

KENDRIYA VIDYALAYA ONGC PANVEL

CLASS XII

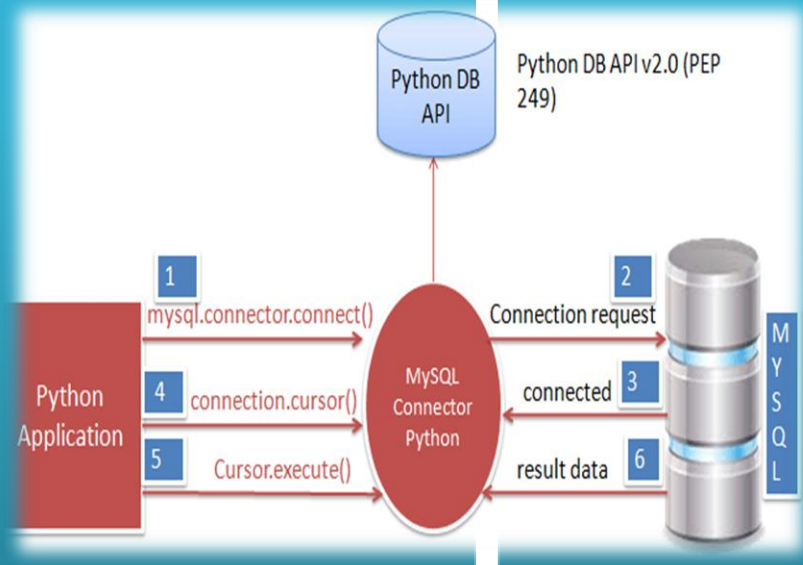
COMPUTER SCIENCE

PRACTICAL LIST

FOR SESSION 2020-21

PREPARED BY:

MRS. SHRUTI SRIVASTAVA, PGT (CS), KV ONGC PANVEL



INDEX

Sr. No.	Name of the Experiment/Activity	Date of Experiment	Data of Submission
1.	Write a function to check prime, calculate factorial and Fibonacci series.	07/07/2020	14/07/2020
2.	Program to perform various operations on the list.	14/07/2020	21/07/2020
3.	Program to create a package for calculation of area of triangle, rectangle and circle using various modules.	21/07/2020	28/07/2020
4.	Program to read a text file and count and display different types of characters in the file.	28/07/2020	04/08/2020
5.	Write a Python program to find smallest word, biggest word words of specific length, count the occurrence of specific words from a text file.	04/08/2020	11/08/2020
6.	Write a program to read text file and display and count different lines from text file.	11/08/2020	18/08/2020
7.	Write a program to append, count and display specific records in text file.	18/08/2020	25/08/2020
8.	Write a program to modify contents of the text file and copy contents from one file to another file based on condition.	25/08/2020	31/08/2020
9.	Write a program to create a binary file using list and perform search operation on it.	31/08/2020	07/09/2020
10.	Write a program to create a binary file using dictionary and perform read, append and search operations on it.	07/09/2020	14/09/2020

Class XII Computer Science Practical List for Session 2020-21

11.	Write a program to perform update operation on the binary file.	14/09/2020	21/09/2020
12.	Write a program to perform delete operation on the binary file.	21/09/2020	26/09/2020
13.	Write a menu driven program to create csv file and append student records such as roll number, marks, total and percentage.	26/09/2020	30/09/2020
14.	Write a program to create csv file and perform search operation on csv file using different criteria.	30/09/2020	05/10/2020
15.	Write a program to copy contents containing phonebook of a csv file into another csv file.	05/10/2020	10/10/2020
16.	Write a program to implement stack in Python using list and perform Push and Pop operations on it.	10/10/2020	12/10/2020
17.	Implementation of various SQL commands on different tables in SQL database.	12/10/2020	19/10/2020
18.	Write a program to connect with database, create table student, store records of students and display records.	19/10/2020	24/10/2020
19.	Write a program to connect with database and search records using (+) operator, format {} and %s and display the matched records.	24/10/2020	28/10/2020
20.	Write a program to connect with database and update and delete the record from database.	28/10/2020	02/11/2020

Practical 1-Write a function to check prime, calculate factorial and Fibonacci series.

Solution:

```
def prime():
    num=int(input("\nEnter the number:-"))
    if num>1:
        for i in range(2,num):
            if num%i==0:
                print(num,"is not a prime number")
                break
        else:
            print(num,"is a prime number")

def fact():
    num=int(input("Enter the number:-"))
    fact1=1
    if num<0:
        print("Factorial does not exist for negetive number")
    elif num==0:
        print("The factorial of 0 is 1")
    else:
        for i in range(1,num+1):
            fact1=fact1*i
        print("The factorial of",num,"is",fact1)

def fibo(num):
    f=[]
    if num<0:
        print("Incorrect input")
    elif num==0:
        return 0
    elif num==1:
        return 1
```

```
#code continue
```

```
    else:
        a,b=0,1
        f.append(a)
        f.append(b)
        for i in range(0,n-2):
            c=a+b
            f.append(c)
            a,b=b,c
        return f
opt='y'
while opt=='y':
    print("\nMenu")
    print("1)To check if a given number is prime or not")
    print("2)To find the the factorial of a number")
    print("3)To calculate the fibonacci series upto nth term")
    ch=int(input("Enter the Option You Choose(1,2,3):-"))
    if ch==1:
        prime()
    elif ch==2:
        fact()
    elif ch==3:
        n=int(input("\nEnter number of terms of fibonacci series:-"))
        print("The fibonnaci series of ",n,"terms is",fibo(n))
    else:
        print("Wrong choice")
        break
opt=input("\nDo you want to continue(y/n):")
```

Class XII Computer Science Practical List for Session 2020-21

Output:

Menu

```
1)To check if a given number is prime or not
2)To find the the factorial of a number
3)To calculate the fibonacci series upto nth term
Enter the Option You Choose(1,2,3):-1
```

```
Enter the number:-13
13 is a prime number
```

```
Do you want to continue(y/n):y
```

Menu

```
1)To check if a given number is prime or not
2)To find the the factorial of a number
3)To calculate the fibonacci series upto nth term
Enter the Option You Choose(1,2,3):-1
```

```
Enter the number:-24
24 is not a prime number
```

```
Do you want to continue(y/n):y
```

Menu

```
1)To check if a given number is prime or not
2)To find the the factorial of a number
3)To calculate the fibonacci series upto nth term
Enter the Option You Choose(1,2,3):-2
Enter the number:-7
The factorial of 7 is 5040
```

```
Do you want to continue(y/n):y
```

Menu

```
1)To check if a given number is prime or not
2)To find the the factorial of a number
3)To calculate the fibonacci series upto nth term
Enter the Option You Choose(1,2,3):-3
```

```
Enter number of terms of fibonacci series:-8
The fibonnaci series of 8 terms is [0, 1, 1, 2, 3, 5, 8, 13]
```

```
Do you want to continue(y/n):n
```

Practical 2-Program to perform following operations on the list-

1. To find sum and average of the list
2. Double the odd values and half even values of a list
3. Left shift operation on list by given number
4. Right shift operation on list by given number

Solution:

```
def sumavglist(list1):
    sum1 = 0
    for n in list1:
        sum1 = sum1 +n
    average = sum1 / len(list1)
    print("\nOriginal list",list1)
    print ("Sum of list element is : ", sum1)
    print ("Average of list element is ", average )

def changelist(list1):
    print("\nOriginal list",list1)
    for j in range(len(list1)):
        if(list1[j] % 2 == 0):
            list1[j]=list1[j]/2
        else:
            list1[j]=list1[j]*2

    print("\nThe list after Double the odd values and half even values")
    print(list1)

def LShift(list1,n):
    print("\nOriginal list",list1)
    L=len(list1)
    for x in range(0,n):
        y=list1[0]
        for i in range(0,L-1):
            list1[i]=list1[i+1]
        list1[L-1]=y
    print("\nList after left shift by ",n,"is=",list1)

def RShift(list1,n):
    print("\nOriginal list",list1)
    L=len(list1)
    for x in range(0,n):
        y=list1[L-1]
        for i in range(L-1,0,-1):
            list1[i]=list1[i-1]
        list1[0]=y
    print("\nList after right shift by ",n,"is=",list1)
```

#function call

```
my_list = [11,22,33,44,55,66,77]
choice = 0
while True :
    print("\nMenu")
    print("1. To find sum and average of the list ")
    print("2. Double the odd values and half even values of a list ")
    print("3. Left shift operation on list")
    print("4. Right shift operation on list")
    print("5. Exit ")
    choice = int(input("Enter the choice (1-5) : "))
    if choice == 1 : #To find sum and average of the given list.
        sumavglist(my_list)

    elif choice == 2 :
        changelst(my_list)

    elif choice == 3 :
        my_list = [11,22,33,44,55,66,77]
        m=int(input("Enter the value by which all elements of the list are shifted to left:"))
        LShift(my_list,m)

    elif choice == 4 :
        my_list = [11,22,33,44,55,66,77]
        m=int(input("Enter the value by which all elements of the list are shifted to right:"))
        RShift(my_list,m)

    elif choice==5:
        break
    else :
        print("Invalid choice")
        print("\n Press any key to continue.....")
        ch = input()
```


Class XII Computer Science Practical List for Session 2020-21

Output:

Menu

1. To find sum and average of the list
2. Double the odd values and half even values of a list
3. Left shift operation on list
4. Right shift operation on list
5. Exit

Enter the choice (1-5) : 1

Original list [11, 22, 33, 44, 55, 66, 77]

Sum of list element is : 308

Average of list element is 44.0

Menu

1. To find sum and average of the list
2. Double the odd values and half even values of a list
3. Left shift operation on list
4. Right shift operation on list
5. Exit

Enter the choice (1-5) : 2

Original list [11, 22, 33, 44, 55, 66, 77]

The list after Double the odd values and half even values
[22, 11.0, 66, 22.0, 110, 33.0, 154]

Menu

1. To find sum and average of the list
2. Double the odd values and half even values of a list
3. Left shift operation on list
4. Right shift operation on list
5. Exit

Enter the choice (1-5) : 3

Enter the value by which all elements of the list are shifted to left:2

Original list [11, 22, 33, 44, 55, 66, 77]

List after left shift by 2 is= [33, 44, 55, 66, 77, 11, 22]

Menu

1. To find sum and average of the list
2. Double the odd values and half even values of a list
3. Left shift operation on list
4. Right shift operation on list
5. Exit

Enter the choice (1-5) : 4

Enter the value by which all elements of the list are shifted to right:3

Original list [11, 22, 33, 44, 55, 66, 77]

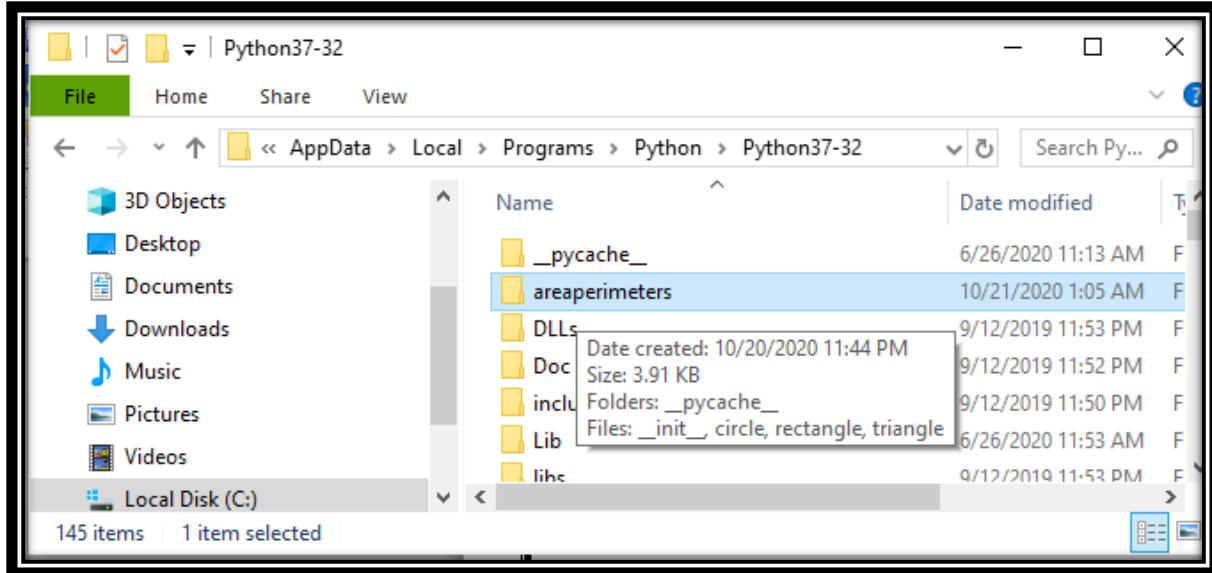
List after right shift by 3 is= [55, 66, 77, 11, 22, 33, 44]

Class XII Computer Science Practical List for Session 2020-21

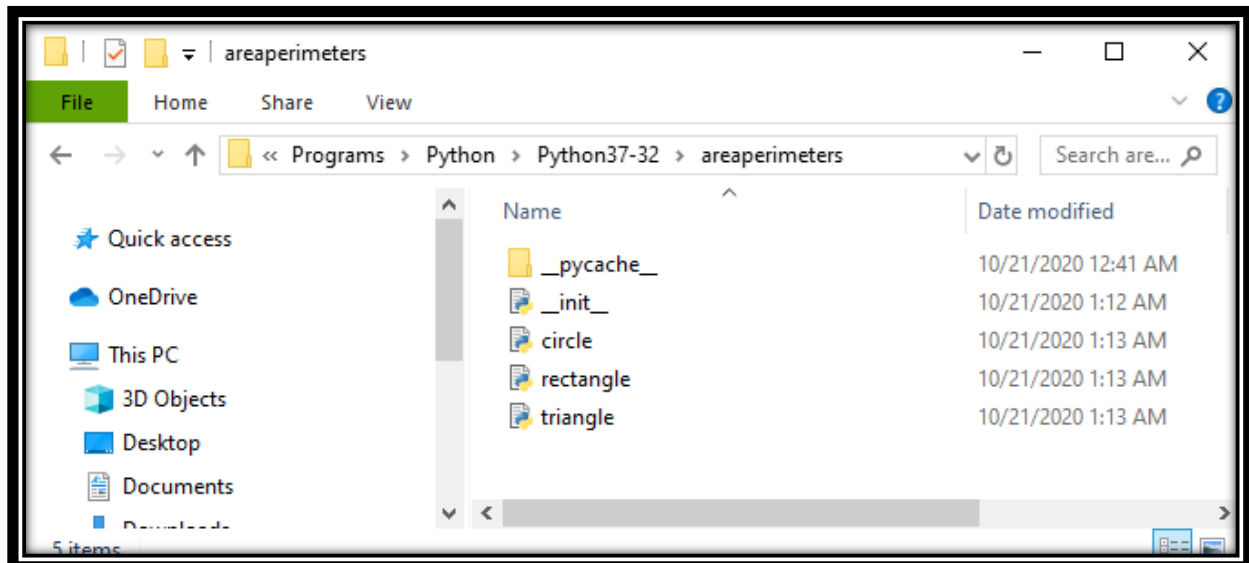
Practical 3- Write a program to create a package for calculation of area of triangle, rectangle and circle using various modules.

Solution:

1. Create a folder <areaperimeters> in Python37-32 folder



2. Create 3 modules for triangle, rectangle and circle and `__init__.py` file inside folder areaperimeters



triangle.py

```
'''Write functions to create a module for calculation of area and perimeter of triangle. '''
```

```
def area():
    print('FOR AREA OF TRIANGLE!!!')
    b=int(input('ENTER THE BASE ='))
    h=int(input('ENTER THE HEIGHT='))
    area=(b*h)/2
    print('THE AREA OF TRIANGLE IS ::',area)

def perimeter():
    print('FOR PERIMETER OF TRIANGLE!!!')
    a=int(input('ENTER THE LENGTH OF SIDE1 ='))
    b=int(input('ENTER THE LENGTH OF SIDE2 ='))
    c=int(input('ENTER THE LENGTH OF SIDE3 ='))
    peri=(a+b+c)
    print('THE PERIMETER OF TRIANGLE IS ::',peri)
```

rectangle.py

```
'''Write function to create a module for calculation of area and perimeter of rectangle. '''
```

```
def area():
    print('FOR AREA OF RECTANGLE!!!')
    a=int(input('ENTER THE BASE ='))
    b=int(input('ENTER THE LENGTH='))
    area=a*b
    print('THE AREA OF RECTANGLE IS ::',area)

def perimeter():
    print('FOR PERIMETER OF RECTANGLE!!!')
    a=int(input('ENTER THE BASE ='))
    b=int(input('ENTER THE LENGTH='))
    peri=2*(a+b)
    print('THE PERIMETER OF RECTANGLE IS ::',peri)
```

circle.py

```
'''Functions to create a module for calculation
of area and perimeter of circle.'''

def area():
    print('FOR AREA OF CIRCLE!!!')
    r=int(input('ENTER THE RADIUS ='))
    area=3.14*(r**2)
    print('AREA OF CIRCLE::',area)

def perimeter():
    print('FOR PERIMETER OF CIRCLE!!!')
    r=int(input('ENTER THE RADIUS ='))
    peri=3.14*(r*2)
    print('PERIMETER OF CIRCLE::',peri)
```

shape.py

```
'''Write a program to create a package for calculation of area of traingle,
rectangle and circle using various modules.'''
#importing modules from package
from areaperimeters import triangle,rectangle,circle

print("*****Area calculation*****")
triangle.area()
print()
rectangle.area()
print()
circle.area()
print("*****Area calculation*****")
print("\n*****Perimeter calculation*****")
triangle.perimeter()
print()
rectangle.perimeter()
print()
circle.perimeter()
print("*****Perimeter calculation*****")
```

Output:

```
*****Area calculation*****  
FOR AREA OF TRIANGLE!!!  
ENTER THE BASE =12  
ENTER THE HEIGHT=12  
THE AREA OF TRIANGLE IS :: 72.0
```

```
FOR AREA OF RECTANGLE!!!  
ENTER THE BASE =10  
ENTER THE LENGTH=12  
THE AREA OF RECTANGLE IS :: 120
```

```
FOR AREA OF CIRCLE!!!  
ENTER THE RADIUS =12  
AREA OF CIRCLE:: 452.16  
*****Area calculation*****
```

```
*****Perimeter calculation*****  
FOR PERIMETER OF TRIANGLE!!!  
ENTER THE LENGTH OF SIDE1 =3  
ENTER THE LENGTH OF SIDE2 =4  
ENTER THE LENGTH OF SIDE3 =5  
THE PERIMETER OF TRIANGLE IS :: 12
```

```
FOR PERIMETER OF RECTANGLE!!!  
ENTER THE BASE =44  
ENTER THE LENGTH=24  
THE PERIMETER OF RECTANGLE IS :: 136
```

```
FOR PERIMETER OF CIRCLE!!!  
ENTER THE RADIUS =7  
PERIMETER OF CIRCLE:: 43.96  
*****Perimeter calculation*****
```

Class XII Computer Science Practical List for Session 2020-21

Practical 4-Read a text file and display the number of digits, alphabets, vowels, consonants, uppercase , lowercase characters, spaces and other characters in the file.

Solution:

```
def writefile():      #function to write into file
    f=open("poem.txt","w")
    str1="""1000s of Emotions in the Eyes,\nMany are there to Hear the Truth of my Life,
Everyone has a Hope that One Day I'll Shine,\nJust not able to make a Decision,
What to take Further...What to leave Behind!"""
    f.write(str1)
    f.close()

def readfile():      #function to read from file
    f=open("poem.txt","r")
    s=f.read()
    L1=len(s)
    print("Contents of the file poem.txt are:\n")
    print(s)
    print("\nSize of the file is=",L1,"Bytes")

def countchar():
    f1=open("poem.txt","r") #open file in read mode,transfer contents in fl
    alpha,lcase,ucase,digit,sp=0,0,0,0,0 #initialize all variables to 0
    space,vowel,consonant=0,0,0
    for line in f1:      #check each line in fl
        words=line.split() #convert contents of fl into list of words by splitting at space
        space=space+len(words)-1
        #print(words)
        for i in words: #check for each word
            for letter in i: #check for each character in the word
                if(letter.isdigit()): #check whether letter is a digit
                    digit=digit+1
                if(letter.isalpha()): #check whether letter is a alpabet
                    alpha=alpha+1
                    if(letter in ['a','e','i','o','u','A','E','I','O','U']):
                        vowel=vowel+1
                    else:
                        consonant=consonant+1
                if(letter.isupper()): #check whether letter is a uppercase
                    ucase=ucase+1
                if(letter.islower()): #check whether letter is a lowercase
                    lcase=lcase+1
                if(letter.isalnum()==False): #check whether letter is a special character
                    sp=sp+1
    print("No. of alphabets:",alpha)
    print("No. of uppercase alphabets:",ucase)
    print("No. of lowercase alphabets:",lcase)
    print("No. of vowels:",vowel)
    print("No. of consonants:",consonant)
    print("No. of digits:",digit)
    print("No. of special character:",sp)
    print("No. of spaces:",space)
```

```
#function call  
writefile()  
readfile()  
countchar()
```

Output:

Contents of the file poem.txt are:

```
1000s of Emotions in the Eyes,  
Many are there to Hear the Truth of my Life,  
Everyone has a Hope that One Day I'll Shine,  
Just not able to make a Decision,  
What to take Further...What to leave Behind!
```

Size of the file is= 199 Bytes

No. of alphabets: 148

No. of uppercase alphabets: 18

No. of lowercase alphabets: 130

No. of vowels: 62

No. of consonants: 86

No. of digits: 4

No. of special character: 9

No. of spaces: 34

Practical 5-Write a Python program to find smallest word, biggest word words of specific length, count the occurrence of specific words from a text file.

Solution:

```
def readfile():
    f1=open("train.txt","r")
    content=f1.read()
    print("The Content Of The File Is:\n")
    print(content)

def smallest():
    f1=open("train.txt","r")
    small=""
    length=50
    content=f1.read()
    lines=content.split()
    for word in lines:
        if len(word)<length:
            small=word
            length=len(small)
    print("Smallest word in the file is: ",small)
    f1.close()

def biggest():
    f1=open("train.txt","r")
    big=""
    content=f1.read()
    lines=content.split()
    for word in lines:
        if len(word)>len(big):
            big=word
    print("The Biggest Word in the file is: ",big)
    f1.close()

def findword():
    f1=open("train.txt","r")
    lstword=[]
    content=f1.read()
    wlen=int(input("Enter the length of the word to find:"))
    lines=content.split()
    for word in lines:
        if len(word)==wlen:
            lstword.append(word)
    print("The list of words with",wlen,"characters in the file are\n",lstword)
    f1.close()
```


Class XII Computer Science Practical List for Session 2020-21

```
def countword():
    f1=open("train.txt","r")    #open file
    cnt1,cnt2,cnt3,cnt4=0,0,0,0    #initialize the variable to 0
    userword=input('Enter your word to search:')

    for line in f1:
        words=line.split()
        for i in words:
            if(i=="the"):
                cnt1=cnt1+1
            if(i=="train"):
                cnt2=cnt2+1
            if i in ['Train','train']:
                cnt3=cnt3+1
            if(i==userword):
                cnt4=cnt4+1
    print("No. of occurrence of the word (the) is:",cnt1)
    print("No. of occurrence of the word (train) is:",cnt2)
    print("No. of occurrence of the word (Train or train) is:",cnt3)
    print("No. of occurrence of the word ",userword,"is=",cnt4)

readfile()    #function call to display contents of file
while True:
    print("\nMenu")
    print("1.Display smallest word in the file")
    print("2.Display biggest word in the file")
    print("3.Display words with specific length")
    print("4.Count the occurrence of specific words")
    ch=int(input("Enter Your Choice:"))
    if ch==1:
        smallest()
    elif ch==2:
        biggest()
    elif ch==3:
        findword()
    elif ch==4:
        countword()
    else:
        break
```

Output:

The Content Of The File Is:

The Vande Bharat Express Train 18 is an Indian semi high-speed rail electric multiple unit (EMU).
The train was designed and built indigenously by the Integral Coach Factory (ICF) at the Chennai in a span of 18 months.

Menu

- 1.Display smallest word in the file
- 2.Display biggest word in the file
- 3.Display words with specific length
- 4.Count the occurrence of specific words

Enter Your Choice:1

Smallest word in the file is: a

Menu

- 1.Display smallest word in the file
- 2.Display biggest word in the file
- 3.Display words with specific length
- 4.Count the occurrence of specific words

Enter Your Choice:2

The Biggest Word in the file is: indigenously

Menu

- 1.Display smallest word in the file
- 2.Display biggest word in the file
- 3.Display words with specific length
- 4.Count the occurrence of specific words

Enter Your Choice:3

Enter the length of the word to find:4

The list of words with 4 characters in the file are
['semi', 'rail', 'unit', 'span']

Menu

- 1.Display smallest word in the file
- 2.Display biggest word in the file
- 3.Display words with specific length
- 4.Count the occurrence of specific words

Enter Your Choice:4

Enter your word to search:and

No. of occurrence of the word (the) is: 2

No. of occurrence of the word (train) is: 1

No. of occurrence of the word (Train or train) is: 2

No. of occurrence of the word and is= 1

Program 6- Write a program to perform following operations on text file

1. Display 1st and last line of file
2. Read the file contents line by line with each word separated by #
3. Count and display lines start with 'O' or 'o'
4. Count and display line end with 'a'

Solution:

```
def readfirstlastline():
    f=open("Corona.txt","r")
    contents=f.readlines()
    print("\nNumber of lines in the file are:",len(contents))
    print("First line:",contents[0],end='')
    print("Last line:",contents[len(contents)-1])
    f.close()

def readlinebyline():
    f=open("Corona.txt")
    line=" "
    print("\nFile contents with each word seperated by #")
    while line:
        line=f.readline()
        words=line.split()
        for w in words:
            print(w+'#',end='')
        print()
    f.close()

def startwith():
    f=open("Corona.txt","r")
    contents=f.readlines()
    count=0
    print("\nLines start with 'O' or 'o' in file")
    for line in contents:
        if line[0]=='O' or line[0]=='o':
            count=count+1
            print(line,end='')
    print("\nNo. of lines start with ('O' or 'o') are::",count)
```

Class XII Computer Science Practical List for Session 2020-21

```
def endwith():
    f=open("Corona.txt",'r')
    contents=f.readlines()
    count=0
    print("\nLines end with 'a' in file")
    for line in contents:
        if line[-2]=='a':
            count=count+1
            print(line,end='')
    print("No. of lines end with ('a') are::",count)
    f.close()

f=open("Corona.txt",'r')
print("*****Contents of file Corona.txt*****")
text=f.read()
print(text)
print("*****Contents of file Corona.txt*****")
f.close()

while True:
    print("Menu")
    print("1.Display first and last line of file")
    print("2.Read file line by line with each word seperated by #")
    print("3.Count and display lines start with 'O' or 'o'")
    print("4.Count and display line endwith 'a'")
    ch=int(input("Enter Your Choice:"))
    if ch==1:
        readfirstlastline()
    elif ch==2:
        readlinebyline()
    elif ch==3:
        startwith()
    elif ch==4:
        endwith()
    else:
        break
```

Class XII Computer Science Practical List for Session 2020-21

Output:

```
*****Contents of file Corona.txt*****
```

```
O Corona O Corona
Jaldi se tum Go na
Social Distancing ka palan karona
sabse 1 meter ki duri rakhona
Lockdown mein ghar me ho to
online padhai karona
```

```
*****Contents of file Corona.txt*****
```

```
Menu
```

```
1.Display first and last line of file
2.Read file line by line with each word seperated by #
3.Count and display lines start with 'O' or 'o'
4.Count and display line endwith 'a'
```

```
Enter Your Choice:1
```

```
Number of lines in the file are: 6
```

```
First line: O Corona O Corona
```

```
Last line: online padhai karona
```

```
Enter Your Choice:2
```

```
File contents with each word seperated by #
```

```
O#Corona#O#Corona#
Jaldi#se#tum#Go#na#
Social#Distancing#ka#palan#karona#
sabse#1#meter#ki#duri#rakhona#
Lockdown#mein#ghar#me#ho#to#
online#padhai#karona#
```

```
Enter Your Choice:3
```

```
Lines start with 'O' or 'o' in file
```

```
O Corona O Corona
online padhai karona
```

```
No. of lines start with ('O' or 'o') are:: 2
```

```
Enter Your Choice:4
```

```
Lines end with 'a' in file
```

```
O Corona O Corona
Jaldi se tum Go na
Social Distancing ka palan karona
sabse 1 meter ki duri rakhona
online padhai karona
No. of lines end with ('a') are:: 5
```

Class XII Computer Science Practical List for Session 2020-21

Program 7- Write a program to append, count and display specific records in text file.

Solution:

```
def appendfile():
    print("The contents of the file before appending records")
    readfile()
    myfile=open("books.txt","a")
    ans='y'
    while ans=='y':
        bname=input("Enter book name:")
        author=input("Enter author name:")
        bookprice=int(input("Enter book's price:"))
        brec="Book Name: "+bname+",Author Name: "+author+",Price: "+str(bookprice)+"\n"
        myfile.write(brec)
        print("Records appended successfully")
        ans=input("Add more?:")
    myfile.close()

def readfile():
    mfile=open("books.txt","r")
    records=mfile.read()
    print("-----")
    print(records,end='')
    print("-----")
    mfile.close()

def display():
    file=open("books.txt","r")
    line=file.readline()
    authornm=input("Enter the name of Author:")
    flag=0
    count=0
    while line:
        if authornm in line:
            print(line)
            flag=1
            count=count+1
        line=file.readline()
    if (flag==0):
        print("There are no books of Author ",authornm)
    else:
        print("There are",count,"books of Author",authornm)
    file.close()

while True:
    print("\nMenu")
    print("1.Append records in text file")
    print("2.Read all records from text file")
    print("3.Count and display specific record ")
    ch=int(input("Enter your choice:"))
    if ch==1:
        appendfile()
    elif ch==2:
        readfile()
    elif ch==3:
        display()
    else:
        break
```

Class XII Computer Science Practical List for Session 2020-21

Output:

Menu

- 1.Append records in text file
- 2.Read all records from text file
- 3.Count and display specific record

Enter your choice:1

The contents of the file before appending records

Book Name:Wings of Fire,Author Name:APJ Abdul Kalam,Price:182
Book Name:I Dare!,Author Name:Kiran Bedi,Price:600
Book Name:Creating Leadership,Author Name:Kiran Bedi,Price:200
Book Name:Train to Pakistan,Author Name:Khushwant Singh,Price:250
Book Name:Delhi: A Novel,Author Name:Khushwant Singh,Price:400
Book Name: The Delhi 6 Box Set, Khushwant Singh,Price: 400

Enter book name:India of my Dreams

Enter author name:Mahatma Gandhi

Enter book's price:345

Records appended successfully

Add more?:n

Menu

- 1.Append records in text file
- 2.Read all records from text file
- 3.Count and display specific record

Enter your choice:2

Book Name:Wings of Fire,Author Name:APJ Abdul Kalam,Price:182
Book Name:I Dare!,Author Name:Kiran Bedi,Price:600
Book Name:Creating Leadership,Author Name:Kiran Bedi,Price:200
Book Name:Train to Pakistan,Author Name:Khushwant Singh,Price:250
Book Name:Delhi: A Novel,Author Name:Khushwant Singh,Price:400
Book Name: The Delhi 6 Box Set, Khushwant Singh,Price: 400
Book Name: India of my Dreams,Author Name: Mahatma Gandhi,Price: 345

Menu

- 1.Append records in text file
- 2.Read all records from text file
- 3.Count and display specific record

Enter your choice:3

Enter the name of Author:Kiran Bedi

Book Name:I Dare!,Author Name:Kiran Bedi,Price:600

Book Name:Creating Leadership,Author Name:Kiran Bedi,Price:200

There are 2 books of Author Kiran Bedi

#Practical 8- Write a program to modify contents of the file and copy contents from one file to another file based on condition.

#Solution:

```
def displayfile(filename):
    f=open(filename,"r")
    content1=f.read()
    print(content1) # Printing contents of file
    f.close()

def copyfiles():
    infile=open("source.txt","r") # Opening Source.txt in read mode
    outfile=open("target.txt","w") # Opening target.txt in write mode

    while True:
        str1=infile.readline() # Readline to read the first line and put it in str1
        if len(str1)==0: # Checking if the file is empty
            break
        if str1[0]=='@': # Checking if the line starts with @
            continue
        outfile.write(str1)
    infile.close()
    outfile.close()
    print("Contents of Source.txt")
    displayfile("source.txt")
    print("\nContents of target.txt after copying lines except start with @")
    displayfile("target.txt")

def copylines_r():
    infile=open("source.txt","r") # Opening Source.txt in read mode
    outfile=open("target.txt","w") # Opening target.txt in write mode

    while True:
        line=infile.readline() # Readline to read the first line and put it in str1
        if len(line)==0: # Cheaking if the file is empty
            break
        if 'r' in line: # Checking if the line contains r
            outfile.write(line)
    infile.close()
    outfile.close()
    print("\nContents of target.txt after copying lines which contains 'r'")
    displayfile("target.txt")

def replacestring():
    fin = open("source.txt", "r")
    data = fin.read()
    data = data.replace('line', 'word')
    fin.close()

    fin = open("source.txt", "w")
    fin.write(data)
    fin.close()
    print("\nContents of Source.txt after replacing string 'line' by 'word' using single file")
    displayfile("source.txt")
```


Class XII Computer Science Practical List for Session 2020-21

```
import os
def replacechar():
    f1=open("source.txt","r+") #open file in read mode,transfer contents in f1
    f2=open("temp.txt","w")
    rec=f1.read()
    for word in rec:
        for letter in word:
            if(letter=='@'): #check whether letter is a digit
                f2.write(letter.replace('@','#'))
            else:
                f2.write(letter)
    f1.close()
    f2.close()
    os.remove('source.txt')
    os.rename('temp.txt','source.txt')
    print("\nContents of source.txt after replacing character @ by # using temporary file")
    displayfile("source.txt")

str1="First line\n@Second line\nThird line\n@Fourth line"
f=open("Source.txt","w")
f.write(str1)
f.close()
#print("1.Copy a text file onto other except the lines starting with @")
copyfiles()
#print("2. Copy only those lines from one file which contains 'r' into other file")
copylines_r()
#print("3.Modify contents of file by replacing string in same file")
replacestring()
#print("4.Modify contents of file by replacing character using temporary file")
replacechar()
```

Output:

Contents of Source.txt

```
First line
@Second line
Third line
@Fourth line
```

Contents of target.txt after copying lines except start with @

```
First line
Third line
```

Contents of target.txt after copying lines which contains 'r'

```
First line
Third line
@Fourth line
```

Contents of Source.txt after replacing string 'line' by 'word' using single file

```
First word
@Second word
Third word
@Fourth word
```

Contents of source.txt after replacing character @ by # using temporary file

```
First word
#Second word
Third word
#Fourth word
```

Class XII Computer Science Practical List for Session 2020-21

Practical 9-Write a program to create a binary file with roll number and name using list. Search for a given roll number and display the name, if not found display appropriate message.

Solution:

```
import pickle
def createfile():
    f=open("student.dat","wb")
    studentlist=[]
    ans='y'
    while ans=='y':
        print()
        RollNo=int(input("Enter Rollno : "))
        Sname=input("Enter Name :")
        rec=[RollNo,Sname]
        studentlist.append(rec)
        ans=input("Do you want to add more(y/n):")
    pickle.dump(studentlist,f)
    f.close()

def readfile():
    print("\nContents of the file:")
    with open("student.dat","rb") as f:
        while True:
            try:
                r=pickle.load(f)
                print(r)
            except EOFError:
                break
    return r
    f.close()

def searchRec():
    fobj=open("student.dat","rb")
    rec=readfile()
    print()
    ans='y'
    while ans=='y':
        flag=False
        print()
        rno=int(input("Enter roll number to search:"))
        for s in rec:
            if s[0]==rno:
                print("Record found!!! \n Rollno:",s[0],"Name:",s[1])
                flag=True
                break
        if flag==False:
            print("Sorry! Roll number ", rno," not found in the file.")
        ans=input("Do you want to search other record(y/n):")
    fobj.close()

createfile() #function call for create binary file
searchRec() #function call for search roll number
```

Output:

```
Enter Rollno : 1
Enter Name :Anuj
Do you want to add more(y/n):y
```

```
Enter Rollno : 2
Enter Name :Sanjay
Do you want to add more(y/n):y
```

```
Enter Rollno : 6
Enter Name :Krishna
Do you want to add more(y/n):n
```

```
Contents of the file:
[[1, 'Anuj'], [2, 'Sanjay'], [6, 'Krishna']]
```

```
Enter roll number to search:2
Record found!!!
Rollno: 2 Name: Sanjay
Do you want to search other record(y/n):y
```

```
Enter roll number to search:1
Record found!!!
Rollno: 1 Name: Anuj
Do you want to search other record(y/n):y
```

```
Enter roll number to search:8
Sorry! Roll number 8 not found in the file.
Do you want to search other record(y/n):n
```

#Practical 10-Write a program to create a binary file using dictionary and perform read, append and search operations on it.

#Solution

```
def readfile():
    print("Contents of the file:")
    with open("employee.dat","rb+") as f:
        while True:
            try:
                r=pickle.load(f)
                print(r)
            except EOFError:
                break
    f.close()

def appendfile():
    myfile=open("employee.dat","ab")
    n=int(input("\nHow many records do you want to append:"))
    for i in range(n):
        eno=int(input("Enter employee no:"))
        enm=input("Enter employee name:")
        esal=int(input("Enter Salary:"))
        rec={'Empno':eno,'Empname':enm,'Salary':esal}
        pickle.dump(rec,myfile) #write 1 record into file
    myfile.close()

def createfile():
    myfile=open("employee.dat","wb")
    n=int(input("\nHow many records do you write:"))
    for i in range(n):
        eno=int(input("Enter employee no:"))
        enm=input("Enter employee name:")
        esal=int(input("Enter Salary:"))
        rec={'Empno':eno,'Empname':enm,'Salary':esal}
        pickle.dump(rec,myfile) #write 1 record into file
    myfile.close()
```

Class XII Computer Science Practical List for Session 2020-21

```
def countrec():
    f=open("employee.dat","rb")
    num = 0
    print("\nDetails of employees with salary between 20000 and 40000")
    while True:
        try:
            rec=pickle.load(f)
            if rec['Salary'] >= 20000 and rec['Salary']<=40000:
                print(rec)
                num = num + 1
        except EOFError:
            break
    f.close()
    return num

ans='y'
while ans=='y':
    print("\nMenu")
    print("1.Create a binary file")
    print("2.Read records from binary file")
    print("3.Append records to binary file")
    print("4.Count and display records from binary file")
    ch=int(input("Enter your choice:"))
    if ch==1:
        createfile()
    elif ch==2:
        readfile()
    elif ch==3:
        appendfile()
    elif ch==4:
        t=countrec()
        print("No. of employees whose salary between 20000 and 40000:",t)
    elif ch==5:
        break
    else:
        print("Invalid option")
    ans=input("Do you want to continue(y/n):")
```

Class XII Computer Science Practical List for Session 2020-21

Output:

Menu

```
1.Create a binary file
2.Read records from binary file
3.Append records to binary file
4.Count and display records from binary file
Enter your choice:1
```

How many records do you write:2

```
Enter employee no:1234
Enter employee name:Sudhir
Enter Salary:28000
Enter employee no:2343
Enter employee name:Arun
Enter Salary:15000
Do you want to continue(y/n):y
```

Enter your choice:3

How many records do you want to append:2

```
Enter employee no:1245
Enter employee name:Pramod
Enter Salary:35000
Enter employee no:1526
Enter employee name:Suraj
Enter Salary:20000
Do you want to continue(y/n):y
```

Enter your choice:2

Contents of the file:

```
{'Empno': 1234, 'Empname': 'Sudhir', 'Salary': 28000}
{'Empno': 2343, 'Empname': 'Arun', 'Salary': 15000}
{'Empno': 1245, 'Empname': 'Pramod', 'Salary': 35000}
{'Empno': 1526, 'Empname': 'Suraj', 'Salary': 20000}
Do you want to continue(y/n):n
```

Enter your choice:4

Details of employees with salary between 20000 and 40000

```
{'Empno': 1234, 'Empname': 'Sudhir', 'Salary': 28000}
{'Empno': 1245, 'Empname': 'Pramod', 'Salary': 35000}
{'Empno': 1526, 'Empname': 'Suraj', 'Salary': 20000}
No. of employees whose salary between 20000 and 40000: 3
Do you want to continue(y/n):y
```

Class XII Computer Science Practical List for Session 2020-21

Practical 11-Write a program to create a binary file with roll number, name, and marks and update the binary file by input a roll number and updates the marks.

Solution:

```
import pickle
def intro():
    print()
    print('Choose option')
    print('1.Enter details')
    print('2.Update Exsisting')
    print('3.Show existing Data')
    ch=int(input('Choose option no.: '))
    if ch==1:
        enter()
    if ch==2:
        update()
    if ch==3:
        show()

def enter():
    lst=[] #Empty list for storing Details
    n=int(input('Enter no. of students: ')) #Number of students for creating loop
    myfile=open('test1.dat','wb') #Opening File in Binary file only in writing mode
    for i in range(n): #Loop for number of students
        roll=int(input('Enter Roll No.: ')) #Roll number input
        name=input('Enter Name: ') #Name input
        marks=int(input('Enter Marks: ')) #Marks input
        details=[roll]+[str(name)]+[marks] #Adding all the elements in list form
        lst.append(details) #Adding the above list in The empty list 'lst'
        print("Details sucessfully Entered",details) #Confirmation message with details preview
        print() #printing empty line(useless)
    pickle.dump(lst,myfile) #Dumping all the elements after the completion of loop
    myfile.close() #Closing the File
    q=input('Continue ? : ')
    if q=='y'or q=='Y':
        intro()
```

Class XII Computer Science Practical List for Session 2020-21

```
def update():
    myfile=open('test1.dat','rb+')           #Opening file to Read and write binary file.
    rec=pickle.load(myfile)                 #All the content are now present in variable 'rec'
    rno1st=[]                               #Empty list for roll numbers
    updatelst=[]                            #Empty list for updated records
    flag=False
    for r in rec:
        rno1st.append(r[0])                #'r' contain elements of 'rec' one by one
        print('Data Found',rec)            #Adding first element of 'r' in Empty list 'rno1st'
        print('Available roll no. for updation',rno1st) #printing all the Data present in file
        rollno=int(input('Enter the Roll no. to update: ')) #printing the list of first elements of rno1st
        newmarks=int(input('Enter the updated marks: ')) #Roll number input for updation
        for r in rec:                       #Updated marks input
            k=r[0]                           #'r' contain elements of 'rec' one by one
            if k==rollno:                    #'k' contains integer form of first element of r
                r[2]=newmarks                #checking whether k is equal to rollno
                updatelst.append(r)          #Changing the third element of rec with the updated marks
                flag=True                    #Adding the whole r list in another list 'updatelst'
            else:
                updatelst.append(r)          #if 'k' is not equal to 'rollno'
        myfile.close()                       #Adding 'r' in list 'updatelst' without any updation
        myfile=open('test1.dat','wb+')       #closing file
        pickle.dump(updatelst,myfile)        #opening in wb mode for removing existings data
        myfile.close()                       #Dumping the new list 'updatelst' in the existings file
        if flag==True:                       #closing file
            print("Details sucessfully Updated",updatelst)
        else:
            print("Sorry no such record found in the file")
        q=input('Continue ?: ')
        if q=='y'or q=='Y':
            intro()

def show():
    myfile=open('test1.dat','rb')           #Opening file to Read in binary file.
    rec=pickle.load(myfile)                 #All the content are now present in variable 'rec'
    for r in rec:
        print(r)                            #'r' contain elements of 'rec' one by one
        q=input('Continue ?: ')            #printing records rowwise
        if q=='y'or q=='Y':
            intro()

intro()                                     #Calling funtion for starting programme
```


Class XII Computer Science Practical List for Session 2020-21

Output:

```
Choose option
1.Enter details
2.Update Exsisting
3.Show existing Data
Choose option no.: 1
Enter no. of students: 3
Enter Roll No.: 1
Enter Name: Amit
Enter Marks: 78
Details sucessfully Entered [1, 'Amit', 78]

Enter Roll No.: 2
Enter Name: Akshay
Enter Marks: 70
Details sucessfully Entered [2, 'Akshay', 70]

Enter Roll No.: 3
Enter Name: Raj
Enter Marks: 80
Details sucessfully Entered [3, 'Raj', 80]

Continue ?: y

Choose option
1.Enter details
2.Update Exsisting
3.Show existing Data
Choose option no.: 3
[1, 'Amit', 78]
[2, 'Akshay', 70]
[3, 'Raj', 80]
Continue ?: y

Choose option
1.Enter details
2.Update Exsisting
3.Show existing Data
Choose option no.: 2
Data Found [[1, 'Amit', 78], [2, 'Akshay', 70], [3, 'Raj', 80]]
Available roll no. for updation [1, 2, 3]
Enter the Roll no. to update: 2
Enter the updated marks: 75
Details sucessfully Updated [[1, 'Amit', 78], [2, 'Akshay', 75], [3, 'Raj', 80]]

Continue ?: y

Choose option
1.Enter details
2.Update Exsisting
3.Show existing Data
Choose option no.: 2
Data Found [[1, 'Amit', 78], [2, 'Akshay', 75], [3, 'Raj', 80]]
Available roll no. for updation [1, 2, 3]
Enter the Roll no. to update: 6
Enter the updated marks: 45
Sorry no such record found in the file
Continue ?: n
///
```

Practical 12-Write a program to create a binary file with roll number, name and mark and delete a particular record.

Solution:

```
import pickle
def intro():
    print()
    print('Choose option')
    print('1.Enter details')
    print('2.Delete Exsisting')
    print('3.Show existing Data')
    ch=int(input('Choose option no.: '))
    if ch==1:
        enter()
    if ch==2:
        deleterec()
    if ch==3:
        show()

def enter():
    lst=[]
    n=int(input('Enter no. of students: '))
    myfile=open('test1.dat','wb')
    for i in range(n):
        roll=int(input('Enter Roll No.: '))
        name=input('Enter Name: ')
        marks=int(input('Enter Marks: '))
        details=[roll]+[str(name)]+[marks]
        lst.append(details)
        print("Details sucessfully Entered",details)
        print()
    pickle.dump(lst,myfile)
    myfile.close()
    q=input('Continue ?: ')
    if q=='y'or q=='Y':
        intro()
```

Class XII Computer Science Practical List for Session 2020-21

```
def deleterec():
    myfile=open('test1.dat','rb+')
    rec=pickle.load(myfile)
    rno1st=[]
    updatelst=[]
    flag=False
    for r in rec:
        rno1st.append(r[0])
    print('Data Found',rec)
    print('Available roll no. for deletion',rno1st)
    rollno=int(input('Enter the Roll no. to delete: '))
    for r in rec:
        k=r[0]
        if k!=rollno:
            updatelst.append(r)
        elif k==rollno:
            flag=True
    myfile.close()
    myfile=open('test1.dat','wb+')
    pickle.dump(updatelst,myfile)
    myfile.close()
    if flag==True:
        print("Details sucessfully deleted",updatelst)
    else:
        print("Sorry no such record found in the file")
    q=input('Continue ?: ')
    if q=='y'or q=='Y':
        intro()

def show():
    myfile=open('test1.dat','rb')
    rec=pickle.load(myfile)
    for r in rec:
        print(r)
    q=input('Continue ?: ')
    if q=='y'or q=='Y':
        intro()

intro()
```

Class XII Computer Science Practical List for Session 2020-21

Output:

Choose option

- 1.Enter details
- 2.Delete Exsisting
- 3.Show existing Data

Choose option no.: 3

[1, 'Amit', 78]

[2, 'Akshay', 75]

[3, 'Raj', 80]

Continue ?: y

Choose option

- 1.Enter details
- 2.Delete Exsisting
- 3.Show existing Data

Choose option no.: 2

Data Found [[1, 'Amit', 78], [2, 'Akshay', 75], [3, 'Raj', 80]]

Available roll no. for deletion [1, 2, 3]

Enter the Roll no. to delete: 2

Details sucessfully deleted [[1, 'Amit', 78], [3, 'Raj', 80]]

Continue ?: y

Choose option

- 1.Enter details
- 2.Delete Exsisting
- 3.Show existing Data

Choose option no.: 2

Data Found [[1, 'Amit', 78], [3, 'Raj', 80]]

Available roll no. for deletion [1, 3]

Enter the Roll no. to delete: 5

Sorry no such record found in the file

Continue ?: n

#Practical 13-Write a menu driven program to create csv file and append student records such as roll number, marks, total and percentage.

Solution:

```
import csv
def enterdata():
    f=open("stud1.csv", "a", newline='')
    w=csv.writer(f)
    l=["RollNo", "Name", "Total", "Percentage"]
    w.writerow(l)
    n=int(input("Enter no. of students:"))
    for i in range(n):
        r=int(input("\nEnter roll no:  "))
        name=input("Enter name:  ")
        m1=int(input("Enter marks in Eng:  "))
        m2=int(input("Enter marks in Phy:  "))
        m3=int(input("Enter marks in Chem:  "))
        m4=int(input("Enter marks in Maths:  "))
        m5=int(input("Enter marks in CS:  "))
        tot=m1+m2+m3+m4+m5
        per=tot/5
        rec=[r, name, tot, per]
        w.writerow(rec)
    f.close()

def display():
    f=open("stud1.csv", "r")
    st=csv.reader(f)
    for i in st:
        print(i)
    f.close()

while True:
    print("MENU \n 1- Enter Data \n 2- Display Data \n 3-Exit\n")
    ch=int(input("Enter choice:"))
    if ch==1:
        enterdata()
    if ch==2:
        display()
    if ch==3:
        break
```

Output:

```
MENU
```

- 1- Enter Data
- 2- Display Data
- 3-Exit

```
Enter choice:2
```

```
['RollNo', 'Name', 'Total', 'Percentage']  
['1', 'Parth', '319', '63.8']  
['2', 'Ayush', '266', '53.2']
```

```
MENU
```

- 1- Enter Data
- 2- Display Data
- 3-Exit

```
Enter choice:1
```

```
Enter no. of students:1
```

```
Enter roll no: 3
```

```
Enter name: Anuj
```

```
Enter marks in Eng: 60
```

```
Enter marks in Phy: 67
```

```
Enter marks in Chem: 78
```

```
Enter marks in Maths: 77
```

```
Enter marks in CS: 68
```

```
MENU
```

- 1- Enter Data
- 2- Display Data
- 3-Exit

```
Enter choice:2
```

```
['RollNo', 'Name', 'Total', 'Percentage']  
['1', 'Parth', '319', '63.8']  
['2', 'Ayush', '266', '53.2']  
['RollNo', 'Name', 'Total', 'Percentage']  
['3', 'Anuj', '350', '70.0']
```

```
MENU
```

- 1- Enter Data
- 2- Display Data
- 3-Exit

```
Enter choice:3
```

```
>>>
```

Practical 14-Write a program to create csv file and perform search operation on csv file using different criteria.

Solution:

```
import csv
f=open("emp.csv","w",newline='')      #open file in write mode
a=[[101,'Harsh',25000,'HR'],
   [102,'Sanjay',24000,'sales'],
   [103,'Amit',27000,'accounts'],
   [104,'Sonali',26000,'HR'],
   [105,'Sahil',23000,'accounts'],
   [106,'Simran',25000,'HR'],
   [107,'Raj',24000,'sales'],
   [108,'Nikil',27000,'accounts'],
   [109,'Rahul',26000,'HR'],
   [110,'Shreya',25000,'accounts']]
writer=csv.writer(f,delimiter=',')
for i in a:
    writer.writerow(i)
f.close()                               #close the file

def searchname():
    f=open("emp.csv","r")               #open the file in read mode
    reader=csv.reader(f)                #read all content of f
    totalCount=0
    count=0
    print("\nDetails of employees whose name begin with 'S' are:")
    for row in reader:
        totalCount+=1
        name=row[1]                     #assigning first row of record(name column in file) to name
        if(name[0].lower()=='s'):       #comparing s with names first character
            count+=1                    #count the no of records with name with letter s
            print(row)
    print("Number of names begin with 'S' out of ",totalCount," employees are:",count)
    f.close()                           #close the file

def searchsalary():
    f=open("emp.csv","r")               #open the file in read mode
    reader=csv.reader(f)                #read all content of f
    totalCount=0
    count=0
    print("\nEmployees getting salary > 25000 are:")
    for row in reader:
        totalCount+=1
        salary=row[2]                   #assigning first row of record(name column in file) to name
        if(int(salary) > 25000):        #comparing s with names first character
            count+=1                    #count the no of records with name with letter s
            print(row)
    print("Number of Employees getting salary > 25000 out of",totalCount," employees are:",count)
    f.close()                           #close the file

searchname()
searchsalary()
```

Output:

Details of employees whose name begin with 'S' are:

```
['102', 'Sanjay', '24000', 'sales']
```

```
['104', 'Sonali', '26000', 'HR']
```

```
['105', 'Sahil', '23000', 'accounts']
```

```
['106', 'Simran', '25000', 'HR']
```

```
['110', 'Shreya', '25000', 'accounts']
```

Number of names begin with 'S' out of 10 employees are: 5

Employees getting salary > 25000 are:

```
['103', 'Amit', '27000', 'accounts']
```

```
['104', 'Sonali', '26000', 'HR']
```

```
['108', 'Nikil', '27000', 'accounts']
```

```
['109', 'Rahul', '26000', 'HR']
```

Number of Employees getting salary > 25000 out of 10 employees are: 4

#Practical 15-Write a program to copy contents containing phonebook of a csv file into another csv file.

Solution:

```
import csv
def copycsv(sourcefile,targetfile):
    f1=open(sourcefile,'r')
    f2=open(targetfile,'w',newline='')
    rec=csv.reader(f1)
    wobj=csv.writer(f2,delimiter="@")
    rowheader=['Name','City','Phoneno']
    wobj.writerow(rowheader)
    city=input("\n Enter city of customers to copy details:")
    for row in rec:
        s=str(row)
        lst=s.split('@')
        if lst[1]==city:
            wobj.writerow(row)
    f1.close()
    f2.close()

def writecsvfile():
    f=open('user.csv','w',newline='')
    wobj=csv.writer(f,delimiter="@")
    rowheader=['Name','City','Phoneno']
    wobj.writerow(rowheader)
    n=int(input("How many records you want to write:"))
    for i in range(n):
        name=input("\nEnter Name:")
        address=input("Enter City:")
        phoneno=int(input("Enter Phone no.:"))
        row=[name,address,phoneno]
        wobj.writerow(row)
    print("Data written sucessfully")
    f.close()

def readcsvfile(filename):
    f=open(filename) #open file
    rec=csv.reader(f) #csv.reader() to read the content of the file
    for row in rec:
        print(row)
    f.close()

writecsvfile()
print("\nDetails of user.csv file")
readcsvfile('user.csv')
copycsv('user.csv','customer.csv')
print("\nDetails of customer.csv file")
readcsvfile('customer.csv')
```

Output:

How many records you want to write:3

Enter Name:Ashish

Enter City:Mumbai

Enter Phone no.:7895674350

Enter Name:Pranav

Enter City:Pune

Enter Phone no.:9765934550

Enter Name:Sonali

Enter City:Mumbai

Enter Phone no.:7605407088

Data written sucessfully

Details of user.csv file

['Name@City@Phoneno']

['Ashish@Mumbai@7895674350']

['Pranav@Pune@9765934550']

['Sonali@Mumbai@7605407088']

Enter city of customers to copy details:Mumbai

Details of customer.csv file

['Name@City@Phoneno']

['Ashish@Mumbai@7895674350']

['Sonali@Mumbai@7605407088']

Practical 16-Write a program to implement stack in Python using list and perform Push and Pop operations on it.

Solution:

```
def isempty(stack):
    if len(stack)==0:
        return True
    else:
        return False

def push(stack):
    data=int(input("\nEnter a number to push:"))
    stack.append(data)
    top=len(stack)+1
    print("After pushing",data,"Stack status is:")
    display(stack)

def pop(stack):
    if isempty(stack):
        print("Stack is Empty")
    else:
        if len(stack)==0:
            top=None
        else:
            val=stack.pop()
            top=len(stack)-1
            print("\nDeleted item was:",val)
            print("After popping",val,"Stack status is:")
            display(stack)

def peek(stack):
    if isempty(stack):
        return "Underflow"
    else:
        top=len(stack)-1
        return stack[top]
```

```
def display(stack):
    if isempty(stack):
        print("Stack is empty")
    else:
        top=len(stack)-1
        print(stack[top], "<==(Top) ")
        while(top>0):
            print(stack[top-1], "<==")
            top=top-1
        print()

stack=[]
top=None
while True:
    print('\nMenu for Stack implementation')
    print('1.Push')
    print('2.Pop ')
    print('3.Display Peek')
    print('4.Display Stack')
    print('5.Quit')
    ch= int(input('Enter your choice:'))
    if ch == 1:
        push(stack)
    elif ch == 2:
        pop(stack)
    elif ch==3:
        val=peek(stack)
        if val=="Underflow":
            print("Stack is Empty")
        else:
            print("Top item :",val)
    elif ch==4:
        print("Stack status is:")
        display(stack)
    elif ch==5:
        break
```

Output:

```
Menu for Stack implementation
```

- 1.Push
- 2.Pop
- 3.Display Peek
- 4.Display Stack
- 5.Quit

```
Enter your choice:1
```

```
Enter a number to push:10
```

```
After pushing 10 Stack status is:
```

```
10 <==(Top)
```

```
Enter your choice:1
```

```
Enter a number to push:20
```

```
After pushing 20 Stack status is:
```

```
20 <==(Top)
```

```
10 <==
```

```
Menu for Stack implementation
```

- 1.Push
- 2.Pop
- 3.Display Peek
- 4.Display Stack
- 5.Quit

```
Enter your choice:3
```

```
Top item : 20
```

```
Enter your choice:1
```

```
Enter a number to push:30
```

```
After pushing 30 Stack status is:
```

```
30 <==(Top)
```

```
20 <==
```

```
10 <==
```

Class XII Computer Science Practical List for Session 2020-21

Menu for Stack implementation

1.Push

2.Pop

3.Display Peek

4.Display Stack

5.Quit

Enter your choice:2

Deleted item was: 30

After popping 30 Stack status is:

20 <==(Top)

10 <==

Menu for Stack implementation

1.Push

2.Pop

3.Display Peek

4.Display Stack

5.Quit

Enter your choice:2

Deleted item was: 20

After popping 20 Stack status is:

10 <==(Top)

Menu for Stack implementation

1.Push

2.Pop

3.Display Peek

4.Display Stack

5.Quit

Enter your choice:2

Deleted item was: 10

After popping 10 Stack status is:

Stack is empty

Class XII Computer Science Practical List for Session 2020-21

Practical 17- Implementation of various SQL commands on the table.
Solution:

T_ID	Name	Age	Department	Date_of_join	Salary	Gender
1	Jugal	34	Computer Sc	10/01/2017	12000	M
2	Sharmila	31	History	24/03/2008	20000	F
3	Sandeep	32	Mathematics	12/12/2016	30000	M
4	Sangeeta	35	History	01/07/2015	40000	F
5	Rakesh	42	Mathematics	05/09/2007	25000	M
6	Shyam	50	History	27/06/2008	30000	M
7	Shiv Om	44	Computer Sc	25/02/2017	21000	M
8	Shalakra	33	Mathematics	31/07/2018	20000	F

P_ID	Department	Place
1	History	Agra
2	Mathematics	Raipur
3	Computer Science	Delhi

```
mysql> use s1;
```

Database changed

```
mysql> create table teacher(T_ID int, Name varchar(10),Age int,  
Department varchar(15),Date_of_join date, Salary int, Gender char(1));
```

```
mysql> insert into teacher values(1,'Jugal',34,'Computer Sc','2017-01-  
10',12000,'M');
```

```
mysql> insert into teacher values(2,'Sharmila',31,'History','2018-03-  
24',20000,'F');
```

```
mysql> insert into teacher values(3,'Sandeep',32,'Mathematics','2016-12-  
12',30000,'M');
```

Class XII Computer Science Practical List for Session 2020-21

```
mysql> insert into teacher values(4,'Sangeeta',35,'History','2015-07-01',40000,'F');
```

```
mysql> insert into teacher values(5,'Rakesh',42,'Mathematics','2007-09-05',25000,'M');
```

```
mysql> insert into teacher values(6,'Shyam',50,'History','2008-06-27',30000,'M');
```

```
mysql> insert into teacher values(7,'Shiv Om',44,'Computer Sc','2017-02-25',21000,'M');
```

```
mysql> insert into teacher values(8,'Shalakra',33,'Mathematics','2018-07-31',20000,'F');
```

```
mysql> create table posting(P_ID int,Department varchar(20),Place varchar(15));
```

```
mysql> insert into posting values(1,'History','Agra');
```

```
mysql> insert into posting values(2,'Mathematics','Raipur');
```

```
mysql> insert into posting values(3,'Computer Science','Delhi');
```

SQL queries:

- 1) Write SQL query to change datatype of Place to varchar(20).

```
mysql> alter table posting
-> modify place varchar(20);
Query OK, 0 rows affected (0.20 sec)
Records: 0 Duplicates: 0 Warnings: 0
```


- 2) Write SQL query to add new column state to table posting.

```
mysql> alter table posting
-> add state varchar(20);
Query OK, 0 rows affected (0.76 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> desc posting;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| P_ID       | int(11)       | YES  |     | NULL    |      |
| Department | varchar(20)   | YES  |     | NULL    |      |
| place      | varchar(20)   | YES  |     | NULL    |      |
| state      | varchar(20)   | YES  |     | NULL    |      |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

- 3) Write SQL query to change the name of column Date_of_Join to DOJ in table teacher.

```
mysql> alter table teacher
-> change Date_of_join DOJ date;
Query OK, 0 rows affected (0.27 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> desc teacher;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| T_ID       | int(11)       | YES  |     | NULL    |      |
| Name       | varchar(10)   | YES  |     | NULL    |      |
| Age        | int(11)       | YES  |     | NULL    |      |
| Department | varchar(15)   | YES  |     | NULL    |      |
| DOJ        | date          | YES  |     | NULL    |      |
| Salary     | int(11)       | YES  |     | NULL    |      |
| Gender     | char(1)       | YES  |     | NULL    |      |
+-----+-----+-----+-----+-----+-----+
7 rows in set (0.06 sec)
```

- 4) Write SQL query to increase salary of all teachers by 1000.

```
mysql> update teacher
-> set salary=salary+1000;
Query OK, 8 rows affected (0.15 sec)
Rows matched: 8 Changed: 8 Warnings: 0
```

5) Write SQL query to delete column state from table posting.

```
mysql> alter table posting drop state;
Query OK, 0 rows affected (1.00 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> desc posting;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| P_ID          | int(11)       | YES  |     | NULL    |      |
| Department    | varchar(20)   | YES  |     | NULL    |      |
| place        | varchar(20)   | YES  |     | NULL    |      |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

6) Write SQL query to show all the information of history teachers.

```
mysql> select * from teacher where department='History';
+-----+-----+-----+-----+-----+-----+-----+
| T_ID | Name      | Age | Department | DOJ          | Salary | Gender |
+-----+-----+-----+-----+-----+-----+-----+
| 2    | Sharmila | 31  | History    | 2018-03-24  | 21000  | F      |
| 4    | Sangeeta | 35  | History    | 2015-07-01  | 41000  | F      |
| 6    | Shyam    | 50  | History    | 2008-06-27  | 31000  | M      |
+-----+-----+-----+-----+-----+-----+-----+
3 rows in set (0.07 sec)
```

7) Write SQL query to list the names of female teachers who are Mathematics department.

```
mysql> select name from teacher
-> where department='Mathematics' and Gender='F';
+-----+
| name      |
+-----+
| Shalakha |
+-----+
1 row in set (0.05 sec)
```

8) Write SQL query to show the names of all teachers with their date of joining in ascending order.

```
mysql> select name,doj from teacher
-> order by doj asc;
+-----+-----+
| name      | doj          |
+-----+-----+
| Rakesh    | 2007-09-05  |
| Shyam     | 2008-06-27  |
| Sangeeta  | 2015-07-01  |
| Sandeep   | 2016-12-12  |
| Jugal     | 2017-01-10  |
| Shiv Om   | 2017-02-25  |
| Sharmila  | 2018-03-24  |
| Shalakha  | 2018-07-31  |
+-----+-----+
8 rows in set (0.04 sec)
```

9) Write SQL query to display name, age, salary of all male teachers.

```
mysql> select name,age,salary from teacher where gender='M';
```

name	age	salary
Jugal	34	13000
Sandeep	32	31000
Rakesh	42	26000
Shyam	50	31000
Shiv Om	44	22000

```
5 rows in set (0.00 sec)
```

10) Write a query to display name, bonus for each teacher where bonus is 10% of salary.

```
mysql> select name,salary*0.10 as 'Bonus' from teacher;
```

name	Bonus
Jugal	1300.00
Sharmila	2100.00
Sandeep	3100.00
Sangeeta	4100.00
Rakesh	2600.00
Shyam	3100.00
Shiv Om	2200.00
Shalakra	2100.00

```
8 rows in set (0.05 sec)
```

11) Write SQL query to count number of teacher in each department.

```
mysql> select department,count(*)
-> from teacher
-> group by department;
```

department	count(*)
Computer Sc	2
History	3
Mathematics	3

```
3 rows in set (0.06 sec)
```

12) Write SQL query to display maximum and minimum date of joining.

```
mysql> select max(doj),min(doj) from teacher;
+-----+-----+
| max(doj) | min(doj) |
+-----+-----+
| 2018-07-31 | 2007-09-05 |
+-----+-----+
1 row in set (0.04 sec)
```

13) Write SQL query to display name, department of all teachers whose place of posting is “Raipur”.

```
mysql> select name,T. Department,place
-> from Teacher T, Posting P
-> where T.Department=P.Department and
-> place='Raipur';
+-----+-----+-----+
| name | Department | place |
+-----+-----+-----+
| Sandeep | Mathematics | Raipur |
| Rakesh | Mathematics | Raipur |
| Shalakra | Mathematics | Raipur |
+-----+-----+-----+
3 rows in set (0.00 sec)
```

14) Write SQL query to display average salary of Computer Science department teachers.

```
mysql> select department,avg(salary)
-> from teacher
-> where department="Computer Sc"
-> group by department;
+-----+-----+
| department | avg(salary) |
+-----+-----+
| Computer Sc | 17500.0000 |
+-----+-----+
1 row in set (0.03 sec)
```

15) Write SQL query to display name, department and salary of all teachers working in Agra.

```
mysql> select name,T.department,salary
-> from teacher T,posting P
-> where T.department=P.department and
-> place='Agra';
+-----+-----+-----+
| name | department | salary |
+-----+-----+-----+
| Sharmila | History | 21000 |
| Sangeeta | History | 41000 |
| Shyam | History | 31000 |
+-----+-----+-----+
3 rows in set (0.00 sec)
```

Class XII Computer Science Practical List for Session 2020-21

16) Display details of all teachers in ascending order of their salary and descending order of their age.

```
mysql> select * from teacher order by salary asc, age desc;
```

T_ID	Name	Age	Department	DOJ	Salary	Gender
1	Jugal	34	Computer Sc	2017-01-10	13000	M
8	Shalakra	33	Mathematics	2018-07-31	21000	F
2	Sharmila	31	History	2018-03-24	21000	F
7	Shiv Om	44	Computer Sc	2017-02-25	22000	M
5	Rakesh	42	Mathematics	2007-09-05	26000	M
6	Shyam	50	History	2008-06-27	31000	M
3	Sandeep	32	Mathematics	2016-12-12	31000	M
4	Sangeeta	35	History	2015-07-01	41000	F

```
8 rows in set (0.00 sec)
```

17) Display details of employees whose name start with 'S' and ends with 'a'

```
mysql> select * from teacher where name like 'S%a';
```

T_ID	Name	Age	Department	DOJ	Salary	Gender
2	Sharmila	31	History	2018-03-24	21000	F
4	Sangeeta	35	History	2015-07-01	41000	F
8	Shalakra	33	Mathematics	2018-07-31	21000	F

```
3 rows in set (0.01 sec)
```

18) Write SQL query to display total salary drawn by mathematics teachers.

```
mysql> select department, sum(salary)
-> from teacher
-> group by department
-> having department='Mathematics';
```

department	sum(salary)
Mathematics	78000

```
1 row in set (0.00 sec)
```

Practical 18-Write a program to connect with database, create table student, store records of students and display records.

Solution:

```
import mysql.connector as sqltor
mycon=sqltor.connect(host="localhost",user="root",passwd="tiger")
mycur=mycon.cursor()

def displayall():
    mycur.execute("select * from student")
    data=mycur.fetchall()
    count=mycur.rowcount
    print("Total number of rows retrived in resultset:",count)
    for row in data:
        print(row)

def displaymany(n):
    mycur.execute("select * from student")
    data=mycur.fetchmany(n)
    count=mycur.rowcount
    print("Total number of rows retrived in resultset:",count)
    for row in data:
        print(row)

def menu():
    print("\n*****MENU*****")
    print("1. Add record using static data")
    print("2. Add records using custom data")
    print("3. Display records")

def insertstatic():
    inquiry="INSERT INTO student(rollno,Name,marks,section,project)\
        VALUES('{}','{}','{}','{}','{}').format(101,'Ruhani',76.80,'A','Pending')
    mycur.execute(inquiry)
    mycon.commit()
    print("\n*****INSERT ROW INTO TABLE*****")
    print("Record added sucessfully")
    displayall()
```

Class XII Computer Science Practical List for Session 2020-21

```
def insertcustom():
    print("\n*****INSERT ROWS INTO TABLE*****")
    ch='y'
    while(ch=='y'):
        rn=int(input("Enter rollno::"))
        nm=input("Enter the name of the student::")
        marks=float(input("Enter marks::"))
        section=input("Enter the section::")
        project=input("Enter the project status::")
        inrow=(rn,nm,marks,section,project)
        inquiry="INSERT INTO student(rollno,Name,marks,section,project)\
                VALUES(%s,%s,%s,%s,%s)"
        mycur.execute(inquiry,inrow)
        print("Record added successfully")
        ch=input("Do you want to insert more records::")
        displayall()

mycur.execute("drop database school") # drop database if already exist
mycur.execute("create database school")
mycur.execute("use school")

query="create table student (rollno int,Name varchar(10),marks float(4,2),\
        section char(1), project varchar(15))"
mycur.execute(query)
print("Table created successfully")
mycur.execute("desc student")
print("Structure of the created table")
for x in mycur:
    print(x)

op='y'
while(op=='y'):
    menu()
    choice=int(input("Enter your choice::"))
    if(choice==1):
        insertstatic()
    if(choice==2):
        insertcustom()
    if(choice==3):
        totrow=int(input("How many records you want read from table student::"))
        displaymany(totrow)
    op=input("Do you want to continue::")

mycur.execute("drop table student") # drop table if already exist
mycon.close()
```

Output:

```
Table created successfully
Structure of the created table
('rollno', 'int(11)', 'YES', '', None, '')
('Name', 'varchar(10)', 'YES', '', None, '')
('marks', 'float(4,2)', 'YES', '', None, '')
('section', 'char(1)', 'YES', '', None, '')
('project', 'varchar(15)', 'YES', '', None, '')

*****MENU*****
1. Add record using static data
2. Add records using custom data
3. Display records
Enter your choice::1

*****INSERT ROW INTO TABLE*****
Record added successfully
Total number of rows retrived in resultset: 1
(101, 'Ruhani', 76.8, 'A', 'Pending')
Do you want to continue::y

*****MENU*****
1. Add record using static data
2. Add records using custom data
3. Display records
Enter your choice::2

*****INSERT ROWS INTO TABLE*****
Enter rollno::102
Enter the name of the student::Rohan
Enter marks::78
Enter the section::B
Enter the project status::Pending
Record added successfully

Do you want to insert more records::y
Total number of rows retrived in resultset: 2
(101, 'Ruhani', 76.8, 'A', 'Pending')
(102, 'Rohan', 78.0, 'B', 'Pending')
Enter rollno::103
Enter the name of the student::Mohit
Enter marks::88
Enter the section::B
Enter the project status::Completed
Record added successfully

Do you want to insert more records::n
Total number of rows retrived in resultset: 3
(101, 'Ruhani', 76.8, 'A', 'Pending')
(102, 'Rohan', 78.0, 'B', 'Pending')
(103, 'Mohit', 88.0, 'B', 'Completed')
Do you want to continue::y
```


Class XII Computer Science Practical List for Session 2020-21

Program 19- Write a program to connect with database and search records in the SQL table using (+) operator,%s and format{ }and display the matched records.

Solution:

Structure and records of table 'teacher' saved in database 's1'

```
mysql> desc teacher;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| T_ID       | int(11)       | YES  |     | NULL    |      |
| Name       | varchar(10)   | YES  |     | NULL    |      |
| Age        | int(11)       | YES  |     | NULL    |      |
| Department | varchar(15)   | YES  |     | NULL    |      |
| DOJ        | date          | YES  |     | NULL    |      |
| Salary     | int(11)       | YES  |     | NULL    |      |
| Gender     | char(1)       | YES  |     | NULL    |      |
+-----+-----+-----+-----+-----+-----+
7 rows in set (0.86 sec)

mysql> select * from teacher;
+-----+-----+-----+-----+-----+-----+-----+
| T_ID | Name      | Age | Department | DOJ       | Salary | Gender |
+-----+-----+-----+-----+-----+-----+-----+
| 1    | Jugal    | 34  | Computer Sc | 2017-01-10 | 13000  | M      |
| 2    | Sharmila | 31  | History     | 2018-03-24 | 21000  | F      |
| 3    | Sandeep  | 32  | Mathematics | 2016-12-12 | 31000  | M      |
| 4    | Sangeeta | 35  | History     | 2015-07-01 | 41000  | F      |
| 5    | Rakesh   | 42  | Mathematics | 2007-09-05 | 26000  | M      |
| 6    | Shyam    | 50  | History     | 2008-06-27 | 31000  | M      |
| 7    | Shiv Om  | 44  | Computer Sc | 2017-02-25 | 22000  | M      |
| 8    | Shalakra | 33  | Mathematics | 2018-07-31 | 21000  | F      |
+-----+-----+-----+-----+-----+-----+-----+
8 rows in set (0.02 sec)
```

Class XII Computer Science Practical List for Session 2020-21

```
import mysql.connector as ms
mycon=ms.connect(host='localhost',user='root',passwd='tiger',database='s1')
if mycon.is_connected()==False:
    print("Error connecting to MySQL Database")
mycursor=mycon.cursor()

def displayall():
    mycursor.execute("select * from teacher")
    data=mycursor.fetchall()
    count=mycursor.rowcount
    print("Total number of rows retrived in resultset:",count)
    print("Records of teacher table are:")
    for row in data:
        print(row)

def searchtid():
    tid=int(input("\nEnter teacher id of the teacher:"))
    query="select * from teacher where T_ID="+str(tid,)
    mycursor.execute(query)
    data=mycursor.fetchone()
    if data!=None:
        print(data)
    else:
        print("\nNo teacher with T_ID number:",tid)

def searchdeptage():
    tdept=input("\nEnter the department to search:")
    tage=int(input("\nEnter age criteria to search:"))
    query="select * from teacher where department=%s and age=%s"%(tdept,tage)
    mycursor.execute(query)
    data=mycursor.fetchall()
    count=mycursor.rowcount
    print("Total records of department",tdept,"and age=",tage,"are=",count)
    print("Retrived records:")
    if data!=None:
        for row in data:
            print(row)
    else:
        print("\nNo matching records found")
```

Class XII Computer Science Practical List for Session 2020-21

```
def searchdeptsal():
    dept=input("\nEnter the department to search:")
    tsalary=int(input("\nEnter salary of the teacher to search:"))
    query="select * from teacher where department='{department}'\
        and salary>={salary}".format(salary=tsalary,department=dept)
    mycursor.execute(query)
    data=mycursor.fetchall()
    count=mycursor.rowcount
    print("Total records of",dept,"with salary greater than",tsalary,"are=",count)
    print("Retrived records:")
    if data!=None:
        for row in data:
            print(row)
    else:
        print("\nNo matching records found")

def menu():
    print("\n*****MENU*****")
    print("1. Display all records")
    print("2. Search record using + operator")
    print("3. Serach record using %s")
    print("4. Search record using format{}")

op='y'
while (op=='y') :
    menu()
    choice=int(input("Enter your choice:"))
    if (choice==1):
        displayall()
    if (choice==2):
        searchtid()
    if (choice==3):
        searchdeptage()
    if (choice==4):
        searchdeptsal()
    op=input("Do you want to continue:")
```

Class XII Computer Science Practical List for Session 2020-21

Output:

```
*****MENU*****
1. Display all records
2. Search record using + operator
3. Serach record using %s
4. Search record using format{}
Enter your choice::1
Total number of rows retrived in resultset: 8
Records of teacher table are:
(1, 'Jugal', 34, 'Computer Sc', datetime.date(2017, 1, 10), 13000, 'M')
(2, 'Sharmila', 31, 'History', datetime.date(2018, 3, 24), 21000, 'F')
(3, 'Sandeep', 32, 'Mathematics', datetime.date(2016, 12, 12), 31000, 'M')
(4, 'Sangeeta', 35, 'History', datetime.date(2015, 7, 1), 41000, 'F')
(5, 'Rakesh', 42, 'Mathematics', datetime.date(2007, 9, 5), 26000, 'M')
(6, 'Shyam', 50, 'History', datetime.date(2008, 6, 27), 31000, 'M')
(7, 'Shiv Om', 44, 'Computer Sc', datetime.date(2017, 2, 25), 22000, 'M')
(8, 'Shalakra', 33, 'Mathematics', datetime.date(2018, 7, 31), 21000, 'F')
Do you want to continue::y
```

```
*****MENU*****
1. Display all records
2. Search record using + operator
3. Serach record using %s
4. Search record using format{}
Enter your choice::2

Enter teacher id of the teacher::2
(2, 'Sharmila', 31, 'History', datetime.date(2018, 3, 24), 21000, 'F')
Do you want to continue::y
```

```
*****MENU*****
1. Display all records
2. Search record using + operator
3. Serach record using %s
4. Search record using format{}
Enter your choice::2

Enter teacher id of the teacher::15

No teacher with T_ID number: 15
Do you want to continue::y
```

Class XII Computer Science Practical List for Session 2020-21

```
*****MENU*****
```

1. Display all records
2. Search record using + operator
3. Search record using %s
4. Search record using format{}

```
Enter your choice::3
```

```
Enter the department to search::'History'
```

```
Enter age criteria to search::35
```

```
Total records of department 'History' and age= 35 are= 1
```

```
Retrieved records:
```

```
(4, 'Sangeeta', 35, 'History', datetime.date(2015, 7, 1), 41000, 'F')
```

```
Do you want to continue::y
```

```
*****MENU*****
```

1. Display all records
2. Search record using + operator
3. Search record using %s
4. Search record using format{}

```
Enter your choice::4
```

```
Enter the department to search::Mathematics
```

```
Enter salary of the teacher to search::20000
```

```
Total records of Mathematics with salary greater than 20000 are= 3
```

```
Retrieved records:
```

```
(3, 'Sandeep', 32, 'Mathematics', datetime.date(2016, 12, 12), 31000, 'M')
```

```
(5, 'Rakesh', 42, 'Mathematics', datetime.date(2007, 9, 5), 26000, 'M')
```

```
(8, 'Shalakha', 33, 'Mathematics', datetime.date(2018, 7, 31), 21000, 'F')
```

```
Do you want to continue::y
```

Program 20-Write a program to connect with database and update and delete the record from database. (Database:s1,table:teacher)

Solution:

```
import mysql.connector as sqltor
mycon=sqltor.connect(host="localhost",user="root",passwd="tiger",database="s1")
mycur=mycon.cursor()

def displayall():
    mycur.execute("select * from teacher")
    data=mycur.fetchall()
    count=mycur.rowcount
    print("Total number of rows retrived in resultset:",count)
    for row in data:
        print(row)

def displayone(tid):
    mycur.execute("select * from teacher where T_ID=%s" %(tid,))
    data=mycur.fetchone()
    print(data)

def menu():
    print("\n*****MENU*****")
    print("1.Display records")
    print("2.Delete record")
    print("3.Update record")

def deleterec():
    print("\n*****DELETE ROW FROM TABLE*****")
    print("\nRecords of teacher table before deletion:")
    displayall()
    tid=int(input("\nEnter teacher id whose record you want to delete:"))
    data1=(tid,)
    delquery="DELETE FROM teacher WHERE T_ID=%s"
    mycur.execute(delquery,data1)
    count=mycur.rowcount
    if count==0:
        print("Sorry! teacher id not found in table")
    else:
        mycon.commit()
        print("Rows affected:",mycur.rowcount)
        print("\nRecords of teacher table after deletion:")
        displayall()
```

Class XII Computer Science Practical List for Session 2020-21

```
def updaterec():
    print("\n*****UPDATE ROW INTO TABLE*****")
    print("\nRecords of teacher table before updation:")
    tid=int(input("\nEnter teacher id whose record you want to UPDATE:"))
    query="Select * from teacher where T_ID=%s"%(tid)
    mycur.execute(query)
    result=mycur.fetchall()
    if mycur.rowcount==0:
        print("Sorry!Teacher with ", tid,"not found")
    else:
        for row in result:
            print(row)
        print("You can change only Department and Salary")
        dept=input("Enter new department,leave blank if not want to change:")
        if dept==' ':
            dept=str(row[3])
            print(dept)
        try:
            sal=int(input("Enter new salary,leave leave blank if not want to change:"))
        except:
            sal=row[5]
        updatequery="UPDATE teacher SET department='%s',salary=%s WHERE T_ID=%s"%(dept,sal,tid)
        mycur.execute(updatequery)
        mycon.commit()
        print("Rows affected:",mycur.rowcount)
        print("\nContents of updated record:")
        displayone(tid)

op='y'
while (op=='y') :
    menu ()
    choice=int(input("\nEnter your choice:"))
    if (choice==1):
        displayall ()
    if (choice==2):
        deleterec ()
    if (choice==3):
        updaterec ()
    op=input("Do you want to continue:")

mycon.close ()
```

Class XII Computer Science Practical List for Session 2020-21

Output:

```
*****MENU*****
```

- 1.Display records
- 2.Delete record
- 3.Update record

```
Enter your choice::1
```

```
Total number of rows retrived in resultset: 7
```

```
(1, 'Jugal', 34, 'Computer Sc', datetime.date(2017, 1, 10), 13000, 'M')
(2, 'Sharmila', 31, 'History', datetime.date(2018, 3, 24), 21000, 'F')
(3, 'Sandeep', 32, 'Mathematics', datetime.date(2016, 12, 12), 31000, 'M')
(4, 'Sangeeta', 35, 'History', datetime.date(2015, 7, 1), 41000, 'F')
(5, 'Rakesh', 42, 'Mathematics', datetime.date(2007, 9, 5), 26000, 'M')
(6, 'Shyam', 50, 'History', datetime.date(2008, 6, 27), 31000, 'M')
(7, 'Shiv Om', 44, 'Computer Sc', datetime.date(2017, 2, 25), 22000, 'M')
```

```
Do you want to continue::y
```

```
*****MENU*****
```

- 1.Display records
- 2.Delete record
- 3.Update record

```
Enter your choice::2
```

```
*****DELETE ROW FROM TABLE*****
```

```
Records of teacher table before deletion:
```

```
Enter teacher id whose record you want to delete::7
```

```
Rows affected:: 1
```

```
Records of teacher table after deletion:
```

```
Total number of rows retrived in resultset: 6
```

```
(1, 'Jugal', 34, 'Computer Sc', datetime.date(2017, 1, 10), 13000, 'M')
(2, 'Sharmila', 31, 'History', datetime.date(2018, 3, 24), 21000, 'F')
(3, 'Sandeep', 32, 'Mathematics', datetime.date(2016, 12, 12), 31000, 'M')
(4, 'Sangeeta', 35, 'History', datetime.date(2015, 7, 1), 41000, 'F')
(5, 'Rakesh', 42, 'Mathematics', datetime.date(2007, 9, 5), 26000, 'M')
(6, 'Shyam', 50, 'History', datetime.date(2008, 6, 27), 31000, 'M')
```

```
Do you want to continue::y
```


Class XII Computer Science Practical List for Session 2020-21

*****MENU*****

- 1.Display records
- 2.Delete record
- 3.Update record

Enter your choice::2

*****DELETE ROW FROM TABLE*****

Records of teacher table before deletion:

Enter teacher id whose record you want to delete::10

Sorry! teacher id not found in table

Do you want to continue::y

*****MENU*****

- 1.Display records
- 2.Delete record
- 3.Update record

Enter your choice::3

*****UPDATE ROW INTO TABLE*****

Records of teacher table before updation:

Enter teacher id whose record you want to UPDATE::6

(6, 'Shyam', 50, 'History', datetime.date(2008, 6, 27), 31000, 'M')

You can change only Department and Salary

Enter new department,leave blank if not want to change::English

Enter new salary,leave blank if not want to change::35000

Rows affected:: 1

Contents of updated record:

(6, 'Shyam', 50, 'English', datetime.date(2008, 6, 27), 35000, 'M')

Do you want to continue::y

*****MENU*****

- 1.Display records
- 2.Delete record
- 3.Update record

Enter your choice::3

*****UPDATE ROW INTO TABLE*****

Records of teacher table before updation:

Enter teacher id whose record you want to UPDATE::5

(5, 'Rakesh', 42, 'Mathematics', datetime.date(2007, 9, 5), 26000, 'M')

You can change only Department and Salary

Enter new department,leave blank if not want to change::

Mathematics

Enter new salary,leave blank if not want to change::30000

Rows affected:: 1

Contents of updated record:

(5, 'Rakesh', 42, 'Mathematics', datetime.date(2007, 9, 5), 30000, 'M')

Do you want to continue::n