cursor()

            createTable ="""CREATE TABLE IF NOT EXISTS FASHION(CID
VARCHAR(20),DRESSVARCHAR(30),AMOUNT VARCHAR(30),BILL VARCHAR(30))"""

            cursor.execute(createTable)

            print("""
                1. Shirts          > 1500 Rs.
                2. T-Shirts        > 300 Rs.
                3. Pants           > 2000 Rs.
                4. Jeans            > 4000 Rs.
                5. Tassel top       > 500 Rs.

                *   Gown           > 3000 Rs
                *   Western dress - > 3000 Rs.
```

8.Skirts > 400 Rs.

9.Trousers > 200 Rs.

10.InnerWear > 30 Rs.

```
""")
dress=int(input("Enter the your Choice wear: "))
quantity=int(input("How many you want to buy: "))

if dress==1:
    print("\nShirts"
    )
    fashionbill =
quantity*1500 elif dress==2:
    print("\nT-Shirts")
    fashionbill = quantity*300
elif dress==3:
    print("\nPant")
    fashionbill =
quantity*2000 elif dress==4:
    print("\nJeans")
    fashionbill = quantity
*4000 elif dress==5:
    print("\nTassel top")
    fashionbill = quantity*500
elif dress==6:
    print("\nGown")
    fashionbill = quantity *3000
```




```
elif dress==7:

    print("\nWestern dress")

    fashionbill = quantity*3000

elif dress==8:

    print("\nSkirt")

    fashionbill = quantity * 400

elif dress==9:

    print("\nTrousers")

    fashionbill = quantity * 200

elif dress==10:

    print("\nInnerWear")

    fashionbill = quantity * 30

else:

    print("Sorry ,May Be You Are Giving Me Wrong Input, Please Try Again !!! ")return

    sql= "INSERT INTO FASHION VALUES(%s,%s,%s,%s)"

    values=(cid,dress,quantity,fashionbill)

    cursor.execute(sql,values)

    cursor.execute("COMMIT")

    print("\n\n#####")

    print("\nYOU SELECT ITEM NO : ",dress,"\nYOUR QUANTITY IS : ",quantity," ITEMS","\nTHANK

YOU FOR SHOPPING VISIT AGAIN!!!" )

    print("\nYour Total Bill Is : ",fashionbill)

    print("\n\n#####")

    cursor.close()

else:

    print("\nERROR ESTABLISHING MYSQL CONNECTION !")

def totalAmount():

    global cid

    customer=searchCustomer()if
```



customer:

global grandTotal

global roomrent

global restaurentbill

global fashionbill

global gamingbill

if myConnection:

 cursor=myConnection.cursor()

 createTable = """CREATE TABLE IF NOT EXISTS TOTAL(CID
VARCHAR(20),C_NAME VARCHAR(30),ROOMRENT INT ,RESTAURENTBILL INT
,GAMINGBILL INT,FASHIONBILLINT,TOTALAMOUNT INT)"""

 cursor.execute(createTable)

 sql= "INSERT INTO TOTAL VALUES(%s,%s,%s,%s,%s,%s,%s)"

 name = input("Enter Customer Name : ")

 grandTotal=roomrent + restaurentbill + fashionbill + gamingbill

 values= (cid,name,roomrent,restaurentbill , gamingbill,fashionbill,grandTotal)

 cursor.execute(sql,values)

 cursor.execute("COMMIT")

 cursor.close()

print("\n **** CROWN PLAZA MIAMI **** CUSTOMER BIILING ****")

print("\n CUSTOMER NAME : " ,name)

print("\nROOM RENT : Rs. ",roomrent)

print("\nRESTAURENT BILL : Rs. ",restaurentbill)

print("\nFASHION BILL : Rs. ",fashionbill)

print("\nGAMING BILL : Rs. ",gamingbill)

print("_____")



```
print("\nTOTAL AMOUNT    : Rs. ",grandTotal)

cursor.close()

else:

    print("\nERROR ESTABLISHING MYSQL CONNECTION !")

def searchOldBill():

    global cid

    customer=searchCustomer()

    if customer:

        if myConnection:

            cursor=myConnection.cursor()

            sql="SELECT * FROM TOTAL WHERE CID= %s"

            cursor.execute(sql,(cid,))

            data=cursor.fetchall()

            if data:

                print(data)

            else:

                print("Record Not Found Try Again !")

            cursor.close()

        else:

            print("\nSomething Went Wrong ,Please Try Again !")

def

searchCustomer():

    global cid

    if myConnection:

        cursor=myConnection.cursor()

        cid=input("ENTER CUSTOMER ID : ")

        sql="SELECT * FROM C_DETAILS WHERE CID= %s"
```



```
cursor.execute(sql,(cid,))
data=cursor.fetchall()

if data:

    print(data)

    return

    True

else:

    print("Record Not Found Try Again !")

    return False

cursor.close()

else:

    print("\nSomthing Went Wrong ,Please Try Again !")

myConnection = MySQLconnectionCheck ()

if myConnection:

    MySQLconnection

    () while(True):

        print("""

            1--->Enter Customer Detais

            2--->Booking Record

            3--->Calculate Room Rent

            4--->Calculate Restaurant Bill

            5--->Calculate Gaming Bill

            6--->Calculate Fashion store Bill

            7--->Display Customer Details

            8--->GENERATE TOTAL BILL AMOUNT

            9--->GENERATE OLD BILL

            10--->EXIT    """)

        choice = int(input("Enter Your Choice"))
```



```
if choice ==
    1:
        userEntry()
elif choice ==2:
        bookingRecord()

elif choice ==3:
        roomRent()
elif choice ==4:
        Restaurent()
elif choice
==5:
        Gaming()
elif choice==6:

Fashion()

        elifchoice==7:
                searchCustomer()
elif choice ==8:
        totalAmount()
elif choice ==9:
        searchOldBill()
        elif choice==10:
                break
        else:

print("Sorry ,May Be You Are Giving Me Wrong Input, Please Try Again !!! ")

else:

        print("\n Error ESTABLISHING MYSQL CONNECTION!")
```

THE DATABASE

```
| TOTALAMOUNT | int(11) | YES | NULL |
+-----+
7 rows in set (0.00 sec)

mysql> SELECT * FROM C_DETAILS;
+-----+
| CID | C_NAME | C_ADDRESS | C_AGE | C_COUNTRY | P_NO | C_EMAIL |
+-----+
| C101 | Suzy Sabastian | 45 street won | 25 | Australia | 76789-32456 | Suzy123@gm
ail.com |
| C101 | Suzy Sabastian | 45 street won | 25 | Australia | 76789-32456 | Suzy123@gm
ail.com |
| C102 | Harry Potter | Hogwarts | 29 | USA | 78906-23456 | Harry34@gm
ail.com |
| C103 | Emma Watson | 23 pine street | 28 | Canada | 76549-45678 | Emma2345@g
mail.com |
| C104 | Selena Gomez | 228 Park Ave | 26 | USA | 74325-12121 | Selena45@g
mail.com |
| C105 | Justin Bieber | new pine street | 24 | USA | 74256-67654 | justbib@gm
ail.com |
+-----+
6 rows in set (0.00 sec)

mysql> _
```

```
7 rows in set (0.07 sec)

mysql> SELECT * FROM FASHION;
+-----+
| CID | DRESS | AMOUNT | BILL |
+-----+
| C101 | 5 | 2 | 1000 |
| C102 | 8 | 1 | 400 |
| C103 | 9 | 2 | 400 |
| C104 | 4 | 1 | 4000 |
| C105 | 10 | 1 | 30 |
+-----+
5 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM GAMING;
+-----+
| CID | GAMES | HOURS | GAMING_BILL |
+-----+
| C101 | 2 | 1 | 100 |
| C102 | 4 | 2 | 800 |
| C103 | 3 | 2 | 500 |
| C104 | 4 | 2 | 800 |
| C105 | 6 | 3 | 1050 |
+-----+
5 rows in set (0.01 sec)

mysql>
```

```
mysql> SELECT * FROM BOOKING_RECORD;
+-----+
| CID | CHECK_IN | CHECK_OUT |
+-----+
| C101 | 2020-01-11 | 2020-01-16 |
| C101 | 2020-01-11 | 2020-01-16 |
| C102 | 2020-01-06 | 2020-01-08 |
| C103 | 2019-12-25 | 2019-12-31 |
| C104 | 2020-01-01 | 2020-01-05 |
| C105 | 2020-02-14 | 2020-02-16 |
+-----+
6 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM ROOM_RENT;
+-----+
| CID | ROOM_CHOICE | NO_OF_DAYS | ROOMNO | ROOMRENT |
+-----+
| C101 | 1 | 6 | 420 | 60000 |
| C101 | 1 | 6 | 420 | 60000 |
| C102 | 2 | 3 | 520 | 15000 |
| C103 | 1 | 7 | 620 | 70000 |
| C104 | 1 | 5 | 820 | 50000 |
| C105 | 1 | 3 | 345 | 30000 |
+-----+
6 rows in set (0.00 sec)

mysql>
```



```
| BILL      | varchar(30) | YES |      | NULL      |      |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.99 sec)
```

```
mysql> SELECT * FROM RESTAURENT;
+-----+-----+-----+-----+
| CID | CUISINE | QUANTITY | BILL |
+-----+-----+-----+-----+
| C101 | 3       | 2         | 1500 |
| C101 | 3       | 2         | 1500 |
| C102 | 2       | 1         | 500  |
| C103 | 2       | 2         | 1000 |
| C104 | 2       | 3         | 1500 |
| C105 | 3       | 4         | 3000 |
+-----+-----+-----+-----+
6 rows in set (0.18 sec)
```

```
mysql>
```

```
| C104 | 2       | 3         | 1500 |
| C105 | 3       | 4         | 3000 |
+-----+-----+-----+-----+
6 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM TOTAL;
+-----+-----+-----+-----+-----+-----+
| CID | C_NAME          | ROOMRENT | RESTAURENTBILL | GAMINGBILL | FASHIONBILL |
+-----+-----+-----+-----+-----+-----+
| C101 | Suzy Sabestian | 60000    | 1500           | 800        | 3000        |
| C102 | Harry Potter   | 15000    | 500            | 800        | 1500        |
| C103 | Emma Watson    | 70000    | 1000           | 350        | 0           |
| C104 | Selena Gomez   | 50000    | 1500           | 100        | 6000        |
| C105 | Justin Bieber  | 30000    | 3000           | 900        | 12000       |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

```
mysql> _
```

DATABASE TABLE STRUCTURE

```
7 rows in set (0.02 sec)

mysql> DESCRIBE FASHION;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| CID   | varchar(20) | YES |  | NULL |  |
| DRESS | varchar(30) | YES |  | NULL |  |
| AMOUNT | varchar(30) | YES |  | NULL |  |
| BILL  | varchar(30) | YES |  | NULL |  |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> DESCRIBE GAMING;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| CID   | varchar(20) | YES |  | NULL |  |
| GAMES | varchar(30) | YES |  | NULL |  |
| HOURS | varchar(30) | YES |  | NULL |  |
| GAMING_BILL | varchar(30) | YES |  | NULL |  |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql>

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

mysql> USE HMS;
Database changed
mysql> DESCRIBE ROOM_RENT;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| CID   | varchar(20) | YES |  | NULL |  |
| ROOM_CHOICE | int(11) | YES |  | NULL |  |
| NO_OF_DAYS | int(11) | YES |  | NULL |  |
| ROOMNO | int(11) | YES |  | NULL |  |
| ROOMRENT | int(11) | YES |  | NULL |  |
+-----+-----+-----+-----+-----+-----+
5 rows in set (1.52 sec)

mysql> DESCRIBE RESTAURENT;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| CID   | varchar(20) | YES |  | NULL |  |
| CUISINE | varchar(30) | YES |  | NULL |  |
| QUANTITY | varchar(30) | YES |  | NULL |  |
| BILL  | varchar(30) | YES |  | NULL |  |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.10 sec)

mysql>
```



```
+-----+-----+-----+-----+-----+
4 rows in set (0.10 sec)

mysql> DESCRIBE TOTAL;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| CID        | varchar(20) | YES  |     | NULL    |       |
| C_NAME     | varchar(30) | YES  |     | NULL    |       |
| ROOMRENT   | int(11)     | YES  |     | NULL    |       |
| RESTAURENTBILL | int(11)     | YES  |     | NULL    |       |
| GAMINGBILL | int(11)     | YES  |     | NULL    |       |
| FASHIONBILL | int(11)     | YES  |     | NULL    |       |
| TOTALAMOUNT | int(11)     | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
7 rows in set (0.04 sec)

mysql>
```

Output of the Project

```
ENTER MYSQL SERVER'S USERNAME : root
ENTER MYSQL SERVER'S PASSWORD : root
CONGRATULATIONS ! YOUR MYSQL CONNECTION HAS BEEN ESTABLISHED !

1--->Enter Customer Details
2--->Booking Record
3--->Calculate Room Rent
4--->Calculate Restaurant Bill
5--->Calculate Gaming Bill
6--->Calculate Fashion store Bill
7--->Display Customer Details
8--->GENERATE TOTAL BILL AMOUNT
9--->GENERATE OLD BILL
10--->EXIT
Enter Your Choice1
Enter Customer Identification Number : C101
Enter Customer Name : Suzy Sabastian
Enter Customer Address : 45 street won
Enter Customer Age : 25
Enter Customer Country : Australia
Enter Customer Contact Number : 76789-32456
Enter Customer Email : Suzy123@gmail.com
New Customer Entered In The System Successfully !

1--->Enter Customer Details
2--->Booking Record
3--->Calculate Room Rent
4--->Calculate Restaurant Bill
```

```
File Edit Shell Debug Options Window Help
Enter Customer Address : 45 street won
Enter Customer Age : 25
Enter Customer Country : Australia
Enter Customer Contact Number : 76789-32456
Enter Customer Email : Suzy123@gmail.com
New Customer Entered In The System Successfully !

1--->Enter Customer Details
2--->Booking Record
3--->Calculate Room Rent
4--->Calculate Restaurant Bill
5--->Calculate Gaming Bill
6--->Calculate Fashion store Bill
7--->Display Customer Details
8--->GENERATE TOTAL BILL AMOUNT
9--->GENERATE OLD BILL
10--->EXIT
Enter Your Choice2
ENTER CUSTOMER ID : C101
[('C101', 'Suzy Sabastian', '45 street won', '25', 'Australia', '76789-32456', 'Suzy123@gmail.com')]

Enter Customer CheckIN Date [ YYYY-MM-DD ] : 2020-01-11
Enter Customer CheckOUT Date [ YYYY-MM-DD ] : 2020-01-16
CHECK-IN AND CHECK-OUT ENTRY MADE SUCCESSFULLY !

1--->Enter Customer Details
2--->Booking Record
3--->Calculate Room Rent
4--->Calculate Restaurant Bill
5--->Calculate Gaming Bill
6--->Calculate Fashion store Bill
7--->Display Customer Details
8--->GENERATE TOTAL BILL AMOUNT
9--->GENERATE OLD BILL
10--->EXIT
Enter Your Choice
```

```
Enter Your Choice3
ENTER CUSTOMER ID : C101
[('C101', 'Suzy Sabastian', '45 street won', '25', 'Australia', '76789-32456', '
Suzy123@gmail.com')]
```

```
##### We have The Following Rooms For You #####
```

1. Ultra Royal -----> 10000 Rs.
2. Royal -----> 5000 Rs.
3. Elite -----> 3500 Rs.
4. Budget -----> 2500 USD

```
Enter Your Option : 1
```

```
Enter Customer Room No : 420
```

```
Enter No. Of Days : 6
```

```
Ultra Royal Room Rent : 60000
```

```
Thank You , Your Room Has Been Booked For : 6 Days
```

```
Your Total Room Rent is : Rs. 60000
```

- 1--->Enter Customer Details
- 2--->Booking Record
- 3--->Calculate Room Rent
- 4--->Calculate Restaurant Bill
- 5--->Calculate Gaming Bill
- 6--->Calculate Fashion store Bill
- 7--->Display Customer Details
- 8--->GENERATE TOTAL BILL AMOUNT
- 9--->GENERATE OLD BILL
- 10--->EXIT

```
Enter Your Choice4
```

```
ENTER CUSTOMER ID : C101
```

```
[('C101', 'Suzy Sabastian', '45 street won', '25', 'Australia', '76789-32456', '
Suzy123@gmail.com')]
```

1. Vegetarian Combo -----> 300 Rs.
2. Non-Vegetarian Combo -----> 500 Rs.
3. Vegetarian & Non-Vegetarian Combo -----> 750 Rs.

```
Enter Your Cusine : 3
```

```
Enter Quantity : 2
```

```
SO YOU HAVE ORDER: Vegetarian & Non-Vegetarian Combo
```

```
Your Total Bill Amount Is : Rs. 1500
```

```
**** WE HOPE YOU WILL ENJOY YOUR MEAL ***
```

- 1--->Enter Customer Details
- 2--->Booking Record
- 3--->Calculate Room Rent
- 4--->Calculate Restaurant Bill
- 5--->Calculate Gaming Bill
- 6--->Calculate Fashion store Bill
- 7--->Display Customer Details
- 8--->GENERATE TOTAL BILL AMOUNT
- 9--->GENERATE OLD BILL
- 10--->EXIT

```
Enter Your Choice
```

```
Enter Your Choice5
ENTER CUSTOMER ID : C101
[('C101', 'Suzy Sabastian', '45 street won', '25', 'Australia', '76789-32456', '
Suzy123@gmail.com'), ('C101', 'Suzy Sabastian', '45 street won', '25', 'Australi
a', '76789-32456', 'Suzy123@gmail.com')]
```

- | | |
|------------------------|-------------------|
| 1. Table Tennis | -----> 150 Rs./HR |
| 2. Bowling | -----> 100 Rs./HR |
| 3. Snooker | -----> 250 Rs./HR |
| 4. VR World Gaming | -----> 400 Rs./HR |
| 5. Video Games | -----> 300 Rs./HR |
| 6. Swimming Pool Games | -----> 350 Rs./HR |
| 7. Exit | |

```
Enter What Game You Want To Play : 4
Enter No Of Hours You Want To Play : 2
```

```
*****
YOU HAVE SELECTED TO PLAY : VR World Gaming
Your Total Gaming Bill Is : Rs. 800
FOR : 2 HOURS
*** WE HOPE YOU WILL ENJOY YOUR GAME ***
```

```
*****
1--->Enter Customer Details
2--->Booking Record
3--->Calculate Room Rent
4--->Calculate Restaurant Bill
5--->Calculate Gaming Bill
6--->Calculate Fashion store Bill
7--->Display Customer Details
8--->GENERATE TOTAL BILL AMOUNT
9--->GENERATE OLD BILL
10--->EXIT
```

```
Enter Your Choice6
ENTER CUSTOMER ID : C101
[('C101', 'Suzy Sabastian', '45 street won', '25', 'Australia', '76789-32456', '
Suzy123@gmail.com'), ('C101', 'Suzy Sabastian', '45 street won', '25', 'Australi
a', '76789-32456', 'Suzy123@gmail.com')]
```

- | | |
|------------------|-----------------|
| 1. Shirts | -----> 1500 Rs. |
| 2. T-Shirts | -----> 300 Rs. |
| 3. Pants | -----> 2000 Rs. |
| 4. Jeans | -----> 4000 Rs. |
| 5. Tassel top | -----> 500 Rs. |
| 6. Gown | -----> 3000 Rs. |
| 7. Western dress | -----> 3000 Rs. |
| 8. Skirts | -----> 400 Rs. |
| 9. Trousers | -----> 200 Rs. |
| 10. InnerWear | -----> 30 Rs. |

```
Enter the your Choice wear: 7
How many you want to buy: 1
```

Western dress

```
*****
YOU SELECT ITEM NO : 7
YOUR QUANTITY IS : 1 ITEMS
THANK YOU FOR SHOPPING VISIT AGAIN!!!
```

Your Total Bill Is : 3000

```
*****
```

```
1--->Enter Customer Details
2--->Booking Record
3--->Calculate Room Rent
4--->Calculate Restaurant Bill
5--->Calculate Gaming Bill
6--->Calculate Fashion store Bill
7--->Display Customer Details
8--->GENERATE TOTAL BILL AMOUNT
9--->GENERATE OLD BILL
10--->EXIT
Enter Your Choice7
ENTER CUSTOMER ID : C101
[('C101', 'Suzy Sabastian', '45 street won', '25', 'Australia', '76789-32456', '
Suzy123@gmail.com'), ('C101', 'Suzy Sabastian', '45 street won', '25', 'Australi
a', '76789-32456', 'Suzy123@gmail.com')]
```

```
1--->Enter Customer Details
2--->Booking Record
3--->Calculate Room Rent
4--->Calculate Restaurant Bill
5--->Calculate Gaming Bill
6--->Calculate Fashion store Bill
7--->Display Customer Details
8--->GENERATE TOTAL BILL AMOUNT
9--->GENERATE OLD BILL
10--->EXIT
Enter Your Choice8
ENTER CUSTOMER ID : C101
[('C101', 'Suzy Sabastian', '45 street won', '25', 'Australia', '76789-32456', '
Suzy123@gmail.com'), ('C101', 'Suzy Sabastian', '45 street won', '25', 'Australi
a', '76789-32456', 'Suzy123@gmail.com')]
Enter Customer Name : Suzy Sabastian

**** CROWN PLAZA MIAMI **** CUSTOMER BILLING ****

CUSTOMER NAME : Suzy Sabastian

ROOM RENT : Rs. 60000
RESTAURENT BILL : Rs. 1500
FASHION BILL : Rs. 3000
GAMING BILL : Rs. 800
-----
TOTAL AMOUNT : Rs. 65300
```

File Edit Shell Debug Options Window Help

```
ROOM RENT      : Rs.   60000
RESTAURENT BILL : Rs.   1500
FASHION BILL   : Rs.   3000
GAMING BILL    : Rs.    800
```

```
TOTAL AMOUNT   : Rs.   65300
```

```
1--->Enter Customer Details
2--->Booking Record
3--->Calculate Room Rent
4--->Calculate Restaurant Bill
5--->Calculate Gaming Bill
6--->Calculate Fashion store Bill
7--->Display Customer Details
8--->GENERATE TOTAL BILL AMOUNT
9--->GENERATE OLD BILL
10--->EXIT
```

Enter Your Choice9

ENTER CUSTOMER ID : C101

```
[('C101', 'Suzy Sabastian', '45 street won', '25', 'Australia', '76789-32456', '
Suzy123@gmail.com'), ('C101', 'Suzy Sabastian', '45 street won', '25', 'Australi
a', '76789-32456', 'Suzy123@gmail.com')]
[('C101', 'Suzy Sabastian', 60000, 1500, 800, 3000, 65300)]
```

ADVANTAGES AND DISADVANTAGES:

ADVANTAGES:

1. This project has made it easy for a hotel to manage and record the details of their customers.
2. It has also helped the management keep in track the amenities used by the customers.
3. The room rent for customers who have stayed in the hotel has been calculated.
4. Bills have been calculated separately for each department of the hotel. Like bill for gaming place, restaurant, fashion store has been calculated separately
5. By this categorizing the hotel management would be able to give their customers a detailed bill.
6. The management would have a rough idea on the services which they have provided to their customers and such detailed bills would avoid any further confusions.

DISADVANTAGES:

1. This project has not been able to record the details of the employees working in the hotel
2. It has been successful only in customer management for a hotel
3. The users of this project are only the hotel management people.
4. It has not been able to directly serve the customers of the hotel i.e. online booking, ordering food or accessing the amenities of the hotel just by staying at the room.
5. It has not been able to track the current status of the customer in the hotel
6. This project been useful only for recording data of the customers who are staying in the hotel but not for the people who just dine in.

CONCLUSION:

The project has enlightened us on the most important part of computer science, the practical and applicative part of it.

This project is designed to improve operational efficiency and is the fastest and reliable way to bill customers that saves both time and money. This software generates invoice quickly and hence it is easy to use and is adaptable to different formats.

The ability of putting together a project with the fundamental knowledge of MySQL and Python truly delights us.

We have learnt a lot from this project,

- The importance of removing bugs from a program
- The importance of following correct syntax.

- Learnt to make a user friendly system

- Maintaining a minimum margin of error

- Understanding the criteria of implementation process

Most importantly we understood how delightful it is to come Up with and idea and implement it.

FURTHER DEVELOPMENTAL AREAS

- The list has fixed prices and services, this can be developed further for adding new items and changing prices.
- Extracting important data such as goods and service tax rates, service tax rates etc. from government websites on the internet
- Using logical algorithms to adjust prices for inflation.
- Creating a user data base for storing information such as past purchases, discount coupons, reservations in hotel etc.
- Creating a management system for salary of the employees working at the hotel.
- Can be developed for tracking available rooms for customers.
- Can be used for tracking orders of all necessities which were bought by the hotel.
- Can be developed to include online booking systems for customers to book the room at home

BIBLIOGRAPHY:

- www.python.org
- www.tutorialpoint.com
- www.pythonchallenge.com
- www.learnpython.org