

```

from os import system
import re
import mysql.connector

# making Connection
con = mysql.connector.connect(
    host="localhost", user="root", password="roshhh@5678",
    database="employee")

# make a regular expression
# for validating an Email
regex = r'\b[A-Za-z0-9._%+-]+@[A-Za-z0-9.-]+\.[A-Z|a-z]{2,}\b'
# for validating an Phone Number
Pattern = re.compile("^[9]{1}([234789]{1})([0-9]{8})$")

# Function to Add_Employ
def Add_stud():
    print("{:>60}".format("-->>Add student Record<<--"))
    Id = input("Enter student Id: ")
    # checking If student Id is Exit Or Not
    if (check_student(Id) == True):
        print("student ID Already Exists\nTry Again..")
        press = input("Press Any Key To Continue..")
        Add_stud()
    Name = input("Enter student Name: ")
    # checking If student Name is Exit Or Not
    if (check_student_name(Name) == True):
        print("student Name Already Exists\nTry Again..")
        press = input("Press Any Key To Continue..")
        Add_stud()
    Email_Id = input("Enter student Email ID: ")
    if(re.fullmatch(regex, Email_Id)):
        print("Valid Email")
    else:
        print("Invalid Email")
        press = input("Press Any Key To Continue..")
        Add_stud()
    Phone_no = input("Enter student Phone No.: ")
    if(Pattern.match(Phone_no)):
        print("Valid Phone Number")
    else:

```

```

        print("Invalid Phone Number")
        press = input("Press Any Key To Continue..")
        Add_stud()
    Address = input("Enter student Address: ")
    std = input("Enter student std: ")
    data = (Id, Name, Email_Id, Phone_no, Address, std)
    # Instering student Details in
    # the student (empdata) Table
    sql = 'insert into empdata3 values(%s,%s,%s,%s,%s,%s) '
    c = con.cursor()

    # Executing the sql Query
    c.execute(sql, data)

    # Commit() method to make changes in the table
    con.commit()
    print("Successfully Added student Record")
    press = input("Press Any Key To Continue..")
    menu()

# Function To Check if student With
# given Name Exist or not
def check_student_name(student_name):
    # query to select all Rows from
    # student(empdata) table
    sql = 'select * from empdata3 where Name=%s'

    # making cursor buffered to make
    # rowcount method work properly
    c = con.cursor(buffered=True)
    data = (student_name,)

    # Execute the sql query
    c.execute(sql, data)

    # rowcount method to find number
    # of rowa with given values
    r = c.rowcount
    if r == 1:
        return True
    else:
        return False

```

```

# Function To Check if Employee With
# given Id Exist or not
def check_student(student_id):
    # query to select all Rows from
    # student(empdata) table
    sql = 'select * from empdata3 where Id=%s'

    # making cursor buffered to make
    # rowcount method work properly
    c = con.cursor(buffered=True)
    data = (student_id,)

    # Execute the sql query
    c.execute(sql, data)

    # rowcount method to find number
    # of rows with given values
    r = c.rowcount
    if r == 1:
        return True
    else:
        return False

# Function to Display_Employ
def Display_stud():
    print("{:>60}".format("--> Display student Record <--"))
    # query to select all rows from student (empdata) Table
    sql = 'select * from empdata3'
    c = con.cursor()

    # Executing the sql query
    c.execute(sql)

    # Fetching all details of all the Employees
    r = c.fetchall()
    for i in r:
        print("student Id: ", i[0])
        print("student Name: ", i[1])
        print("student Email Id: ", i[2])
        print("student Phone No.: ", i[3])
        print("student Address: ", i[4])
        print("student std: ", i[5])

```

```

        print("\n")
    press = input("Press Any key To Continue..")
    menu()

# Function to Update_stud
def Update_stud():
    print("{:>60}".format("-->> Update student Record <<--\n"))
    Id = input("Enter student Id: ")
    # checking If student Id is Exit Or Not
    if(check_student(Id) == False):
        print("student Record Not exists\nTry Again")
        press = input("Press Any Key To Continue..")
        menu()
    else:
        Email_Id = input("Enter student Email ID: ")
        if(re.fullmatch(regex, Email_Id)):
            print("Valid Email")
        else:
            print("Invalid Email")
            press = input("Press Any Key To Continue..")
            Update_stud()
        Phone_no = input("Enter student Phone No.: ")
        if(Pattern.match(Phone_no)):
            print("Valid Phone Number")
        else:
            print("Invalid Phone Number")
            press = input("Press Any Key To Continue..")
            Update_stud()
        Address = input("Enter student Address: ")
        std = input("enter std: ")
        # Updating student details in empdata Table
        sql = 'UPDATE empdata3 set Email_Id = %s, Phone_no = %s,
Address = %s, std = %s where Id = %s'
        data = (Email_Id, Phone_no, Address, std, Id)
        c = con.cursor()

        # Executing the sql query
        c.execute(sql, data)

        # commit() method to make changes in the table
        con.commit()
        print("Updated student Record")
        press = input("Press Any Key To Continue..")

```

```

        menu()

# Function to Promote_stud
def Promote_stud():
    print("{:>60}".format("-->> increment student Record <<--\n"))
    Id = input("Enter student Id: ")
    # checking If student Id is Exit Or Not
    if(check_student(Id) == False):
        print("student Record Not exists\nTry Again")
        press = input("Press Any Key To Continue..")
        menu()
    else:
        std = int(input("Enter promote std: "))
        #query to fetch std of student with given data
        sql = 'select std from empdata3 where Id=%s'
        data = (Id,)
        c = con.cursor()

        #executing the sql query
        c.execute(sql, data)

        #fetching salary of Employee with given Id
        r = c.fetchone()
        t = r[0]+std

        #query to update salary of Employee with given id
        sql = 'update empdata3 set std = %s where Id = %s'
        d = (t, Id)

        #executing the sql query
        c.execute(sql, d)

        #commit() method to make changes in the table
        con.commit()
        print("student Promoted")
        press = input("Press Any key To Continue..")
        menu()

# Function to studnet_Employ
def Remove_student():
    print("{:>60}".format("-->> Remove student Record <<--\n"))
    Id = input("Enter student Id: ")
    # checking If student Id is Exit Or Not

```

```

if(check_student(Id) == False):
    print("student Record Not exists\nTry Again")
    press = input("Press Any Key To Continue..")
    menu()
else:
    #query to delete student from empdata table
    sql = 'delete from empdata3 where Id = %s'
    data = (Id,)
    c = con.cursor()

    #executing the sql query
    c.execute(sql, data)

    #commit() method to make changes in the empdata table
    con.commit()
    print("student Removed")
    press = input("Press Any key To Continue..")
    menu()

# Function to Search_stud
def Search_stud():
    print("{:>60}".format("-->> Search student Record <--\n"))
    Id = input("Enter student Id: ")
    # checking If student Id is Exit Or Not
    if(check_student(Id) == False):
        print("student Record Not exists\nTry Again")
        press = input("Press Any Key To Continue..")
        menu()
    else:
        #query to search Employee from empdata table
        sql = 'select * from empdata3 where Id = %s'
        data = (Id,)
        c = con.cursor()

        #executing the sql query
        c.execute(sql, data)

        #fetching all details of all the employee
        r = c.fetchall()
        for i in r:
            print("student Id: ", i[0])
            print("student Name: ", i[1])
            print("student Email Id: ", i[2])

```

```

        print("student Phone No.: ", i[3])
        print("student Address: ", i[4])
        print("student std: ", i[5])
        print("\n")
        press = input("Press Any key To Continue..")
        menu()

# Menu function to display menu
def menu():
    system("cls")
    print("{:>60}".format("*****"))
    print("{:>60}".format("-->> student Management System <<--"))
    print("{:>60}".format("*****"))
    print("1. Add student")
    print("2. Display student Record")
    print("3. Update student Record")
    print("4. Promote student Record")
    print("5. Remove student Record")
    print("6. Search student Record")
    print("7. Exit\n")
    print("{:>60}".format("-->> Choice Options: [1/2/3/4/5/6/7] <<--"))

    ch = int(input("Enter your Choice: "))
    if ch == 1:
        system("cls")
        Add_stud()
    elif ch == 2:
        system("cls")
        Display_stud()
    elif ch == 3:
        system("cls")
        Update_stud()
    elif ch == 4:
        system("cls")
        Promote_stud()
    elif ch == 5:
        system("cls")
        Remove_student()
    elif ch == 6:
        system("cls")
        Search_stud()
    elif ch == 7:
        system("cls")

```

```

        print("{:>60}".format("Have A Nice Day :"))
        exit(0)
    else:
        print("Invalid Choice!")
        press = input("Press Any key To Continue..")
        menu()

# Calling menu function
menu()

from os import system
import re
import mysql.connector

# making Connection
con = mysql.connector.connect(
    host="localhost", user="root", password="roshhh@5678",
    database="employee")

# make a regular expression
# for validating an Email
regex = r'\b[A-Za-z0-9._%+-]+@[A-Za-z0-9.-]+\.[A-Z|a-z]{2,}\b'
# for validating an Phone Number
Pattern = re.compile("^( [9]{1} ) ( [234789]{1} ) ( [0-9]{8} ) $")

# Function to Add_Employ
def Add_stud():
    print("{:>60}".format("-->>Add student Record<<--"))
    Id = input("Enter student Id: ")
    # checking If student Id is Exit Or Not
    if (check_student(Id) == True):
        print("student ID Already Exists\nTry Again..")
        press = input("Press Any Key To Continue..")
        Add_stud()
    Name = input("Enter student Name: ")
    # checking If student Name is Exit Or Not
    if (check_student_name(Name) == True):
        print("student Name Already Exists\nTry Again..")
        press = input("Press Any Key To Continue..")
        Add_stud()
    Email_Id = input("Enter student Email ID: ")

```



```

if(re.fullmatch(regex, Email_Id)):
    print("Valid Email")
else:
    print("Invalid Email")
    press = input("Press Any Key To Continue..")
    Add_stud()
Phone_no = input("Enter student Phone No.: ")
if(Pattern.match(Phone_no)):
    print("Valid Phone Number")
else:
    print("Invalid Phone Number")
    press = input("Press Any Key To Continue..")
    Add_stud()
Address = input("Enter student Address: ")
std = input("Enter student std: ")
data = (Id, Name, Email_Id, Phone_no, Address, std)
# Instering student Details in
# the student (empdata) Table
sql = 'insert into empdata3 values(%s,%s,%s,%s,%s,%s)'
c = con.cursor()

# Executing the sql Query
c.execute(sql, data)

# Commit() method to make changes in the table
con.commit()
print("Successfully Added student Record")
press = input("Press Any Key To Continue..")
menu()

# Function To Check if student With
# given Name Exist or not
def check_student_name(student_name):
    # query to select all Rows from
    # student(empdata) table
    sql = 'select * from empdata3 where Name=%s'

    # making cursor buffered to make
    # rowcount method work properly
    c = con.cursor(buffered=True)
    data = (student_name,)

    # Execute the sql query

```

```

c.execute(sql, data)

# rowcount method to find number
# of rowa with given values
r = c.rowcount
if r == 1:
    return True
else:
    return False

# Function To Check if Employee With
# given Id Exist or not
def check_student(student_id):
    # query to select all Rows from
    # student(empdata) table
    sql = 'select * from empdata3 where Id=%s'

    # making cursor buffered to make
    # rowcount method work properly
    c = con.cursor(buffered=True)
    data = (student_id,)

    # Execute the sql query
    c.execute(sql, data)

    # rowcount method to find number
    # of rowa with given values
    r = c.rowcount
    if r == 1:
        return True
    else:
        return False

# Function to Display_Employ
def Display_stud():
    print("{:>60}".format("--> Display student Record <--"))
    # query to select all rows from student (empdata) Table
    sql = 'select * from empdata3'
    c = con.cursor()

    # Executing the sql query
    c.execute(sql)

```

```

# Fetching all details of all the Employees
r = c.fetchall()
for i in r:
    print("student Id: ", i[0])
    print("student Name: ", i[1])
    print("student Email Id: ", i[2])
    print("student Phone No.: ", i[3])
    print("student Address: ", i[4])
    print("student std: ", i[5])
    print("\n")
press = input("Press Any key To Continue..")
menu()

# Function to Update_stud
def Update_stud():
    print("{:>60}".format("-->> Update student Record <--\n"))
    Id = input("Enter student Id: ")
    # checking If student Id is Exit Or Not
    if(check_student(Id) == False):
        print("student Record Not exists\nTry Again")
        press = input("Press Any Key To Continue..")
        menu()
    else:
        Email_Id = input("Enter student Email ID: ")
        if(re.fullmatch(regex, Email_Id)):
            print("Valid Email")
        else:
            print("Invalid Email")
            press = input("Press Any Key To Continue..")
            Update_stud()
        Phone_no = input("Enter student Phone No.: ")
        if(Pattern.match(Phone_no)):
            print("Valid Phone Number")
        else:
            print("Invalid Phone Number")
            press = input("Press Any Key To Continue..")
            Update_stud()
        Address = input("Enter student Address: ")
        std = input("enter std: ")
        # Updating student details in empdata Table
        sql = 'UPDATE empdata3 set Email_Id = %s, Phone_no = %s,
Address = %s, std = %s where Id = %s'

```

```

data = (Email_Id, Phone_no, Address, std, Id)
c = con.cursor()

# Executing the sql query
c.execute(sql, data)

# commit() method to make changes in the table
con.commit()
print("Updated student Record")
press = input("Press Any Key To Continue..")
menu()

# Function to Promote_stud
def Promote_stud():
    print("{:>60}".format("-->> increment student Record <--\n"))
    Id = input("Enter student Id: ")
    # checking If student Id is Exit Or Not
    if(check_student(Id) == False):
        print("student Record Not exists\nTry Again")
        press = input("Press Any Key To Continue..")
        menu()
    else:
        std = int(input("Enter promote std: "))
        #query to fetch std of student with given data
        sql = 'select std from empdata3 where Id=%s'
        data = (Id,)
        c = con.cursor()

        #executing the sql query
        c.execute(sql, data)

        #fetching salary of Employee with given Id
        r = c.fetchone()
        t = r[0]+std

        #query to update salary of Employee with given id
        sql = 'update empdata3 set std = %s where Id = %s'
        d = (t, Id)

        #executing the sql query
        c.execute(sql, d)

        #commit() method to make changes in the table

```

```

        con.commit()
        print("student Promoted")
        press = input("Press Any key To Continue..")
        menu()

# Function to studnet_Employ
def Remove_student():
    print("{:>60}".format("--> Remove student Record <--\n"))
    Id = input("Enter student Id: ")
    # checking If student Id is Exit Or Not
    if(check_student(Id) == False):
        print("student Record Not exists\nTry Again")
        press = input("Press Any Key To Continue..")
        menu()
    else:
        #query to delete student from empdata table
        sql = 'delete from empdata3 where Id = %s'
        data = (Id,)
        c = con.cursor()

        #executing the sql query
        c.execute(sql, data)

        #commit() method to make changes in the empdata table
        con.commit()
        print("student Removed")
        press = input("Press Any key To Continue..")
        menu()

# Function to Search_stud
def Search_stud():
    print("{:>60}".format("--> Search student Record <--\n"))
    Id = input("Enter student Id: ")
    # checking If student Id is Exit Or Not
    if(check_student(Id) == False):
        print("student Record Not exists\nTry Again")
        press = input("Press Any Key To Continue..")
        menu()
    else:
        #query to search Employee from empdata table
        sql = 'select * from empdata3 where Id = %s'
        data = (Id,)
        c = con.cursor()

```

```

        #executing the sql query
        c.execute(sql, data)

        #fetching all details of all the employee
        r = c.fetchall()
        for i in r:
            print("student Id: ", i[0])
            print("student Name: ", i[1])
            print("student Email Id: ", i[2])
            print("student Phone No.: ", i[3])
            print("student Address: ", i[4])
            print("student std: ", i[5])
            print("\n")
        press = input("Press Any key To Continue..")
        menu()

# Menu function to display menu
def menu():
    system("cls")
    print("{:>60}".format("*****"))
    print("{:>60}".format("-->> student Management System <<--"))
    print("{:>60}".format("*****"))
    print("1. Add student")
    print("2. Display student Record")
    print("3. Update student Record")
    print("4. Promote student Record")
    print("5. Remove student Record")
    print("6. Search student Record")
    print("7. Exit\n")
    print("{:>60}".format("-->> Choice Options: [1/2/3/4/5/6/7] <<--"))

    ch = int(input("Enter your Choice: "))
    if ch == 1:
        system("cls")
        Add_stud()
    elif ch == 2:
        system("cls")
        Display_stud()
    elif ch == 3:
        system("cls")
        Update_stud()
    elif ch == 4:

```

```
        system("cls")
        Promote_stud()
    elif ch == 5:
        system("cls")
        Remove_student()
    elif ch == 6:
        system("cls")
        Search_stud()
    elif ch == 7:
        system("cls")
        print("{:>60}".format("Have A NIce Day :"))
        exit(0)
    else:
        print("Invalid Choice!")
        press = input("Press Any key To Continue..")
        menu()

# Calling menu function
menu()
```