;

LAB ASSIGNMENT
DATABASE LAB
DUE DATE: 04/04/2022
TKM COLLEGE OF ENGINEERING

GROUP 6

1. Consider the database for a banking enterprise. Write the queries for the below questions.

   (i) Create the followingtables

Table	Attributes
Customer	cid,cname,loc,sex,dob
Bank_brn	bcode,bloc,bsate
Deposit	Dacno,dtype,ddate,damt
Loan	Lacno,ltype,ldate,lamt
Accounts_in	Bcode,cid
Depositor	cid,dacno
Borrower	cid,lacno

   (ii) Include necessary constraints.
   (iii) Tables are created under the database 'bank'
   (iv) Display all the tables in bank database
   (v) Describe the structure of all tables
   (vi) Delete tables

Script:

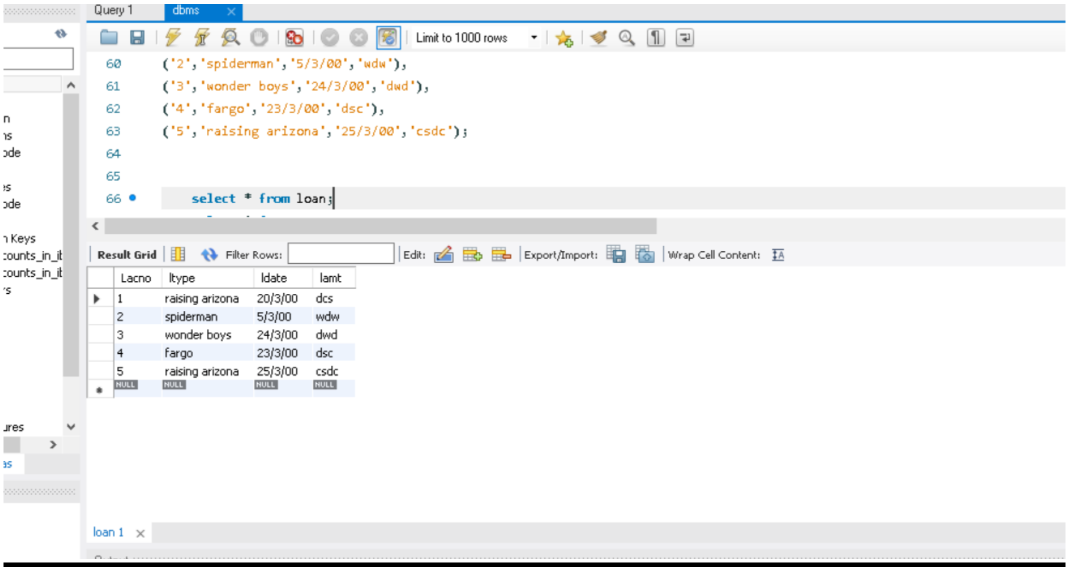
```
create database bank;
use bank;
CREATE TABLE Customer(
   cid varchar(255)NOT NULL,
   cname varchar(255)NOT NULL,
   loc varchar(255)NOT NULL,
   sex varchar(255)NOT NULL,
   dob varchar(255)NOT NULL,
   primary key(cid)

   );
     INSERT INTO Customer(cid,cname,loc,sex,dob)
```

```
VALUES ('1','raising arizona','sgh','f','20/3/00'),
('2','spiderman','kkz','m','5/3/00'),
('3','wonder boys','fds','f','24/3/00'),
('4','fargo','64w','m','23/3/00'),
('5','raising arizona','uy','f','25/3/00'),
('6','wonder boys','utds','m','16/3/00');

CREATE TABLE Bank_brn(
   bcode varchar(255)NOT NULL,
    bloc varchar(255)NOT NULL,
    bsate varchar(255)NOT NULL,

    primary key(bcode)

    );
      INSERT INTO Bank_brn(bcode,bloc,bsate)
VALUES ('5','raising arizona','sgh'),
('11','spiderman','kkz'),
('10','wonder boys','fds'),
('9','fargo','64w'),
('6','raising arizona','bgf');

CREATE TABLE Deposit(
   Dacno varchar(255)NOT NULL,
   dtype  varchar(255)NOT NULL,
   ddate varchar(255)NOT NULL,
   damt varchar(255)NOT NULL,
    primary key(Dacno)

    );
      INSERT INTO Deposit(Dacno,dtype,ddate,damt)
VALUES ('1','raising arizona','20/3/00','dcs'),
('2','spiderman','5/3/00','wdw'),
('3','wonder boys','24/3/00','dwd'),
('4','fargo','23/3/00','dsc'),
('5','raising arizona','25/3/00','csdc');

CREATE TABLE Loan(
   Lacno varchar(255)NOT NULL,
   ltype varchar(255)NOT NULL,
   ldate varchar(255)NOT NULL,
   lamt varchar(255)NOT NULL,
    primary key(Lacno)

    );
```

```
      INSERT INTO Loan(Lacno,ltype,ldate,lamt)
VALUES ('1','raising arizona','20/3/00','dcs'),
('2','spiderman','5/3/00','wdw'),
('3','wonder boys','24/3/00','dwd'),
('4','fargo','23/3/00','dsc'),
('5','raising arizona','25/3/00','csdc');



CREATE TABLE Accounts_in(
    Bcode varchar(255)NOT NULL,
    cid varchar(255)NOT NULL,
    FOREIGN KEY (cid) REFERENCES Customer(cid),
     FOREIGN KEY (Bcode) REFERENCES Bank_brn(Bcode)

    );


CREATE TABLE Depositor(
    cid varchar(255)NOT NULL,
    dacno varchar(255)NOT NULL,
   FOREIGN KEY (cid) REFERENCES Customer(cid),
     FOREIGN KEY (dacno) REFERENCES Deposit(dacno)
    );

    CREATE TABLE Borrower(
    cid varchar(255)NOT NULL,
    lacno varchar(255)NOT NULL,
   FOREIGN KEY (cid) REFERENCES Customer(cid),
     FOREIGN KEY (lacno) REFERENCES Loan(lacno)
    );

    select * from loan;
    select * from Customer;
    select * from Bank_brn;
    select * from Deposit;

SHOW TABLES;
drop  table Borrower;
```

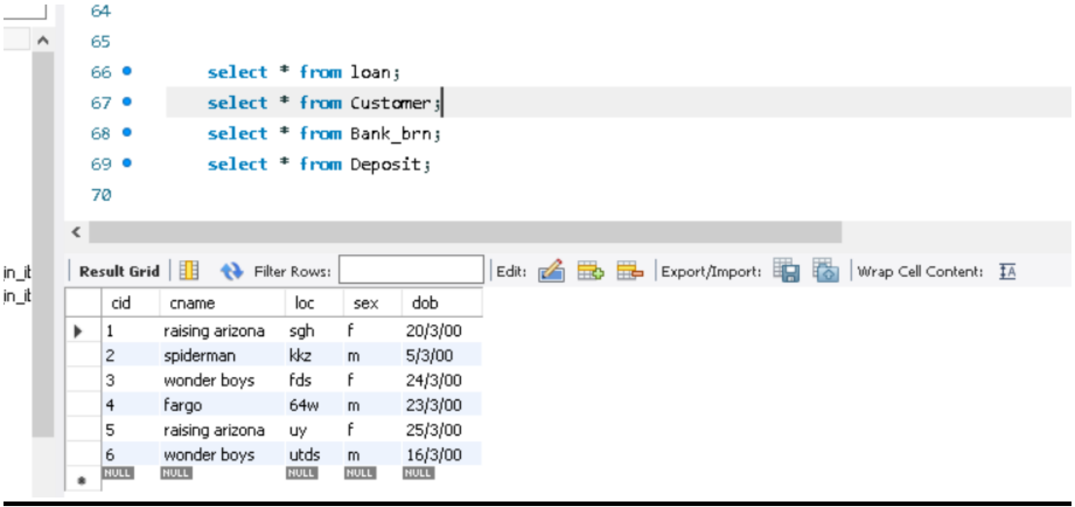
Result: output is obtained successfully.

output:



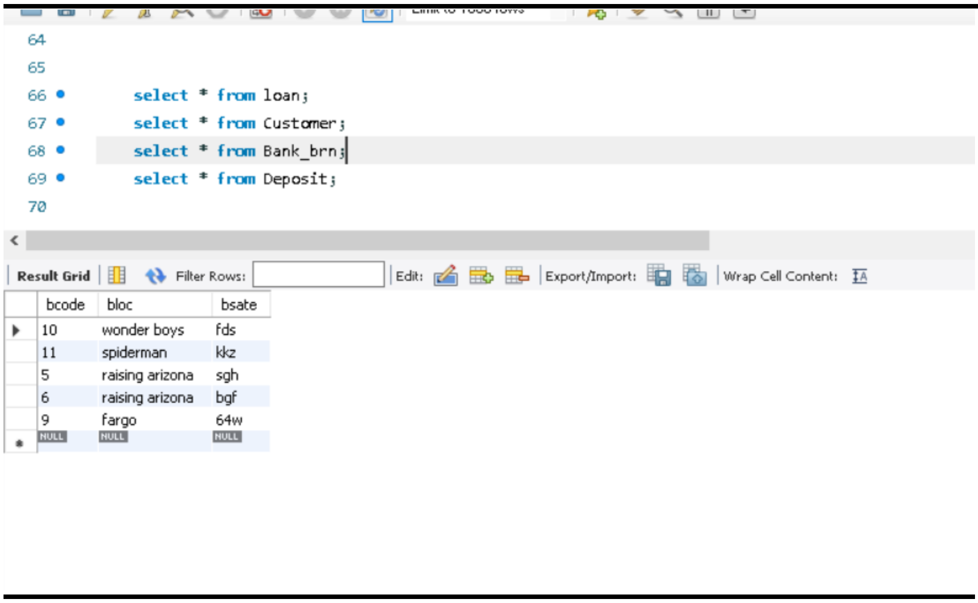
```
('2','spiderman','5/3/00','wdw'),
('3','wonder boys','24/3/00','dwd'),
('4','fargo','23/3/00','dsc'),
('5','raising arizona','25/3/00','csdc');



    select * from loan;
```

Lacno	ltype	ldate	lamt
1	raising arizona	20/3/00	dcs
2	spiderman	5/3/00	wdw
3	wonder boys	24/3/00	dwd
4	fargo	23/3/00	dsc
5	raising arizona	25/3/00	csdc
NULL	NULL	NULL	NULL



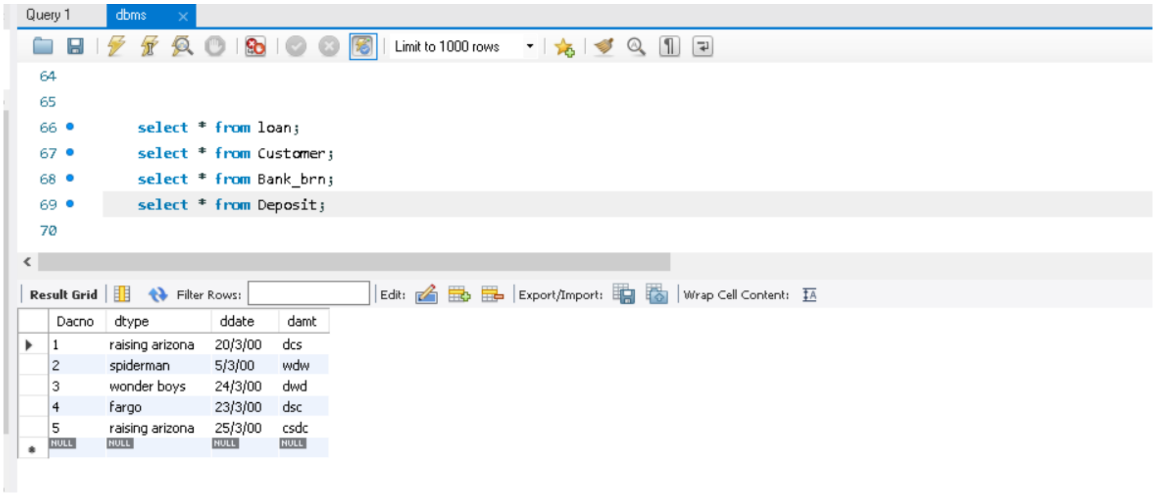
```


    select * from loan;
    select * from Customer;
    select * from Bank_brn;
    select * from Deposit;

```

cid	cname	loc	sex	dob
1	raising arizona	sgh	f	20/3/00
2	spiderman	kkz	m	5/3/00
3	wonder boys	fds	f	24/3/00
4	fargo	64w	m	23/3/00
5	raising arizona	uy	f	25/3/00
6	wonder boys	utds	m	16/3/00
NULL	NULL	NULL	NULL	NULL



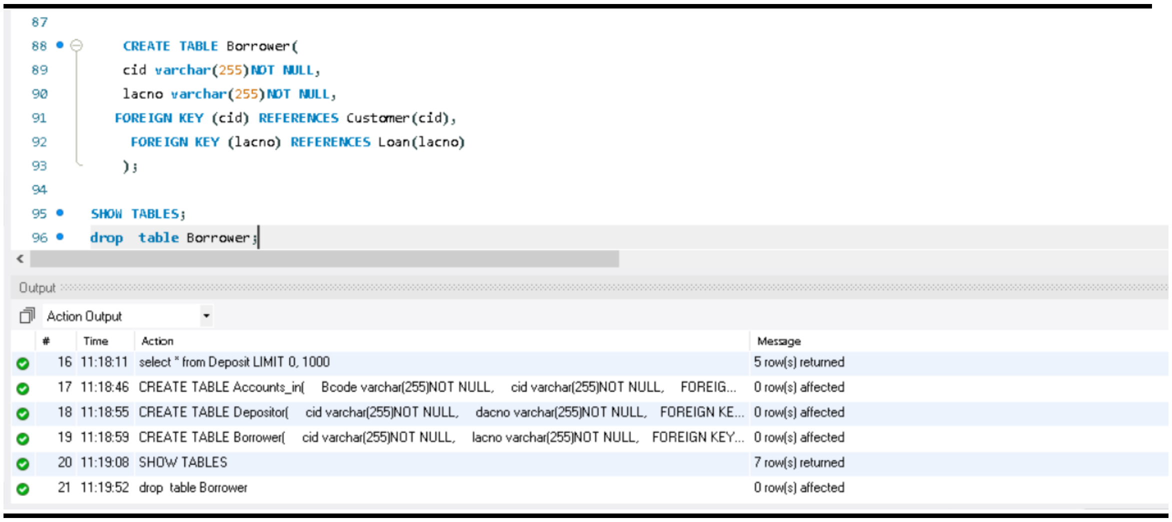
```


    select * from loan;
    select * from Customer;
    select * from Bank_brn;
    select * from Deposit;

```

bcode	bloc	bsate
10	wonder boys	fds
11	spiderman	kkz
5	raising arizona	sgh
6	raising arizona	bgf
9	fargo	64w
NULL	NULL	NULL



```sql
select * from loan;
select * from Customer;
select * from Bank_brn;
select * from Deposit;
```

Dacno	dtype	ddate	damt
1	raising arizona	20/3/00	dcs
2	spiderman	5/3/00	wdw
3	wonder boys	24/3/00	dwd
4	fargo	23/3/00	dsc
5	raising arizona	25/3/00	csdc
NULL	NULL	NULL	NULL



```sql
CREATE TABLE Borrower(
cid varchar(255)NOT NULL,
lacno varchar(255)NOT NULL,
FOREIGN KEY (cid) REFERENCES Customer(cid),
 FOREIGN KEY (lacno) REFERENCES Loan(lacno)
);

SHOW TABLES;
drop table Borrower;
```

```
#   Time      Action                                                                                              Message
16  11:18:11  select * from Deposit LIMIT 0, 1000                                                                 5 row(s) returned
17  11:18:46  CREATE TABLE Accounts_in(   Bcode varchar(255)NOT NULL,   cid varchar(255)NOT NULL,   FOREIG...     0 row(s) affected
18  11:18:55  CREATE TABLE Depositor(   cid varchar(255)NOT NULL,   dacno varchar(255)NOT NULL,   FOREIGN KE...   0 row(s) affected
19  11:18:59  CREATE TABLE Borrower(   cid varchar(255)NOT NULL,   lacno varchar(255)NOT NULL,   FOREIGN KEY...   0 row(s) affected
20  11:19:08  SHOW TABLES                                                                                         7 row(s) returned
21  11:19:52  drop  table Borrower                                                                                0 row(s) affected
```

2.Consider the database for a college. Write the query for the following.

- Insert at least 5 tuples into each table.
- List the details of students in the ascending order of date of birth
- Display the details of students from computer department
- List the faculties in the descending order of salary
- Display the total number of students in each department
- Display the total number of faculties in each department with salary greater than 25000

<u>Script:</u>

create database co;

use co;

create table student1(

```
rollno int,
name varchar(20),
dept varchar(20),
dob date
);
INSERT INTO student1(rollno,name,dob,dept) values('101','Amit','2000-03-12','computer');
INSERT INTO student1(rollno,name,dob,dept) values('102','Suji','2000-08-02','computer');
INSERT INTO student1(rollno,name,dob,dept) values('103','Anu','1999-12-5','engineering');
INSERT INTO student1(rollno,name,dob,dept) values('104','Karthi','1998-09-24','maths');
INSERT INTO student1(rollno,name,dob,dept) values('105','Kevin','2001-06-30','engineering');

create table faculty(
f_id int,
name varchar(20),
dept varchar(20),
salary int
);
INSERT INTO faculty(f_id,name,dept,salary) values('121','Arya','computer',20000);
INSERT INTO  faculty(f_id,name,dept,salary) values('122','Anoop','computer',15000);
INSERT INTO  faculty(f_id,name,dept,salary) values('123','Manju','engineering',45000);
INSERT INTO  faculty(f_id,name,dept,salary) values('124','Shiji','maths',20000);
INSERT INTO  faculty(f_id,name,dept,salary) values('125','Sabitha','engineering',35000);



select * from student1 order by dob asc;
```

select * from student1 where dept='computer';

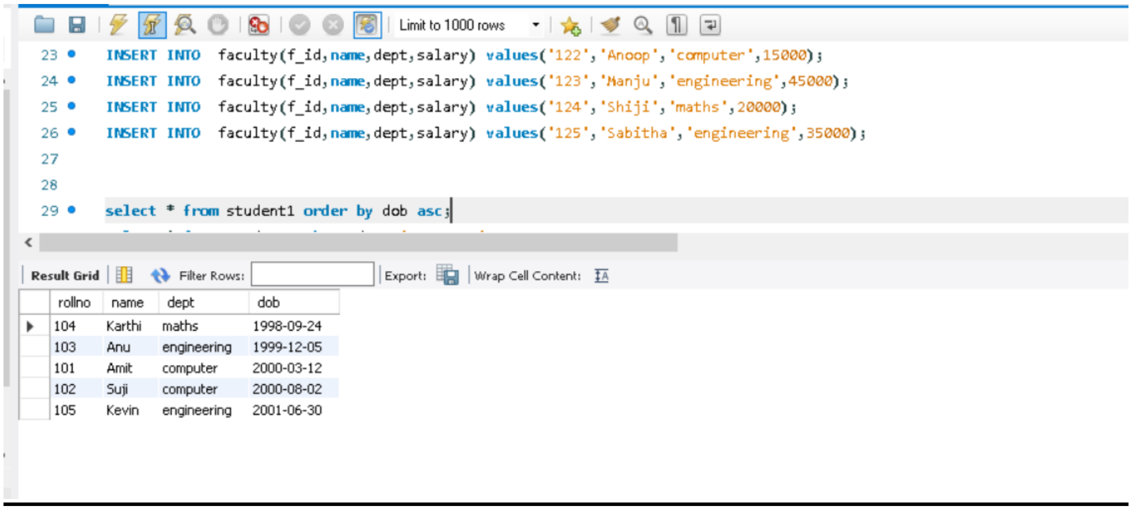
select * from faculty order by salary desc;

select count(rollno),dept from student1 group by dept;

select count(f_id),dept,salary from faculty where salary>25000;

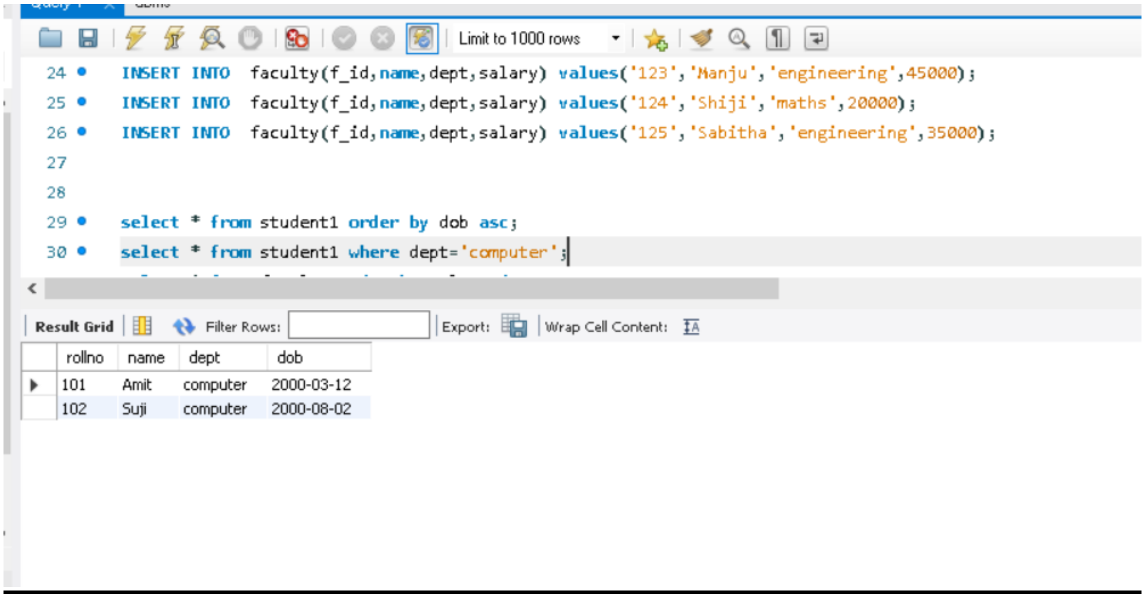
Result: output is obtained successfully.

output:



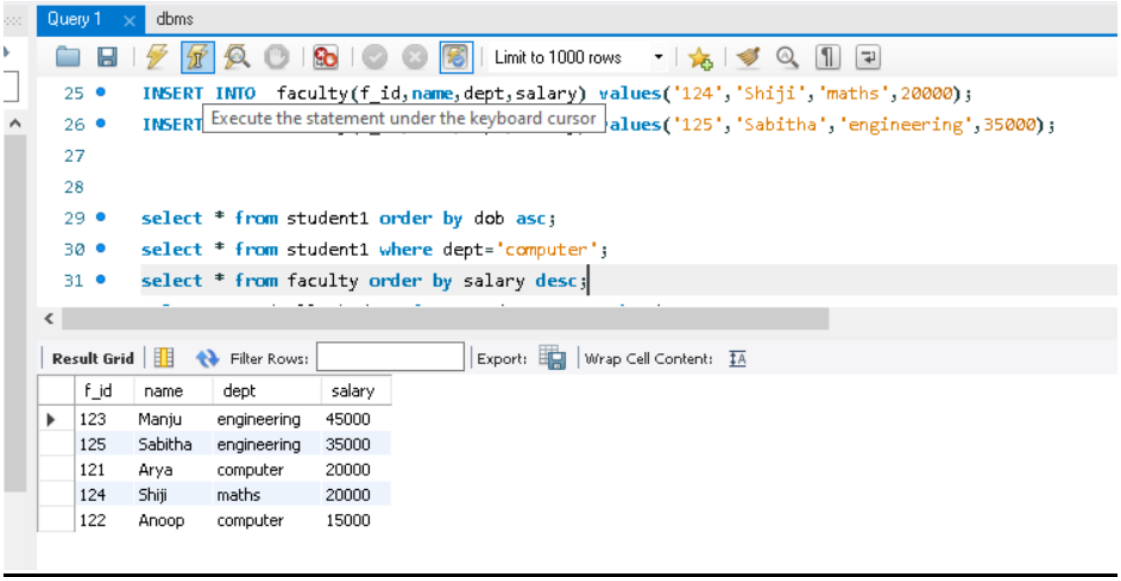
```
INSERT INTO  faculty(f_id,name,dept,salary) values('122','Anoop','computer',15000);
INSERT INTO  faculty(f_id,name,dept,salary) values('123','Manju','engineering',45000);
INSERT INTO  faculty(f_id,name,dept,salary) values('124','Shiji','maths',20000);
INSERT INTO  faculty(f_id,name,dept,salary) values('125','Sabitha','engineering',35000);


select * from student1 order by dob asc;
```

rollno	name	dept	dob
104	Karthi	maths	1998-09-24
103	Anu	engineering	1999-12-05
101	Amit	computer	2000-03-12
102	Suji	computer	2000-08-02
105	Kevin	engineering	2001-06-30



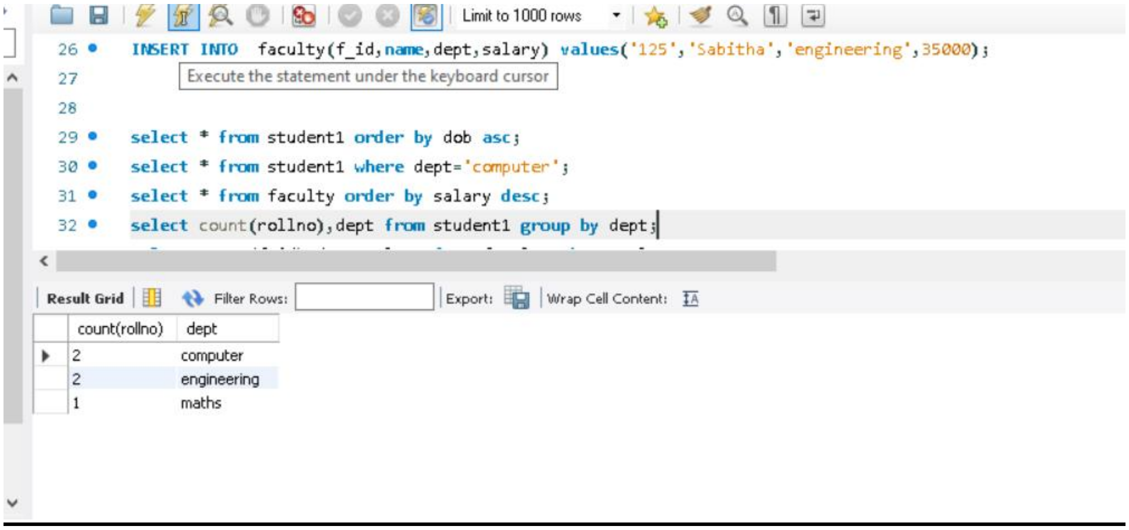
```
INSERT INTO  faculty(f_id,name,dept,salary) values('123','Manju','engineering',45000);
INSERT INTO  faculty(f_id,name,dept,salary) values('124','Shiji','maths',20000);
INSERT INTO  faculty(f_id,name,dept,salary) values('125','Sabitha','engineering',35000);


select * from student1 order by dob asc;
select * from student1 where dept='computer';
```

rollno	name	dept	dob
101	Amit	computer	2000-03-12
102	Suji	computer	2000-08-02



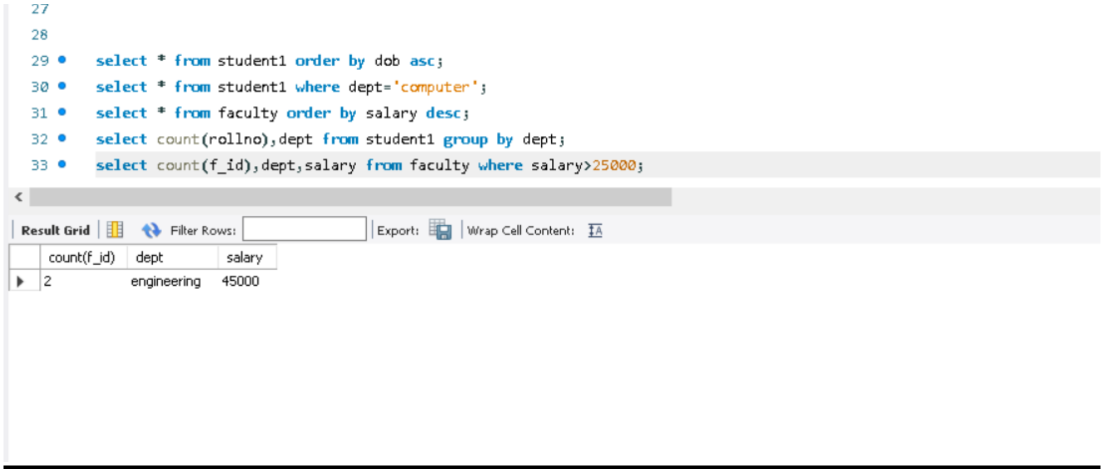
```
INSERT INTO  faculty(f_id,name,dept,salary) values('124','Shiji','maths',20000);
INSERT                                            alues('125','Sabitha','engineering',35000);


select * from student1 order by dob asc;
select * from student1 where dept='computer';
select * from faculty order by salary desc;
```

f_id	name	dept	salary
123	Manju	engineering	45000
125	Sabitha	engineering	35000
121	Arya	computer	20000
124	Shiji	maths	20000
122	Anoop	computer	15000



```
INSERT INTO  faculty(f_id,name,dept,salary) values('125','Sabitha','engineering',35000);


select * from student1 order by dob asc;
select * from student1 where dept='computer';
select * from faculty order by salary desc;
select count(rollno),dept from student1 group by dept;
```

count(rollno)	dept
2	computer
2	engineering
1	maths



```


select * from student1 order by dob asc;
select * from student1 where dept='computer';
select * from faculty order by salary desc;
select count(rollno),dept from student1 group by dept;
select count(f_id),dept,salary from faculty where salary>25000;
```

count(f_id)	dept	salary
2	engineering	45000

MANDATORY ASSIGNMENT FOR ALL GROUPS:

Queries using aggregate functions(COUNT,AVG,MIN,MAX,SUM),Group by,Order by,Having.

E_ID	E_NAME	AGE	SALARY
101	ANU	22	9000
102	Shane	29	8000
103	Rohan	34	6000
104	Scott	44	10000
105	Tiger	35	8000
106	Alex	27	7000
107	Abhi	29	8000

(i) Create Employee table containing all Records.

(ii) Count number of employee names from employee table.

(iii) Find the Maximum age from employee table

(iv) Find the Minimum age from employee table.

(v) Display the Sum of age employee table.

(vi) Display the Average of age from Employee table

(vii) Create a View for age in employee table

(viii) Display views

(ix) Find grouped salaries of employees.

(x) Find salaries of employee in Ascending Order

(xi) Find salaries of employee in Descending Order

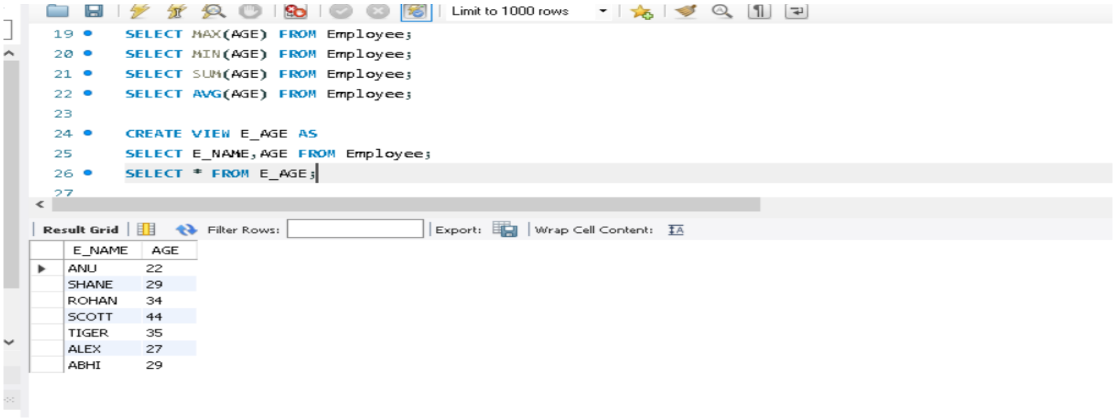
Script:

```
CREATE DATABASE EMPLOYEE;

USE EMPLOYEE;

CREATE TABLE Employee(

E_ID INT(25),

E_NAME VARCHAR(25),

AGE INT(10),

SALARY FLOAT(10),

PRIMARY KEY(E_ID)

);
```

```
INSERT INTO Employee VALUES (101,"ANU",22,9000);

INSERT INTO Employee VALUES (102,"SHANE",29,8000);

INSERT INTO Employee VALUES (103,"ROHAN",34,6000);

INSERT INTO Employee VALUES (104,"SCOTT",44,10000);

INSERT INTO Employee VALUES (105,"TIGER",35,8000);

INSERT INTO Employee VALUES (106,"ALEX",27,7000);

INSERT INTO Employee VALUES (107,"ABHI",29,8000);

SELECT * FROM Employee ;

SELECT COUNT(*) FROM Employee;

SELECT MAX(AGE) FROM Employee;

SELECT MIN(AGE) FROM Employee;

SELECT SUM(AGE) FROM Employee;

SELECT AVG(AGE) FROM Employee;


CREATE VIEW E_AGE AS

SELECT E_NAME,AGE FROM Employee;

SELECT * FROM E_AGE;


SELECT E_NAME,SALARY FROM Employee GROUP BY E_NAME;


SELECT SALARY FROM Employee ORDER BY SALARY ASC;


SELECT SALARY FROM Employee ORDER BY SALARY DESC;
```

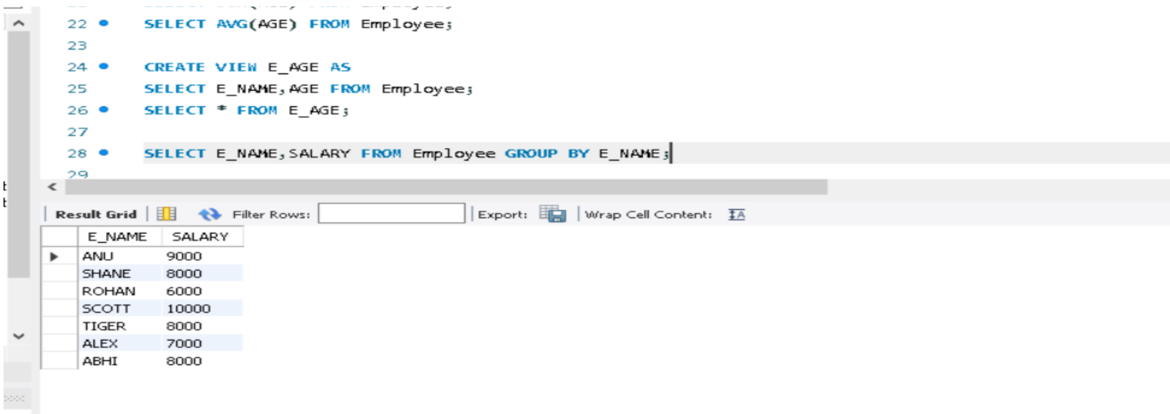
Result: output is obtained successfully.

output:



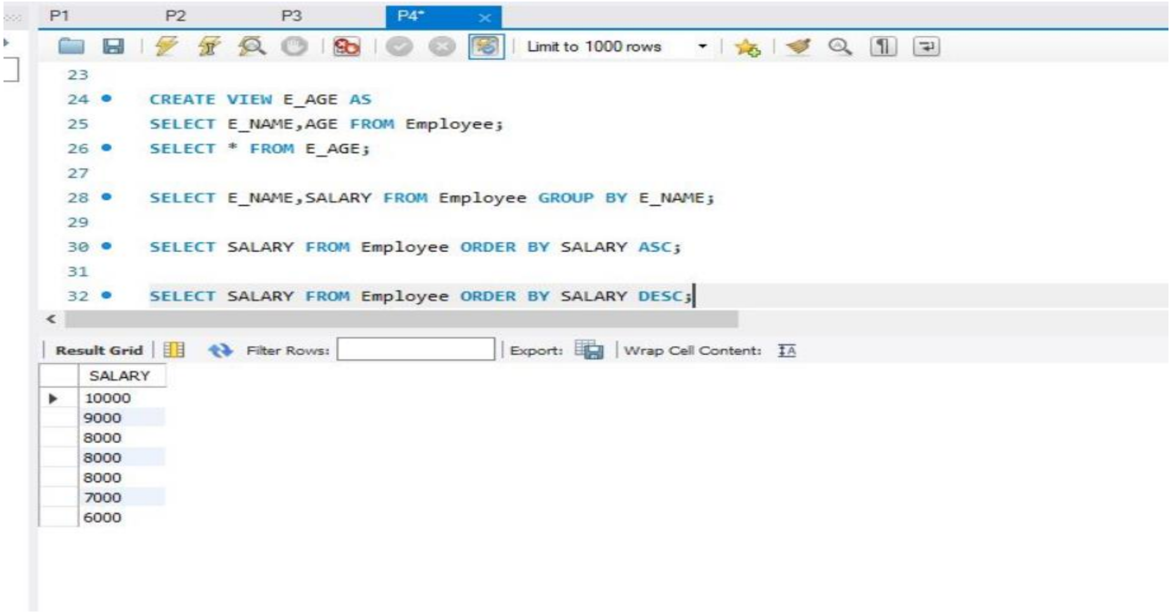
```
SELECT MAX(AGE) FROM Employee;
SELECT MIN(AGE) FROM Employee;
SELECT SUM(AGE) FROM Employee;
SELECT AVG(AGE) FROM Employee;

CREATE VIEW E_AGE AS
SELECT E_NAME,AGE FROM Employee;
SELECT * FROM E_AGE;
```

E_NAME	AGE
ANU	22
SHANE	29
ROHAN	34
SCOTT	44
TIGER	35
ALEX	27
ABHI	29



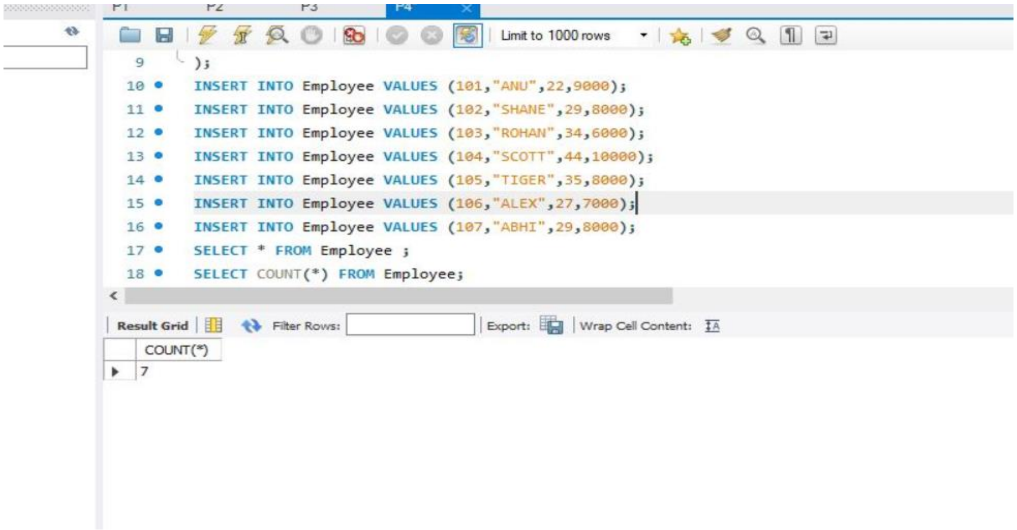
```
SELECT AVG(AGE) FROM Employee;

CREATE VIEW E_AGE AS
SELECT E_NAME,AGE FROM Employee;
SELECT * FROM E_AGE;

SELECT E_NAME,SALARY FROM Employee GROUP BY E_NAME;
```

E_NAME	SALARY
ANU	9000
SHANE	8000
ROHAN	6000
SCOTT	10000
TIGER	8000
ALEX	7000
ABHI	8000



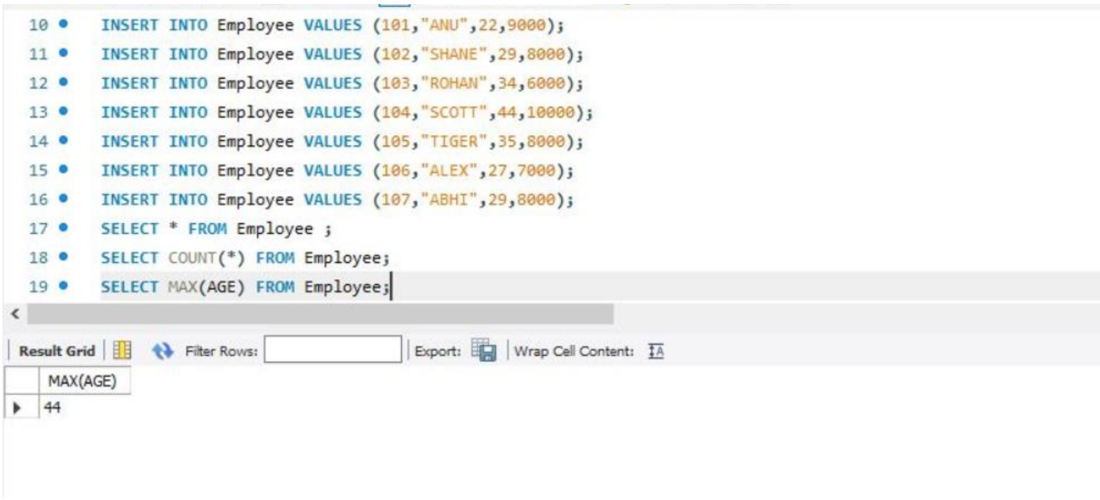
```
CREATE VIEW E_AGE AS
SELECT E_NAME,AGE FROM Employee;
SELECT * FROM E_AGE;

SELECT E_NAME,SALARY FROM Employee GROUP BY E_NAME;

SELECT SALARY FROM Employee ORDER BY SALARY ASC;

SELECT SALARY FROM Employee ORDER BY SALARY DESC;
```

SALARY
10000
9000
8000
8000
8000
7000
6000



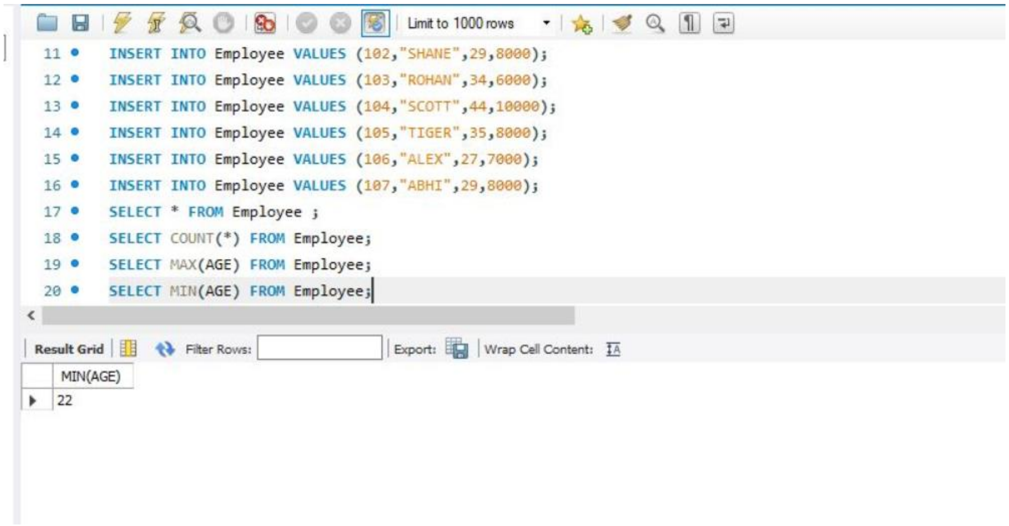
```
);
INSERT INTO Employee VALUES (101,"ANU",22,9000);
INSERT INTO Employee VALUES (102,"SHANE",29,8000);
INSERT INTO Employee VALUES (103,"ROHAN",34,6000);
INSERT INTO Employee VALUES (104,"SCOTT",44,10000);
INSERT INTO Employee VALUES (105,"TIGER",35,8000);
INSERT INTO Employee VALUES (106,"ALEX",27,7000);
INSERT INTO Employee VALUES (107,"ABHI",29,8000);
SELECT * FROM Employee ;
SELECT COUNT(*) FROM Employee;
```

```
COUNT(*)
7
```



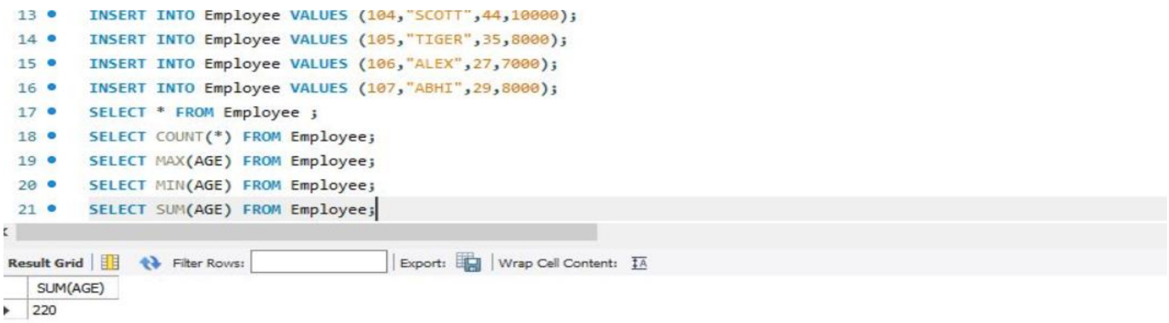
```
INSERT INTO Employee VALUES (101,"ANU",22,9000);
INSERT INTO Employee VALUES (102,"SHANE",29,8000);
INSERT INTO Employee VALUES (103,"ROHAN",34,6000);
INSERT INTO Employee VALUES (104,"SCOTT",44,10000);
INSERT INTO Employee VALUES (105,"TIGER",35,8000);
INSERT INTO Employee VALUES (106,"ALEX",27,7000);
INSERT INTO Employee VALUES (107,"ABHI",29,8000);
SELECT * FROM Employee ;
SELECT COUNT(*) FROM Employee;
SELECT MAX(AGE) FROM Employee;
```

```
MAX(AGE)
44
```



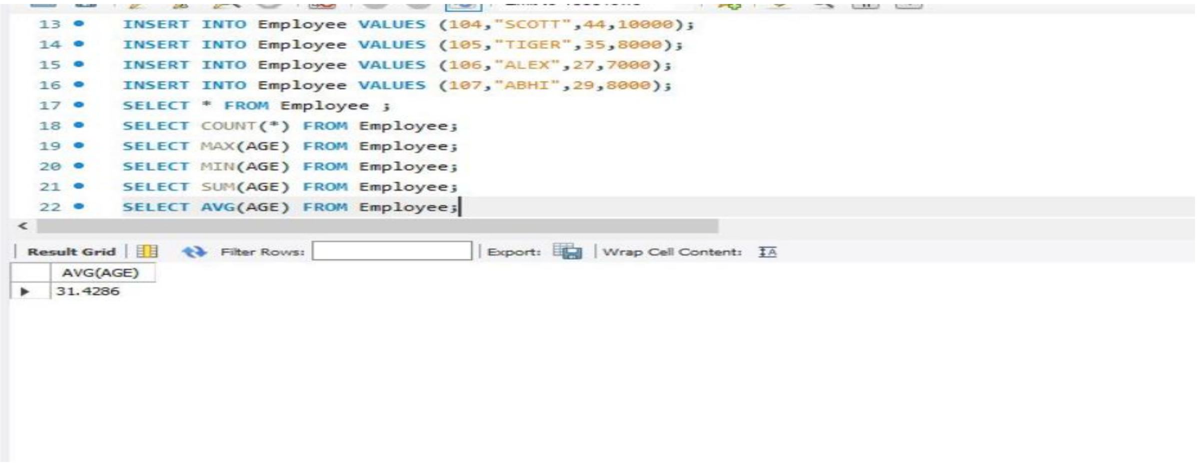
```
INSERT INTO Employee VALUES (102,"SHANE",29,8000);
INSERT INTO Employee VALUES (103,"ROHAN",34,6000);
INSERT INTO Employee VALUES (104,"SCOTT",44,10000);
INSERT INTO Employee VALUES (105,"TIGER",35,8000);
INSERT INTO Employee VALUES (106,"ALEX",27,7000);
INSERT INTO Employee VALUES (107,"ABHI",29,8000);
SELECT * FROM Employee ;
SELECT COUNT(*) FROM Employee;
SELECT MAX(AGE) FROM Employee;
SELECT MIN(AGE) FROM Employee;
```

```
MIN(AGE)
22
```



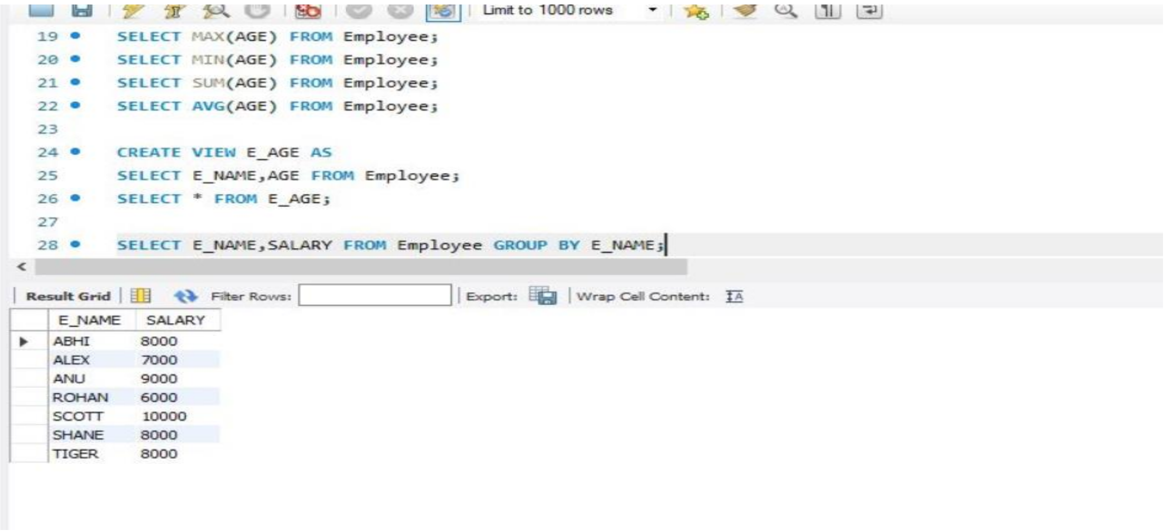
```sql
INSERT INTO Employee VALUES (104,"SCOTT",44,10000);
INSERT INTO Employee VALUES (105,"TIGER",35,8000);
INSERT INTO Employee VALUES (106,"ALEX",27,7000);
INSERT INTO Employee VALUES (107,"ABHI",29,8000);
SELECT * FROM Employee ;
SELECT COUNT(*) FROM Employee;
SELECT MAX(AGE) FROM Employee;
SELECT MIN(AGE) FROM Employee;
SELECT SUM(AGE) FROM Employee;
```

```
SUM(AGE)
220
```



```sql
INSERT INTO Employee VALUES (104,"SCOTT",44,10000);
INSERT INTO Employee VALUES (105,"TIGER",35,8000);
INSERT INTO Employee VALUES (106,"ALEX",27,7000);
INSERT INTO Employee VALUES (107,"ABHI",29,8000);
SELECT * FROM Employee ;
SELECT COUNT(*) FROM Employee;
SELECT MAX(AGE) FROM Employee;
SELECT MIN(AGE) FROM Employee;
SELECT SUM(AGE) FROM Employee;
SELECT AVG(AGE) FROM Employee;
```

```
AVG(AGE)
31.4286
```



```sql
SELECT MAX(AGE) FROM Employee;
SELECT MIN(AGE) FROM Employee;
SELECT SUM(AGE) FROM Employee;
SELECT AVG(AGE) FROM Employee;

CREATE VIEW E_AGE AS
SELECT E_NAME,AGE FROM Employee;
SELECT * FROM E_AGE;

SELECT E_NAME,SALARY FROM Employee GROUP BY E_NAME;
```

```
E_NAME   SALARY
ABHI     8000
ALEX     7000
ANU      9000
ROHAN    6000
SCOTT    10000
SHANE    8000
TIGER    8000
```