

# VELAMMAL MATRIC HR.SEC.SCHOOL, PONNERI

Ponneri, Thiruvallur DT – 601 204

## 12<sup>th</sup> Computer Science Full Syllabus Manual – 2024

S.NO	EXP.NO	DATE	NAME OF EXPRIEMENT	PAGE NO	SIGNATURE
1	PY01 (a)		Calculate Factorial		
2	PY01 (b)		Sum of Series		
3	PY02 (a)		Odd (or) Even Using Class		
4	PY02 (b)		Reverse the String		
5	PY03		Generate values and remove odd numbers		
6	PY04		Generate Prime Numbers and set Operations		
7	PY05		Display String Elements using class		
8	DB06		MySQL Employee Table		
9	DB07		MySQL Student Table		
10	PY08		Python with CSV		
11	PY09		Python with SQL		
12	PY10		Python Graphics with pip		

**PY01 (a) - Calculate Factorial**

**1(a) Write a program to calculate the factorial of the given number using for loop**

**Aim:**

To write a python program to calculate the factorial of the given number using for loop

**Coding:**

```
num = int(input("Enter a Number: "))  
if (num==0):  
    fact = 1  
fact = 1  
for i in range(1,num+1):  
    fact = fact * i  
print("Factorial of ", num, " is ", fact)
```

**Output:**

```
Enter a Number: 6  
Factorial of 6 is 720
```

**Result:**

The given python program executed successfully.

**PY01 (b) - Sum of Series**

**1(b) Write a program to sum the series:**  $\frac{1}{1} + \frac{2^2}{2} + \frac{3^2}{3} + \dots + \frac{n^n}{n}$

**Aim:**

To write a python program to sum the series:  $\frac{1}{1} + \frac{2^2}{2} + \frac{3^2}{3} + \dots + \frac{n^n}{n}$

**Coding:**

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

**Output:**

Enter a value of n: 4

The sum of the series is 76.0

**Result:**

The given python program executed successfully.

**PY02 (a) - Odd or Even****2 (a) Write a program using functions to check whether a number is even or odd****Aim:**

To write a python program using functions to check whether a number is even or odd

**Coding:**

```
def oddeven(a):  
    if (a%2==0):  
        return 0  
    else:  
        return 1  
  
num = int(input("Enter a number: "))  
if (oddeven(num)==0):  
    print("The given number is Even")  
else:  
    print("The given number is Odd")
```

**Output 1:**

Enter a number: 66  
The given number is Even

**Output 2:**

Enter a number: 77  
The given number is Odd

**Result:**

The given python program executed successfully.

**PY02 (b) - Reverse the String**

**2(b) Write a program to create a mirror of the given string. For example, "wel" = "lew"**

**Aim:**

To write a python program to create a mirror of the given string. For example,  
"wel" = "lew"

**Coding:**

```
def rev(str1):  
    str2=""  
    i=len(str1)-1  
    while i>=0:  
        str2+=str1[i]  
        i=i-1  
    return str2
```

```
word = input("\n Enter a String: ")  
print("\n The Mirror image of the given string is: ", rev(word))
```

**(OR)**

```
word = input("Enter a String: ")  
print("The Mirror image of the given string is: ",word[::-1])
```

**Output:**

```
Enter a String: SARO  
The Mirror image of the given string is: ORAS
```

**Result:**

The given python program executed successfully.

**PY03 - Generate values and remove odd numbers**

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

**Aim:**

To write a program to generate values 1 to 10 and then remove all the odd numbers from the list

**Coding:**

```
num1=[]
for i in range(1,11):
    num1.append(i)
print("Numbers from 1 to 10.....\n",num1)
for j, i in enumerate(num1):
    if(i%2==1):
        del num1[j]
print("The values after removed odd numbers.....\n",num1)
```

**(OR)**

```
num1=[]
for i in range(1,11):
    num1.append(i)
print("Numbers from 1 to 10.....\n",num1)
for i in num1:
    if(i%2==1):
        num1.remove(i)
print("The values after removed odd numbers.....\n",num1)
```

**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:**

The given python program executed successfully.

**PY04 – Generate Prime numbers and Set Operations**

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

**Aim:**

To write a Python program that generates a set of prime numbers and another set of odd numbers. Display the result of union, intersection, difference and symmetric difference operations.

**Coding:**

```
odd=set()
for x in range(0,11):
    if(x%2==1):
        odd.add(x)
primes=set()
for i in range(2,10):
    j=2
    f=0
    while j<=i/2:
        if i%j==0:
            f=1
            j=j+1
    if f==0:
        primes.add(i)
print("Odd Numbers: ", odd)
print("Prime Numbers: ", primes)
print("Union: ", odd|primes)
print("Intersection: ", odd&primes)
print("Difference: ", odd-primes)
print("Symmetric Difference: ", odd^primes)
```

**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:**

The given python program executed successfully.



**PY05 – Display sting 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 write a python program to accept a string and print the number of uppercase, lowercase, vowels, consonants and spaces in the given string using Class

**Coding:**

```
class String:
    def __init__(self):
        self.uppercase=0
        self.lowercase=0
        self.vowels=0
        self.consonants=0
        self.spaces=0
        self.string=""
    def getstr(self):
        self.string=str(input("Enter a String: "))
    def count_upper(self):
        for ch in self.string:
            if (ch.isupper()):
                self.uppercase+=1
    def count_lower(self):
        for ch in self.string:
            if (ch.islower()):
                self.lowercase+=1
    def count_vowels(self):
        for ch in self.string:
            if (ch in ('A', 'a', 'e', 'E', 'i', 'I', 'o', 'O', 'u', 'U')):
```

```
        self.vowels+=1
def count_consonants(self):
    for ch in self.string:
        if (ch not in ('A', 'a', 'e', 'E', 'i', 'I', 'o', 'O', 'u', 'U')):
            self.consonants+=1
def count_space(self):
    for ch in self.string:
        if (ch==" "):
            self.spaces+=1
def execute(self):
    self.count_upper()
    self.count_lower()
    self.count_vowels()
    self.count_consonants()
    self.count_space()
def display(self):
    print("The given string contains...")
    print("%d Uppercase letters"%self.uppercase)
    print("%d Lowercase letters"%self.lowercase)
    print("%d Vowels"%self.vowels)
    print("%d Consonants"%self.consonants)
    print("%d Spaces"%self.spaces)

S = String()
S.getstr()
S.execute()
S.display()
```

**Output:**

Enter a String: **Welcome to Computer Science**

The given string contains...

3 Uppercase letters

21 Lowercase letters

10 Vowels

17 Consonants

3 Spaces

**Result:**

The given python program executed successfully.

**DB06 – 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 2018/01/01.

**Aim:**

To Create an Employee Table with the fields

**Empno, Empname, Desig, Dept, Age and Place.**

and performing adding records, modifying table structure, checking null values and selecting records using where clause.

**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&gt; 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	

**vii) Inserting date of joining to each employee:**

```
mysql> update employee set doj = '2010-03-21' where empno=1221;
mysql> update employee set doj = '2012-05-13' where empno=1222;
mysql> update employee set doj = '2017-10-25' where empno=1223;
mysql> update employee set doj = '2018-06-17' where empno=1224;
mysql> update employee set doj = '2018-01-02' where empno=1225;
mysql> update employee set doj = '2017-12-31' where empno=3226;
mysql> update employee set doj = '2015-08-16' 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 emp where empno is null;
```

```
Empty set (0.00 sec)
```

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

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

Empno	Empname	Desig	Dept	Age	Place	doj
1224	Abinaya	Manager	Admin	28	Anna Nagar	2018-06-17
1225	Rahul	Officer	Accounts	31	Anna Nagar	2018-01-02

2 rows in set (0.00 sec)

### Result:

The given MySQL Employee Table Created and Give Details executed successfully.

## DB7 – MySQL Student Table

**7**

Create Student table with following fields and enter data as given in the table below

Field	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 "CSE".
- (ii) List all the students of age 20 and more in ME 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 Student table with given details and enter data as given in the table.

**SQL Queries and Output:****(1) Creating Table - Student**

```
mysql> Create table Student(Reg_No char(5), Sname varchar(20), Age integer(2),
Dept varchar(10), Class char(3));
```

Query OK, 0 rows affected (0.51 sec)

**View table structure:**

```
mysql> desc Student;
```

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 (0.02 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	CS1

4 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)

www.Padasalai.Net

**Result:**

The given MySQL Student Table Created and Give Details executed successfully.

**PY08 – Python with CSV**

**8. Write a program using python to get 10 players 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 write a program using python to get 10 players 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:\pyprogram\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:\pyprogram\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?: Amma

Score: 100

Player Name?: Brother

Score: 98

Player Name?: Captain

Score: 88

Player Name?: Donglee

Score: 55

Player Name?: Enemy

Score: 87

Player Name?: Fighter

Score: 77

Player Name?: Grandpa

Score: 101

Player Name?: Honey

Score: 50

Player Name?: Ironman

Score: 89

Player Name?: Jhonny

Score: 65

Player File created

Enter the name to be searched Enemy

['Enemy', '87']

**Result:**

The given python program executed successfully.

**PY09 - 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
conn=sqlite3.connect("saro.db")
cursor=conn.cursor()
#cursor.execute("DROP TABLE student") (*no need for first time execution)
cursor.execute("CREATE TABLE student(name,age)")
print("Enter the 10 students name 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 the 10 students name and their ages respectively:

Enter Name: Anbu

Enter age: 14

Enter Name: Bala

Enter age: 15

Enter Name: Cameron

Enter age: 17

Enter Name: Donglee

Enter age: 18

Enter Name: Emey

Enter age: 19

Enter Name: Fridee

Enter age: 20

Enter Name: Gabi

Enter age: 21

Enter Name: Happiee

Enter age: 22

Enter Name: Irfon

Enter age: 23

Enter Name: Jonny

Enter age: 24

Displaying All the Records From student Table in Descending order of age

('Jonny', 24)

('Irfon', 23)

('Happiee', 22)

('Gabi', 21)

('Fridee', 20)

('Emey', 19)

('Donglee', 18)

('Cameron', 17)

('Bala', 15)

('Anbu', 14)

### Result:

The given python program executed successfully.



**PY10 - Python Graphics with pip**

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

**Aim:**

To write a program get five marks using list and display the marks in pie chart.

**Coding:**

```
import matplotlib.pyplot as plt
marks=[]
i=0
subjects = ["Tamil","English","Maths","Science","Social"]
while (i<5):
    marks.append(int(input("Enter the mark:")))
    i+=1
for j in range(len(marks)):
    print("{}.{ } Mark = {}".format(j+1, subjects[j],marks[j]))
plt.pie(marks, labels = subjects, autopct = "%.2f ")
plt.show()
```

**Output:**

Enter the mark:97

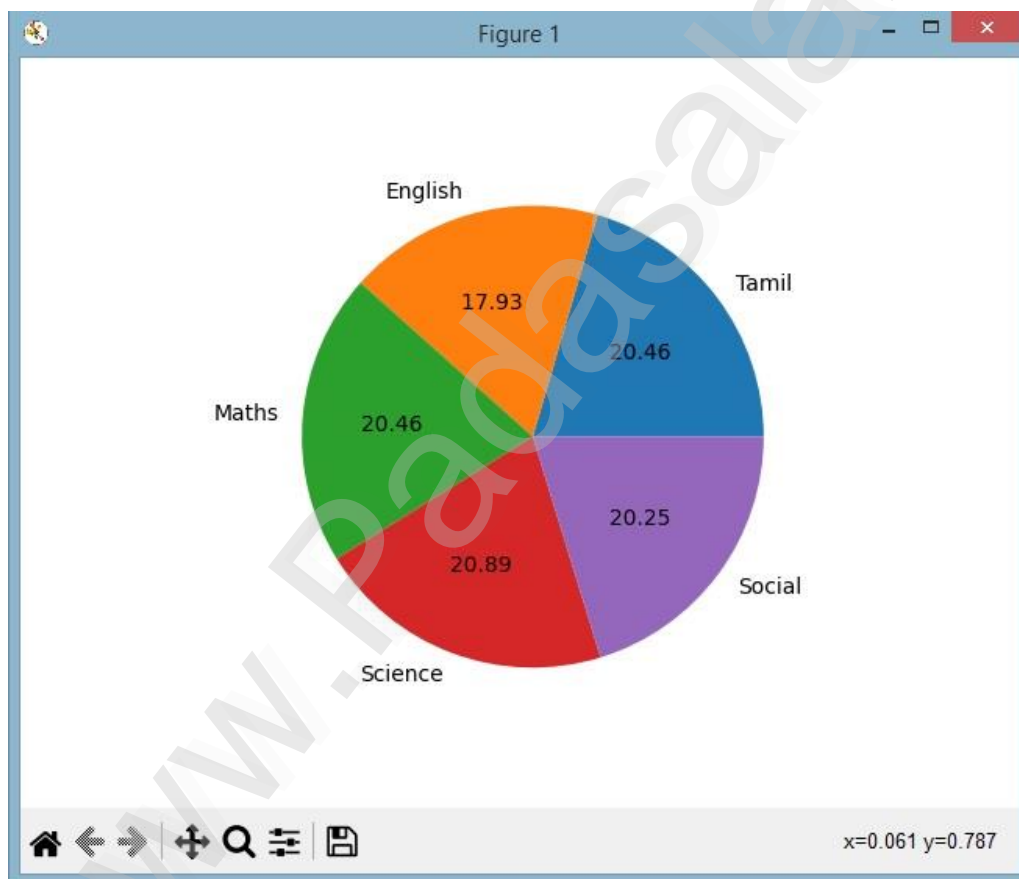
Enter the mark:85

Enter the mark:97

Enter the mark:99

Enter the mark:96

1. Tamil Mark = 97
2. English Mark = 85
3. Maths Mark = 97
4. Science Mark = 99
5. Social Mark = 96

**Result:**

The given python program executed successfully.