COMPUTER SCIENCE PRACTICAL FILE

(Session: 2021-22)

DOON PUBLIC SCHOOL

Name: **NIKHIL SHEORAN**

Class: **12th D**

Roll No.: ____________________

INDEX

PROGRAM-1: LINEAR SEARCH USING FUNCTION

```
def Lsearch(list,x):

    for i in range(len(list)):
        if list[i] == x:
            return True
    return False


# Driver Code

n=int(input("\n Enter No. of elements :"))

list=[]
for i in range(0,n):
    y=input("\n Enter element ")
    list.append(y)
x=(input("\n Enter element to be searched :"))
if Lsearch(list, x):
    print("Element Exist in List")
else:
    print("Element not Exist in List")
```

OUTPUT

```
Enter No. of elements :5

Enter element 1

Enter element 6

Enter element 8

Enter element 24

Enter element 35

Enter element to be searched :11

Element not Exist in List
```

PROGRAM-2: BINARY SEARCH USING FUNCTION

```
def Bsearch(list1,x):
    low = 0
    high = len(list1) - 1
    mid = 0

    while low <= high:
        # for get integer result
        mid = (high + low) // 2
        # Check if n is present at mid
        if list1[mid] < x:
            low = mid + 1
        # If n is greater, compare to the right of mid
        elif list1[mid] > x:
            high = mid - 1
        # If n is smaller, compared to the left of mid
        else:
            return mid
        # element was not present in the list, return -1
    return -1

# Driver Code
n=int(input("\n Enter No. of elements :"))
list=[]
for i in range(0,n):
    y=input("\n Enter element ")
    list.append(y)
x=(input("\n Enter element to be searched :"))
pos=Bsearch(list, x)
if (pos>-1):
    print("Element Exist in List at position ",pos+1)
else:
    print("Element not Exist in List")
```

OUTPUT

```
Enter No. of elements :5
 Enter element 1
 Enter element 3
 Enter element 5
 Enter element 7
 Enter element 9
Enter element to be searched :7
Element Exist in List at position  4
```

PROGRAM-3: BUBBLE SORT USING FUNCTION

```
def bubbleSort(arr):
    n = len(arr)

    # Traverse through all array elements
    for i in range(n-1):
    # range(n) also work but outer loop will repeat one time more than
needed.

        # Last i elements are already in place
        for j in range(0, n-i-1):

            # traverse the array from 0 to n-i-1
            # Swap if the element found is greater
            # than the next element
            if arr[j] > arr[j+1] :
                arr[j], arr[j+1] = arr[j+1], arr[j]

# Driver Code

n=int(input("\n Enter No. of elements :"))
list=[]
for i in range(0,n):
    y=input("\n Enter element ")
    list.append(y)

bubbleSort(list)
print("Final List after Sorting : \n", list)
```

OUTPUT

```
Enter No. of elements :6

Enter element 312

Enter element 11

Enter element 123

Enter element 454

Enter element 1

Final List after Sorting :

 ['1', '11', '123', '312', '454']
```

PROGRAM-4: INSERTION SORT USING FUNCTION

```
# Function to do insertion sort

def insertion_Sort(arr):

    # Traverse through 1 to len(arr)
    for i in range(1, len(arr)):

        key = arr[i]

        # Move elements of arr[0..i-1], that are
        # greater than key, to one position ahead
        # of their current position
        j = i-1
        while j >=0 and key < arr[j] :
                arr[j+1] = arr[j]
                j -= 1
        arr[j+1] = key

# Driver Code

n=int(input("\n Enter No. of elements :"))
list=[]
for i in range(0,n):
    y=input("\n Enter element ")
    list.append(y)

insertion_Sort(list)
print("Final List after Sorting : \n", list)
```

OUTPUT

```
 Enter No. of elements :4

 Enter element 61

 Enter element 92

 Enter element 12

 Enter element 4

Final List after Sorting :
 [ '4', '12', '61', '92']
```

PROGRAM-5: PROGRAM TO CHECK IF A NUMBER IS PALINDROME USING FUNCTION

```
#  function to reverse
def rev(n):
   r=0
   while ( n > 0 ):
      d=n%10
      r=r*10 + d
      n = n//10
   return r

# initializing number
test_number = int(input("\n Enter No. :"))

# printing the original number
print ("The original number is : " + str(test_number))

# using math.log() + recursion + list comprehension
# for checking a number is palindrome
res = test_number == rev(test_number)

# printing result
print ("Is the number palindrome ? : " + str(res))
```

OUTPUT

```
Enter No. :11222211
The original number is : 11222211
Is the number palindrome ? : True

Enter No. :321124
The original number is : 321124
Is the number palindrome ? : False
```

PROGRAM-6: PROGRAM TO CHECK IF A NUMBER IS PALINDROME

```
# using math.log() + recursion + list comprehension
import math
def rev(num):
    return int(num != 0) and ((num % 10)*(10**int(math.log(num, 10)))+rev(num // 10))


# initializing number
test_number = int(input("\n Enter No. :"))


# printing the original number
print ("The original number is : " + str(test_number))


# for checking a number is palindrome
res = test_number == rev(test_number)


# printing result
print ("Is the number palindrome ? : " + str(res))
```

OUTPUT

```
Enter No. :12321
The original number is : 12321
Is the number palindrome ? : True
```

PROGRAM-7: PROGRAM TO CHECK IF A NUMBER IS ARMSTRONG NO. OR NOT USING FUNCTION

```
import math
def armstrong(n):
    s=0
    while ( n > 0 ):
        d=n%10
        s=s + math.pow(d,3)
        n = n//10
    return s

# Driver Code
test_number = int(input("\n Enter No. :"))

# printing the original number
print ("The original number is : " + str(test_number))

# for checking a number is armstong no. or not
res = test_number == armstrong(test_number)

# printing result
print ("Is the number armstrong ? : " + str(res))
```

OUTPUT

```
Enter No. :153
The original number is : 153
Is the number armstrong ? : True

Enter No. :123
The original number is : 123
Is the number armstrong ? : False
```

PROGRAM-8: PROGRAM TO CALCULATE FACTORIAL OF A NO. USING FUNCTION

```
def factorial(n):
    res=1
    for i in range(1,n+1):
        res=res*i
    return res
# Driver Code
n=int(input("Input a number to compute the factiorial : "))
print("\n Factorial of No. : ",factorial(n))
```

OUTPUT

```
Input a number to compute the factiorial : 5
 Factorial of No. :  120


Input a number to compute the factiorial : 7
 Factorial of No. :  5040
```

PROGRAM-9: PROGRAM TO CALCULATE FACTORIAL OF A NO. USING RECURSIVE FUNCTION

```
def factorial(n):
    if n == 0:
        return 1
    else:
        return n * factorial(n-1)
n=int(input("Input a number to compute the factiorial : "))
print("\n Factorial of No. : ",factorial(n))
```

OUTPUT

```
Input a number to compute the factiorial : 10
Factorial of No. : 3628800
```

PROGRAM-10: CHECK IF A STRING IS PALINDROME OR NOT USING FUNCTION WHICH RETURNS REVERSE OF A STRING

```
def isPalindrome(s):
    return s == s[::-1]



# Driver code
s = input("\n Enter a string : ")
ans = isPalindrome(s)

if ans:
    print("\n String is palindrome .")
else:
    print("\n String is not palindrome .")
```

OUTPUT

```
Enter a string : naman
 String is palindrome.

Enter a string : nikhil
 String is not palindrome.
```

PROGRAM-11: PRINT FIBONACCI SERIES UPTO N TERMS USING FUNCTION

```
def Fibonacci (n): # return Fibonacci series up to n
    """Return a list containing the Fibonacci series up to n."""
    result = []
    i=1
    a, b = 0, 1
    while i<=n:
        result.append(b)    # see below
        a, b = b, a+b
        i=i+1
    return result

# Driver Program
n = int(input("\n Enter a No. of Terms : "))
print("Fibonacci Series Terms are :")
print(Fibonacci(n))
```

OUTPUT

```
Enter a No. of Terms : 6
Fibonacci Series Terms are :
[1, 1, 2, 3, 5, 8]
```

PROGRAM-12: PRINT FIBONACCI SERIES UPTO N TERMS USING RECURSIVE FUNCTION

```
def Fibonacci(n):
    if n<=0:
        print("Incorrect input")
    # First Fibonacci number is 0
    elif n==1:
        return 0
    # Second Fibonacci number is 1
    elif n==2:
        return 1
    else:
        return Fibonacci(n-1)+Fibonacci(n-2)


# Driver Program
n = int(input("\n Enter a No. of Terms : "))
print("Fibonacci Series Terms are :")
for i in range(1,n+1):
    print(Fibonacci(i),end=' ,')
```

OUTPUT

```
Enter a No. of Terms : 8
Fibonacci Series Terms are :
0 ,1 ,1 ,2 ,3 ,5 ,8 ,13 ,
```

PROGRAM-13: FIND THE OCCURENCE OF ANY WORD IN A STRING

```
def countWord(str1,word):
    s = str1.split()
    count=0
    for w in s:
        if w==word:
            count+=1
    return count


# Driver Code
str1 = input("Enter any sentence :")
word = input("Enter word to search in sentence :")
count = countWord(str1,word)
if count==0:
    print("## Sorry! ",word," not present ")
else:
    print(word," occurs ",count," times")
```

OUTPUT

```
Enter any sentence : Nikhil is a good boy
Enter word to search in sentence : Nikhil
computer  occurs  1  times

Enter any sentence : I like python
Enter word to search in sentence :Nikhil
Sorry! Nikhil not present
```

PROGRAM-14 WRITE 3 LINES INTO TEXT FILE, READ AND DISPLAY FILE CONTENT LINE BY LINE WITH EACH WORD SEPARATED BY ‘#’

```
f = open("file1.txt",'w')
line1 = 'India is my country'
f.write(line1)

line2 = '\nI love python'
f.write(line2)

line3 = '\nPython learning is fun'
f.write(line3)

f.close()
f = open("file1.txt",'r')
for line in f:
    words = line.split()
    for w in words:
        print(w+'#',end='')
    print()
f.close()
```

OUTPUT

```
Hello#I#am#Nikhil#Sheoran#
I#have#made#this#project#
I#like#coding#in#python#
```

PROGRAM-15: WRITE 3 LINES INTO TEXT FILE, READ CONTENT OF FILE AND DISPLAY TOTAL NUMBER OF VOWELS, CONSONANTS, LOWERCASE AND UPPERCASE CHARACTERS

```
f = open("file1.txt",'w')
line1 = 'India is my country'
f.write(line1)
line2 = '\nI love python'
f.write(line2)
line3 = '\nPython learning is fun @'
f.write(line3)

f.close()
f = open("file1.txt",'r')
v,c,u,l,o=0,0,0,0,0
data = f.read()
vowels=['a','e','i','o','u']
for ch in data:
    if ch.isalpha():
        if ch.lower() in vowels:
            v+=1
        else:
            c+=1
    if ch.isupper():
        u+=1
    elif ch.islower():
        l+=1
    elif ch!=' ' and ch!='\n':
        o+=1
print("Total Vowels in file             :",v)
print("Total  Consonants in file              :",c)
print("Total Capital letters  in file           :",u)
print("Total Small letters  in file  :",l)
print("Total  Other than letters              :",o)
f.close()
```

OUTPUT

```
Total Vowels in file           : 16
Total  Consonants in file   : 30
Total Capital letters  in file          : 3
Total Small letters  in file  : 43
Total  Other than letters   : 1
```

PROGRAM-16: CREATE A BINARY FILE TO STORE ROLL NO. AND NAME SEARCH FOR ROLL NO. AND DISPLAY RECORD

```
import pickle
student=[]
f=open('student.dat','wb')
ans='y'
while ans.lower()=='y':
   roll = int(input("Enter Roll Number :"))
   name = input("Enter Name :")
   student.append([roll,name])
   ans=input("Add More ?(Y)")
pickle.dump(student,f)
f.close()
f=open('student.dat','rb')
student=[]
while True:
   try:
      student = pickle.load(f)
   except EOFError:
      break
ans='y'
while ans.lower()=='y':
   found=False
   r = int(input("Enter Roll number to search :"))
   for s in student:
      if s[0]==r:
         print("## Name is :",s[1], " ##")
         found=True
         break
   if not found:
      print("####Sorry! Roll number not found ####")
   ans=input("Search more ?(Y) :")
f.close()
```

OUTPUT

```
Enter Roll Number :1
Enter Name :NIKHIL
Add More ?(Y)Y
Enter Roll Number :2
Enter Name :SHEORAN
Add More ?(Y)N
Enter Roll number to search :1
## Name is : NIKHIL  ##
```

PROGRAM-17: CREATE A BINARY FILE TO STORE ROLL NO. AND NAME SEARCH FOR ROLL NO. AND UPDATE RECORD IF FOUND OTHERWISE "ROLL NO. NOT FOUND"

```
import pickle
student=[]
f=open('student.dat','wb')
ans='y'
while ans.lower()=='y':
    roll = int(input("Enter Roll Number :"))
    name = input("Enter Name :")
    marks = int(input("Enter Marks :"))
    student.append([roll,name,marks])
    ans=input("Add More ?(Y)")
pickle.dump(student,f)
f.close()
f=open('student.dat','rb+')
student=[]
while True:
    try:
        student = pickle.load(f)
    except EOFError:
        break
while ans.lower()=='y':
    found=False
    r = int(input("Enter Roll number to update :"))
    for s in student:
        if s[0]==r:
            print("## Name is :",s[1], " ##")
            print("## Current Marks is :",s[2]," ##")
            m = int(input("Enter new marks :"))
            s[2]=m
            print("## Record Updated ##")
            found=True
            break
    if not found:
        print("####Sorry! Roll number not found ####")
    ans=input("Update more ?(Y) :")
f.close()
```

OUTPUT

```
Enter Roll Number :1
Enter Name : NIKHIL
Enter Marks :90
Add More ?(Y)Y
Enter Roll Number :2
Enter Name :VISHU
Enter Marks :80
Add More ?(Y)N
Enter Roll number to update :1
## Name is : NIKHIL  ##
## Current Marks is : 90  ##
Enter new marks :95
## Record Updated ##
Update more ?(Y) :N
```

PROGRAM-18: CREATE CSV FILE AND STORE EMP NO., NAME, SALARY AND SEARCH ANY EMP NO. AND DISPLAY NAME, SALARY AND IF NOT FOUND APPROPRIATE MESSAGE.

```
import csv
with open('myfile.csv',mode='w') as csvfile:
    mywriter = csv.writer(csvfile,delimiter=',')
    ans='y'
    while ans.lower()=='y':
        eno=int(input("Enter Employee Number "))
        name=input("Enter Employee Name ")
        salary=int(input("Enter Employee Salary :"))
        mywriter.writerow([eno,name,salary])
        print("## Data Saved... ##")
        ans=input("Add More ?")
ans='y'
with open('myfile.csv',mode='r') as csvfile:
    myreader = csv.reader(csvfile,delimiter=',')
    while ans=='y':
        found=False
        e = int(input("Enter Employee Number to search :"))
        for row in myreader:
            if len(row)!=0:
                if int(row[0])==e:
                    print("============================")
                    print("NAME    :",row[1])
                    print("SALARY :",row[2])
                found=True
                break
        if not found:
            print("=========================")
            print("          EMPNO NOT FOUND")
            print("=========================")
            ans = input("Search More ? (Y)")
```

OUTPUT

```
    Enter Employee Number 1
    Enter Employee Name SAGAR
    Enter Employee Salary :200
    ## Data Saved... ##
    Add More ?Y
    Enter Employee Number 2
    Enter Employee Name DAKSH
    Enter Employee Salary :7000
    ## Data Saved... ##
    Add More ?N
    Enter Employee Number to search :2
    ===========================
    NAME          : DAKSH
    SALARY : 7000
    Enter Employee Number to search :3
    =========================
             EMPNO NOT FOUND
    =========================
Search More ? (Y)N
```

PROGRAM -19: PROGRAM TO IMPLEMENT STACK IN PYTHON USING LIST

```
def isEmpty(S):
    if len(S)==0:
        return True
    else:
        return False
def Push(S,item):
    S.append(item)
    top=len(S)-1
def Pop(S):
    if isEmpty(S):
        return "Underflow"
    else:
        val = S.pop()
        if len(S)==0:
            top=None
        else:
            top=len(S)-1
        return val
def Peek(S):
    if isEmpty(S):
        return "Underflow"
    else:
        top=len(S)-1
        return S[top]
def Show(S):
    if isEmpty(S):
        print("Sorry No items in Stack ")
    else:
        t = len(S)-1
        print("(Top)",end=' ')
        while(t>=0):
            print(S[t],"<==",end=' ')
            t-=1
        print()
# main begins here
S=[]    #Stack
top=None
while True:
    print("**** STACK DEMONSTRATION ******")
    print("1. PUSH ")
    print("2. POP")
    print("3. PEEK")
    print("4. SHOW STACK ")
    print("0. EXIT")
    ch = int(input("Enter your choice :"))
    if ch==1:
        val = int(input("Enter Item to Push :"))
        Push(S,val)
    elif ch==2:
        val = Pop(S)
        if val=="Underflow":
            print("Stack is Empty")
        else:
           print("\nDeleted Item was :",val)
    elif ch==3:
        val = Peek(S)
        if val=="Underflow":
            print("Stack Empty")
        else:
            print("Top Item :",val)
    elif ch==4:
        Show(S)
    elif ch==0:
        print("Bye")
        break
```

OUTPUT

```
 **** STACK DEMONSTRATION ******
1. PUSH
2. POP
3. PEEK
4. SHOW STACK
0. EXIT
Enter your choice :1
Enter Item to Push :10
**** STACK DEMONSTRATION ******
1. PUSH
2. POP
3. PEEK
4. SHOW STACK
0. EXIT
Enter your choice :1
Enter Item to Push :20
**** STACK DEMONSTRATION ******
1. PUSH
2. POP
3. PEEK
4. SHOW STACK
0. EXIT
```

```
Enter your choice :1
Enter Item to Push :30
**** STACK DEMONSTRATION ******
1. PUSH
2. POP
3. PEEK
4. SHOW STACK
0. EXIT
Enter your choice :4
(Top) 30 <== 20 <== 10 <==
**** STACK DEMONSTRATION ******
1. PUSH
2. POP
3. PEEK
4. SHOW STACK
0. EXIT
Enter your choice :2

Deleted Item was : 30
**** STACK DEMONSTRATION ******
1. PUSH
2. POP
3. PEEK
4. SHOW STACK
0. EXIT
Enter your choice :3
Top Item : 20
**** STACK DEMONSTRATION ******
1. PUSH
2. POP
3. PEEK
4. SHOW STACK
0. EXIT
Enter your choice :0
Bye
```

PROGRAM-20: PROGRAM TO IMPLEMENT QUEUE IN PYTHON USING LIST

```
def isEmpty(Q):
   if len(Q)==0:
      return True
   else:
      return False
def Enqueue(Q,item):
   Q.append(item)
   if len(Q)==1:
      front=rear=0
   else:
      rear=len(Q)-1
def Dequeue(Q):
   if isEmpty(Q):
      return "Underflow"
   else:
      val = Q.pop(0)
   if len(Q)==0:
      front=rear=None
   return val
def Peek(Q):
   if isEmpty(Q):
      return "Underflow"
   else:
      front=0
      return Q[front]
def Show(Q):
   if isEmpty(Q):
      print("Sorry No items in Queue ")
   else:
      t = len(Q)-1
      print("(Front)",end=' ')
      front = 0
      i=front
      rear = len(Q)-1
      while(i<=rear):
         print(Q[i],"==>",end=' ')
         i+=1
      print()
# Driver Code
Q=[]   #Queue
front=rear=None
while True:
   print("**** QUEUE DEMONSTRATION ******")
```

```
    print("1. ENQUEUE ")
    print("2. DEQUEUE")
    print("3. PEEK")
    print("4. SHOW QUEUE ")
    print("0. EXIT")
    ch = int(input("Enter your choice :"))
    if ch==1:
        val = int(input("Enter Item to Insert :"))
        Enqueue(Q,val)
    elif ch==2:
        val = Dequeue(Q)
        if val=="Underflow":
            print("Queue is Empty")
        else:
            print("\nDeleted Item was :",val)
    elif ch==3:
        val = Peek(Q)
        if val=="Underflow":
            print("Queue Empty")
        else:
            print("Front Item :",val)
    elif ch==4:
         Show(Q)
    elif ch==0:
        print("Bye")
        break
```

OUTPUT

```
**** QUEUE DEMONSTRATION ******
1. ENQUEUE
2. DEQUEUE
3. PEEK
4. SHOW QUEUE
0. EXIT
Enter your choice :1
Enter Item to Insert :10
**** QUEUE DEMONSTRATION ******
1. ENQUEUE
2. DEQUEUE
3. PEEK
4. SHOW QUEUE
0. EXIT
Enter your choice :1
Enter Item to Insert :20
```

```
**** QUEUE DEMONSTRATION ******
1. ENQUEUE
2. DEQUEUE
3. PEEK
4. SHOW QUEUE
0. EXIT
Enter your choice :1
Enter Item to Insert :30
**** QUEUE DEMONSTRATION ******
1. ENQUEUE
2. DEQUEUE
3. PEEK
4. SHOW QUEUE
0. EXIT
Enter your choice :4
(Front) 10 ==> 20 ==> 30 ==>
**** QUEUE DEMONSTRATION ******
1. ENQUEUE
2. DEQUEUE
3. PEEK
4. SHOW QUEUE
0. EXIT
Enter your choice :2

Deleted Item was : 10
**** QUEUE DEMONSTRATION ******
1. ENQUEUE
2. DEQUEUE
3. PEEK
4. SHOW QUEUE
0. EXIT
Enter your choice :3
Front Item : 20
**** QUEUE DEMONSTRATION ******
1. ENQUEUE
2. DEQUEUE
3. PEEK
4. SHOW QUEUE
0. EXIT
Enter your choice :
```

PROGRAM-21: READ ALL RECORDS AND DISPAY USING MYSQL CONNECTIVITY

```
#  use of fetchall() function
import mysql.connector
mydb = mysql.connector.connect(   host= 'localhost',   user='root',
password='12345',
  database = 'new'  )
if mydb.is_connected():
    print("Successfully Connected to MYSQL database")
else:
    print("Error Connecting to MYSQL database")
cur = mydb.cursor();
cur.execute("select * from student;")
data = cur.fetchall()
count=cur.rowcount
print("Total No. of Rrows retrieved :",count)
for row in data:
     print(row[0],"  ",row[1],"  ",row[2])
mydb.close()
```

OUTPUT

```
Successfully Connected to MYSQL database
Total No. of Rrows retrieved : 4
1   'NIKHIL'    90
2   'SAGAR'   5
3   'VISHU'  80
4   'DAKSH'    100
```

PROGRAM -22: READ RECORDS ONE BY ONE USING FETCHONE() AND DISPAY USING MYSQL CONNECTIVITY

```
#  use of fetchone() function
import mysql.connector
mydb = mysql.connector.connect(  host= 'localhost',  user='root', 
password='12345',
  database = 'new')
if mydb.is_connected():
    print("Successfully Connected to MYSQL database")
else:
    print("Error Connecting to MYSQL database")
cur = mydb.cursor();
cur.execute("select * from student;")
data = cur.fetchone()
count=cur.rowcount
print("Total No. of Rrows retrieved :",count)
print(data)
data = cur.fetchone()
count=cur.rowcount
print("Total No. of Rrows retrieved :",count)
print(data)
data = cur.fetchone()
count=cur.rowcount
print("Total No. of Rrows retrieved :",count)
print(data)
data = cur.fetchone()
count=cur.rowcount
print("Total No. of Rrows retrieved :",count)
print(data)
mydb.close()
```

OUTPUT

```
Successfully Connected to MYSQL database
Total No. of Rrows retrieved : 1
(1, ‘NIKHIL’, 90)
Total No. of Rrows retrieved : 2
(2, ‘SAGAR’, 5)
Total No. of Rrows retrieved : 3
(3, ‘VISHU’, 80)
Total No. of Rrows retrieved : 4
(4, ‘DAKSH’, 100)
```

PROGRAM-23: PROGRAM TO READ RECORDS ONE BY ONE USING FETCHMANY() AND DISPAY USING MYSQL CONNECTIVITY

```
#  use of fetchmany() function

import mysql.connector

mydb = mysql.connector.connect(  host= 'localhost',  user='root',
password='12345',

  database = 'new')

if mydb.is_connected():

    print("Successfully Connected to MYSQL database")

else:

    print("Error Connecting to MYSQL database")

cur = mydb.cursor();

cur.execute("select * from student;")

data = cur.fetchmany(3)

count=cur.rowcount

print("Total No. of Rrows retrieved :",count)

for row in data:

    print(row)

 mydb.close()
```

OUTPUT

```
Successfully Connected to MYSQL database

Total No. of Rrows retrieved : 3

(1, 'NIKHIL', 90)

(2, 'SAGAR', 5)

(3, 'VISHU’, 80)
```

PROGRAM-24: PROGRAM TO READ ALL RECORDS USING FETCHALL() USING WHERE CLAUSE AND DISPAY USING MYSQL CONNECTIVITY

```
import mysql.connector

mydb = mysql.connector.connect(  host= 'localhost',  user='root',
password='12345',

  database = 'new')

if mydb.is_connected():

    print("Successfully Connected to MYSQL database")

else:

    print("Error Connecting to MYSQL database")

cur = mydb.cursor();

st="select * from student where marks < %s" %(70,)

cur.execute(st)

data = cur.fetchall()

count=cur.rowcount

print("Total No. of Rows retrieved :",count)

for row in data:

    print(row)

 mydb.close()
```

OUTPUT

```
Successfully Connected to MYSQL database

Total No. of Rows retrieved : 2

(1, 'NIKHIL', 90)

(2, 'SAGAR', 5)
```

PROGRAM-25: PROGRAM TO CONNECT WITH DATABASE AND STORE RECORD OF EMPLOYEE AND DISPLAY RECORDS.

```
import mysql.connector as mycon
con = mycon.connect(host='localhost',user='root',password="12345")
cur = con.cursor()
cur.execute("create database if not exists company")
cur.execute("use company")
cur.execute("create table if not exists employee(empno int, name varchar(20), dept varchar(20),salary int)")
con.commit()
choice=None
while choice!=0:
    print("1. ADD RECORD ")
    print("2. DISPLAY RECORD ")
    print("0. EXIT")
    choice = int(input("Enter Choice :"))
    if choice == 1:
        e = int(input("Enter Employee  Number  :"))
        n = input("Enter Name :")
        d =  input("Enter  Department  :")
        s = int(input("Enter Salary :"))
        query="insert into employee values({},'{}','{}',{})".format(e,n,d,s)
        cur.execute(query)
        con.commit()
        print("## Data Saved ##")
    elif choice == 2:
        query="select * from employee"
        cur.execute(query)
        result = cur.fetchall()
        print("%10s"%"EMPNO","%20s"%"NAME","%15s"%"DEPARTMENT", "%10s"%"SALARY")
        for row in result:
            print("%10s"%row[0],"%20s"%row[1],"%15s"%row[2],"%10s"%row[3])
    elif choice==0:
        con.close()
        print("## Bye!! ##")
    else:
        print("## INVALID CHOICE ##")
```

OUTPUT

```
1. ADD RECORD
2. DISPLAY RECORD
0. EXIT
Enter Choice :1
Enter Employee  Number  :1
Enter Name :AMIT
Enter  Department  :PHYSICS
Enter Salary :20000
## Data Saved ##
1. ADD RECORD
2. DISPLAY RECORD
0. EXIT
Enter Choice :1
Enter Employee  Number  :2
Enter Name :SUMIT
Enter  Department  :CHEMISTRY
Enter Salary :30000
## Data Saved ##
1. ADD RECORD
2. DISPLAY RECORD
0. EXIT
Enter Choice :2
   EMPNO             NAME      DEPARTMENT         SALARY
     1               SAGAR        DANCE              10
     2               VISHU       ENGLISH           30000
1. ADD RECORD
2. DISPLAY RECORD
0. EXIT
Enter Choice :0
## Bye!! ##
```

PROGRAM-26: CONNECT WITH DATABASE AND SEARCH EMPLOYEE NUMBER IN TABLE EMPLOYEE AND DISPLAY RECORD, IF EMPNO NOT FOUND DISPLAY PROPER MESSAGE.

```
import mysql.connector as mycon
con = mycon.connect(host='localhost',user='root',password="12345",
database="company")
cur = con.cursor()
print("#"*40)
print("EMPLOYEE SEARCHING FORM")
print("#"*40)
print("\n\n")
ans='y'
while ans.lower()=='y':
   eno = int(input("ENTER EMPNO TO SEARCH :"))
   query="select * from employee where empno={}".format(eno)
   cur.execute(query)
   result = cur.fetchall()
   if cur.rowcount==0:
      print("Sorry! Empno not found ")
   else:
      print("%10s"%"EMPNO", "%20s"%"NAME","%15s"%"DEPARTMENT",
"%10s"%"SALARY")
      for row in result:
         print("%10s"%row[0],"%20s"%row[1],"%15s"%row[2],"%10s"%row[3])
   ans=input("SEARCH MORE (Y) :")
```

OUTPUT

```
########################################
EMPLOYEE SEARCHING FORM
########################################

ENTER EMPNO TO SEARCH :1
    EMPNO               NAME     DEPARTMENT    SALARY
       1                SAGAR       DANCE            10
SEARCH MORE (Y) :y
ENTER EMPNO TO SEARCH :2
    EMPNO               NAME     DEPARTMENT    SALARY
       2                VISHU       ENGLISH        30000
SEARCH MORE (Y) :n
```

PROGRAM-27: CONNECT WITH DATABASE AND UPDATE THE EMPLOYEE RECORD OF ENTERED EMPNO.

```
import mysql.connector as mycon
con = mycon.connect(host='localhost',user='root',password="12345", database="company")
cur = con.cursor()
print("EMPLOYEE UPDATION FORM")
ans='y'
while ans.lower()=='y':
    eno = int(input("ENTER EMPNO TO UPDATE :"))
    query="select * from employee where empno={}".format(eno)
    cur.execute(query)
    result = cur.fetchall()
    if cur.rowcount==0:
        print("Sorry! Empno not found ")
    else:
        print("%10s"%"EMPNO","%20s"%"NAME", "%15s"%"DEPARTMENT", "%10s"%"SALARY")
        for row in result:
            print("%10s"%row[0],"%20s"%row[1],"%15s"%row[2],"%10s"%row[3])
        choice=input("\n## ARE YOUR SURE TO UPDATE ? (Y) :")
        if choice.lower()=='y':
            print("== YOU CAN UPDATE ONLY DEPT AND SALARY ==")
            print("== FOR EMPNO AND NAME CONTACT ADMIN ==")
            d = input("ENTER NEW DEPARTMENT,(LEAVE BLANK IF NOT WANT TO CHANGE )")
            if d=="":
                d=row[2]
            try:
                s = int(input("ENTER NEW SALARY,(LEAVE BLANK IF NOT WANT TO
CHANGE ) "))
            except:
                s=row[3]
            query="update employee set dept='{}',salary={} where
empno={}".format(d,s,eno)
            cur.execute(query)
            con.commit()
            print("## RECORD UPDATED ## ")
        ans=input("UPDATE MORE (Y) :")
```

OUTPUT

```
    EMPLOYEE UPDATION FORM
    ENTER EMPNO TO UPDATE :1
      EMPNO          NAME     DEPARTMENT    SALARY
        1             SAGAR       DANCE           10
    ## ARE YOUR SURE TO UPDATE ? (Y) :Y
    == YOU CAN UPDATE ONLY DEPT AND SALARY ==
    == FOR EMPNO AND NAME CONTACT ADMIN ==
    ENTER NEW DEPARTMENT,(LEAVE BLANK IF NOT WANT TO CHANGE )
CLEANER
    ENTER NEW SALARY,(LEAVE BLANK IF NOT WANT TO CHANGE ) 20
    ## RECORD UPDATED ##
    UPDATE MORE (Y) :N
```

PROGRAM-28: CONNECT WITH DATABASE AND DELETE THE RECORD OF ENTERED EMPLOYEE NUMBER

```
    import mysql.connector as mycon
    con = mycon.connect(host='localhost',user='root',password="12345",
database="company")
    cur = con.cursor()
    print("EMPLOYEE DELETION FORM")
    ans='y'
    while ans.lower()=='y':
       eno = int(input("ENTER EMPNO TO DELETE :"))
       query="select * from employee where empno={}".format(eno)
       cur.execute(query)
       result = cur.fetchall()
       if cur.rowcount==0:
          print("Sorry! Empno not found ")
       else:
          print("%10s"%"EMPNO","%20s"%"NAME", "%15s"%"DEPARTMENT",
"%10s"%"SALARY")
          for row in result:
             print("%10s"%row[0],"%20s"%row[1],"%15s"%row[2],"%10s"%row[3])
             choice=input("\n## ARE YOUR SURE TO DELETE ? (Y) :")
             if choice.lower()=='y':
                query="delete from employee where empno={}".format(eno)
                cur.execute(query)
                con.commit()
                print("=== RECORD DELETED SUCCESSFULLY! ===")
       ans=input("DELETE MORE ? (Y) :")
```

OUTPUT

```
EMPLOYEE DELETION FORM
ENTER EMPNO TO DELETE :2
    EMPNO               NAME     DEPARTMENT    SALARY
       2                VISHU       ENGLISH       30000

## ARE YOUR SURE TO DELETE ? (Y) :Y
=== RECORD DELETED SUCCESSFULLY! ===
DELETE MORE ? (Y) :Y
ENTER EMPNO TO DELETE :2
Sorry! Empno not found DELETE MORE ? (Y) :N
```

THANK YOU