# Experiment Index

**PYTHON WITH DJANGO LAB 102393CS**

|  |  |  |
| --- | --- | --- |
| **Student’s Name** | **Branch & Semester** | **Class Roll No.** |
| **RITESH CHATURVEDI** | **CSE 3rd SEMESTER** | **A59** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **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 |  |  |

# Experiment Index

**PYTHON WITH DJANGO LAB 102393CS**

|  |  |  |
| --- | --- | --- |
| **Student’s Name** | **Branch & Semester** | **Class Roll No.** |
| **RITESH CHATURVEDI** | **CSE 3rd SEMESTER** | **A59** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 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.   1. Input the Name and Dept details from the user. 2. Input marks of 5 subjects (out of 100) and store the aggregate in AggMarks. 3. Calculate Aggregate percentage out of 500 and store in AggMarks. 4. Display all the contents | 130 |  |  |

# Experiment Index

**PYTHON WITH DJANGO LAB 102393CS**

|  |  |  |
| --- | --- | --- |
| **Student’s Name** | **Branch & Semester** | **Class Roll No.** |
| **RITESH CHATURVEDI** | **CSE 3rd SEMESTER** | **A59** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 9.9 | 14/11/22 | WAP to build a database table to hold  Name, Email and Address of 5 customers.   1. Input the Name, Email and Address. 2. 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:   1. Search any customer by Account number and display its details 2. Edit any customer's address 3. Deposit/Withdraw amount from the account. 4. Display data of all the customers | 147 |  |  |

**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(): |

P a g e

|  |
| --- |
| 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) |

|  |
| --- |
| 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()  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() |

|  |
| --- |
| 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) |

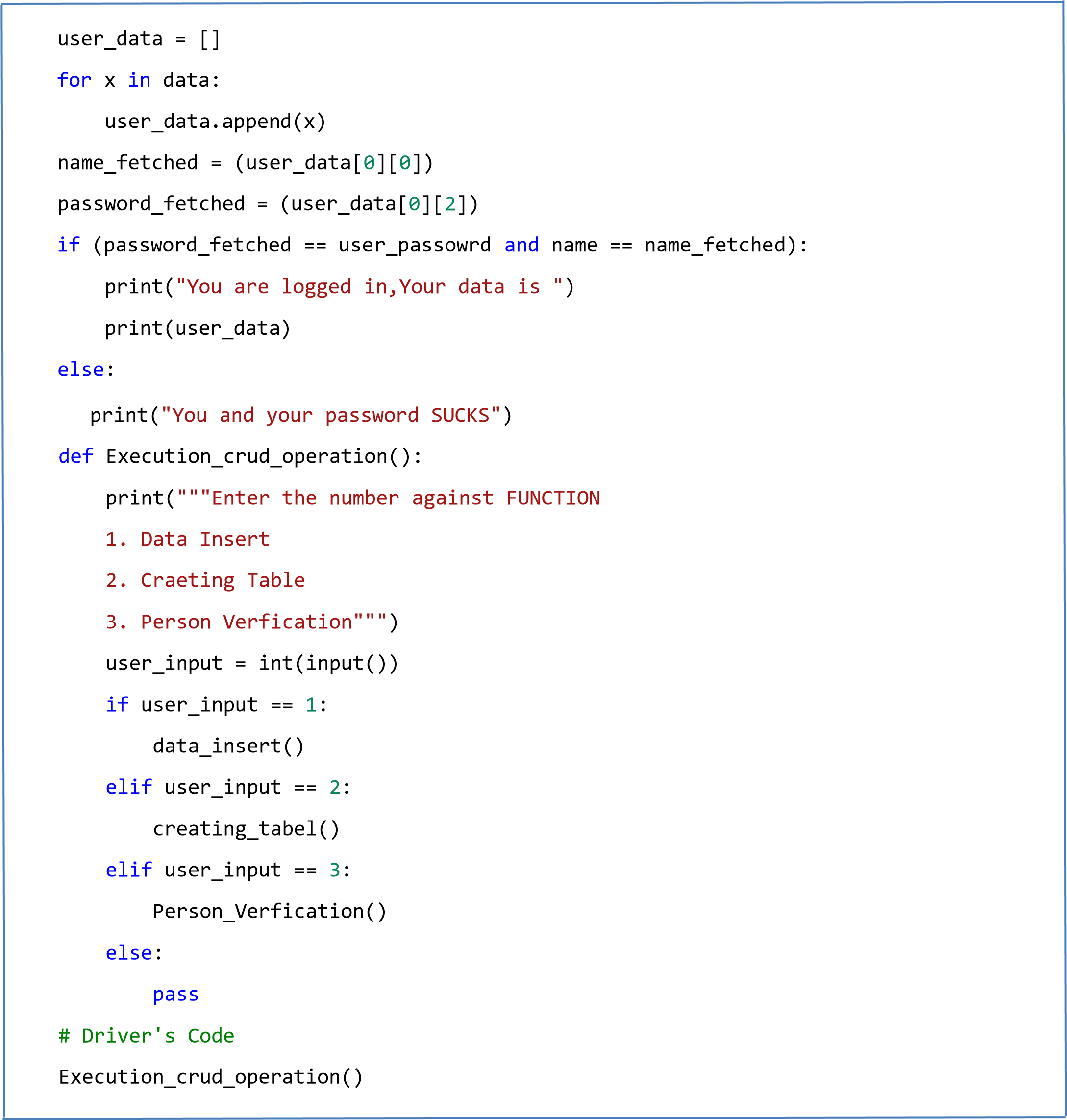
**Date: 21/11/2022 Experiment No: 9.4 Aim:**

**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") |





**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')] |

**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:"

))

student\_database

(

same

as

it

,

required

for

further

execution)

#

give

data\_base

=

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"

)

117**|**

|  |
| --- |
| 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 =  customber\_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="customber\_da tabase") mycursor = mydb.cursor()  table\_name = str(input("Enter the name of the table:")) # give table\_name =  customber\_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) 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 Customber  Address 5.Delete Customber 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() | 4.Update |

**Output:**

|  |
| --- |
| Enter the number against FUNCTION  1. Creat Database 2.Creat Table 3.Insert Customber  4.UpdateAddress 5.Delete Customber  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 Customber  4.UpdateAddress 5.Delete Customber 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 Customber  4.UpdateAddress 5.Delete Customber 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 Customber  4.UpdateAddress 5.Delete Customber 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:**

**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  Student Data 5.Delete Student Data 6.Fetch All Data  Studnet  6 | 4.Update  7.Search |
| (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, 'Nikhil', 'CSE', '3', 80, 81, 85, 89, 75) | 4.Update  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:**

1. **Search any customer by Account number and display its details**
2. **Edit any customer's address**
3. **Deposit/Withdraw amount from the account.**
4. **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() customber\_data = [] for x in data:  customber\_data.append(x)  account\_balance = customber\_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 Customber 4.Update Address 5.Withdraw Money 6.Fetch All Data 7.Fetch Customber 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 Customber 4.Update Address 5.Withdraw Money 6.Fetch All Data 7.Fetch Customber 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 Customber 4.Update Address 5.Withdraw Money 6.Fetch All Data 7.Fetch Customber 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 Customber 4.Update Address 5.Withdraw Money 6.Fetch All Data 7.Fetch Customber 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 Customber 4.Update Address 5.Withdraw Money 6.Fetch All Data 7.Fetch Customber 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)