Bhilai Institute of Technology, Durg

Department of Computer Science & Engg.

Experiment Index

PYTHON WITH DJANGO LAB 102393CS

Student's Name	Branch & Semester	Class Roll No.
RITESH CHATURVEDI	CSE 3rd SEMESTER	A27

Experiment No.	Date	Aim of Experiment	Page No.	Signature	Remarks
9.1	14/11/22	WAP to demonstrate CRUD operations.	99		
9.2	14/11/22	WAP to read student's name, five subject names and their corresponding marks and store it in a database table for 5 students. Display that table contents.	103		
9.3	14/11/22	WAP to read student's name, five subject names and their corresponding marks and store it in a database table for 5 students. Display subject names with maximum and minimum marks of every student.	108		
9.4	14/11/22	WAP to create a Login validator. Use database table to hold all users and their passwords. The existing users should be able to login by entering correct username and password.	113		
9.5	14/11/22	WAP to create a Login validator. Use database table to hold all user names, their passwords and a secret key. The existing users should be able to login by entering correct username and password. Also the existing users can view and update their passwords by entering the right secret key .	117		

Bhilai Institute of Technology, Durg

Department of Computer Science & Engg.

Experiment Index

PYTHON WITH DJANGO LAB 102393CS

Student's Name	Branch & Semester	Class Roll No.
RAJEEV MANDLE	CSE 3rd SEMESTER	A27

9.6	14/11/22	WAP to build a database table to hold Name, Dept, Salary, DA and Gross of 5 employees. i. Input the Name, Dept and Salary details from the user. ii. Calculate DA as 20% of the Salary iii. Gross = Salary + DA iv. Display all the contents of the database table.	122		
9.7	14/11/22	WAP to build a database table to hold Name, Dept, Salary, DA and Gross of 5 employees. i. Input the Name, Dept and Salary details from the user. ii. Calculate DA as 20% of the Salary iii. Gross = Salary + DA. iv. Search any employee and display its details.	126		
9.8	14/11/22	WAP to build a database table to hold Name, Dept, AggMarks, AggPer and Div for 5 students. i. Input the Name and Dept details from the user. ii. Input marks of 5 subjects (out of 100) and store the aggregate in AggMarks. iii. Calculate Aggregate percentage out of 500 and store in AggMarks. iv. Display all the contents	130		

Bhilai Institute of Technology, Durg

Department of Computer Science & Engg.

Experiment Index

PYTHON WITH DJANGO LAB 102393CS

Student's Name	Branch & Semester	Class Roll No.
RAJEEV MANDLE	CSE 3rd SEMESTER	A27

9.9	14/11/22	WAP to build a database table to hold Name, Email and Address of 5 customers. i. Input the Name, Email and Address. ii. Provide facility to search any customer data using Email as the key. iii. Provide facility to update any customer address using Email as the key. iv. Provide facility to delete any customer data using Email as the key. v. Display all the contents	134		
9.10	14/11/22	Write a program to build a Student Information System. Use database table to store the data. The data base should contain the Enrollment Number, Student Name, Branch, Semester, Marks of 5 subjects. Give the facility to perform add, delete, search, edit and View All procedures on the database.	139		
9.11	14/11/22	Create a database of 5 Customers in a Bank. Every customer has Account number, Name, Address of branch and Balance amount. Use database table to store the data. Provide following options to the user: a. Search any customer by Account number and display its details b. Edit any customer's address c. Deposit/Withdraw amount from the account. d. Display data of all the customers	147		

Bhilai Institute of Technology, Durg.

Department of Computer Science and Engineering

PYTHON WITH DJANGO LAB 102393CS **CRN: A27**

Date: 21/11/2022

Experiment No: 9.1 Aim:

WAP to demonstrate CRUD operations.

Code:

```
def creat_database():
    import mysql.connector mydb =
    mysql.connector.connect(host="localhost",user="root",password="")
    mycursor = mydb.cursor()
    data_base = str(input("Enter the name of data base:"))
    querry ="CREATE DATABASE IF NOT EXISTS %s" %(data_base)       # give data_base =
student_database (same as it , required for further execution) mycursor.execute(querry)


def creating_tabel():
    import mysql.connector mydb
    =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") # give table_name = student_table_2 (same as it , required for further execution)
mycursor = mydb.cursor()
    table_name = str(input("Enter the name of the table"))
    querry = "CREATE TABLE IF NOT EXISTS %s (Student_Name VARCHAR(255), Subject_1
int(5), Subject_2 int(5), Subject_3 int(5),Subject_4 int(5),Subject_5
int(5))"%(table_name)
    mycursor.execute(querry)


def data_insertion():
```

Page

```
    import  mysql.connector
    import numpy as np mydb
    =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") mycursor = mydb.cursor() Name = input("Enter your Name") marks = [] print("Enter
your 5 Subjects marks") for i in range (0,5): a = input() marks.append(a)
    Subject_1 = marks[0] Subject_2
    = marks[1] Subject_3 =
    marks[2]
    Subject_4 = marks[3]
    Subject_5 = marks[4] np.sort(marks)
    querry = "INSERT INTO student_table_2 VALUES ('{}' , '{}' , '{}' , '{}' , '{}' ,
'{}')".format(Name , Subject_1 , Subject_2 , Subject_3 , Subject_4 , Subject_5 )
    mycursor.execute(querry) mydb.commit()
    print("Data has been INSERTED")


def fetch_data():
    import mysql.connector
    mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") mycursor = mydb.cursor()
```

```
    mycursor.execute("SELECT * FROM student_table_2")
    data = mycursor.fetchall()
    for x in data:
        print(x)
    print("Data has been FETCHED OUT")

def Execution_crud_operation():
    print("""Enter the number against FUNCTION
    1.Creat Database
    2.Creat Table
    3.Insert   Data   4.Fetch
    Data""")   user_input   =
    int(input())           if
    user_input      ==      1:
    creat_database()
    elif user_input == 2:
        creating_tabel()
    elif user_input == 3:
        data_insertion()
    elif user_input == 4:
        fetch_data()
    else:
        pass

# Driver's Code
Execution_crud_operation()
```

Output:

```
Enter the number against FUNCTION
        1.Creat Database
        2.Creat     Table
        3.Insert     Data
        4.Fetch Data
3
Enter your Name Utkrash ThakurEnter your
5 Subjects marks 32
31
36
35
33
Data has been INSERTED


Enter the number against FUNCTION
   1.Creat Database
   2.Creat
   Table 3.Insert
   Data 4.Fetch
   Data
4
('Asmi', 34, 33, 31, 37, 38)
('Utkrash Thakur', 32, 31, 36, 35, 33)
```

Date: 21/11/2022

Experiment No: 9.2 Aim:

WAP to read student's name, five subject names and their corresponding marks andstore it in a database table for 5 students. Display that table contents.

Code:

```
def creat_database():
    import mysql.connector mydb =
    mysql.connector.connect(host="localhost",user="root",password="")
    mycursor = mydb.cursor()
    data_base = str(input("Enter the name of data base:")) # give data_base =
student_database (same as it , required for further execution) querry
    ="CREATE DATABASE IF NOT EXISTS %s" %(data_base)
    mycursor.execute(querry)


def creating_tabel():
    import mysql.connector mydb
    =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") mycursor = mydb.cursor()
    table_name = str(input("Enter the name of Table:")) # give
table_name = student_table (same as it , required for further execution)
    querry = "CREATE TABLE IF NOT EXISTS %s (Student_ID int(5) ,Student_Name
VARCHAR(255), Subject_1 int(5), Subject_2 int(5), Subject_3 int(5),Subject_4
int(5),Subject_5 int(5),Maximum_Marks int(5), Minimum_Marks int(5))"%(table_name)
```

```
    mycursor.execute(querry)


def      data_insertion():
    import  mysql.connector
    import numpy as np mydb
    =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") mycursor = mydb.cursor()
    querry_1 = ("SELECT Student_ID FROM student_table")
    mycursor.execute(querry_1) data =
    mycursor.fetchall() id_list = []
    for x in data:
        a = list(x) id_list.append((a))
    id = id_list[-1][0] + 1 print(id_list
    , id)
    Name = input("Enter your Name")
    marks = [] print("Enter your 5
    Subjects marks") for i in range
    (0,5): a = input() marks.append(a)
    Subject_1 = marks[0]
    Subject_2 = marks[1] Subject_3
    = marks[2] Subject_4 =
    marks[3]
    Subject_5 = marks[4]
```

```
np.sort(marks) max_marks
```

```
    = marks[0] min_marks =

    marks[4]

    querry = "INSERT INTO student_table VALUES ('{}' , '{}' , '{}' , '{}' , '{}' , '{}'
, '{}' , '{}' , '{}')".format(id , Name , Subject_1 , Subject_2 , Subject_3 , Subject_4
, Subject_5 , max_marks , min_marks )

    mycursor.execute(querry) mydb.commit()

    print("Data has been Inserted")


def   update   ():   import

    mysql.connector

    mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") mycursor = mydb.cursor()

    id = input("Enter the ID od the Student:") Name

    = input("Enter the name of the student:")

    sql = "UPDATE student_table SET Student_ID = '{}' WHERE Student_Name =
'{}'".format(id , Name)

    mycursor.execute(sql) mydb.commit()

    print("Data has been Updated")


def    delete():    import

    mysql.connector

    mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") mycursor = mydb.cursor()

    id = input("Enter the ID od the Student:")
```

```
    queery = "DELETE FROM student_table WHERE Student_ID = '{}'".format(id)
    mycursor.execute(queery) mydb.commit()
    print("Data has been Deleted")


def fetch_data():
    import mysql.connector
    mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") mycursor = mydb.cursor()
    mycursor.execute("SELECT * FROM student_table")
    data = mycursor.fetchall() for x in data:
        print(x)


def Execution_crud_operation():
    print("""Enter the number against FUNCTION
    1. Creat Database 2.Creat Table 3.Insert Data 4.Update Data 5.Delete
Data     6.Fetch     Data""")
    user_input = int(input())
    if   user_input   ==   1:
    creat_database()
    elif user_input == 2: creating_tabel()
    elif user_input == 3: data_insertion()
    elif user_input == 4:
        update ()
```

```
    elif user_input == 5:
        delete()
    elif user_input == 6:
        fetch_data()
    else:
        pass


# Driver's Code
Execution_crud_operation()
```

Output:

```
Enter the number against FUNCTION
        1.Creat Database
        2.Creat     Table
        3.Insert     Data
        4.Fetch Data
3
Enter your Name Utkrash ThakurEnter your
5 Subjects marks 32
31
36
35
33
Data has been INSERTED

Enter the number against FUNCTION
   1.Creat Database
   2.Creat
   Table 3.Insert
   Data 4.Fetch
   Data
4
('Asmi', 34, 33, 31, 37, 38)
('Utkrash Thakur', 32, 31, 36, 35, 33)
```

Date: 21/11/2022

Experiment No: 9.3 Aim:

WAP to read student's name, five subject names and their corresponding marks and store it in a database table for 5 students. Display subject names with maximum andminimum marks of every student.

Code:

```
def creat_database():
    import mysql.connector mydb =
    mysql.connector.connect(host="localhost",user="root",password="")
    mycursor = mydb.cursor()
    data_base = str(input("Enter the name of data base:")) # give data_base =
student_database (same as it , required for further execution) querry
    ="CREATE DATABASE IF NOT EXISTS %s" %(data_base)
    mycursor.execute(querry)


def creating_tabel():
    import mysql.connector mydb
    =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") mycursor = mydb.cursor()
    table_name = str(input("Enter the name of the table:")) # give table_name =
student_table (same as it , required for further execution)
    querry="CREATE TABLE %s (Student_ID int(5) ,Student_Name VARCHAR(255), Subject_1
int(5), Subject_2 int(5), Subject_3 int(5),Subject_4 int(5),Subject_5
int(5),Maximum_Marks int(5), Minimum_Marks int(5))"(table_name)
```

```
112| Pmycursor.execute(querry)
```

```
def      data_insertion():
    import  mysql.connector
    import numpy as np mydb
    =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") mycursor = mydb.cursor()
    id = 6   #static data , updatee mannulay ever time
    Name = input("Enter your Name") marks = []
    print("Enter your 5 Subjects marks")
    for i in range (0,5): a = input()
        marks.append(a)
    Subject_1 = marks[0]
    Subject_2 = marks[1] Subject_3
    = marks[2]
    Subject_4   =   marks[3]
    Subject_5   =   marks[4]
    marks = np.sort(marks)
    max_marks =marks[-1] min_marks
    = marks[0]
    querry = "INSERT INTO student_table VALUES ('{}' , '{}' , '{}' , '{}' , '{}' , '{}'
, '{}' , '{}' , '{}')".format(id , Name , Subject_1 , Subject_2 , Subject_3 , Subject_4
, Subject_5 , max_marks , min_marks )
    mycursor.execute(querry) mydb.commit()
```

Bhilai Institute of Technology, Durg.

Department of Computer Science and Engineering

PYTHON WITH DJANGO LAB 102393CS **CRN: A27**


```
def update():  import
```

```
    mysql.connector
    mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") mycursor = mydb.cursor() id = input("Enter the ID od the Student:") Name =
input("Enter the name of the student:")
    sql = "UPDATE student_table SET Student_ID = '{}' WHERE Student_Name =
'{}'".format(id , Name)
    mycursor.execute(sql) mydb.commit()


def    delete():    import
    mysql.connector
    mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") mycursor = mydb.cursor() id = input("Enter the ID od the Student:") queery =
"DELETE FROM student_table WHERE Student_ID = '{}'".format(id) mycursor.execute(queery)
mydb.commit()


def fetch_data():
    import mysql.connector
    mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") mycursor = mydb.cursor()
    mycursor.execute("SELECT * FROM student_table")
```

```
    data = mycursor.fetchall()
    for x in data:
        print(x)

def Execution_crud_operation():
    print("""Enter the number against FUNCTION
    1.Creat Database
    2. Creat Table
    3.Insert Data
    4.Update Data
    5.Delete   Data   6.Fetch
    Data""")   user_input   =
    int(input())           if
    user_input     ==     1:
    creat_database()
    elif user_input == 2:
        creating_tabel()
    elif user_input == 3:
        data_insertion()
    elif user_input == 4:
        update ()
    elif user_input == 5:
        delete()
    elif user_input == 6:
        fetch_data()
    else: pass
```

```
# Driver's Code
Execution_crud_operation()
```

Output:

```
Enter the number against FUNCTION
        1.Creat Database
        2.Creat     Table
        3.Insert     Data
        4.Fetch Data
3
Enter your Name Utkrash ThakurEnter your
5 Subjects marks 32
31
36
35
33
Data has been INSERTED


Enter the number against FUNCTION
   1.Creat Database
   2.Creat
   Table 3.Insert
   Data 4.Fetch
   Data
4
('Asmi', 34, 33, 31, 37, 38)
('Utkrash Thakur', 32, 31, 36, 35, 33)
```

Date: 21/11/2022

Experiment No: 9.4 Aim:

WAP to create a Login validator. Use database table to hold all users and their passwords. The existing users should be able to login by entering correct username and password.

Code:

```
def creat_database():
    import mysql.connector mydb =
    mysql.connector.connect(host="localhost",user="root",password="")
    mycursor = mydb.cursor()
    data_base = str(input("Enter the name of data base:")) # give data_base =
student_database (same as it , required for further execution) querry
    ="CREATE DATABASE IF NOT EXISTS %s" %(data_base)
    mycursor.execute(querry)


def data_insert():
    import mysql.connector mydb
    =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") mycursor = mydb.cursor()
    Name = input("Enter your Name")
    Date_of_birth = input("Enter your Date of birth")
    Password = input("Enter your Password")
    Confirm_Password = input("Enter your Confirm Password")
```

```
    querry = "INSERT INTO data_verfication VALUES ('{}' , '{}' , '{}' , '{}')".format(
Name , Date_of_birth , Password , Confirm_Password )
    if Password == Confirm_Password:
        mycursor.execute(querry)
        mydb.commit()
        print("Insertion of entered DATA is Settled")
    else:
        print("Someting went !!! Try AGAIN")

    def creating_tabel():
          import mysql.connector
          mydb = mysql.connector.connect(host="localhost",user="root",
    password="",database="student_database")
          mycursor = mydb.cursor()
          table_name = str(input("Enter the name of the table:"))
    # give table_name = data_verfication (same as it , required for
    further execution)
          querry="CREATE TABLE IF NOT EXISTS %s (Name VARCHAR(225) , Date_of_Birth
    VARCHAR(20) , Password VARCHAR(20) , Confirm_Password VARCHAR(20))"
          %(table_name)
          mycursor.execute(querry)


def Person_Verfication():
    import mysql.connector
    mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base")
    mycursor = mydb.cursor()
    name = input("Enter your name")
    user_passowrd = input("Enter your password:")
    querry = "SELECT * FROM data_verfication WHERE Name = '{}'".format(name)
    mycursor.execute(querry)
    data = mycursor.fetchall()
```

```
user_data = []
for x in data:
    user_data.append(x)
name_fetched = (user_data[0][0])
password_fetched = (user_data[0][2])
if (password_fetched == user_passowrd and name == name_fetched):
    print("You are logged in,Your data is ")
    print(user_data)
else:
   print("You and your password SUCKS")
def Execution_crud_operation():
    print("""Enter the number against FUNCTION
    1. Data Insert
    2. Craeting Table
    3. Person Verfication""")
    user_input = int(input())
    if user_input == 1:
        data_insert()
    elif user_input == 2:
        creating_tabel()
    elif user_input == 3:
        Person_Verfication()
    else:
        pass
# Driver's Code
Execution_crud_operation()
```

Output:

```
Enter the number against FUNCTION
    1. Data Insert
    2. Person Verfication
1
Enter your Name Shubham
Enter your Date of birth 23 Sep 2003Enter your
Password @123
Enter your Confirm Password @123 Insertion of entered
DATA is Settled

Enter the number against FUNCTION
    1. Data Insert
    2. Person Verfication
2
Enter your name Shubham Enter your
password:@123
You are logged in,Your data is [('Shubham', '23 Sep 2003', '@123',
'@123')]
```

Bhilai Institute of Technology, Durg.

Department of Computer Science and Engineering

PYTHON WITH DJANGO LAB 102393CS **CRN: A27**

Date: 21/11/2022

Experiment No: 9.5 Aim:

WAP to create a Login validator. Use database table to hold all user names, their passwords and a secret key. The existing users should be able to login by entering correct username and password. Also the existing users can view and update theirpasswords by entering the right secret key .

Code:

```
def creat_database():
    import mysql.connector
    mydb = mysql.connector.connect(host="localhost",user="root",password="")
    mycursor = mydb.cursor()
    data_base = str(input("Enter the name of data base:"))          # give data_base =
student_database (same as it , required for further execution)
    querry ="CREATE DATABASE IF NOT EXISTS %s" %(data_base)
    mycursor.execute(querry)


def data_insert():
    import mysql.connector
    mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base")
    mycursor = mydb.cursor()
    Name = input("Enter your Name")
    Date_of_birth = input("Enter your Date of birth")
    Password = input("Enter your Password")
    Confirm_Password = input("Enter your Confirm Password")
```

```
    querry = "INSERT INTO data_verfication VALUES ('{}' , '{}' , '{}' , '{}')".format(
Name , Date_of_birth , Password , Confirm_Password )
    if Password == Confirm_Password:
        mycursor.execute(querry) mydb.commit()
        print("Insertion of entered DATA is Settled")
    else: print("Someting went !!! Try
        AGAIN")

def   update   ():   import
    mysql.connector mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") mycursor = mydb.cursor() new_password = input("Enter your password:")
    sql_1 = "UPDATE data_verfication SET Password = '{}'".format(new_password) sql_2 =
    "UPDATE data_verfication SET Confirm_Password = '{}'".format(new_password)
    mycursor.execute(sql_1) mycursor.execute(sql_2)
    mydb.commit()

def Person_Verfication():
    import  mysql.connector  mydb
    =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") mycursor = mydb.cursor() name = input("Enter your name")
    user_passowrd = input("Enter your password:")
```

```
    querry = "SELECT * FROM data_verfication WHERE Name =
    '{}'".format(name) mycursor.execute(querry) data = mycursor.fetchall()
    user_data = []
    for x in data:
        user_data.append(x)
    name_fetched = (user_data[0][0]) password_fetched
    = (user_data[0][2])
    if (password_fetched == user_passowrd and name == name_fetched):
        print("Your are logged in")
        print("""Enter 1 for knowing your data or enter 2 to Set new password""")
        user_input = int(input())
        if user_input == 1:
            print("Your data is")
            print(user_data)
        elif user_input == 2:
            update()
            print("Password Updated Sucessfully")
    else:
        print("You and your password SUCKS")


def Execution_crud_operation():
    print("""Enter the number against FUNCTION
    1.Data Insert
    2.Person Verfication""") user_input = int(input())
    if user_input == 1: data_insert()
```

```
    elif user_input == 2:
        Person_Verfication()
    else:
        pass


# Driver's Code
Execution_crud_operation()
```

Output:

```
Enter the number against FUNCTION
    1. Data Insert
    2. Person Verfication
1
Enter your Name RITESH CHATURVEDIEnter your Date
of birth23 Sep 2003Enter your Password @NYC Enter
your Confirm Password @NYCInsertion of entered
DATA is

Enter the number against FUNCTION
    1. Data Insert
    2. Person Verfication
2
Enter your nameShubham AnandEnter your
password:@NYC You are logged in
Enter 1 for knowing your data or enter 2 to Set new password 2
Enter your password:@LA Password Updated
Sucessfully
Enter your name Shubham Enter your
password:@123
```

```
You are logged in,Your data is [('Shubham', '23 Sep 2003', '@123', '@123')]
```

Date: 21/11/2022

Experiment No: 9.6 Aim:

WAP to build a database table to hold Name, Dept, Salary, DA and Gross of 5 employees.
i. Input the Name, Dept and Salary details from the user.
ii. Calculate DA as 20% of the Salary iii. Gross = Salary + DA iv. Display all the contents of the database table.

Code:

```
def creat_database():
    import mysql.connector mydb =
    mysql.connector.connect(host="localhost",user="root",password="")
    mycursor = mydb.cursor()
    data_base = str(input("Enter the name of data base:"))     # give data_base =
employee_database (same as it , required for further execution) querry
    ="CREATE DATABASE IF NOT EXISTS %s" %(data_base)
    mycursor.execute(querry)


#creating table def
creat_tabel():
    import mysql.connector
    mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="employee_dat
abase") mycursor = mydb.cursor()
    table_name = str(input("Enter the name of the table:"))      # give table_name =
employee_data_table (same as it , required for further execution)
```

```
    querry = "CREATE TABLE IF NOT EXISTS %s (Employee_Name VARCHAR(225)
,Employee_Department VARCHAR(255), Salary INT(10), DA INT(5), Gross
INT(5))"%(table_name) mycursor.execute(querry)


# creat_tabel()


# data insertion
def data_insertion():
    import mysql.connector
    mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="employee_dat
abase") mycursor = mydb.cursor()
    Name = input("Enter your Name")
    Deapartment = input("Enter your Department")
    Salary = int(input("Enter your Salary"))
    DA= int((Salary * 0.2)) Gross
    = Salary + DA
    querry = "INSERT INTO employee_data_table VALUES ('{}' , '{}' , '{}' , '{}' , '{}'
)".format(Name ,Deapartment , Salary , DA , Gross )
    mycursor.execute(querry) mydb.commit()
# data_insertion()


#data fetachtion from row
def fetch_data():
    import mysql.connector
```

```
    mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="employee_dat
abase") mycursor = mydb.cursor() querry = "SELECT * FROM employee_data_table"
mycursor.execute(querry) data = mycursor.fetchall()
    for   x   in   data:
print(x)              #
fetch_data_employee()


def Execution_crud_operation():
    print("""Enter the number against FUNCTION
    1.Creat Database 2.Creat Table 3.Insert Data 4.Fetch Data of Employee""") user_input
    = int(input())
    if user_input == 1: creat_database()
    elif user_input == 2: creat_tabel()
    elif user_input == 3: data_insertion()
    elif user_input == 4:
        fetch_data()
    else:
        pass

Execution_crud_operation()
```

Output:

```
Enter the number against FUNCTION
1.Creat Database      2.Creat Table      3.Insert Data
4. Featch ALL Data From DataBase          5.Fetch Data of Employee
3
Enter your Name RITESH CHATURVEDI
Enter your Department Human ResourceEnter your Salary60000

Enter the number against FUNCTION
    1. Creat Database          2.Creat Table 3.Insert Data 4.Featch ALL Data From DataBase
       5.Fetch Data of Employee5
Enter the name of Employee, u looking for:Shubham Anand('Shubham Anand', Human Resource,
60000, 12000, 72000)
```

Date: 21/11/2022

Experiment No: 9.7 Aim:

WAP to build a database table to hold Name, Dept, Salary, DA and Gross of 5 employees.
i)Input the Name, Dept and Salary details from the user.
ii)Calculate DA as 20% of the Salary iii)Gross = Salary + DA. iv)Search any employee and display its details.

Code:

```
def creat_database():
    import mysql.connector
    mydb = mysql.connector.connect(host="localhost",user="root",password="") mycursor
    = mydb.cursor()
    data_base = str(input("Enter the name of data base:"))      # give data_base =
employee_database (same as it , required for further execution) querry
    ="CREATE DATABASE IF NOT EXISTS %s" %(data_base)
    mycursor.execute(querry)


#creating table def
creat_tabel():
    import mysql.connector
    mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="employee_dat
abase") mycursor = mydb.cursor()
    table_name = str(input("Enter the name of the table:"))      # give table_name =
employee_data_table (same as it , required for further execution)
```

```
    querry = "CREATE TABLE %s (Employee_Name VARCHAR(225) ,Employee_Department
VARCHAR(255), Salary INT(10), DA INT(5), Gross INT(5))"%(table_name)
    mycursor.execute(querry)
# creat_tabel()


# data insertion
def data_insertion():
    import mysql.connector mydb
    =
mysql.connector.connect(host="localhost",user="root",password="",database="employee_dat
abase") mycursor = mydb.cursor()
    Name = input("Enter your Name")
    Deapartment = input("Enter your Department")
    Salary = int(input("Enter your Salary"))
    DA= int((Salary * 0.2)) Gross
    = Salary + DA
    querry = "INSERT INTO employee_data_table VALUES ('{}' , '{}' , '{}' , '{}' , '{}'
)".format(Name ,Deapartment , Salary , DA , Gross ) mycursor.execute(querry)
    mydb.commit()
# data_insertion()


#data fetachtion from row def
fetch_data_employee():
    import mysql.connector
    mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="employee_dat
abase") mycursor = mydb.cursor()
```

```
    Name = input("Enter the name of Employee, u looking for:")
    querry = "SELECT * FROM employee_data_table WHERE Employee_Name
    =
'{}'".format(Name)
    mycursor.execute(querry)
    data = mycursor.fetchall()
    for   x   in   data:
print(x)              #
fetch_data_employee()

def Execution_crud_operation():
    print("""Enter the number against FUNCTION
    1.Creat Database 2.Creat Table 3.Insert Data 4.Fetch Data of Employee""")
    user_input = int(input())
    if user_input == 1:
        creat_database()
    elif user_input == 2:
        creat_tabel()
    elif user_input == 3:
        data_insertion()
    elif user_input == 4:
        fetch_data_employee()
    else:
        pass

Execution_crud_operation()
```

Output:

```
 Enter the number against FUNCTION
1.Creat Database       2.Creat Table         3.Insert Data
5. Featch ALL Data From DataBase                    5.Fetch Data of Employee
 3
 Enter your Name RITESH CHATURVEDI
 Enter your Department Human ResourceEnter your Salary60000

 Enter the number against FUNCTION
       1. Creat Database            2.Creat Table 3.Insert Data 4.Featch ALL Data From DataBase
          5.Fetch Data of Employee5
 Enter the name of Employee, u looking for:Shubham Anand('Shubham Anand', Human Resource,
 60000, 12000, 72000)
```

Date: 21/11/2022

Experiment No: 9.8 Aim:

WAP to build a database table to hold Name, Dept, AggMarks, AggPer and Div for 5students.

Code:

```
def creat_database():
    import mysql.connector
    mydb = mysql.connector.connect(host="localhost",user="root",password="")
    mycursor = mydb.cursor() data_base = str(input("Enter the name of data
    base:"))
    querry ="CREATE DATABASE IF NOT EXISTS %s" %(data_base)          # give data_base =
student_database (same as it , required for further execution)
    mycursor.execute(querry)
# creat_database()
def creating_tabel():
    import mysql.connector mydb
    =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") mycursor = mydb.cursor()
    table_name = str(input("Enter the name of the table:"))      # give table_name =
student_table_result (same as it , required for further execution) querry
    = "CREATE TABLE student_table_result (Student_Name
VARCHAR(255),Student_Depart VARCHAR(225), Subject_1 FLOAT(5), Subject_2 FLOAT(5),
Subject_3 FLOAT(5),Subject_4 FLOAT(5),Subject_5 FLOAT(5), Aggregated_Marks FLOAT(5),
Aggregated_Percentage FLOAT(5))"%s(table_name)
```

```
    mycursor.execute(querry)


# creating_tabel()


def data_insertion():
    import mysql.connector mydb
    =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") mycursor = mydb.cursor()
    Name = input("Enter your Name:")
    Department = input("Enter your Department:")
    marks = [] print("Enter your 5 Subjects
    marks") for i in range (0,5): a =
    int(input()) marks.append(a)
    Subject_1 = marks[0]
    Subject_2 = marks[1]
    Subject_3 = marks[2] Subject_4
    = marks[3]
    Subject_5 = marks[4]
    Aggregated_Marks = sum(marks)
    Aggregated_Percentage = (Aggregated_Marks / 500) * 100
    querry = "INSERT INTO student_table_result VALUES ('{}' , '{}' , '{}' , '{}' , '{}'
, '{}' , '{}' , '{}' , '{}')".format( Name , Department, Subject_1 , Subject_2 ,
Subject_3 , Subject_4 , Subject_5 , Aggregated_Marks , Aggregated_Percentage )
    mycursor.execute(querry) mydb.commit()
```

```
# data_insertion()


def fetch_data():
    import mysql.connector
    mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") mycursor = mydb.cursor()
    querry = "SELECT * FROM student_table_result"
mycursor.execute(querry) data =
mycursor.fetchall() for x in data: print(x) #
fetch_data()


def Execution_crud_operation():
    print("""Enter the number against FUNCTION
    1.Creat Database 2.Creat Table 3.Insert Student 4.Fetch Data """)
    user_input = int(input()) if user_input == 1: creat_database()
    elif user_input == 2: creating_tabel()
    elif user_input == 3: data_insertion()
    elif user_input == 4:
        fetch_data() else:
```

```
        pass


Execution_crud_operation()
```

Output:

```
Enter the number against FUNCTION
1. Creat Database    2.Creat Table    3.Insert Data    4.Fetch Data
3
Enter your Name:Saurabh
Enter your Department:CSE
Enter your 5 Subjects marks73
66
78
65
80

Enter the number against FUNCTION
      1. Creat Database          2.Creat Table          3.Insert Student 4.Fetch Data
4
('Saurabh', 'CSE', 73.0, 66.0, 78.0, 65.0, 80.0, 362.0, 72.4)
```

Date: 21/11/2022

Experiment No: 9.9 Aim:

WAP to build a database table to hold Name, Dept, AggMarks, AggPer and Div for 5students.

Code:

```
def creat_database():
    import mysql.connector
    mydb = mysql.connector.connect(host="localhost",user="root",password="") mycursor
    = mydb.cursor()
    data_base = str(input("Enter the name of data base:"))           # give data_base =
custember_database (same as it , required for further execution) querry
    ="CREATE DATABASE IF NOT EXISTS %s" %(data_base)
    mycursor.execute(querry)


# creat_database()


def creating_tabel():
    import mysql.connector mydb
    =
mysql.connector.connect(host="localhost",user="root",password="",database="custember_da
tabase") mycursor = mydb.cursor()
    table_name = str(input("Enter the name of the table:"))      # give table_name =
custember_data_table (same as it , required for further execution)
```

```
    querry = "CREATE TABLE %s (Customber_Name VARCHAR(225) ,Customber_Email
VARCHAR(225), Customber_Address VARCHAR(225) , PRIMARY KEY
(Customber_Email))"%(table_name) mycursor.execute(querry)


# creating_tabel()


def data_insertion():
    import mysql.connector mydb
    =
mysql.connector.connect(host="localhost",user="root",password="",database="customber_da
tabase") mycursor = mydb.cursor()
    Name = input("Enter your Name")
    Email = input("Enter your Email") Address
    = input("Enter your Address")
    querry = "INSERT INTO customber_data_table VALUES ('{}' , '{}' , '{}')".format(Name
, Email , Address) mycursor.execute(querry)
    mydb.commit() #


data_insertion()


def   update   ():   import
    mysql.connector mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="customber_da
tabase") mycursor = mydb.cursor()
    Email = input("Enter your Email:")
```

```
    New_address = input("Enter your New Address:")
    queery = "UPDATE customber_data_table SET Customber_Address = '{}' WHERE
Customber_Email = '{}'".format(New_address,Email) mycursor.execute(queery)
    mydb.commit() #


update()


def    delete():    import
    mysql.connector
    mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="customber_da
tabase") mycursor = mydb.cursor()
    Email = input("Enter your Email:")
    queery = "DELETE FROM customber_data_table WHERE Customber_Email =
'{}'".format(Email)
    mycursor.execute(queery) mydb.commit()
    print("Customber has been succesfully deleted ")


# delete()


def fetch_data():
    import  mysql.connector  mydb
    =
mysql.connector.connect(host="localhost",user="root",password="",database="customber_da
tabase") mycursor = mydb.cursor() queery = "SELECT * FROM customber_data_table"
```

```
    mycursor.execute(queery)                                    4.Update
    data = mycursor.fetchall()
    for x in data:

print(x)         #

fetch_data()



def Execution_crud_operation():
    print("""Enter the number against FUNCTION
    1. Creat Database 2.Creat Table 3.Insert Custember
Address  5.Delete  Custember  6.Fetch  Data  """)
    user_input = int(input()) if user_input == 1:
    creat_database()
    elif user_input == 2:
        creating_tabel()
    elif user_input == 3:
        data_insertion()
    elif user_input == 4:
        update()
    elif user_input == 5:
        delete()
    elif user_input == 6:
        fetch_data()
    else:
        pass
Execution_crud_operation()
```

Output:

```
Enter the number against FUNCTION
       1. Creat Database 2.Creat Table 3.Insert Custember
          4.UpdateAddress      5.Delete Custember
          6.Fetch Data
3
Enter your NameRITESH CHATURVEDI
Enter your
Emailshubham@gmail.comEnter
your AddressRaipur

Enter the number against FUNCTION
     1.Creat Database          2.Creat Table          3.Insert Custember
            4.UpdateAddress 5.Delete Custember 6.Fetch Data
4
Enter your
Email:shubham@gmail.comEnter your
New Address:Durg

Enter the number against FUNCTION
     1.Creat Database          2.Creat Table          3.Insert Custember
            4.UpdateAddress 5.Delete Custember 6.Fetch Data
5
Enter your Email:nikhil@gmail.com
Customer has been successfully deleted

Enter the number against FUNCTION
     1.Creat Database          2.Creat Table          3.Insert Custember
            4.UpdateAddress 5.Delete Custember 6.Fetch Data
6
(Utkrash, 'UT@gmail.com', 'Lower MAnhatton St-4 Near WTC')
('Shubham Anand', 'shubham@gmail.com', 'Durg')
```

Date: 21/11/2022

Experiment No: 9.10 Aim:

Bhilai Institute of Technology, Durg.

Department of Computer Science and Engineering

PYTHON WITH DJANGO LAB 102393CS **CRN: A27**

Write a program to build a Student Information System. Use database table to store the data. The data base should contain the Enrollment Number, Student Name, Branch, Semester, Marks of 5 subjects. Give the facility to perform add, delete, search, edit and View All procedures on the database.

Code:

```
def creat_database():
    import mysql.connector
    mydb = mysql.connector.connect(host="localhost",user="root",password="") mycursor
    = mydb.cursor()
    data_base = str(input("Enter the name of data base:"))                # give
table_name = student_database (same as it , required for further execution)
    querry ="CREATE DATABASE %s" %(data_base) mycursor.execute(querry)


def creating_tabel():
    import mysql.connector
    mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") mycursor = mydb.cursor()
    table_name = str(input("Enter the name of the table:"))              # give
table_name = student_table_3 (same as it , required for further execution)
```

```
    querry_1 = "CREATE TABLE IF NOT EXISTS %s (Enrollment_Number int(5) ,Student_Name
VARCHAR(255), Branch VARCHAR(5), Semester VARCHAR(5), Subject_1 INT(5), Subject_2
INT(5), Subject_3 INT(5),Subject_4 INT(5),Subject_5 INT(5))"%(table_name)
    mycursor.execute(querry_1)


def data_insertion():
    import mysql.connector mydb
    =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") mycursor = mydb.cursor()
    querry_1 = ("SELECT Enrollment_Number FROM student_table_3")
    mycursor.execute(querry_1) data = mycursor.fetchall()
    id_list = [] for x in data:
        a = list(x)
        id_list.append(list(a)) id
    = id_list[-1][0] + 1
    Name = input("Enter your Name:")
    Branch = input("Enter your Branch:") Semester
    = input("Enter yout Semester:") marks = []
    print("Enter your 5 Subjects marks")
    for i in range (0,5): a = input()
        marks.append(a)
    Subject_1 = marks[0]
    Subject_2 = marks[1]
```

```
    Subject_3 = marks[2]
    Subject_4 = marks[3] Subject_5
    = marks[4]
    querry = "INSERT INTO student_table_3 VALUES ('{}' , '{}' , '{}' , '{}' , '{}' ,
'{}' , '{}' , '{}' , '{}')".format(id , Name , Branch , Semester , Subject_1 ,
Subject_2 , Subject_3 , Subject_4 , Subject_5)
    mycursor.execute(querry) mydb.commit()
    print("Data has been Inserted")


def   update   ():   import
    mysql.connector
    mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") mycursor = mydb.cursor()
    Enrollment_Number = input("Enter the Enrollment_Number of the Student:")
    Name = input("Enter the name of the student:") new_branch = input("Enter
    the edited branch:") new_semester = input("Enter the edited SEMESTER:")
    new_Subject_1 = input("Enter the edited marks for Subject 1:")
    new_Subject_2 = input("Enter the edited marks for Subject 2:")
    new_Subject_3 = input("Enter the edited marks for Subject 3:")
    new_Subject_4 = input("Enter the edited marks for Subject 4:")
    new_Subject_5 = input("Enter the edited marks for Subject 5:")


    sql_1 = "UPDATE student_table_3 SET Subject_1 = '{}' WHERE Enrollment_Number =
'{}'".format(new_Subject_1 , Enrollment_Number)
    sql_2 = "UPDATE student_table_3 SET Subject_2 = '{}' WHERE Enrollment_Number =
'{}'".format(new_Subject_2 , Enrollment_Number)
```

```
    sql_3 = "UPDATE student_table_3 SET Subject_3 = '{}' WHERE Enrollment_Number =
'{}'".format(new_Subject_3 , Enrollment_Number)
    sql_4 = "UPDATE student_table_3 SET Subject_4 = '{}' WHERE Enrollment_Number =
'{}'".format(new_Subject_4 , Enrollment_Number)
    sql_5 = "UPDATE student_table_3 SET Subject_5 = '{}' WHERE Enrollment_Number =
'{}'".format(new_Subject_5 , Enrollment_Number) sql_6 = "UPDATE student_table_3
    SET Branch = '{}' WHERE Enrollment_Number =
'{}'".format(new_branch , Enrollment_Number) sql_7 = "UPDATE student_table_3 SET
Semester = '{}' WHERE Enrollment_Number = '{}'".format(new_semester ,
Enrollment_Number)


    mycursor.execute(sql_1) mycursor.execute(sql_2)
    mycursor.execute(sql_3) mycursor.execute(sql_4)
    mycursor.execute(sql_5) mycursor.execute(sql_6)
    mycursor.execute(sql_7)


    mydb.commit()
    print("Data has been Updated")


def    delete():    import
    mysql.connector mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") mycursor = mydb.cursor()
    Enrollment_Number = input("Enter the Enrollment_Number of Student:") queery
    = "DELETE FROM student_table_3 WHERE Enrollment_Number =
'{}'".format(Enrollment_Number)
```

```
    mycursor.execute(queery) mydb.commit()
    print("Data has been Deleted")


def fetch_data_all():
    import mysql.connector mydb
    =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") mycursor = mydb.cursor() mycursor.execute("SELECT * FROM student_table_3") data
= mycursor.fetchall() for x in data: print(x)


def search_studnet():
    import mysql.connector
    mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="student_data
base") mycursor = mydb.cursor()
    Name = input("Enter the name of the student: ")
    queery = "SELECT * FROM student_table_3 WHERE Student_Name = '{}'".format(Name)
    mycursor.execute(queery) data = mycursor.fetchall() for x in data:
        print(x)


def Execution_crud_operation():
    print("""Enter the number against FUNCTION
```

```
    1.Creat Database 2.Creat Table 3.Insert Data 4.Update Student
Data  5.Delete  Student  Data  6.Fetch  All  Data  7.Search  Studnet""")
    user_input = int(input()) if user_input == 1: creat_database()
    elif user_input == 2:
        creating_tabel()
    elif user_input == 3:
        data_insertion()
    elif user_input == 4:
        update ()
    elif user_input == 5:
        delete()
    elif user_input == 6 :
        fetch_data_all()
    elif user_input == 7:
        search_studnet()
    else:
        pass

# Driver's Code
Execution_crud_operation()
```

Output:

```
Enter the number against FUNCTION
    1.Creat Database    2.Creat
Table  3.Insert Data 4.Update
Student Data 5.Delete Student
Data 6.Fetch All Data
7.Search Studnet
3
Enter your Name:Harsh
Enter your Branch:CSE
Enter yout Semester:3
Enter your 5 Subjects marks
88 87
78
81
83
Data has been Inserted

Enter the number against FUNCTION
       1. Creat Database 2.Creat Table  3.Insert Data 4.Update
Student Data 5.Delete Student Data 6.Fetch All Data  7.Search Studnet
4
Enter the Enrollment_Number of the Student:5
Enter the name of the student:Harsh
Enter the edited branch:CIVIL
Enter the edited  SEMESTER:4
Enter the edited marks for Subject 1:86
Enter the edited marks for Subject 2:71
Enter the edited marks for Subject 3:73
Enter the edited marks for Subject 4:77
Enter the edited marks for Subject 5:79
Data has been Updated

Enter the number against FUNCTION
```

```
    1.Creat Database     2.Creat Table 3.Insert Data    4.Update
    Student Data 5.Delete Student Data 6.Fetch All Data    7.Search
Studnet
6
(2, 'Aditya', 'Civil', '2', 80, 70, 75, 85, 90)
(3, 'Shubham', 'CSE', '3', 89, 87, 88, 83, 81)
(4, 'Nikhil', 'CSE', '3', 80, 81, 85, 89, 75)
(5, 'Harsh', 'CIVIL', '4', 86, 71, 73, 77, 79)

Enter the number against FUNCTION
    1.Creat Database   2.Creat Table 3.Insert Data
Student Data 5.Delete Student Data 6.Fetch All Data
Studnet
7
Enter the name of the student: Nikhil                    4.Update
(4, 'Nikhil', 'CSE', '3', 80, 81, 85, 89, 75)               7.Search
```

Date: 21/11/2022

Experiment No: 9.11 Aim:

Create a database of 5 Customers in a Bank. Every customer has Account number, Name, Address of branch and Balance amount. Use database table to store the data.Provide following options to the user:
a. Search any customer by Account number and display its details
b. Edit any customer's address
c. Deposit/Withdraw amount from the account.
d. Display data of all the customers

Code:

```
def creat_database():
    import mysql.connector
    mydb = mysql.connector.connect(host="localhost",user="root",password="")
    mycursor = mydb.cursor()
    data_base = str(input("Enter the name of data base:"))     # give data_base =
bank_database (same as it , required for further execution)
    querry ="CREATE DATABASE IF NOT EXISTS %s" %(data_base)
    mycursor.execute(querry)


# creat_database()


def creating_tabel():
    import mysql.connector mydb
    =
mysql.connector.connect(host="localhost",user="root",password="",database="bank_databas
e")
```

```
    mycursor = mydb.cursor()
    table_name = str(input("Enter the name of the table:"))      # give table_name =
bank_cust_table (same as it , required for further execution) querry = "CREATE TABLE IF
    NOT EXISTS %s (Account_Number int(20) ,Name VARCHAR(255),
Address_of_Branch VARCHAR(255),Balance_amount FLOAT(10))"%(table_name)
    mycursor.execute(querry)


# creating_tabel()


def data_insertion():
    import mysql.connector mydb
    =
mysql.connector.connect(host="localhost",user="root",password="",database="bank_databas
e") mycursor = mydb.cursor()
    Account_nummber = int(input("Enter the account number"))
    Name = input("Enter your Name")
    Address_of_Branch = input("Enter your address of branch") Balance_amount
    = int(input("Enter your amount"))
    querry = "INSERT INTO bank_cust_table VALUES ('{}' , '{}' , '{}' ,
'{}')".format(Account_nummber , Name , Address_of_Branch , Balance_amount )

mycursor.execute(querry) mydb.commit() # data_insertion()


def fetch_data_one():
    import mysql.connector
    mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="bank_databas
e")
```

```
    mycursor = mydb.cursor()
    Account_Number = int(input("Enter your account number:"))
    querry = "SELECT * FROM bank_cust_table WHERE Account_Number =
'{}'".format(Account_Number)
mycursor.execute(querry) data
= mycursor.fetchall() for x in
data: print(x) #
fetch_data_one()


def   withdraw():   import
    mysql.connector mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="bank_databas
e") mycursor = mydb.cursor()
    Account_Number     =     int(input("Enter     Account_Number:"))
    withdraw_amount  =  int(input("Enter  the  amount  for  withdraw:"))
    querry_1 = "SELECT * FROM bank_cust_table WHERE Account_Number =
'{}'".format(Account_Number)
    mycursor.execute(querry_1)
    data = mycursor.fetchall()
    custom ber_data = [] for x
    in data:
        custom ber_data.append(x)
    account_balance = custom ber_data[0][3]


    if account_balance >= withdraw_amount:
        Net_Balance = account_balance - withdraw_amount
```

```
        querry_2 = "UPDATE bank_cust_table SET Balance_amount =
'{}'".format(Net_Balance)
        mycursor.execute(querry_2) mydb.commit()
    else: print("You are “Broke” !!!
        ")


# withdraw()


def   update   ():   import
    mysql.connector mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="bank_databas
e") mycursor = mydb.cursor()
    Account_Number = int(input("Enter Account_Number:")) New_Address
    = input("Enter your new address:")
    sql = "UPDATE bank_cust_table SET Address_of_Branch = '{}' WHERE Account_Number =
'{}'".format( Account_Number , New_Address ) mycursor.execute(sql)
    mydb.commit()


def fetch_data_all():
    import  mysql.connector  mydb
    =
mysql.connector.connect(host="localhost",user="root",password="",database="bank_databas
e") mycursor = mydb.cursor() mycursor.execute("SELECT * FROM bank_cust_table") data =
mycursor.fetchall()
```

```
    for x in data:

print(x) #

fetch_data_all()


def fetch_data_coloumn():
    import mysql.connector
    mydb =
mysql.connector.connect(host="localhost",user="root",password="",database="bank_databas
e") mycursor = mydb.cursor() name = input("Enter your Name") queery = "SELECT * FROM
bank_cust_table WHERE Name = '{}'".format(name) mycursor.execute(queery) data =
mycursor.fetchall()
    for x in data:
        print(x)


def Execution_crud_operation():
    print("""Enter the number against FUNCTION
    1. Creat Database 2.Creat Table 3.Insert Customber 4.Update
Address 5.Withdraw Money 6.Fetch All Data 7.Fetch Customber Data""") user_input =
    int(input())
    if user_input == 1: creat_database()
    elif user_input == 2: creating_tabel()
    elif user_input == 3: data_insertion()
```

```
    elif user_input == 4:
        update ()
    elif user_input == 5:
        withdraw()
    elif user_input == 6:
        fetch_data_all()
    elif user_input == 7:
        fetch_data_coloumn()
    else:
        pass


Execution_crud_operation()
```

Output:

```
Enter the number against FUNCTION
    1.Creat Database 2.Creat Table 3.Insert Custember 4.Update Address
5.Withdraw Money 6.Fetch All Data 7.Fetch Custember Data 3
Enter the account number5
Enter your NameRITESH CHATURVEDI
Enter your address of branch270 Park Ave., New York, NY 10017
Enter your amount90000

Enter the number against FUNCTION
    1.Creat Database 2.Creat Table 3.Insert Custember 4.Update Address
5.Withdraw Money 6.Fetch All Data 7.Fetch Custember Data 4
Enter Account_Number:5
```

```
Enter your new address:100 North Tryon Street, Charlotte , NC 28255
```

```
Enter the number against FUNCTION
    1. Creat Database 2.Creat Table 3.Insert Custombe r 4.Update Address
5.Withdraw Money 6.Fetch All Data 7.Fetch Custombe r Data 5
Enter Account_Number:5
Enter the amount for withdraw:20000
```

```
Enter the number against FUNCTION
    1.Creat Database 2.Creat Table 3.Insert Custombe r 4.Update Address
5.Withdraw Money 6.Fetch All Data 7.Fetch Custombe r Data 6
(2, 'Shubham Anand', 'Wall Street St-10 NYC', 70000.0)
(5, 'Shubham Anand', '270 Park Ave., New York, NY 10017', 70000.0)
```

```
Enter the number against FUNCTION
    1.Creat Database 2.Creat Table 3.Insert Custombe r 4.Update Address
5.Withdraw Money 6.Fetch All Data 7.Fetch Custombe r Data 7
Enter your NameRITESH CHATURVEDI
(2, 'Shubham Anand', 'Wall Street St-10 NYC', 70000.0)
(5, 'Shubham Anand', '270 Park Ave., New York, NY 10017', 70000.0)
```