COMPUTER SCIENCE WITH PYTHON
PRACTICAL FILE

NAME:

CLASS: XII

SECTION:

PRACTICAL FILE- COMPUTER SCIENCE (083)

LIST OF PRACTICALS (2025-26) CLASS-XII

Programming Language: Python

S. No.	NAME OF PRACTICAL
1	Write a Program using functions to enter the string and to check if it's palindrome
2	Write a function that takes a character and returns True if it is a vowel , False otherwise.
3	Write a program that inputs two tuples and creates a third that contains all elements of the first followed by all the elements of the second.
4	Write a program using function to create a dictionary, which accepts the marks in five different subjects with subject as key and marks as value. Display the dictionary and the subjects getting marks 90 and above.
5	Write a program using function that takes one argument (a positive integer) and reports if the argument is prime or not.
6	Write a program to generate random numbers between 1 to 6 and check whether a user won a lottery or not.
7	Write a program using function to generate Fibonacci series up to n. Pass the value of 'n' to a function and display all the numbers of the series in the function as a list elements.
8	Write a program to copy the records of that students having marks greater than 95 from binary file "marks dat" into the "toppers.dat" file.
9	Write a program that reads a text file and then creates a new file where each character's case is inverted.
10	Read a text file and display the number of vowels/ consonants/ uppercase/ lowercase characters in the file.
11	Write a function in PYTHON to count and display the number of words starting with alphabet 'A' or 'a' present in a text file "LINES.TXT". Example: If the file "LINES.TXT" contains the following lines, "A boy is playing there. There is a playground. An aeroplane is in the sky. Are you getting it?" The function should display the output as 5 .
12	Consider a binary file "Emp.dat" containing details such as empno: ename:salary(separator':').Write a python function to display details of those employees who are earning between 20000 and 40000.
13	Write a program that will write a string in binary file "school.dat" and display the words of the string in reverse order.
14	Write a program to read data from a text file DATA.TXT, and display word, which have maximum/minimum characters.
15	Write the function definition for WORD4CHAR() in PYTHON to read the content of a text file FUN.TXT, and display all those words, which have four characters in it. Example: If the content of the file Fun.TXT is as follows:

	"When I was a small child, I used to play in the garden with my grand mom. Those days were amazingly funful and I remember all the moments of that time" The function WORD4CHAR() should display the following: "When used play with days were that time"
16	Write a definition for function BUMPER() in PYTHON to read each object of a binary file GIFTS.DAT, find and display details of those gifts, which have remarks as "ON DISCOUNT". Assume that the file GIFTS.DAT is created with the help of lists of following type: (ID, Gift, Remarks, Price**)**
17	Create a binary file "batsman.dat" with the fields:b_No,Name,runs,rank and write a function count() in python which accepts the rank from user and counts the number of the file having that rank.
18	Create a binary file with roll number, name and marks. Input a roll number and update the marks.
19	Following is the structure of each record in a data file named "PRODUCT.DAT". {"prod_code":value, "prod_desc":value, "stock":value} The values for prod_code and prod_desc are strings, and the value for stock is an integer. Write a function in PYTHON to update the file with a new value of stock. The stock and the product_code, whose stock is to be updated, are to be input during the execution of the function.
20	Write a Python program to get 10 items' details (itemno,name,price,category) from the user and create a CSV file(Items.csv).
21	Write a menu-based program to perform the operation on stack in python.
22	Write a program to connect Python with MySQL using database connectivity and perform the following operations on data in database: Fetch, Update and delete the data
23	**Consider the following tables STORE and SUPPLIERS and answer (a) and (b) parts of this question:** **Table: - Store** **ItemNo. Iteam Scode Qty Rate LastBuy** 2005 Sharpener Classic 23 60 8 31-Jun-09 2003 Ball Pen 0.25 22 50 25 01-Feb-10 2002 Gel Pen Premium 21 150 12 24-Feb-10 2006 Gel Pen Classic 21 250 20 11-Mar-09 2001 Eraser Small 22 220 6 19-Jan-09 2004 Eraser Big 22 110 8 02-Dec-09 2009 Ball Pen 0.5 21 180 18 03-Nov-09 **Table:- Supplier** **Scode Sname** 21 Premium Stationers 23 Soft Plastics 22 Tetra Supply **A.Write SQL commands for the following statements:** (i) To display details of all the items in the Store table in ascending order of LastBuy. (ii) To display ItemNo and Item name of those items from Store table, whose Rate is more than 15 Rupees. (iii) To display the details of those items whose Supplier code (Scode) is 22 or Quantity in Store (Qty) is more than 110 from the table Store. **(iv) To display Minimum Rate of items for each Supplier individually as per Scode from the table Store** **B.Give the output of the following SQL queries:** (i) SELECT COUNT(DISTINCT Scode) FROM Store; (ii) SELECT Rate*Qty FROM Store WHERE ItemNo=2004;

(iii) SELECT Item,Sname FROM Store S, Suppliers P WHERE S.Scode=P.Scode AND ItemNo=2006;

(iv)SELECT MAX(LastBuy) FROM Store;

24 **Consider the following table Item and Customer.**

Table:- ITEM

i_ID	ItemName	Manufacturer	Price
PC01	Personal Computer	ABC	35000
LC05	Laptop	ABC	55000
Pc03	Personal Computer	XYZ	32000
Pc06	Personal Computer	COMP	37000
Lc03	Laptop	PQR	57000

Table : CUSTOMER

C_ID	CustomerName	City	I_ID
01	N Roy	Delhi	LC03
06	H Singh	Mumbai	PC03
12	R Pandey	Delhi	PC06
15	C Sharma	Delhi	LC03
16	K Agarwal	Banglore	PC01

A. **Write SQL commands for the following statements:**

I. To display the details of those Customer whose City is Delhi.

II. To display the details of Item whose Price is in the range of 3500 to 55000 (Both values included).

III. To display the CustomerName, City from table Customer, and ItemName and Price from table Item with their corresponding matching I_ID.

IV. To increase the Price of all Items by 1000 in the table Item.

B. **Give the output of the following SQL queries:**

V. SELECT DISTINCT City FROM Customer;

VI. SELECT ItemName, Max(Price), Count(*) FROM Item GROUP BY ItemName;

VII. SELECT CustomerName, Manufacturer From Item, Customer WHERE Item.I_Id=Customer.I_Id;

VIII. SELECT ItemName, Price * 100 FROM Item WHERE Manufacturer = 'ABC';

25 **Consider the following tables.**

TABLE : SENDER

SenderID	SenderName	SenderName	SenderCity
ND01	R Jain	2,ABC Appts	New Delhi
MU02	H Sinha	12, Newtown	Mumbai
MU15	S Jha	27/A, Park Street	Mumbai
ND50	T Prasad	122-K, SDA	New Delhi

TABLE : RECIPIENT

RecID	SenderID	RecName	RecAddress	ReCity
KO05	ND01	R Bajpayee	5, Central Avenue	Kolkata
ND08	MU02	S Mohan	116, A vihar	New Delhi
MU19	ND01	H singh	2a, Andheri east	Mumbai

MU32	MU15	P K Swamy	B5, c S Terminus	Mumbai
ND48	ND50	S Tirupathi	13, B1 d, Mayur vihar	New Delhi

A. **Write SQL commands for the following statements:**

I. To display the names of all Senders from Mumbai.
II. To display the RecID, SenderName, SenderAddress, RecName, RecAddess for every Recipient
III. To display Recipient detail in ascending order of RecName
IV. To display number of Recipients from each city

B. **Give the output of the following SQL queries:**

I. SELECT DISTINCT Sendercity FROM Sender;
II. SELECT A.SenderName, B.RecName FROM Sender A, Recipient B WHERE A.SenderID=B.SenderID AND B.RecCity='Mumbai';
III. SELECT RecName, RecAddress FROM Recipient WHERE recCity NOT IN('Mumbai', 'Kolkata');
IV. SELECT RecID, RecName FROM Recipient WHERE SenderID='MU02' OR SenderID='ND50';

26 **Consider the following tables FURNITURE and ARRIVALS: write SQL commands for the questions (a) to (f) and give outputs for SQL queries (i) to (iv) :**

TABLE : FURNITURE

NO	ITEMNAME	TYPE	DATEOFSTOCK	PRICE	DISCOUNT
1	White lotus	Double Bed	23/02/02	30000	25
2	Pink feather	Baby cot	20/01/02	7000	20
3	Dolphin	Baby cot	19/02/02	9500	20
4	Decent	Office Table	01/01/02	25000	30
5	Comfort zone	Double Bed	12/01/02	25000	25
6	Donald	Baby cot	24/02/02	6500	15
7	Royal Finish	Office Table	20/02/02	18000	30
8	Royal tiger	Sofa	22/02/02	31000	30
9	Econo sitting	Sofa	13/12/01	9500	25
10	Eating Paradise	Dining Table	19/02/02	11500	25

TABLE : ARRIVALS

NO	ITEMNAME	TYPE	DATEOFSTOCK	PRICE	DISCOUNT
11	Wood Comfort	Double Bed	23/03/03	25000	25
12	Old Fox	Sofa	20/02/03	17000	20
13	Micky	Baby cot	21/02/03	7500	15

A. **Write SQL commands for the following statements:**

a) To show all information about the Baby cots from the FURNITURE table.
b) To list the ITEMNAME which are priced at more than 15000 from the FURNITURE table.
c) To list ITEMNAME and TYPE of those items, in which DATEOFSTOCK is before 22/01/02 from the FURNITURE table in descending order of ITEMNAME.
d) To display ITEMNAME and DATEOFSTOCK of those items, in which the DISCOUNT percentage is more than 25 form FURNITURE table.

e) To count the number of items, whose TYPE is "Sofa" from FURNITURE table.
f) To insert a new row in the ARRIVALS table with the following data:
   a. 14, 'Velvet touch', 'Double bed', {25/03/03}, 25000,30

B. **Give the output of following SQL statement:**

NOTE : Outputs of the below mentioned queries should be based on original data given in both the tables, i.e., without considering the insertion done in (f) part of this question :

I. Select COUNT (distinct TYPE) from FURNITURE;
II. Select MAX(DISCOUT) form FURNITURE, ARRIVALS;
III. Select AVG(DISCOUT) form FURNITURE where TYPE = 'Baby cot';
IV. Select SUM(PRICE) from FURNITURE where DATEOFSTOCK<{12/02/02};

27 **Study the following tables DOCTOR and SALARY and write SQL commands for the questions (i) to (iv) and give outputs for SQL queries (v) to (vi) :**

TABLE : DOCTOR

ID	NAME	DEPT	SEX	EXPERIENCE
101	John	ENT	M	12
104	Smith	ORTHOPEDIC	M	5
107	George	CARDIOLOGY	M	10
114	Lara	SKIN	F	3
109	K George	MEDICINE	F	9
105	Johnson	ORTHOPEDIC	M	10
117	Lucy	ENT	F	3
111	Bill	MEDICINE	F	12
130	Morphy	ORTHOPEDIC	M	15

TABLE : SALARY

ID	BASIC	ALLOWANCE	CONSULTATION
101	12000	1000	300
104	23000	2300	500
107	32000	4000	500
114	12000	5200	100
109	42000	1700	200
105	18900	1690	300
130	21700	2600	300

A. **Write SQL commands for the following statements:**

(i) Display NAME of all doctors who are in "MEDICINE" having more than 10 year experience from the table DOCTOR.
(ii) Display the average salary of all doctors working in "ENT" department using the tables DOCTOR and SALARY. Salary=BASIC + ALLOWANCE.
(iii) Display the minimum ALLOWANCE of female doctors.
(iv) Display the highest consultation fee among all male doctor

B. **Give the output of following SQL statement:**

(v) SELECT count(*) from DOCTOR where SEX="F".
(vi) SELECT NAME, DEPT, BASIC from DOCTOR Salary WHERE DEPT="ENT" AND DOCTORID=SALARY.ID

Q1 #Write a Program using functions to enter the string and to check if it's palindrome

Code

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

s = input("Enter a string: ")
if check_palindrome(s):
        print("It's a palindrome.")
else:
        print("It's not a palindrome.")
```

OUTPUT

```
Enter a string: madam
It's a palindrome.

====== RESTART: C:/Users/user/AppData/Local/Programs/Python/Python311/a.py =====
Enter a string: computer
It's not a palindrome.
```

Q2 #Write a function that takes a character and returns True if it is a vowel , False otherwise.

Code

```
def checkvowel(x):
    if (x == 'a' or x == 'e' or x == 'i' or x == 'o' or x == 'u'):
        print("Vowel")
    else:
        print("not Vowel")

ch=input('enter a character (in lower case)to check if it is vowel or not ')
checkvowel(ch)
```

OUTPUT

```
enter a character (in lower case)to check if it is vowel or not c
not Vowel

================ RESTART: C:/Users/user/AppData/Local/Programs/Python/Python311/a.py ================
enter a character (in lower case)to check if it is vowel or not e
Vowel
```

Q3 #Write a program that inputs two tuples and creates a third that contains all elements of the first followed by all the elements of the second.

Code

```
tup1 = eval(input("Enter the elements of first tuple: "))
tup2 = eval(input("Enter the elements of second tuple: "))
tup3 = tup1 + tup2
print(tup3)
```

OUTPUT

```
Enter the elements of first tuple: 1,3,5,7,9
Enter the elements of second tuple: 2,4,6,8,10
(1, 3, 5, 7, 9, 2, 4, 6, 8, 10)
```

Q4 #Write a program using function to create a dictionary, which accepts the marks in five different subjects with subject as key and marks as value. Display the dictionary and the subjects getting marks 90 and above.

Code

```
def display_marks90(marks_dict):
    for i in marks_dict:
        if marks_dict[i]>90:
            print(i,marks_dict[i])

marks_dict = {}
for i in range(5):
    subject = input("Enter the name of subject : ")
    marks = int(input("Enter the marks : "))
    marks_dict[subject] = marks

print("\nDictionary where marks above 90 are:")
display_marks90(marks_dict)
```

OUTPUT

```
Enter the name of subject : English
Enter the marks : 90
Enter the name of subject : Chemistry
Enter the marks : 87
Enter the name of subject : Physics
Enter the marks : 95
Enter the name of subject : Maths
Enter the marks : 98
Enter the name of subject : Computer Science
Enter the marks : 89

Dictionary where marks above 90 are:
Physics 95
Maths 98
```

Q5 #Write a program using function that takes one argument (a positive integer) and reports if the argument is prime or not.

Code

```
def check_prime(n):
    if (n == 1):
        return False
    elif (n == 2):
        return True
    else:
        for x in range(2, n):
            if (n % x == 0):
                return False
```

```
        return True

n=int(input('enter any no'))
if check_prime(n):
    print('prime')
else:
    print('not prime')
```

OUTPUT

```
enter any no12
not prime

========== RESTART: C:/Users/user/AppData/Local/Programs/Python/Python311/a.py ==========
enter any no13
prime
```

Q6 #Write a program to generate random numbers between 1 to 6 and check whether a user won a lottery or not.

Code

```
import random

def generate_lottery_number(guess):
  # Generate a random 6-digit lottery number
    lottery_number = random.randint(1, 6)
    print('lottery number',lottery_number)
    if guess>=1 and guess<=6:
        if guess==lottery_number:
            print('won lottery')
        else:
            print('not won lottery')
    else:
        print('Invalid, enter no between 1 and 6')

guess_no=int(input('enter any no between 1 and 6'))
generate_lottery_number(guess_no)
```

OUTPUT

```
enter any no between 1 and 62
lottery number 5
not won lottery

========== RESTART: C:/Users/user/AppData/Local/Programs/Python/Python311/a.py ==========
enter any no between 1 and 65
lottery number 1
not won lottery

========== RESTART: C:/Users/user/AppData/Local/Programs/Python/Python311/a.py ==========
enter any no between 1 and 62
lottery number 3
not won lottery

========== RESTART: C:/Users/user/AppData/Local/Programs/Python/Python311/a.py ==========
enter any no between 1 and 66
lottery number 6
won lottery
```

Q7 #Write a program using function to generate Fibonacci series up to n. Pass the value of 'n' to a #function and displayall the numbers of the series in the function as a list elements.

Code

```
def fibo(n):
   n1, n2 = 0, 1
   print("Fibonacci sequence:")
   print(n1)
   print(n2)
   count = 0
   while count < n-2:
      c=n1+n2
      print(c)
      n1=n2
      n2=c
      count+=1

nterms = int(input("How many terms? "))
fibo(nterms)
```

OUTPUT

```
How many terms? 8
Fibonacci sequence:
0
1
1
2
3
5
8
13
```

Q8 # Write a program to copy the records of that students having marks greater than 95 from binary file "marks dat" into the "toppers.dat" file.

Code

```
import pickle
def fun():
    f=open("marks.DAT","rb")
    f1=open("toppers.DAT","wb")
    while True:
        try:
            r=pickle.load(f)
            if(r[2]>95:
                pickle.dump(r,f1)
        except:
            break
    print(c)
    f.close()
    f1.close()
```

Q9 #Write a program that reads a text file and then creates a new file where each character's case is inverted.

Code

```
f=open('file.txt','w')
f.write('hello world')
f.close()

f1 = open("output1.txt", "w")
with open("file.txt", "r") as myfile:
    data = myfile.read()
for i in data.split():
    data_1 = i[::-1]
    f1.write(data_1)
    f1.write(' ')
f1.close()

with open('output1.txt','r') as f2:
    print(f2.read())
```

OUTPUT

```
olleh dlrow
```

Q10 #Read a text file and display the number of vowels/ consonants/ uppercase/ lowercase characters in the file.

Code

```
f=open('file.txt','w')
f.write('Hello World')
f.close()

v=0
upp=0
low=0
c=0
with open("file.txt", "r") as f1:
    d=f1.read()
    print('file content is: ', d)
    for i in d:
        if i.islower():
            low=low+1
        elif i.isupper():
            upp=upp+1
        if i.isalpha():
            if i.lower() in 'aeiou':
                v=v+1
            else:
                c=c+1
    print('vowels',v)
    print('consonants',c)
    print('upper case',upp)
```

```
    print('lower case',low)
```

OUTPUT

```
file content is:  Hello World
vowels 3
consonants 7
upper case 2
lower case 8
```

Q11 #Write a function in PYTHON to count and display the number of words starting with alphabet ‘A’ or ‘a’ present in a text file “LINES.TXT”.

Code

```
f=open('LINES.TXT','w')
f.write('A boy is playing there. There is a playground. An aeroplane is in the sky.\nAre you getting it?')
f.close()

with open("LINES.TXT", "r") as f1:
    d=f1.read()
    print('file content is: ', d)
    count=0
    for i in d.split():
        if i[0]=='A' or i[0]=='a':
            print(i)
            count=count+1
    print('total number of words starting with A or a',count)
```

OUTPUT

```
file content is:  A boy is playing there. There is a playground. An aeroplane is in the sky.
Are you getting it?
A
a
An
aeroplane
Are
total number of words starting with A or a 5
```

Q12 #Consider a binary file “Emp.dat” containing details such as empno: ename:salary(separator‘:’).Write a python function to display details of those employees who are earning between 20000 and 40000.

Code

```
import pickle
def add():
    f=open('EMP.dat','wb')
    n=int(input('how many records to add'))
    for i in range(n):
        en=int(input('enter employee no'))
        ename=input('enter employee name')
        sal=int(input('enter employee salary'))
```

```
        pickle.dump([en,ename,sal],f)
    f.close()

def show():
    f=open('EMP.dat','rb')
    while True:
        try:
            d=pickle.load(f)
            if d[2]>=20000 and d[2]<=40000:
                print(d)
        except:
            pass
    f.close()

add()
show()
```

OUTPUT

```
how many records to add3
enter employee no1
enter employee namefsff
enter employee salary34000
enter employee no2
enter employee namefdddg
enter employee salary23000
enter employee no4
enter employee namedgdgg
enter employee salary16000
[1, 'fsff', 34000]
[2, 'fdddg', 23000]
```

Q13 #Write a program that will write a string in binary file "school.dat" and display the words of the string in reverse order.

Code

```
import pickle
def add():
    f=open('school.dat','wb')
    n=int(input('how many strings to add'))
    for i in range(n):
        s=input('enter string')
        pickle.dump(s,f)
    f.close()

def show():
    f=open('school.dat','rb')
    while True:
        try:
            d=pickle.load(f)
            for i in d.split():
                print(i[::-1],end=' ')
```

```
        except:
            pass
    f.close()

add()
show()
```

OUTPUT

```
how many strings to add3
enter string this is my file
enter string to check whether it is showing
enter string content in reverse order
siht si ym elif ot kcehc rehtehw ti si gniwohs tnetnoc ni esrever redro
```

Q14 #Write a program to read data from a text file DATA.TXT, and display word, which have maximum/minimum characters.

Code

```
f=open('DATA.TXT','w')
f.write('A boy is playing there There is a playground An aeroplane is in the sky Are you getting it')
f.close()


with open("DATA.TXT", "r") as f1:
    g=[]
    d=f1.read()
    print('file content is',d)
    for i in d.split():
        g.append(len(i))
    m=max(g)
    n=min(g)
    y=[]
    z=[]
    for i in d.split():
        if len(i)==m:
            y.append(i)
        elif len(i)==n:
            z.append(i)
        else:
            continue
    print('-'*100)
    print('maximum characters word : ',y)
    print('minimum characters word : ',z)
```

OUTPUT

```
file content is A boy is playing there There is a playground An aeroplane is in the sky Are you getting it
----------------------------------------------------------------------------------------------------
maximum characters word :  ['playground']
minimum characters word :  ['A', 'a']
```

Q15 #Write the function definition for WORD4CHAR() in PYTHON to read the content of a text file FUN.TXT, and display all those words, which have four characters in it.
Example:
If the content of the file Fun.TXT is as follows:
“When I was a small child, I used to play in the garden with my grand mom. Those days were amazingly funful and I remember all the moments of that time”
The function WORD4CHAR() should display the following:
“When used play with days were that time”

Code:

```
f=open('FUN.TXT','w')
f.write('When I was a small child, I used to play in the garden with my grand mom. Those days were
\namazingly funful and I remember all the moments of that time?')
f.close()

def WORD4CHAR():
    with open("FUN.TXT", "r") as f1:
        d=f1.read()
        print('file content is',d)
        print('-'*100)
        print('All these words, have four characters')
        for i in d.split():
            if len(i)==4:
                print(i)


WORD4CHAR()
```

OUTPUT

```
file content is When I was a small child, I used to play in the garden with my grand mom. Those days were
amazingly funful and I remember all the moments of that time?
----------------------------------------------------------------------------------------------------
All these words, have four characters
When
used
play
with
mom.
days
were
that
```

Q16 # Write a definition for function BUMPER() in PYTHON to read each object of a binary file GIFTS.DAT, find and display details of those gifts, which have remarks as “ON DISCOUNT”. Assume that the file GIFTS.DAT is created with the help of lists of following type:
(ID, Gift, Remarks, Price)

Code

```
import pickle
def add():
    with open("GIFTS.DAT", "wb") as f:
```

```
        while True:
            Gid=int(input('enter gift id'))
            gname=input('enter gift name')
            rem=input('enter remarks')
            pr=int(input('enter gift price'))
            pickle.dump([Gid,gname,rem,pr],f)
            ch=input('want to add more')
            if ch=='n' or ch=='N':
                break

def BUMPER():
    with open("GIFTS.DAT", "rb") as f:
        try:
            while True:
                gift = pickle.load(f)
                if gift[2] == "ON DISCOUNT":
                    print(gift)
        except EOFError:
            pass

add()
BUMPER()
```

OUTPUT

```
enter gift id1
enter gift nameflower
enter remarksON DISCOUNT
enter gift price560
want to add morey
enter gift id2
enter gift nameRacing car
enter remarksOUT OF STOCK
enter gift price670
want to add morey
enter gift id3
enter gift namePlaying Cards
enter remarksON DISCOUNT
enter gift price200
want to add moren
[1, 'flower', 'ON DISCOUNT', 560]
[3, 'Playing Cards', 'ON DISCOUNT', 200]
```

Q17 # Create a binary file “batsman.dat” with the fields:b_No,Name,runs,rank and write a function count() in python which accepts the rank from user and counts the number of the file having that rank.

Code

```
import pickle
def add():
    with open("batsman.dat", "wb") as f:
        while True:
            b_no=int(input('enter batsman id'))
            name=input('enter  name')
            run=input('enter runs scored by him')
            rank=int(input('enter his rank'))
```

```
            pickle.dump([b_no,name,run,rank],f)
            ch=input('want to add more')
            if ch=='n' or ch=='N':
                break
def count_rank(rank_val):
    with open("batsman.dat", "rb") as f:
        count = 0
        try:
            while True:
                b = pickle.load(f)
                if b[3] == rank_val:
                    count += 1
        except EOFError:
            pass
    print("Count:", count)


add()
rank=int(input('enter batsman rank'))
count_rank(rank)
```

OUTPUT

```
enter batsman id1
enter  namevirat
enter runs scored by him2000
enter his rank3
want to add morey
enter batsman id2
enter  namerohit
enter runs scored by him2300
enter his rank4
want to add morey
enter batsman id3
enter  nameshubham
enter runs scored by him1500
enter his rank12
want to add morey
enter batsman id4
enter  nameshivam
enter runs scored by him2300
enter his rank4
want to add moren
enter batsman rank4
Count: 2
```

Q18 # Create a binary file with roll number, name and marks. Input a roll number and update the marks.

Code

```
import pickle
def add():
    with open("student.dat", "wb") as f:
        while True:
            rno=int(input('enter rollno'))
            name=input('enter  name')
            marks=int(input('enter marks'))
            pickle.dump([rno,name,marks],f)
            ch=input('want to add more')
            if ch=='n' or ch=='N':
                break

def up(roll):
    new_data = []
    with open("student.dat", "rb") as f:
        try:
            while True:
                rec = pickle.load(f)
                if rec[0] == roll:
                    rec[2] = int(input("Enter new marks: "))
                    new_data.append(rec)
                else:
                    new_data.append(rec)
        except EOFError:
            pass

    with open("student.dat", "wb") as f:
        for rec in new_data:
            pickle.dump(rec, f)

def show():
    with open("student.dat", "rb") as f:
        try:
            while True:
                rec = pickle.load(f)
                print(rec)
        except:
            pass
add()
roll = int(input("Enter roll number to update: "))
up(roll)
show()
```

OUTPUT

```
enter rollno1
enter  nameajay
enter marks78
want to add morey
enter rollno2
enter  namesanjay
enter marks83
want to add morey
enter rollno3
enter  namerohit
enter marks92
want to add moren
Enter roll number to update: 1
Enter new marks: 82
[1, 'ajay', 82]
[2, 'sanjay', 83]
[3, 'rohit', 92]
```

Q19 # Following is the structure of each record in a data file named ”PRODUCT.DAT”.
{"prod_code":value, "prod_desc":value, "stock":value}
The values for prod_code and prod_desc are strings, and the value for stock is an integer. Write a function in PYTHON to update the file with a new value of stock. The stock and the product_code, whose stock is to be updated, are to be input during the execution of the function.

Code

```
import pickle

f=open('product.dat','wb')
while True:
    a=int(input('enter product code'))
    b=input('enter product description')
    c=int(input('enter stock value'))
    d={'prod_code':a,'prod_desc':b,'stock':c}
    pickle.dump(d,f)
    ch=input('do you wanna add more')
    if ch=='n':
        break
f.close()

def update_stock(p,s):
    updated = []
    with open("PRODUCT.DAT", "rb") as f:
        try:
            while True:
                item = pickle.load(f)
                if item['prod_code']==p:
                    item["stock:"]=s
                    updated.append(item)
        except EOFError:
            pass

    with open("PRODUCT.DAT", "wb") as f:
        for item in updated:
            pickle.dump(item, f)
```

```
def display():
    with open("PRODUCT.DAT", "rb") as f:
        try:
            while True:
                item = pickle.load(f)
                print(item)
        except EOFError:
            pass

p=int(input('enter new product code'))
s=int(input('enter new stock value'))
update_stock(p,s)
display()
```

OUTPUT

```
enter product code1
enter product descriptionpen
enter stock value34
do you wanna add morey
enter product code2
enter product descriptionpencil
enter stock value23
do you wanna add morey
enter product code3
enter product descriptionscale
enter stock value10
do you wanna add moren
enter new product code1
enter new stock value45
{'prod_code': 1, 'prod_desc': 'pen', 'stock': 34, 'stock:': 45}
```

Q20 # Write a Python program to get 10 items' details (itemno,name,price,category) from the user and create a CSV file(Items.csv).

Code

```
import csv
with open("Items.csv", "w", newline='') as f:
    writer = csv.writer(f)
    writer.writerow(["ItemNo", "Name", "Price", "Category"])
    for _ in range(10):
        data = input("Enter itemno, name, price, category: ").split(',')
        writer.writerow(data)
with open('Items.csv','r') as f1:
    d=csv.reader(f1)
    for i in d:
        print(i)
```

OUTPUT

```
Enter itemno, name, price, category: 1,pen,45,stationary
Enter itemno, name, price, category: 2,pencil,20,stationary
Enter itemno, name, price, category: 3,scale,30,stationary
Enter itemno, name, price, category: 4,toy,300, gift
Enter itemno, name, price, category: 5,wallet,389,gift
Enter itemno, name, price, category: 6,eraser,10,stationary
Enter itemno, name, price, category: 7,shirt,400,garment
Enter itemno, name, price, category: 8,trouser,560,garment
Enter itemno, name, price, category: 9,crocin,12,medicine
Enter itemno, name, price, category: 10,atta,300,grocery
['ItemNo', 'Name', 'Price', 'Category']
['1', 'pen', '45', 'stationary']
['2', 'pencil', '20', 'stationary']
['3', 'scale', '30', 'stationary']
['4', 'toy', '300', ' gift']
['5', 'wallet', '389', 'gift']
['6', 'eraser', '10', 'stationary']
['7', 'shirt', '400', 'garment']
['8', 'trouser', '560', 'garment']
['9', 'crocin', '12', 'medicine']
['10', 'atta', '300', 'grocery']
```

Q21 # Write a menu-based program to perform the operation on stack in python.

Code

```
stack = []

while True:
    print("1.Push 2.Pop 3.Display 4.Exit")
    ch = int(input("Enter choice: "))
    if ch == 1:
        stack.append(input("Enter item: "))
    elif ch == 2:
        if stack: print("Popped:", stack.pop())
        else: print("Empty stack")
    elif ch == 3:
        print(stack)
    else:
        break
```

OUTPUT

```
1.Push 2.Pop 3.Display 4.Exit
Enter choice: 1
Enter item: 3
1.Push 2.Pop 3.Display 4.Exit
Enter choice: 1
Enter item: 4
1.Push 2.Pop 3.Display 4.Exit
Enter choice: 1
Enter item: 5
1.Push 2.Pop 3.Display 4.Exit
Enter choice: 3
['3', '4', '5']
1.Push 2.Pop 3.Display 4.Exit
Enter choice: 2
Popped: 5
1.Push 2.Pop 3.Display 4.Exit
Enter choice: 3
['3', '4']
1.Push 2.Pop 3.Display 4.Exit
Enter choice: 4
```

Q22 # Write a program to connect Python with MySQL using database connectivity and perform the following operations on data in database: Fetch, Update and delete the data

Code

```
import mysql.connector
con=mysql.connector.connect(host='localhost',passwd='1234',user='root',database='school')
cur=con.cursor()
#update
s=input('enter designation')
cur.execute("update emp set sal=sal+sal*0.20 where designation='{}' ".format(s))
con.commit()
#fetch
cur.execute('select * from emp')
r=cur.fetchone()
for i in r:
    print(i)
z=cur.fetchmany(2)
for i in z:
    print(i)
```

OUTPUT

```
enter designationclerk
(Decimal('7369'), 'SMITH', 'CLERK', Decimal('7902'), datetime.date(1980, 12, 17), Decimal('960.00'), None, Decimal('20'))
(Decimal('7499'), 'ALLEN', 'SALESMAN', Decimal('7698'), datetime.date(1981, 2, 20), Decimal('1600.00'), Decimal('300.00'),
Decimal('30'))
(Decimal('7521'), 'WARD', 'SALESMAN', Decimal('7698'), datetime.date(1981, 2, 22), Decimal('1250.00'), Decimal('500.00'),
Decimal('30'))
(Decimal('7654'), 'MARTIN', 'SALESMAN', Decimal('7698'), datetime.date(1981, 9, 28), Decimal('1250.00'), Decimal('1400.0
0'), Decimal('30'))
(Decimal('7788'), 'SCOTT', 'ANALYST', Decimal('7566'), datetime.date(1982, 12, 9), Decimal('3000.00'), None, Decimal('20'))
(Decimal('7839'), 'KING', 'PRESIDENT', None, datetime.date(1981, 11, 17), Decimal('5000.00'), None, Decimal('10'))
(Decimal('7844'), 'TURNER', 'SALESMAN', Decimal('7698'), datetime.date(1981, 9, 8), Decimal('1500.00'), Decimal('0.00'), D
ecimal('30'))
(Decimal('7876'), 'ADAMS', 'CLERK', Decimal('7788'), datetime.date(1983, 1, 12), Decimal('1320.00'), None, Decimal('20'))
(Decimal('7900'), 'JAMES', 'CLERK', Decimal('7698'), datetime.date(1981, 12, 3), Decimal('1140.00'), None, Decimal('30'))
(Decimal('7902'), 'FORD', 'ANALYST', Decimal('7566'), datetime.date(1981, 12, 3), Decimal('3000.00'), None, Decimal('20'))
(Decimal('7934'), 'MILLER', 'CLERK', Decimal('7782'), datetime.date(1982, 1, 23), Decimal('1560.00'), None, Decimal('10'))
(Decimal('9876'), 'AMAR', 'ANALYST', Decimal('7654'), datetime.date(1991, 4, 12), Decimal('3456.00'), None, Decimal('10'))
```

Q23 Consider the following tables STORE and SUPPLIERS and answer (a) and (b) parts of this question:

Table: - Store

ItemNo.	Item	Scode	Qty	Rate	LastBuy
2005	Sharpener Classic	23	60	8	31-Jun-09
2003	Ball Pen 0.25	22	50	25	01-Feb-10
2002	Gel Pen Premium	21	150	12	24-Feb-10
2006	Gel Pen Classic	21	250	20	11-Mar-09
2001	Eraser Small	22	220	6	19-Jan-09
2004	Eraser Big	22	110	8	02-Dec-09
2009	Ball Pen 0.5	21	180	18	03-Nov-09

Table:- Supplier

Scode	Sname
21	Premium Stationers
23	Soft Plastics
22	Tetra Supply

A.Write SQL commands for the following statements:

(i) To display details of all the items in the Store table in ascending order of LastBuy.

Ans select * from store order by LastBuy;

(ii) To display ItemNo and Item name of those items from Store table, whose Rate is more than 15 Rupees.

Ans select Itemno,Item from store where rate>15;

(iii) To display the details of those items whose Supplier code (Scode) is 22 or Quantity in Store (Qty) is more than 110 from the table Store.

Ans select * from store where scode =22 or qty >110;

(iv) To display Minimum Rate of items for each Supplier individually as per Scode from the table Store

Ans select min(rate) from store group by scode;

B.Give the output of the following SQL queries:

(i) SELECT COUNT(DISTINCT Scode) FROM Store;

COUNT(DISTINCT Scode)
3

(ii) SELECT Rate*Qty FROM Store WHERE ItemNo=2004;

Rate*Qty
880

(iii) SELECT Item,Sname FROM Store S, Suppliers P WHERE S.Scode=P.Scode AND ItemNo=2006;

Item	Sname
Gel Pen Classic	Premium Stationers

(iv) SELECT MAX(LastBuy) FROM Store;

MAX(LastBuy)
24-Feb-10

Q24

Consider the following table Item and Customer.

Table:- ITEM

i_ID	ItemName	Manufacturer	Price
PC01	Personal Computer	ABC	35000
LC05	Laptop	ABC	55000
Pc03	Personal Computer	XYZ	32000
Pc06	Personal Computer	COMP	37000
Lc03	Laptop	PQR	57000

Table : CUSTOMER

C_ID	CustomerName	City	I_ID
01	N Roy	Delhi	LC03
06	H Singh	Mumbai	PC03
12	R Pandey	Delhi	PC06
15	C Sharma	Delhi	LC03
16	K Agarwal	Banglore	PC01

A. Write SQL commands for the following statements:

(i) To display the details of those Customer whose City is Delhi.
Ans select * from customer where city='Delhi';

(ii) To display the details of Item whose Price is in the range of 3500 to 55000 (Both values included).
Ans select * from item where price between 3500 and 55000

(iii) To display the CustomerName, City from table Customer, and ItemName and Price from table Item with their corresponding matching I_ID.
Ans select customername,city,itemname,price from customer,item where item.I_id=customer.I_id;

(iv) To increase the Price of all Items by 1000 in the table Item.
Ans update item set price=price+1000;

B. Give the output of the following SQL queries:

(v) SELECT DISTINCT City FROM Customer;

DISTINCT City
Delhi
Mumbai
Banglore

(vi) SELECT ItemName, Max(Price), Count(*) FROM Item GROUP BY ItemName;

ItemName	Max(Price)	Count(*)
Personal Computer	37000	3
Laptop	57000	2

(vii) SELECT CustomerName, Manufacturer From Item, Customer WHERE Item.I_Id=Customer.I_Id;

CustomerName	Manufacturer
K Agarwal	ABC
H Singh	XYZ
R Pandey	COMP
C Sharma	PQR

(viii) SELECT ItemName, Price * 100 FROM Item WHERE Manufacturer = 'ABC';

ItemName	Price * 100
Personal Computer	3500000
Laptop	5500000

Q25 **Consider the following tables.**

TABLE : SENDER

SenderID	SenderName	SenderAddress	SenderCity
ND01	R Jain	2,ABC Appts	New Delhi
MU02	H Sinha	12, Newtown	Mumbai
MU15	S Jha	27/A, Park Street	Mumbai
ND50	T Prasad	122-K, SDA	New Delhi

TABLE : RECIPIENT

RecID	SenderID	RecName	RecAddress	ReCity
KO05	ND01	R Bajpayee	5, Central Avenue	Kolkata
ND08	MU02	S Mohan	116, A vihar	New Delhi
MU19	ND01	H singh	2a, Andheri east	Mumbai
MU32	MU15	P K Swamy	B5, c S Terminus	Mumbai
ND48	ND50	S Tirupathi	13, B1 d, Mayur vihar	New Delhi

A. Write SQL commands for the following statements:

(i) To display the names of all Senders from Mumbai.
Ans select sendername from sender where sendercity='mumbai';

(ii) To display the RecID, SenderName, SenderAddress, RecName, RecAddess for every Recipient
Ans select recid,sendername,senderaddress,recname,recaddress from sender,recipient where sender.senderid=recipient.senderid;

(iii) To display Recipient detail in ascending order of RecName
Ans select * from recipient order by recname;

(iv) To display number of Recipients from each city
Ans seect count(*) from recipient group by recity;

B. Give the output of the following SQL queries:

v. SELECT DISTINCT Sendercity FROM Sender;

DISTINCT Sendercity

New Delhi
Mumbai

VI. SELECT A.SenderName, B.RecName FROM Sender A, Recipient B WHERE A.SenderID=B.SenderID AND B.RecCity='Mumbai';

SenderName	RecName
R Jain	H singh
S Jha	P K Swamy

VII. SELECT RecName, RecAddress FROM Recipient WHERE recCity NOT IN('Mumbai', 'Kolkata');

RecName	RecAddress
S Mohan	116, A vihar
S Tirupathi	13, B1 d, Mayur vihar

VIII. SELECT RecID, RecName FROM Recipient WHERE SenderID='MU02' OR SenderID='ND50';

RecID	RecName
ND08	S Mohan
ND48	S Tirupathi

Q26 Consider the following tables FURNITURE and ARRIVALS: write SQL commands for the questions (a) to (f) and give outputs for SQL queries (i) to (iv) :

TABLE : FURNITURE

NO	ITEMNAME	TYPE	DATEOFSTOCK	PRICE	DISCOUNT
1	White lotus	Double Bed	23/02/02	30000	25
2	Pink feather	Baby cot	20/01/02	7000	20
3	Dolphin	Baby cot	19/02/02	9500	20
4	Decent	Office Table	01/01/02	25000	30
5	Comfort zone	Double Bed	12/01/02	25000	25
6	Donald	Baby cot	24/02/02	6500	15
7	Royal Finish	Office Table	20/02/02	18000	30
8	Royal tiger	Sofa	22/02/02	31000	30
9	Econo sitting	Sofa	13/12/01	9500	25
10	Eating Paradise	Dining Table	19/02/02	11500	25

TABLE : ARRIVALS

NO	ITEMNAME	TYPE	DATEOFSTOCK	PRICE	DISCOUNT
11	Wood Comfort	Double Bed	23/03/03	25000	25
12	Old Fox	Sofa	20/02/03	17000	20
13	Micky	Baby cot	21/02/03	7500	15

A. Write SQL commands for the following statements:

a) To show all information about the Baby cots from the FURNITURE table.
Ans select * from furniture where type='Baby cot';

b) To list the ITEMNAME which are priced at more than 15000 from the FURNITURE table.
Ans select itemname from furniture where price >15000;

c) To list ITEMNAME and TYPE of those items, in which DATEOFSTOCK is before 22/01/02 from the FURNITURE table in descending order of ITEMNAME.
Ans select itemname,type from furniture where dateofstock<'22/01/02' order by itemname desc;

d) To display ITEMNAME and DATEOFSTOCK of those items, in which the DISCOUNT percentage is more than 25 form FURNITURE table.
Ans select itemname,dateofstock from furniture where discount>25;

e) To count the number of items, whose TYPE is "Sofa" from FURNITURE table.

Ans select count(*) from furniture where type='sofa';

f) To insert a new row in the ARRIVALS table with the following data:
14, 'Velvet touch', 'Double bed', {25/03/03}, 25000,30

Ans insert into arrivals values (14, 'Velvet touch', 'Double bed', '25/03/03', 25000,30);

B. Give the output of following SQL statement:

NOTE : Outputs of the below mentioned queries should be based on original data given in both the tables, i.e., without considering the insertion done in (f) part of this question :

I . Select COUNT (distinct TYPE) from FURNITURE;

COUNT (distinct TYPE)
6

II. Select MAX(DISCOUNT) form FURNITURE;

MAX(DISCOUNT)
30

III. Select AVG(DISCOUNT) form FURNITURE where TYPE = 'Baby cot';

AVG(DISCOUNT)
7666.6666

IV. Select SUM(PRICE) from FURNITURE where DATEOFSTOCK<{12/02/02};

sum(price)
66500

Q27

Study the following tables DOCTOR and SALARY and write SQL commands for the questions (i) to (iv) and give outputs for SQL queries (v) to (vi) :

TABLE : DOCTOR

ID	NAME	DEPT	SEX	EXPERIENCE
101	John	ENT	M	12
104	Smith	ORTHOPEDIC	M	5
107	George	CARDIOLOGY	M	10
114	Lara	SKIN	F	3
109	K George	MEDICINE	F	9
105	Johnson	ORTHOPEDIC	M	10
117	Lucy	ENT	F	3
111	Bill	MEDICINE	F	12
130	Morphy	ORTHOPEDIC	M	15

TABLE : SALARY

ID	BASIC	ALLOWANCE	CONSULTATION
101	12000	1000	300
104	23000	2300	500
107	32000	4000	500
114	12000	5200	100
109	42000	1700	200
105	18900	1690	300
130	21700	2600	300

A. Write SQL commands for the following statements:

(i) Display NAME of all doctors who are in "MEDICINE" having more than 10 year experience from the table DOCTOR.

Ans select name from doctor where dept='medicine' and experience>10;

(ii) Display the average salary of all doctors working in "ENT" department using the tables DOCTOR and SALARY. Salary=BASIC + ALLOWANCE.

Ans select avg(basic+allowance) from doctor,salary where doctor.id=salary.id;

(iii) Display the minimum ALLOWANCE of female doctors.

Ans select min(allownance) from salary,doctor where doctor.id=salary.id and sex='f';

(iv) Display the highest consultation fee among all male doctor

Ans select max(consultation) from salary,doctor where doctor.id=salary.id and sex='m';

B. Give the output of following SQL statement:

(v) SELECT count(*) from DOCTOR where SEX="F".

Count(*)
4

(vi) SELECT NAME, DEPT, BASIC from DOCTOR Salary WHERE DEPT="ENT" AND DOCTORID=SALARY.ID

Name	Dept	Basic
John	ENT	12000