

Computer Science Report

V R Darsh
XII-A

October 5, 2023

SQL

Consider the tables given below and answer the questions that follow:

No	Name	Salary	Zone	Age	Grade	Dept
1	Mukul	30000	West	28	A	10
2	Kritika	35000	Centre	30	A	10
3	Naveen	32000	West	40	NULL	20
4	Uday	38000	North	38	C	30
5	Nupur	32000	East	26	NULL	20
6	Moksh	37000	South	28	B	10
7	Shelly	36000	North	26	A	30

Table:Employee

Dept	DName	MinSal	MaxSal	HOD
10	Sales	25000	32000	1
20	Finance	30000	50000	5
30	Admin	25000	40000	7

Table:Department

Create the table Employee.

```
CREATE TABLE Employee(No int, Name char(20), Salary int, Zone char(10), Age int, Grade char(2), Dept int, primary key(No));
```

Create the table Department.

```
CREATE TABLE Department(Dept int, DName char(10), MinSal int, MaxSal int, HOD int, primary key (Dept));
```

Insert data in the table Employee.

```
INSERT INTO Employee VALUES(1,'Mukul', 30000, 'West',28,'A',10);
INSERT INTO Employee VALUES(2,'Kritika', 35000, 'Centre',30,'A',10);
INSERT INTO Employee VALUES(3,'Naveen', 32000, 'West',40,NULL,20);
INSERT INTO Employee VALUES(4,'Uday', 38000, 'North',38,'C',30);
INSERT INTO Employee VALUES(5,'Nupur', 32000, 'East',26,NULL,20);
INSERT INTO Employee VALUES(6,'Moksh', 37000, 'South',28,'B',10);
INSERT INTO Employee VALUES(7,'Shelly', 36000, 'North',26,'A',30);
```

Insert data in the table Department..

```
INSERT INTO Department VALUES(10,'Sales',25000,32000,1);
INSERT INTO Department VALUES(20,'Finance',40000,50000,5);
INSERT INTO Department VALUES(30,'Admin',25000,40000,7);
```

Display the Salary, Zone, and Grade of all the employees.

```
SELECT Salary,Zone,Grade FROM Employee;
```

Display the name of all the employees along with their annual salaries. The Salary column of the table contains monthly salaries of the employees. The new column should be given the name “Annual Salary”.

```
SELECT Name, Salary*12 as Annual_Salary FROM Employee;
```

Display the details of all the employees who are below 30 years of age.

```
SELECT * FROM Employee where Age<30;
```

Display the names of various zones from the table Employee. A zone name should appear only once.

```
SELECT DISTINCT Zone FROM Employee;
```

Display the details of all the employees who are getting a salary of more than 35000 in the department 30.

```
SELECT * FROM Employee WHERE Dept=30 AND Salary>35000;
```

Display the names and salaries of all the employees who are not working in department 20.

```
SELECT Name, Salary FROM Employee WHERE NOT Dept=20;
```

Display the details of all the employees whose salary is between 32000 and 38000.

```
SELECT * FROM Employee WHERE 32000<Salary AND Salary<38000;
```

Display the names of all the employees who are working in department 20 or 30. (Using IN operator)

```
SELECT Name FROM Employee WHERE Dept IN (20,30);
```

Display the details of all the employees whose salary is between 32000 and 38000.

```
SELECT * FROM Employee WHERE Salary BETWEEN 32000 AND 38000;
```

Display the details of all the employees whose grade is between ‘A’ and ‘C’.

```
SELECT * FROM Employee WHERE Grade BETWEEN 'A' AND 'C';
```

Display the name, salary, and age of all the employees whose names start with ‘M’.

SELECT Name,Salary,Age FROM Employee WHERE Name LIKE 'M%';

Display the name, salary, and age of all the employees whose names contain 'a' in the descending order of their names.

SELECT Name,Salary,Age FROM Employee WHERE Name LIKE '%a%' ORDER BY Name DESC;

Display the details of all the employees whose names contain 'a' as the second character.

SELECT * FROM Employee WHERE Name LIKE '_a%';

Display the highest and the lowest salaries being paid in department 10.

SELECT MAX(Salary),MIN(Salary) FROM Employee WHERE Dept=10;

Display the number of employees working in department 10.

SELECT COUNT(No) FROM Employee WHERE Dept=10;

Display the name and salary of all the employees in the ascending order of their salaries.

SELECT Name, Salary FROM Employee ORDER BY Salary;

Display the total number of employees in each department.

SELECT Dept,COUNT(No) FROM Employee GROUP BY Dept;

Display the highest salary, lowest salary, and average salary of each zone.

SELECT MAX(Salary),MIN(Salary),AVG(Salary) FROM Employee GROUP BY Dept;

Put the grade B for all those whose grade is NULL.

UPDATE Employee SET Grade='B' WHERE Grade IS NULL;

Increase the salary of all the employees above 30 years of age by 10%.

UPDATE Employee SET Salary=Salary+Salary*0.1 WHERE Age > 30;

Delete the records of all the employees whose grade is C and salary is below 30000.

DELETE FROM Employee WHERE Grade='C' AND Salary<30000;

Add another column HireDate of type Date in the Employee table.

ALTER TABLE Employee ADD HireDate date;

Display the name,salary of all the employees who work in Sales department.

SELECT Employee.Name, Employee.Salary FROM Employee JOIN Department ON Department.Dept=Employee.Dept WHERE DName='Sales';

Display the Name and Department Name of all the employees.

```
SELECT Employee.Name, Department.DName FROM Employee JOIN Department ON
Department.Dept=Employee.Dept;
```

Display the names of all the employees whose salary is out of the specified range for the corresponding department.

```
SELECT Employee.Name FROM Employee JOIN Department ON Department.Dept=Employee.Dept
WHERE Salary>MaxSal or Salary<MinSal;
```

Display the name of the department and the name of the corresponding HOD for all the departments.

```
SELECT Department.DName, Employee.Name FROM Employee JOIN Department ON
Department.Dept=Employee.Dept WHERE Department.HOD=Employee.No;
```

OUTPUT

```
[vrdarsh@lenovo ~]$ mysql -u root -p
mysql: Deprecated program name. It will be removed in a future release, use '/
usr/bin/mariadb' instead
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 4
Server version: 11.2.2-MariaDB Arch Linux

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> create database Darsh;
Query OK, 1 row affected (0.115 sec)

MariaDB [(none)]> use Darsh
Database changed
MariaDB [Darsh]> CREATE TABLE Employee(No int, Name char(20), Salary int, Zone
char(10), Age int,Grade char(2), Dept int, primary key(No));
Query OK, 0 rows affected (0.320 sec)

MariaDB [Darsh]> CREATE TABLE Department(Dept int, DName char(10), MinSal int,
MaxSal int, HOD int, primary key (Dept));
Query OK, 0 rows affected (0.296 sec)

MariaDB [Darsh]> INSERT INTO Employee VALUES(1,'Mukul', 30000, 'West',28,'A
',10);
Query OK, 1 row affected (0.080 sec)

MariaDB [Darsh]> INSERT INTO Employee VALUES(2,'Kritika', 35000, 'Centre',30,'
A',10);
Query OK, 1 row affected (0.194 sec)
```

```
MariaDB [Darsh]> INSERT INTO Employee VALUES(3,'Naveen', 32000, 'West',40,NULL,20);
```

```
Query OK, 1 row affected (0.109 sec)
```

```
MariaDB [Darsh]> INSERT INTO Employee VALUES(4,'Uday', 38000, 'North',38,'C',30);
```

```
Query OK, 1 row affected (0.117 sec)
```

```
MariaDB [Darsh]> INSERT INTO Employee VALUES(5,'Nupur', 32000, 'East',26,NULL,20);
```

```
Query OK, 1 row affected (0.108 sec)
```

```
MariaDB [Darsh]> INSERT INTO Employee VALUES(6,'Moksh', 37000, 'South',28,'B',10);
```

```
Query OK, 1 row affected (0.110 sec)
```

```
MariaDB [Darsh]> INSERT INTO Employee VALUES(7,'Shelly', 36000, 'North',26,'A',30);
```

```
Query OK, 1 row affected (0.121 sec)
```

```
MariaDB [Darsh]> INSERT INTO Department VALUES(10,'Sales',25000,32000,1);
```

```
Query OK, 1 row affected (0.042 sec)
```

```
MariaDB [Darsh]> INSERT INTO Department VALUES(20,'Finance',40000,50000,5);
```

```
Query OK, 1 row affected (0.032 sec)
```

```
MariaDB [Darsh]> INSERT INTO Department VALUES(30,'Admin',25000,40000,7);
```

```
Query OK, 1 row affected (0.019 sec)
```

```
MariaDB [Darsh]> SELECT Salary,Zone,Grade FROM Employee;
```

Salary	Zone	Grade
30000	West	A
35000	Centre	A
32000	West	NULL
38000	North	C
32000	East	NULL
37000	South	B
36000	North	A

```
7 rows in set (0.001 sec)
```

```
MariaDB [Darsh]> SELECT Name, Salary*12 as Annual_Salary FROM Employee;
```

Name	Annual_Salary
Mukul	360000
Kritika	420000
Naveen	384000
Uday	456000
Nupur	384000
Moksh	444000

```
| Shelly |          432000 |
+-----+-----+
7 rows in set (0.001 sec)
```

```
MariaDB [Darsh]> SELECT * FROM Employee where Age<30;
+---+---+---+---+---+---+---+
| No | Name   | Salary | Zone  | Age  | Grade | Dept |
+---+---+---+---+---+---+---+
| 1  | Mukul  | 30000  | West  | 28   | A     | 10   |
| 5  | Nupur  | 32000  | East  | 26   | NULL  | 20   |
| 6  | Moksh  | 37000  | South | 28   | B     | 10   |
| 7  | Shelly | 36000  | North | 26   | A     | 30   |
+---+---+---+---+---+---+---+
4 rows in set (0.001 sec)
```

```
MariaDB [Darsh]> SELECT DISTINCT Zone FROM Employee;
+-----+
| Zone  |
+-----+
| West  |
| Centre|
| North |
| East  |
| South |
+-----+
5 rows in set (0.001 sec)
```

```
MariaDB [Darsh]> SELECT * FROM Employee WHERE Dept=30 AND Salary>35000;
+---+---+---+---+---+---+---+
| No | Name   | Salary | Zone  | Age  | Grade | Dept |
+---+---+---+---+---+---+---+
| 4  | Uday   | 38000  | North | 38   | C     | 30   |
| 7  | Shelly | 36000  | North | 26   | A     | 30   |
+---+---+---+---+---+---+---+
2 rows in set (0.001 sec)
```

```
MariaDB [Darsh]> SELECT Name, Salary FROM Employee WHERE NOT Dept=20;
+---+---+
| Name   | Salary |
+---+---+
| Mukul  | 30000  |
| Kritika| 35000  |
| Uday   | 38000  |
| Moksh  | 37000  |
| Shelly | 36000  |
+---+---+
5 rows in set (0.001 sec)
```

```
MariaDB [Darsh]> SELECT * FROM Employee WHERE 32000<Salary AND Salary<38000;
+---+---+---+---+---+---+---+
| No | Name   | Salary | Zone  | Age  | Grade | Dept |
+---+---+---+---+---+---+---+
| 2  | Kritika| 35000  | Centre| 30   | A     | 10   |
| 6  | Moksh  | 37000  | South | 28   | B     | 10   |
+---+---+---+---+---+---+---+
```

```
| 7 | Shelly | 36000 | North | 26 | A | 30 |
+---+-----+-----+-----+-----+-----+
3 rows in set (0.001 sec)
```

MariaDB [Darsh]> SELECT Name FROM Employee WHERE Dept IN (20,30);

```
+-----+
| Name |
+-----+
| Naveen |
| Uday |
| Nupur |
| Shelly |
+-----+
4 rows in set (0.000 sec)
```

MariaDB [Darsh]> SELECT * FROM Employee WHERE Salary BETWEEN 32000 AND 38000;

```
+---+-----+-----+-----+-----+-----+
| No | Name | Salary | Zone | Age | Grade | Dept |
+---+-----+-----+-----+-----+-----+
| 2 | Kritika | 35000 | Centre | 30 | A | 10 |
| 3 | Naveen | 32000 | West | 40 | NULL | 20 |
| 4 | Uday | 38000 | North | 38 | C | 30 |
| 5 | Nupur | 32000 | East | 26 | NULL | 20 |
| 6 | Moksh | 37000 | South | 28 | B | 10 |
| 7 | Shelly | 36000 | North | 26 | A | 30 |
+---+-----+-----+-----+-----+-----+
6 rows in set (0.001 sec)
```

MariaDB [Darsh]> SELECT * FROM Employee WHERE Grade BETWEEN 'A' AND 'C';

```
+---+-----+-----+-----+-----+-----+
| No | Name | Salary | Zone | Age | Grade | Dept |
+---+-----+-----+-----+-----+-----+
| 1 | Mukul | 30000 | West | 28 | A | 10 |
| 2 | Kritika | 35000 | Centre | 30 | A | 10 |
| 4 | Uday | 38000 | North | 38 | C | 30 |
| 6 | Moksh | 37000 | South | 28 | B | 10 |
| 7 | Shelly | 36000 | North | 26 | A | 30 |
+---+-----+-----+-----+-----+-----+
5 rows in set (0.001 sec)
```

MariaDB [Darsh]> SELECT Name,Salary,Age FROM Employee WHERE Name LIKE 'M%';

```
+-----+-----+-----+
| Name | Salary | Age |
+-----+-----+-----+
| Mukul | 30000 | 28 |
| Moksh | 37000 | 28 |
+-----+-----+-----+
2 rows in set (0.000 sec)
```

MariaDB [Darsh]> SELECT Name,Salary,Age FROM Employee WHERE Name LIKE '%a%'
ORDER BY Name DESC;

```
+-----+-----+-----+
| Name | Salary | Age |
+-----+-----+-----+
```


Uday	38000	38
Naveen	32000	40
Kritika	35000	30

3 rows in set (0.001 sec)

```
MariaDB [Darsh]> SELECT Name,Salary,Age FROM Employee WHERE Name LIKE '%a%'
ORDER BY Name DESC;
```

Name	Salary	Age
Uday	38000	38
Naveen	32000	40
Kritika	35000	30

3 rows in set (0.001 sec)

```
MariaDB [Darsh]> SELECT MAX(Salary),MIN(Salary) FROM Employee WHERE Dept=10;
```

MAX(Salary)	MIN(Salary)
37000	30000

1 row in set (0.030 sec)

```
MariaDB [Darsh]> SELECT COUNT(No) FROM Employee WHERE Dept=10;
```

COUNT(No)
3

1 row in set (0.001 sec)

```
MariaDB [Darsh]> SELECT Name, Salary FROM Employee ORDER BY Salary;
```

Name	Salary
Mukul	30000
Naveen	32000
Nupur	32000
Kritika	35000
Shelly	36000
Moksh	37000
Uday	38000

7 rows in set (0.001 sec)

```
MariaDB [Darsh]> SELECT Dept,COUNT(No) FROM Employee GROUP BY Dept;
```

Dept	COUNT(No)
10	3
20	2
30	2

```
+-----+-----+
3 rows in set (0.001 sec)
```

```
MariaDB [Darsh]> SELECT MAX(Salary),MIN(Salary),AVG(Salary) FROM Employee
GROUP BY Dept;
```

```
+-----+-----+-----+
| MAX(Salary) | MIN(Salary) | AVG(Salary) |
+-----+-----+-----+
|          37000 |          30000 | 34000.0000 |
|          32000 |          32000 | 32000.0000 |
|          38000 |          36000 | 37000.0000 |
+-----+-----+-----+
3 rows in set (0.001 sec)
```

```
MariaDB [Darsh]> UPDATE Employee SET Grade='B' WHERE Grade IS NULL;
Query OK, 2 rows affected (0.138 sec)
Rows matched: 2  Changed: 2  Warnings: 0
```

```
MariaDB [Darsh]> UPDATE Employee SET Salary=Salary+Salary*0.3 WHERE Age>30;
Query OK, 2 rows affected (0.267 sec)
Rows matched: 2  Changed: 2  Warnings: 0
```

```
MariaDB [Darsh]> DELETE FROM Employee WHERE Grade='C' AND Salary<300000;
Query OK, 1 row affected (0.112 sec)
```

```
MariaDB [Darsh]> ALTER TABLE Employee ADD HireDate date;
Query OK, 0 rows affected (2.400 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

```
MariaDB [Darsh]> SELECT Employee.Name, Employee.Salary FROM Employee JOIN
Department ON Department.Dept=Employee.Dept WHERE DName='Sales';
```

```
+-----+-----+
| Name      | Salary |
+-----+-----+
| Mukul     | 30000  |
| Kritika   | 35000  |
| Moksh     | 37000  |
+-----+-----+
3 rows in set (0.002 sec)
```

```
MariaDB [Darsh]> SELECT Employee.Name, Department.DName FROM Employee JOIN
Department ON Department.Dept=Employee.Dept;
```

```
+-----+-----+
| Name      | DName   |
+-----+-----+
| Mukul     | Sales   |
| Kritika   | Sales   |
| Naveen    | Finance |
| Nupur     | Finance |
| Moksh     | Sales   |
| Shelly    | Admin   |
+-----+-----+
6 rows in set (0.001 sec)
```

```
MariaDB [Darsh]> SELECT Employee.Name FROM Employee JOIN Department ON  
    Department.Dept=Employee.Dept WHERE Salary>MaxSal or Salary<MinSal;
```

```
+-----+
```

```
| Name      |
```

```
+-----+
```

```
| Kritika   |
```

```
| Nupur     |
```

```
| Moksh     |
```

```
+-----+
```

```
3 rows in set (0.001 sec)
```

```
MariaDB [Darsh]> SELECT Department.DName, Employee.Name FROM Employee JOIN  
    Department ON Department.Dept=Employee.Dept WHERE Department.HOD=Employee.  
    No;
```

```
+-----+-----+
```

```
| DName     | Name     |
```

```
+-----+-----+
```

```
| Sales     | Mukul    |
```

```
| Finance   | Nupur    |
```

```
| Admin     | Shelly   |
```

```
+-----+-----+
```

```
3 rows in set (0.000 sec)
```

Python

1. Write a program to define a user defined function to check a number whether it is palindrome or not. (without using built in function)

```
def func1(x):
    c=str(x)
    b=c[::-1]
    if c==b:
        print("It is a palindrome")
    else:
        print("It is not a palindrome")
a=int(input("Enter Number:"))
func1(a)
```

Output:

Enter Number:12321
It is a palindrome

2. Write a python program by defining a function to sum the sequence given below. Take the input n from the user. $1 + 1/1! + 1/2!! + 1/3! + \dots + 1/n!$

```
def func2(x):
    z=0
    if x==0:
        print("The result is 1")
    for j in range(1,x+1):
        k=1
        for i in range(1,j+1):
            k*=i
        z+=(1/k)
    print(z)\textbf{ }
a=int(input("Enter Number:"))
func2(a)
```

Output:

Enter Number:12
1.7182818282861687

3. Write a program for linear search.

```
def func3(x):
    k=[1,2,3,4,5,6,7,8,9,0]
    o=0
    for i in k:
        o+=1
        if i==x:
            print("Number is present at ",o)
            return i
```

```

    print("Number is not present")
a=int(input("Enter Number to find:"))
func3(a)

```

Output:

Enter Number to find:5
Number is present at 5

4. Write a program for bubble sort.

```

def func4():
    l=[112,23,235,21,563,23]
    n=len(l)
    print("Original List: ",l)
    for i in range(n-1):
        for j in range(n-i-1):
            if l[j]>l[j+1]:
                l[j],l[j+1]=l[j+1],l[j]
    print(l)
func4()

```

Output:

Original List:[112, 23, 235, 21, 563, 23]
[21, 23, 23, 112, 235, 563]

5. Write a program to define a user defined function to find the sum of all elements of a list.

```

def func5(l):
    z=0
    for i in l:
        z+=i
    print("The sum is",z)
l=[192,133,78,643]
func5(l)

```

Output:

The sum is 1046

6. Write a program to define a user defined function to print fibonacci series.

```

def func6(n):
    a=0
    b=1
    i=0
    while i<n:
        print(a,end=', ')
        c=a+b
        a=b

```

```

        b=c
        i+=1
n=int(input("Enter the number of numbers in fibbonaci sequence: "))
print("The sequence is: ")
func6(n)

```

Output:

Enter the number of numbers in fibbonaci sequence: 12
The sequence is:
0,1,1,2,3,5,8,13,21,34,55,89,

7. Write a program to print largest of four numbers entered by user.

```

n=int(input("How many number do you want to enter: "))
l=[]
for i in range(n):
    j=int(input("Enter the number: "))
    if len(l)<4:
        l.append(j)
        continue
    else:
        for k in range(0,4):
            if l[k]<j:
                l[k]=j
            break
print("The greatest 4 numbers are: ",l)

```

Output:

How many number do you want to enter: 6
Enter the number: 18
Enter the number: 76
Enter the number: 32
Enter the number: 123
Enter the number: 1236
Enter the number: 2123
The greatest 4 numbers are: [2123, 76, 32, 123]

8. Write a recursive python program to test if a string is palindrome or not.

```

def func8(a):
    z=0
    for i in range(len(a)//2):
        if a[i]==a[len(a)-i-1]:
            z=1
            pass

    else:
        z=0
        print("It is not a Palindrome")

```

```

        break
    if z==1:
        print("It is a Palindrome")

func8("naman")
func8("akshay")

```

Output:

It is a Palindrome
It is not a Palindrome

9. Write a program to display ASCII code of a character and vice versa.

```

a=input("Enter a character: ")
print("The ASCII value is", ord(a))
b=int(input("Enter a ASCII value: "))
print("The character is", chr(b))

```

Output:

Enter a character:l
The ASCII value is 49
Enter a ASCII value:65
The character is A

10. Write a program to calculate the factorial of an integer.

```

def func10(a):
    for i in range(1,a):
        a*=i
    return a

print(func10(12))

```

Output:

479001600

11. Write a python program by defining a function to reverse a number entered by user without using built-in function.

```

def func11(a):
    j=str(a)
    k=int(j[::-1])
    return k

print(func11(122983))

```

Output:

389221

12. Write a program to find the most common words in a file.

```
def func12(filename):
    k=0
    l=[]
    p="abcdefghijklmnopqrstuvwxyz"
    with open(filename,'r') as f1:
        f=f1.read()
        s=f.split()
        w=''
        for i in s:
            n=0
            p1=0
            if i[-1] in ',.-:':
                i=i[:-1]
            for j in s:
                if j[-1] in ',.-:':
                    if j[:-1].lower()==i.lower():
                        n+=1
                        s.remove(j)
                    pass
                if j.lower()==i.lower():
                    n+=1
                    s.remove(j)
            if n>k:
                k=n+1
                w=i
    return w,k
k2,k3=func12("text12.txt")
print("The most common word is",k2,"with",k3,"count.")
```

text12.txt

It was a good idea. At least, they all thought it was a good idea at the time. Hindsight would reveal that in reality, it was an unbelievably terrible idea, but it would take another week for them to understand that.

Right now, at this very moment. They all agreed that it was the perfect course of action for the current situation.

He wondered if he should disclose the truth to his friends. It would be a risky move.

Yes, the truth would make things a lot easier if they all stayed on the same page, but the truth might fracture the group leaving everything in even more of a mess than it was not telling the truth. It was time to decide which way to go. the.

Output:

The most common word is the with 11 count.

13. Write a program to read a text file line by line and display each word separated by '#'.


```
def func13(filename):
    with open(filename, 'r') as f1:
        f=f1.read()
        s=f.split('\n')
        for i in range(len(s)):
            n=s[i]
            j=n.split()
            for o in j:
                print("#%s"%o, end=' ')
            print(' ')
func13("text12.txt")
```

Output:

```
#It#was#a#good#idea.
#At#least,#they#all#thought#it#was#a#good#idea#at#the#time.#Hindsight
#would#reveal#that#in#reality,#it#was#an#unbelievably#terrible#idea,
#but#it#would#take#another#week#for#them#to#understand#that.
#Right#now,#at#this#very#moment.#They#all#agreed#that#it#was#the
#perfect#course#of#action#for#the#current#situation.
#He#wondered#if#he#should#disclose#the#truth#to#his#friends.#It#would#be#a#risky#move.
#Yes,#the#truth#would#make#things#a#lot#easier#if#they#all#stayed#on#
the#same#page,#but#the#truth#might#fracture#the#group#leaving
#everything#in#even#more#of#a#mess#than#it#was#not#telling#the#truth.
#It#was#time#to#decide#which#way#to#go.#the.
```

14. Create a binary file with the name and roll number of the student and display the data by reading the file.

```
import pickle

def createfile(rno:int,name):
    with open('data.bat','wb') as f:
        pickle.dump([rno,name],f)
createfile(1,'V R Darsh')

def readfile():
    with open('data.bat','rb') as f:
        k=pickle.load(f)
        print(k)
readfile()
```

Output:

```
[1, 'V R Darsh']
```

15. Write a program to update the name of a student by using its roll number in a binary file.

```
import pickle
def updatename(rno:int,newname):
```

```

l=[]
n=0
f=open('data.bat','rb')
while True:
    try:
        l.append(pickle.load(f))
    except EOFError:
        break
for i in l:
    if i[0]==rno:
        i[1]=newname
        n=1
        print("Successfully updated.")
if(n==0):
    print("Roll number not in file.")
    f.close()
else:
    f.close()
    f=open('data.bat','wb')
    for i in l:
        pickle.dump(i,f)
    f.close()
updatename(2,'Karthik')

```

Output:

Successfully updated from Akshay Kher to Karthik

16. Write a program to count the number of times the occurrence of 's', word in a text file.

```

def func16(filename):
    k=0
    l=[]
    p="abcdefghijklmnopqrstuvwxyz"
    with open(filename,'r') as f1:
        f=f1.read()
        for i in f:
            if i=='s':
                k+=1
    return k
k=func16("text12.txt")
print('There are %s \'s\' in this file.'%str(k))

```

Output:

There are 25 's' in this file.

17. Write a program to read data from a csv file and write data to a csv file.

```
import csv
```

```

def writefile(rno,name):
    with open("file1.csv",'a') as f:
        w=csv.writer(f)
        w.writerow([rno,name])
writefile(1,'V R Darsh')
writefile(2,'Akshay Kher')

def readfile():
    with open("file1.csv",'r') as f:
        w=csv.reader(f)
        for i in w:
            print(i)
readfile()

```

Output:

```
['1', 'V R Darsh'] ['2', 'Akshay Kher']
```

18. Write a program to count the number of vowels present in a text file.

```

def func18(filename):
    k=0
    with open(filename,'r') as f1:
        f=f1.read()
        for i in f:
            if i in 'aeiou':
                k+=1
    return k
k=func18("text12.txt")
print('There are %s vowels in this file.'%str(k))

```

Output:

There are 196 vowels in this file.

19. Write a program to search a record using its roll number and display the name of the student in a binary file. If the record is not found then display appropriate message.

```

import pickle
def findname(rno:int):
    l=[]
    n=0
    f=open('data.bat','rb')
    while True:
        try:
            l.append(pickle.load(f))
        except EOFError:
            break
    for i in l:
        if i[0]==rno:
            print(i[0],":",i[1])
            n=1

```

```

        if (n==0):
            print("Roll number not in file.")
        f.close()
findname(1)

```

Output:

1 : V R Darsh

20. Write a program to delete a record from a binary file.

```

import pickle
def deletename(rno:int):
    l=[]
    n=0
    f=open('data.bat','rb')
    while True:
        try:
            l.append(pickle.load(f))
        except EOFError:
            break
    for i in l:
        if i[0]==rno:
            l.remove(i)
            print("Successfully removed")
    if (n==0):
        print("Roll number not in file.")
        f.close()
    else:
        f.close()
        f=open('data.bat','wb')
        for i in l:
            pickle.dump(i,f)
        f.close()
deletename(2)

```

Output:

Successfully removed the record [2, 'Karthik']

21. Write a program to write those lines which have the character 'p', from one text file to another text file.

```

def func21(filename):
    k=0
    l=[]
    with open(filename,'r') as f1:
        f=f1.readlines()
        for i in f:
            print(i)
            for j in i:
                print(j)

```

```

        if j=='p':
            k=1
        if k==1:
            l.append(i)
    print(l)
    with open("output21.txt",'w+') as f1:
        for i in l:
            f1.write(i)
func21("text21.txt")

```

text21.txt

what is your name
 What is this file
 My name is Darsh.
 This is my practical file.
 python is a high level language.

Output File:

This is my practical file.
 python is a high level language.

22. Write a program to count the number of words in a file.

```

def func22(filename):
    k=0
    with open(filename,'r') as f1:
        f=f1.read()
        for i in f:
            k+=1
    return k
print(func22("text22.txt"))

```

text22.txt:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quisque consequat placerat erat eu commodo. Nam congue maximus purus, in blandit ligula viverra eu. Mauris euismod mollis elit, ut semper nisi pellentesque a. Cras lacinia quis libero vitae fringilla. Mauris aliquet dolor vel feugiat gravida. Sed ut aliquam leo. Duis vitae arcu eros. Cras sed justo sed ex volutpat facilisis eu non purus. Praesent pharetra dolor arcu, vitae dictum est facilisis et. Sed ullamcorper urna quis urna convallis, vel vestibulum erat molestie. Sed ac augue consectetur, suscipit augue eget, pharetra felis. Ut ex neque, consequat eu pharetra nec, fermentum in felis. Phasellus suscipit vulputate lacus, nec ornare nisi.

Output:

705