SAINIK SCHOOL AMBIKAPUR

INDEX

PRACTICAL FILE- COMPUTER SCIENCE (083)

LIST OF PRACTICALS (2022-23) CLASS-XII

Programming Language : Python

Sr. No	Name of Practical
SQL-1	SQL Queries for relations MOBILE-MASTER & MOBILE-STOCK
SQL-2	SQL Queries for relations TRAINER & COURSE
1	Input any number from user and calculate factorial of a number
2	Input any number from user and check it is Prime no. or not
3	Write a program to find sum of elements of List recursively
4	Write a program to calculate the nth term of Fibonacci series
5	Program to search any word in given string/sentence
6	Program to read and display file content line by line with each word separated by "#"
7	Program to read the content of file and display the total number of consonants, uppercase, vowels and lower case characters
8	Program to create binary file to store Rollno and Name, Search any Rollno and display name if Rollno found otherwise "Rollno not found"
9	Program to create binary file to store Rollno,Name and Marks and update marks of entered Rollno
10	Program to read the content of file line by line and write it to another file except for the lines contains "a" letter in it.
11	Program to create CSV file and store empno,name,salary and search any empno and display name,salary and if not found appropriate message.
12	Program to generate random number 1-6, simulating a dice
13	Program to implement Stack in Python using List
14	Program to implement Queue in Python using List
15	Program to take 10 sample phishing email, and find the most common word occurring
16	Program to create 21 Stick Game so that computer always wins
17	Program to connect with database and store record of employee and display records.
18	Program to connect with database and search employee number in table employee and display record, if empno not found display appropriate message.
19	Program to connect with database and update the employee record of entered empno.
20	Program to connect with database and delete the record of entered employee number.

SQL-1 : Write SQL queries for (i) to (iv) and find outputs for SQL queries (v) to (viii), which are based on the tables.

TRAINER

TID	TNAME	CITY	HIREDATE	SALARY
101	SUNAINA	MUMBAI	1998-10-15	90000
102	ANAMIKA	DELHI	1994-12-24	80000
103	DEEPTI	CHANDIGARG	2001-12-21	82000
104	MEENAKSHI	DELHI	2002-12-25	78000
105	RICHA	MUMBAI	1996-01-12	95000
106	MANIPRABHA	CHENNAI	2001-12-12	69000

COURSE

CID	CNAME	FEES	STARTDATE	TID
C201	AGDCA	12000	2018-07-02	101
C202	ADCA	15000	2018-07-15	103
C203	DCA	10000	2018-10-01	102
C204	DDTP	9000	2018-09-15	104
C205	DHN	20000	2018-08-01	101
C206	O LEVEL	18000	2018-07-25	105

(i)	Display the Trainer Name, City & Salary in descending order of their Hiredate.
(ii)	To display the TNAME and CITY of Trainer who joined the Institute in the month of December 2001.
(iii)	To display TNAME, HIREDATE, CNAME, STARTDATE from tables TRAINER and COURSE of all those courses whose FEES is less than or equal to 10000.
(iv)	To display number of Trainers from each city.
(v)	SELECT TID, TNAME, FROM TRAINER WHERE CITY NOT IN('DELHI', 'MUMBAI');
(vi)	SELECT DISTINCT TID FROM COURSE;
(vii)	SELECT TID, COUNT(*), MIN(FEES) FROM COURSE GROUP BY TID HAVING COUNT(*)>1;
(viii)	SELECT COUNT(*), SUM(FEES) FROM COURSE WHERE STARTDATE< '2018-09-15';

SQL-2 : Write SQL queries for (i) to (iv) and find outputs for SQL queries (v) to (viii), which are based on the tables.

MobileMaster

M_Id	M_Company	M_Name	M_Price	M_Mf_Date
MB001	Samsung	Galaxy	4500	2013-02-12
MB003	Nokia	N1100	2250	2011-04-15
MB004	Micromax	Unite3	4500	2016-10-17
MB005	Sony	XperiaM	7500	2017-11-20
MB006	Oppo	SelfieEx	8500	2010-08-21

MobileStock

S_Id	M_Id	M_Qty	M_Supplier
S001	MB004	450	New Vision
S002	MB003	250	Praveen Gallery
S003	MB001	300	Classic Mobile Store
S004	MB006	150	A-one Mobiles
S005	MB003	150	The Mobile
S006	MB006	50	Mobile Centre

Write the SQL query for questions from (i) to (iv) & write the output of SQL command for questions from (v) to (viii) given below:-

(i) Display the Mobile company, Mobile name & price in descending order of their manufacturing date.

(ii) List the details of mobile whose name starts with „S".

(iii) Display the Mobile supplier & quantity of all mobiles except „MB003".

(iv) To display the name of mobile company having price between 3000 & 5000.

(v) SELECT M_Id, SUM(M_Qty) FROM MobileStock GROUP BY M_Id;

(vi) SELECT MAX(M_Mf_Date), MIN(M_Mf_Date) FROM MobileMaster;

(vii) SELECT M1.M_Id, M1.M_Name, M2.M_Qty, M2.M_Supplier FROM MobileMaster M1, MobileStock M2 WHERE M1.M_Id=M2.M_Id AND M2.M_Qty>=300;

(viii) SELECT AVG(M_Price) FROM MobileMaster;

Date : Experiment No: 1

Program 1: Input any number from user and calculate factorial of a number

```
# Program to calculate factorial of entered number
num = int(input("Enter any number :"))
fact = 1
n = num             # storing num in n for printing
while num>1:        # loop to iterate from n to 2
        fact = fact * num
        num-=1

print("Factorial of ", n , " is :",fact)
```

OUTPUT

```
Enter any number :6
Factorial of  6  is : 720
```

Date : Experiment No:2

Program 1: Input any number from user and check it is Prime no. or not

```
#Program to input any number from user
#Check it is Prime number of not
import math
num = int(input("Enter any number :"))
isPrime=True
for i in range(2,int(math.sqrt(num))+1):
        if num % i == 0:
                isPrime=False

if isPrime:
        print("## Number is Prime ##")
else:
        print("## Number is not Prime ##")
```

OUTPUT

```
Enter any number :117
## Number is not Prime ##
>>>
Enter any number :119
## Number is not Prime ##
>>>
Enter any number :113
## Number is Prime ##
>>>
Enter any number :7
## Number is Prime ##
>>>
Enter any number :19
## Number is Prime ##
```

Date : Experiment No: 3

Program : Write a program to find sum of elements of List recursively

```
#Program to find sum of elements of list recursively
def findSum(lst,num):
    if num==0:
        return 0
    else:
        return lst[num-1]+findSum(lst,num-1)


mylist = []                 # Empty List
#Loop to input in list
num = int(input("Enter how many number :"))
for i in range(num):
    n = int(input("Enter Element "+str(i+1)+":"))
    mylist.append(n)    #Adding number to list

sum = findSum(mylist,len(mylist))
print("Sum of List items ",mylist, " is :",sum)
```

OUTPUT

```
Enter how many number :6
Enter Element 1:10
Enter Element 2:20
Enter Element 3:30
Enter Element 4:40
Enter Element 5:50
Enter Element 6:60
Sum of List items  [10, 20, 30, 40, 50, 60]  is : 210
```

Date : Experiment No: 4

Program 1: Write a program to calculate the n^{th} term of Fibonacci series

```
#Program to find 'n'th term of fibonacci series
#Fibonacci series : 0,1,1,2,3,5,8,13,21,34,55,89,...
#nth term will be counted from 1 not 0

def nthfiboterm(n):
    if n<=1:
        return n
    else:
        return (nthfiboterm(n-1)+nthfiboterm(n-2))

num = int(input("Enter the 'n' term to find in fibonacci :"))
term =nthfiboterm(num)
print(num,"th term of fibonacci series is :",term)
```

OUTPUT

```
Enter the 'n' term to find in fibonacci :10
10 th term of fibonacci series is : 55
```

Date : Experiment No: 5

Program : Program to search any word in given string/sentence

```
#Program to 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

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 :my computer your computer our computer everyones computer
Enter word to search in sentence :computer
##  computer  occurs  4  times ##

Enter any sentence :learning python is fun
Enter word to search in sentence :java
## Sorry!  java  not present
```

Date : Experiment No: 6

Program 1: Program to read and display file content line by line with each word separated by '#'

```
#Program to read content of file line by line
#and display each word separated by '#'

f = open("file1.txt")

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

NOTE : if the original content of file is:
India is my country
I love python
Python learning is fun

OUTPUT
India#is#my#country#
I#love#python#
Python#learning#is#fun#

Date : Experiment No: 7

Program 1: Program to read the content of file and display the total number of consonants, uppercase, vowels and lower case characters'

```
#Program to read content of file
#and display total number of vowels, consonants, lowercase and uppercase characters

f = open("file1.txt")
v=0
c=0
u=0
l=0
o=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    n    :",c)
print("Total Capital letters in file    :",u)
print("Total Small letters in file      :",l)
print("Total Other than letters         :",o)
f.close()
```

NOTE : if the original content of file is:

India is my country
I love python
Python learning is fun
123@

OUTPUT

```
Total Vowels in file             : 16
Total Consonants in file    n    : 30
Total Capital letters in file    : 2
Total Small letters in file      : 44
Total Other than letters         : 4
```

Date : Experiment No: 8

Program 1: Program to create binary file to store Rollno and Name, Search any Rollno and display name if Rollno found otherwise "Rollno not found"

```
#Program to create a binary file to store Rollno and name
#Search for Rollno and display 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 :")
        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 :Amit
Add More ?(Y)y

Enter Roll Number :2
Enter Name :Jasbir
Add More ?(Y)y

Enter Roll Number :3
Enter Name :Vikral
Add More ?(Y)n

Enter Roll number to search :2
## Name is : Jasbir  ##
Search more ?(Y) :y

Enter Roll number to search :1
## Name is : Amit  ##
Search more ?(Y) :y

Enter Roll number to search :4
####Sorry! Roll number not found ####
Search more ?(Y) :n
```

Date : Experiment No: 9

Program 1: Program to create binary file to store Rollno,Name and Marks and update marks of entered Rollno

```
#Program to create a binary file to store Rollno and name
#Search for Rollno and display 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
ans='y'
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 :Amit
Enter Marks :99
Add More ?(Y)y

Enter Roll Number :2
Enter Name :Vikrant
Enter Marks :88
Add More ?(Y)y

Enter Roll Number :3
Enter Name :Nitin
Enter Marks :66
Add More ?(Y)n

Enter Roll number to update :2
## Name is : Vikrant  ##
## Current Marks is : 88  ##
Enter new marks :90
## Record Updated ##
Update more ?(Y) :y

Enter Roll number to update :2
## Name is : Vikrant  ##
## Current Marks is : 90  ##
Enter new marks :95
## Record Updated ##
Update more ?(Y) :n
```

Date : Experiment No: 10

Program 1: Program to read the content of file line by line and write it to another file except for the lines contains 'a' letter in it.

```
#Program to read line from file and write it to another line
#Except for those line which contains letter 'a'

f1 = open("file2.txt")
f2 = open("file2copy.txt","w")

for line in f1:
        if 'a' not in line:
                f2.write(line)
print("## File Copied Successfully! ##")
f1.close()
f2.close()
```

NOTE: Content of file2.txt

a quick brown fox
one two three four
five six seven
India is my country
eight nine ten
bye!

OUTPUT

```
## File Copied Successfully! ##
```

NOTE: After copy content of file2copy.txt

one two three four
five six seven
eight nine ten
bye!

Date : Experiment No: 11

Program 1: Program to create CSV file and store empno,name,salary and search any empno and display name,salary and if not found appropriate message.

```
import csv
with open('myfile.csv',mode='a') 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)")
```

```
Enter Employee Number 1
Enter Employee Name Amit
Enter Employee Salary :90000
## Data Saved... ##
Add More ?y
Enter Employee Number 2
Enter Employee Name Sunil
Enter Employee Salary :80000
## Data Saved... ##
Add More ?y
Enter Employee Number 3
Enter Employee Name Satya
Enter Employee Salary :75000
## Data Saved... ##
Add More ?n
Enter Employee Number to search :2
============================
NAME   : Sunil
SALARY : 80000
Search More ? (Y)y
Enter Employee Number to search :3
============================
NAME   : Satya
SALARY : 75000
Search More ? (Y)y
Enter Employee Number to search :4
=========================
     EMPNO NOT FOUND
=========================
Search More ? (Y)n
```

Date : Experiment No: 12

Program 1: Program to generate random number 1-6, simulating a dice

```
# Program to generate random number between 1 - 6
# To simulate the dice
import random
import time
print("Press CTRL+C to stop the dice ")
play='y'
while play=='y':
        try:
                while True:
                        for i in range(10):
                                print()
                        n = random.randint(1,6)
                        print(n,end='')
                        time.sleep(.00001)
        except KeyboardInterrupt:
                print("Your Number is :",n)
                ans=input("Play More? (Y) :")
                if ans.lower()!='y':
                        play='n'
                        break
```

OUTPUT

```
4Your Number is : 4
Play More? (Y) :y
Your Number is : 3
Play More? (Y) :y
Your Number is : 2
Play More? (Y) :n
```

Date : Experiment No: 13

Program 1: 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
```

Cont…

```
**** 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 :3
Top Item : 30

**** 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 :4
(Top) 20 <== 10 <==

**** STACK DEMONSTRATION ******
1. PUSH
2. POP
3. PEEK
4. SHOW STACK
0. EXIT
Enter your choice :0
Bye
```

Date : Experiment No: 14

Program 1: 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()
```

Cont…

```
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
```

Cont…

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

Date : Experiment No: 15

Program 1: Program to take 10 sample phishing email, and find the most common word occurring

```
#Program to take 10 sample phishing mail
#and count the most commonly occuring word
phishingemail=[
        "jackpotwin@lottery.com",
        "claimtheprize@mymoney.com",
        "youarethewinner@lottery.com",
        "luckywinner@mymoney.com",
        "spinthewheel@flipkart.com",
        "dealwinner@snapdeal.com"
        "luckywinner@snapdeal.com"
        "luckyjackpot@americanlottery.com"
        "claimtheprize@lootolottery.com"
        "youarelucky@mymoney.com"
        ]
myd={}
for e in phishingemail:
        x=e.split('@')
        for w in x:
                if w not in myd:
                        myd[w]=1
                else:
                        myd[w]+=1
key_max = max(myd,key=myd.get)
print("Most Common Occuring word is :",key_max)
```

OUTPUT

Most Common Occuring word is : mymoney.com

Date : Experiment No: 16

Program 1: Program to create 21 Stick Game so that computer always wins

Rule of Game (Total Sticks = 21):

1) User and Computer both can pick stick one by one
2) Maximum stick both can pick is 4 i.e. 1 to 4
3) Anyone with last stick will be the looser

```
def PrintStick(n):
        print("o "*n)
        print("| "*n)
        print("| "*n)
        print("| "*n)
        print("| "*n)
TotalStick=21
win=False
humanPlayer=True
print("==== Welcome To Stick Picking Game :: Computer Vs User =====")
print("Rule: 1) Both User and Computer can pick sticks between 1 to 4 at a time")
print("      2) Whosoever picks the last stick will be the looser")
print("==== Lets Begin ======")
playerName = input("Enter Your Name :")
userPick=0
PrintStick(TotalStick)
while win==False:
    if humanPlayer==True:
        print("You Can Pick stick between 1 to 4")
        userPick=0
        while userPick<=0 or userPick>4:
            userPick = int(input(playerName +": Enter Number of Stick to Pick"))
        TotalStick=TotalStick - userPick
        humanPlayer=False
        PrintStick(TotalStick)
        print("*"*60)
        input("Press any key...")
    else:
        computerPick = (5-userPick)
        print("Computer Picks : ",computerPick," Sticks ")
        TotalStick=TotalStick -computerPick
        PrintStick(TotalStick)
        if TotalStick==1:
            print("## WINNER : COMPUTER ##")
            win=True
        print("*"*60)
        input("Press any key...")
        humanPlayer=True
```

OUTPUT

```
==== Welcome To Stick Picking Game :: Computer Vs User =====
Rule: 1) Both User and Computer can pick sticks between 1 to 4 at a time
      2) Whosoever picks the last stick will be the looser
==== Lets Begin ======
Enter Your Name :RAJ
o o o o o o o o o o o o o o o o o o o o o o o
| | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | |
You Can Pick stick between 1 to 4
RAJ: Enter Number of Stick to Pick3
o o o o o o o o o o o o o o o o o o o o o
| | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | |
************************************************************
Press any key...
Computer Picks :  2  Sticks
o o o o o o o o o o o o o o o o o o o o o
| | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | |
************************************************************
Press any key...
You Can Pick stick between 1 to 4
RAJ: Enter Number of Stick to Pick4
o o o o o o o o o o o o o o o o o o o
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
************************************************************
Press any key...
Computer Picks :  1  Sticks
o o o o o o o o o o o o o o o o o o
| | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | |
************************************************************
Press any key...
You Can Pick stick between 1 to 4
RAJ: Enter Number of Stick to Pick2
o o o o o o o o o
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |
```

```
************************************************************
Press any key...
Computer Picks :  3  Sticks
o o o o o o
| | | | | |
| | | | | |
| | | | | |
| | | | | |
************************************************************
Press any key...
You Can Pick stick between 1 to 4
RAJ: Enter Number of Stick to Pick3
o o o
| | |
| | |
| | |
| | |
************************************************************
Press any key...
Computer Picks :  2  Sticks
o
|
|
|
|
## WINNER : COMPUTER ##
************************************************************
Press any key...
```

Date : Experiment No: 17

Program 1: Program to connect with database and store record of employee and display records.

```
import mysql.connector as mycon
con = mycon.connect(host='127.0.0.1',user='root',password="admin")
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 :SALES
Enter Salary :9000
## Data Saved ##
1. ADD RECORD
2. DISPLAY RECORD
0. EXIT
Enter Choice :1
Enter Employee Number :2
Enter Name :NITIN
Enter Department :IT
Enter Salary :80000
## Data Saved ##
1. ADD RECORD
2. DISPLAY RECORD
0. EXIT
Enter Choice :2
    EMPNO              NAME     DEPARTMENT     SALARY
        1              AMIT             SALES       9000
        2             NITIN                IT      80000
1. ADD RECORD
2. DISPLAY RECORD
0. EXIT
Enter Choice :0
## Bye!! ##
```

Date : Experiment No: 18

Program 1: Program to connect with database and search employee number in table employee and display record, if empno not found display appropriate message.

```
import mysql.connector as mycon
con = mycon.connect(host='127.0.0.1',user='root',password="admin",
      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              AMIT          SALES       9000
SEARCH MORE (Y) :y
ENTER EMPNO TO SEARCH :2
   EMPNO              NAME     DEPARTMENT     SALARY
       2             NITIN             IT      80000
SEARCH MORE (Y) :y
ENTER EMPNO TO SEARCH :4
Sorry! Empno not found
SEARCH MORE (Y) :n
```

Date : Experiment No: 19

Program 1: Program to connect with database and update the employee record of entered empno.

```
import mysql.connector as mycon
con = mycon.connect(host='127.0.0.1',user='root',password="admin",
      database="company")
cur = con.cursor()
print("#"*40)
print("EMPLOYEE UPDATION FORM")
print("#"*40)
print("\n\n")
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 :2
   EMPNO            NAME     DEPARTMENT     SALARY
      2            NITIN           IT      90000

## 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 )
ENTER NEW SALARY,(LEAVE BLANK IF NOT WANT TO CHANGE )
## RECORD UPDATED ##
UPDATE MORE (Y) :y

ENTER EMPNO TO UPDATE :2
   EMPNO            NAME     DEPARTMENT     SALARY
      2            NITIN           IT      90000

## 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 )SALES
ENTER NEW SALARY,(LEAVE BLANK IF NOT WANT TO CHANGE )
## RECORD UPDATED ##
UPDATE MORE (Y) :Y

ENTER EMPNO TO UPDATE :2
   EMPNO            NAME     DEPARTMENT     SALARY
      2            NITIN          SALES      90000

## 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 )
ENTER NEW SALARY,(LEAVE BLANK IF NOT WANT TO CHANGE ) 91000
## RECORD UPDATED ##
UPDATE MORE (Y) :Y

ENTER EMPNO TO UPDATE :2
   EMPNO            NAME     DEPARTMENT     SALARY
      2            NITIN          SALES      91000

## ARE YOUR SURE TO UPDATE ? (Y) :N
UPDATE MORE (Y) :N
```

Date : Experiment No: 20

Program 1: Program to connect with database and delete the record of entered employee number.

```
import mysql.connector as mycon
con = mycon.connect(host='127.0.0.1',user='root',password="admin",
        database="company")
cur = con.cursor()
print("#"*40)
print("EMPLOYEE DELETION FORM")
print("#"*40)
print("\n\n")
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             NITIN          SALES      91000

## 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
```