HIGHER SECONDARY SECOND YEAR

COMPUTER SCIENCE

PRACTICAL PROGRAMS WITH SOLUTION

2024 - 25

J. KAVITHA, **B.Sc, B.Ed, M.C.A, M.Phil.,**
Computer Instructor Gr - I
GHSS, S.S.KULAM
Coimbatore – 641107.
https://kavikalvi.freeweb.co.in/

INDEX

S No	Question Number	Program Name
1	PY1	(a) Calculate Factorial (b) Sum of Series
2	PY2	(a) Odd or Even (b) Reverse the String
3	PY3	Generate values and remove odd numbers
4	PY4	Generate prime numbers and set operations
5	PY5	Display a string elements - using class
6	DB6	MySQL - Employee table
7	DB7	MySQL - Student table
8	PY8	Python with CSV
9	PY9	Python with SQL
10	PY10	Python Graphics with PIP

PY1(a) – Calculate Factorial

1(a) **Write a program to calculate the factorial of the given number using for loop (Don't use built-in function factorial).**

Aim: To calculate the factorial of the given number using for loop.

Coding:

```
num = int(input("Enter the Number:"))
fact = 1
for i in range(1,num+1):
        fact = fact*i
print("Factorial of",num,"is",fact)
```

Output:

```
Enter the Number: 5
Factorial of 5 is 120
```

Result: Thus the Python program to calculate factorial has been done and the output is verified.

PY1 (b) - Sum of Series

1(b) **Write a program to sum the series 11 /1 + 22/2 + 33/3 + ……. nn/n**

Aim: To calculate the sum of the series : 11 /1 + 22/2 + 33/3 + ……. nn/n

Coding:

```
n = int(input("Enter a value of n:"))
s=0.0
for i in range(1,n+1):
        s=s+(i**i)/i
print("The Sum of Series is ",s)
```

Output:

```
Enter a value of n: 4
The Sum of Series is  76.0
```

Result: Thus the Python program to calculate the sum of the series has been done and the output is verified.

PY2(a) – Odd or Even

2(a) **Write a program using functions to check whether a number is even or odd.**

Aim: To check whether a number is even or odd using user defined function.

Coding:

```
def oddeven(a):
    if(a%2==0):
        return "Even"
    else:
        return "Odd"
num = int(input("Enter a Number:"))
print("The given Number is ",oddeven(num))
```

Output 1: Enter a Number:7

The given Number is Odd

Output 2: Enter a Number:6

The given Number is Even

Result: Thus the Python program to check whether a number is odd or even has been done and the output is verified.

PY2(b) – Reverse the string

2(b) **Write a program to create reverse of the given string. For example, "wel" = "lew". (Don't use string slice with stride operation)**

Aim: To create a reverse of the given string.

Coding:

```
def reverse(str1):
    str2 = ' '
    for i in str1:
        str2 = i + str2
    return str2
word = input("\n Enter a String: ")
print("\n The Reverse of the given string is: ",reverse(word))
```

Output:

Enter a String: school

The Reverse of the given string is: loohcs

Result: Thus the Python program to reverse the string has been done and the output is verified.

PY3 – Generate values and remove odd numbers

3. **Write a program to generate values from 1 to 10 and then remove all the odd numbers from the list.**

   Aim: To generate values from 1 to 10 and then remove all the odd numbers from the list.

   Coding:
```
num = list(range(1,11))
print("Numbers from 1 to 10 ...\n", num)
for i in num:
   if(i%2 ==1):
       num.remove(i)
print("The values after removed odd numbers ...\n", num)
```

   Output:
```
Numbers from 1 to 10...
 [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
The values after removed odd numbers ...
[2, 4, 6, 8, 10]
```

   Result: Thus the Python program to generate values from 1 to 10 and then remove all the odd numbers from the list has been done and the output is verified.

.

PY4 – Generate prime numbers and set operations

4. **Write a Program that generate a set of prime numbers and another set of odd numbers. Display the result of union, intersection, difference and symmetric difference operations.**

Aim: To generate a set of prime numbers and another set of odd numbers, and to display the result of union, intersection, difference and symmetric difference operations.

Coding:

```
odd=set(range(1,10,2))
prime=set()
for i in range(2,10):
    j=2
    f=0
    while (j<=i/2):
        if i%j==0:
            f+=1
        j+=1
    if f==0:
        prime.add(i)
print("Odd Numbers            :",odd)
print("Prime Numbers          :",prime)
print("Union                  :",odd.union(prime))
print("Intersection           :",odd.intersection(prime))
print("Difference             :",odd.difference(prime))
print("Symmetric Difference   :",odd.symmetric_difference(prime))
```

Output:

```
Odd Numbers          : {1, 3, 5, 7, 9}
Prime Numbers        : {2, 3, 5, 7}
Union                : {1, 2, 3, 5, 7, 9}
Intersection         : {3, 5, 7}
Difference           : {1, 9}
Symmetric Difference : {1, 2, 9}
```

Result: Thus the Python program to generate prime numbers and set operations has been done and the output is verified.

PY5 – Display a string elements – using class

5. **Write a program to accept a string and print the number of uppercase, lowercase, vowels, consonants and spaces in the given string using Class.**

Aim: To accept a string and print the number of uppercase, lowercase, vowels, consonants and spaces in the given string using Class.

Coding:

```
class string:
    upper,lower,vowels,consonants,space = 0,0,0,0,0
    def process(self):
        str1=str(input("Enter the String:"))
        for ch in str1:
            if (ch.isupper()):
                self.upper+=1
            if (ch.islower()):
                self.lower+=1
            if (ch in ('A','a','E','e','I','i','O','o','U','u')):
                self.vowels+=1
            if (ch not in ('A','a','E','e','I','i','O','o','U','u', ' ')):
                self.consonants+=1
            if (ch==" "):
                self.space+=1
        print("Uppercase letters    :",self.upper)
        print("Lowercase letters    :",self.lower)
        print("Vowels letters       :",self.vowels)
        print("Consonants letters   :",self.consonants)
        print("Blank Space          :",self.space)
        return
S = string()
S.process()
```

Output:

```
Enter the String: Welcome to Computer Science
Uppercase letters        : 3
Lowercase letters        : 21
Vowels letters           : 10
Consonants letters       : 14
Blank Space              : 3
```

Result: Thus the Python program to display a string elements – using class has been done and the output is verified.

DB6 – MySQL – Employee table

6. Create an Employee Table with the fields Empno, Empname, Desig, Dept, Age and Place. Enter five records into the table.

- **Add two more records to the table.**
- **Modify the table structure by adding one more field namely date of joining.**
- **Check for Null value in doj of any record.**
- **List the employees who joined after 01/01/2018.**

Aim: To Create employee table with Empno, Empname, Desig, Dept, Age and Place fields and manipulate the records as given therein.

SQL QUERIES AND OUTPUT:

(i) Creating Table Employee:

mysql> Create table Employee (Empno integer(4) primary key,
Empname varchar(20), Desig varchar(10), Dept varchar(10),
Age integer(2), Place varchar(10));

(ii) View Table Structure:

mysql> Desc Employee;

Field	Type	Null	Key	Default	Extra
Empno	int(4)	NO	PRI	NULL	
Empname	varchar(20)	YES		NULL	
Desig	varchar(10)	YES		NULL	
Dept	varchar(10)	YES		NULL	
Age	int(2)	YES		NULL	
Place	varchar(10)	YES		NULL	

6 rows in set (0.00 sec)

(iii) Inserting Data into Table:

mysql> Insert into employee values(1221, 'Sidharth', 'Officer', 'Accounts', 45, 'Salem');
mysql> Insert into employee values(1222, 'Naveen', 'Manager', 'Admin', 32, 'Erode');
mysql> Insert into employee values(1223, 'Ramesh', 'Clerk', 'Accounts', 33, 'Ambathur');
mysql> Insert into employee values(1224, 'Abinaya', 'Manager', 'Admin', 28, 'Anna Nagar');
mysql> Insert into employee values(1225, 'Rahul', 'Officer', 'Accounts', 31, 'Anna Nagar');

(iv) Select all the record:

mysql> select * from Employee;

Empno	Empname	Desig	Dept	Age	Place
1221	Sidharth	Officer	Accounts	45	Salem
1222	Naveen	Manager	Admin	32	Erode
1223	Ramesh	Clerk	Accounts	33	Ambathur
1224	Abinaya	Manager	Admin	28	Anna Nagar
1225	Rahul	Officer	Accounts	31	Anna Nagar

5 rows in set (0.00 sec)

(v) Adding two more records:

mysql> Insert into employee values(3226, 'Sona', 'Manager', 'Accounts', 42, 'Erode');
mysql> Insert into employee values(3227, 'Rekha', 'Officer', 'Admin', 34, 'Salem');
mysql> select * from Employee;

Empno	Empname	Desig	Dept	Age	Place
1221	Sidharth	Officer	Accounts	45	Salem
1222	Naveen	Manager	Admin	32	Erode
1223	Ramesh	Clerk	Accounts	33	Ambathur
1224	Abinaya	Manager	Admin	28	Anna Nagar
1225	Rahul	Officer	Accounts	31	Anna Nagar
3226	Sona	Manager	Accounts	42	Erode
3227	Rekha	Officer	Admin	34	Salem

7 rows in set (0.00 sec)

(vi) Adding one more Field:

mysql> Alter table employee add(doj date);
desc employee;

Field	Type	Null	Key	Default	Extra
Empno	int(4)	NO	PRI	NULL	
Empname	varchar(20)	YES		NULL	
Desig	varchar(10)	YES		NULL	
Dept	varchar(10)	YES		NULL	
Age	int(2)	YES		NULL	
Place	varchar(10)	YES		NULL	
doj	date	YES		NULL	

7 rows in set (0.00 sec)

(vii) Inserting date of joining to each employee:

mysql> update employee set doj = '21-03-2010' where empno = 1221;
mysql> update employee set doj = '13-05-2012' where empno = 1222;
mysql> update employee set doj = '25-10-2017' where empno = 1223;
mysql> update employee set doj = '17-16-2018' where empno = 1224;
mysql> update employee set doj = '02-01-2018' where empno = 1225;
mysql> update employee set doj = '31-12-2017' where empno = 3226;
mysql> update employee set doj = '16-08-2015' where empno = 3227;
mysql> select * from Employee;

Empno	Empname	Desig	Dept	Age	Place	Doj
1221	Sidharth	Officer	Accounts	45	Salem	2010-03-21
1222	Naveen	Manager	Admin	32	Erode	2012-05-13
1223	Ramesh	Clerk	Accounts	33	Ambathur	2017-10-25
1224	Abinaya	Manager	Admin	28	Anna Nagar	2018-06-17
1225	Rahul	Officer	Accounts	31	Anna Nagar	2018-01-02
3226	Sona	Manager	Accounts	42	Erode	2017-12-31
3227	Rekha	Officer	Admin	34	Salem	2015-08-16

7 rows in set (0.00 sec)

(viii) Checking null value in doj

mysql> select * from empno is null;
Empty set (0.00 sec)

(ix) List the employees who joined after 01/01/2018.

mysql> select * from emp where doj > '01-01-2018';

Empno	Empname	Desig	Dept	Age	Place	doj
1224	Abinaya	Manager	Admin	28	Anna Nagar	6/17/2018
1225	Rahul	Officer	Accounts	31	Anna Nagar	1/2/2018

2 rows in set (0.00 sec)

Result: Created Employee data table containing fields Empno, Empname, Desig, Dept, Age and Place, recorded entries as given therein and checked output.

DB7 - MySQL – Student table

7. **Create a table with following fields and enter data as given in the table below.**

Field Name	Type	Size
Reg_No	char	5
Sname	varchar	15
Age	int	2
Dept	varchar	10
Class	char	3

Data to be entered :

Reg_No	Sname	Age	Dept	Class
M1001	Harish	19	ME	ME1
M1002	Akash	20	ME	ME2
C1001	Sneha	20	CSE	CS1
C1002	Lithya	19	CSE	CS2
E1001	Ravi	20	ECE	EC1
E1002	Leena	21	EEE	EE1
E1003	Rose	20	ECE	EC2

Then, query the followings :
(i) List the students whose department is "Computer Science".
(ii) List all the students of age 20 and more in Mechanical department.
(iii) List the students department wise.
(iv) Modify the class M2 to M1.
(v) Check for the uniqueness of Register no.
Aim: To Create a Student table based on the given fields and enter the given data. Then, to execute the given queries.

SQL QUERIES AND OUTPUT
(1) Creating Table – Student:
    mysql> Create table Student (Reg_no char(4), Sname varchar(20),
        Age integer(2), Dept varchar(10), Class char(3));
View table structure:

    mysql> Desc Employee;

Field	Type	Null	Key	Default	Extra
Reg_no	Char(5)	YES		NULL	
Sname	varchar(20)	YES		NULL	
Age	int(2)	YES		NULL	
Dept	varchar(10)	YES		NULL	
Class	char(3)	YES		NULL	

5 rows in set (0.00 sec)

(2) Inserting Data into table:

mysql> Insert into Student values('M1001','Harish',19,'ME','ME1');
mysql> Insert into Student values ('M1002','Akash',20,'ME','ME2');
mysql> Insert into Student values('C1001','Sneha',20,'CSE','CS1');
mysql> Insert into Student values('C1002','Lithya',19,'CSE','CS2');
mysql> Insert into Student values('E1001','Ravi',20,'ECE','EC1');
mysql> Insert into Student values('E1002','Leena',21,'EEE','EE1');
mysql> Insert into Student values('E1003','Rose',20,'ECE','EC2');

View all records :

mysql> select * from Student

Reg_no	Sname	Age	Dept	Class
M1001	Harish	19	ME	ME1
M1002	Akash	20	ME	ME2
C1001	Sneha	20	CSE	CS1
C1002	Lithya	19	CSE	CS2
E1001	Ravi	20	ECE	EC1
E1002	Leena	21	EEE	EE1
E1003	Rose	20	ECE	EC2

7 rows in set (0.00 sec)

(3) Other Queries:

(i) List the students whose department is "CSE":

mysql> Select * from Student where Dept='CSE';

Reg_No	Sname	Age	Dept	Class
C1001	Sneha	20	CSE	CS1
C1002	Lithya	19	CSE	CS2

2 rows in set (0.03 sec)

(ii) List all the students of age 20 and more in ME department:

mysql> Select * from Student where Age >=20 and Dept='ME';

Reg_No	Sname	Age	Dept	Class
M1002	Akash	20	ME	ME2

1 row in set (0.02 sec)

(iii) List the students department wise:

mysql> Select * from Student Group by Dept Order by Sname;

Reg_No	Sname	Age	Dept	Class
M1001	Harish	19	ME	ME1
E1002	Leena	21	CSE	EE1
E1001	Ravi	20	ECE	EC1
C1001	Sneha	20	EEE	CS14

rows in set (0.00 sec)

(iv) Modify the class M2 to M1:

mysql> Update Student set Class='ME1' where Class='ME2';

Query OK,

1 row affected (0.11 sec)

Rows matched: 1 Changed: 1 Warnings: 0

mysql> select * from Student;

Reg_No	Sname	Age	Dept	Class
M1001	Harish	19	ME	ME1
M1002	Akash	20	ME	ME2
C1001	Sneha	20	CSE	CS1
C1002	Lithya	19	CSE	CS2
E1001	Ravi	20	ECE	EC1
E1002	Leena	21	EEE	EE1
E1003	Rose	20	ECE	EC2

7 rows in set (0.00 sec)

(v) Check for the uniqueness of Register no.

mysql> Select Distinct Reg_No from Student;

Reg_No
M1001
M1002
C1001
C1002
E1001
E1002
E1003

7 rows in set (0.02 sec)

Result: Creates the Student table based on the given fields and inserts the given data into it. Then, run the given quries and chck the output.

PY8 – Python with CSV

8. **Write a program using python to get 10 players name and their score. Write the input in a csv file. Accept a player name using python. Read the csv file to display the name and the score. If the player name is not found give an appropriate message.**

Aim: To get 10 players name and their score. Write the input in a csv file. Accept a player name using python. Read the csv file to display the name and the score. If the player name is not found give an appropriate message.

Coding:

```
import csv
with open('c:\\pyprg\\player.csv','w') as f:
        w = csv.writer(f)
        n=1
         while (n<=10):
                name = input("Player Name?:" )
                score = int(input("Score: "))
                w.writerow([name,score])
                n+=1
print("Player File created")
f.close()
searchname=input("Enter the name to be searched ")
f=open('c:\\pyprg\\player.csv','r')
reader =csv.reader(f)
lst=[]
for row in reader:
        lst.append(row)
q=0
for row in lst:
        if searchname in row:
                 print(row)
                q+=1
if(q==0):
        print("string not found")
f.close()
```

Output:

```
Player Name?:      Rohit Sharma
Score: 264
Player Name?:      VirenderSehwag
Score: 219
Player Name?:      Sachin Tendulkar
Score: 200
Player Name?:      Dhoni
Score: 190
Player Name?:Sachin Tendulkar
Score: 250
Player Name?:ViratKohli
Score: 148
Player Name?:Ganguly
Score: 158
Player Name?:KapilDev
Score: 175
Player Name?:Amarnath
Score: 148
Player Name?:SunilGavaskar
Score: 200
Player File created
Enter the name to be searched Sachin Tendulkar
['Sachin Tendulkar', '200']
 ['Sachin Tendulkar', '250']
```

Result: The Program was written according to the given procedures and the results were verified.

PY9 – Python with SQL

9. **Create a sql table using python and accept 10 names and age .sort in descending order of age and display.**

Aim: To create a sql table using python and accept 10 names and age. sort in descending order of age and display.

Coding:

```
import sqlite3
connection = sqlite3.connect("info.db")
cursor = connection.cursor()
#cursor.execute("DROP Table student")
cursor.execute("create table student(name, age)")
print("Enter 10 students names and their ages respectively:")
for i in range(10):
        who =[input("Enter Name:")]
        age =[int(input("Enter Age:"))]
        n =len(who)
        for i in range(n):
                cursor.execute("insert into student values (?, ?)", (who[i],age[i]))
cursor.execute("select * from student order by age desc")
print("Displaying All the Records From student Table in Descending order of age")
print (*cursor.fetchall(),sep='\n' )
```

Output:

Enter 10 students names and their ages respectively:
Enter Name:Rama
Enter Age:45
Enter Name:Meena
Enter Age:46
Enter Name:Bala
Enter Age:17
Enter Name:Leena
Enter Age:13
Enter Name:Kannan
Enter Age:24
Enter Name:Sowmiya
Enter Age:25
Enter Name:Sivabalan
Enter Age:52
Enter Name:Kumaran
Enter Age:54
Enter Name:Viswa
Enter Age:19
Enter Name:Melvin
Enter Age:15
Displaying All the Records From student Table in Descending order of age
('Kumaran', 54)
('Sivabalan', 52)
('meena', 46)
('Rama', 45)
('Sowmiya', 25)
('Kannan', 24)
('Viswa', 19)
('Bala', 17)
('Melvin', 15)
('Leena', 13)

Result: The Program was written according to the given procedures and the results were verified.

PY10 – Python Graphics with Pip

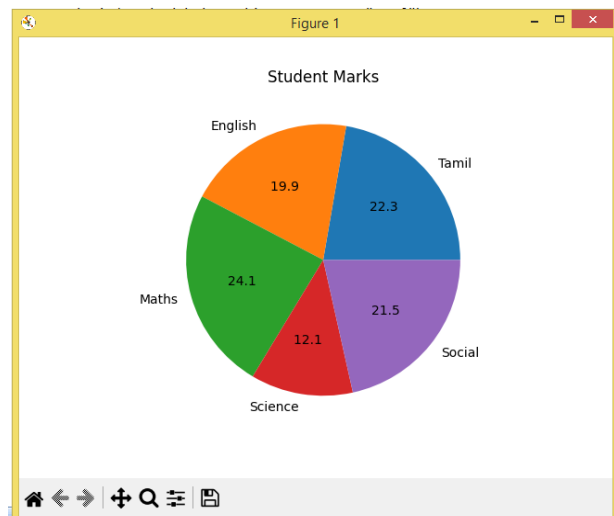**10.** **Write a program to get five marks using list and display the marks in pie chart.**

Aim: To get five marks using list and display the marks in pie chart using matplot.

Coding:

```
import matplotlib.pyplot as plt
marks=[]
i=0
subjects = ["Tamil", "English", "Maths", "Science", "Social"]
while i<5:
    marks.append(int(input("Enter Mark = ")))
    i+=1
for j in range(len(marks)):
    print("{}.{} Mark = {}".format(j+1, subjects[j],marks[j]))
plt.pie (marks, labels = subjects, autopct = "%.1f ")
plt.title("Student Marks")
plt.show()
```

Output:

Enter Mark = 85

Enter Mark = 76

Enter Mark = 92

Enter Mark = 46

Enter Mark = 82

1.Tamil Mark = 85

2.English Mark = 76

3.Maths Mark = 92

4.Science Mark = 46

5.Social Mark = 82

Result: A student's marks in five subjects were taken as input, plotted in a pie chart and the results were checked.

Education Is The
Most Powerful Tool
Which You Can Use
To Change The World.
Good Luck
With Your Education And
Live A Wonderful Life.

J. KAVITHA, B.Sc, B.Ed, M.C.A, M.Phil.,
Computer Instructor Gr - I
GHSS, S.S.KULAM
Coimbatore – 641107.
https://kavikalvi.freeweb.co.in/
✆: **8940762362**