ADBMS LAB
LAB CYCLE-1

SUBMITTED BY:
MOHAMMED SHIBILI O
ROLL NO:228
MCA S2

PROGRAM NO 1:

Consider the database movie

Movies

title	director	myear	rating
Fargo	Coen	1996	8.2
Raising Arizona	Coen	1987	7.6
Spiderman	Raimi	2002	7.4
Wonder Boys	Hanson	2000	7.6

Actors

actor	ayear
Cage	1964
Hanks	1956
Maguire	1975
McDormand	1957

Acts

actor	title
Cage	Raising Arizona
Maguire	Spiderman
Maguire	Wonder Boys
McDormand	Fargo
McDormand	Raising Arizona
McDormand	Wonder Boys

Directors

director	dyear
Coen	1954
Hanson	1945
Raimi	1959

Write following relational algebra queries for a given set of relations.

1. Find movies made after 1997
2. Find movies made by Hanson after 1997

3.Find all movies and their ratings

4.Find all actors and directors

5.Find Coen's movies with McDormannd

CODE:

use movies;

create table movie(title varchar(20) not null primary key,director varchar(20) not null,myear int not null,rating float4 not null);

INSERT INTO movie(title,director,myear,rating)VALUES('fargo','coen',1996,8.2);

INSERT INTO movie(title,director,myear,rating)VALUES('Raising Arizona','coen',1987,7.6);

```
INSERT INTO
movie(title,director,myear,rating)VALUES('Spiderman','Raimi',2002,7.4
);

INSERT INTO movie(title,director,myear,rating)VALUES('Wonder
Boys','Hanson',2000,7.6);

show databases;

use movies;

create table actors(actor varchar(20) not null primary key,ayear int not
null );

INSERT INTO actors(actor,ayear)values('cage',1964);

INSERT INTO actors(actor,ayear)values('hanks',1956);

INSERT INTO actors(actor,ayear)values('maguire',1975);

INSERT INTO actors(actor,ayear)values('mcdormand',1957);

use movies;

create table directors(director varchar(20) not null primary key,dyear int
not null);

INSERT INTO directors(director,dyear)values('coen',1954);

INSERT INTO directors(director,dyear)values('Hanson',1945);

INSERT INTO directors(director,dyear)values('Raimi',1959);

use movies;

create table  acts(actor  varchar(20) not null,title varchar(20) not
null,foreign key(title) references movie(title),foreign key(actor)
references actors(actor));

INSERT INTO acts(actor,title)values('cage','Raising Arizona');

INSERT INTO acts(actor,title)values('maguire','Spiderman');
```

INSERT INTO acts(actor,title)values('maguire','Wonder Boys');

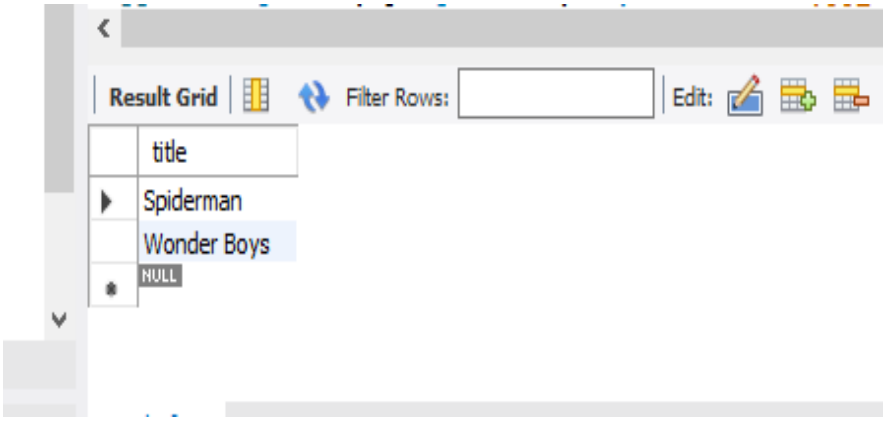
INSERT INTO acts(actor,title)values('mcdormand','fargo');

INSERT INTO acts(actor,title)values('mcdormand','Raising Arizona');

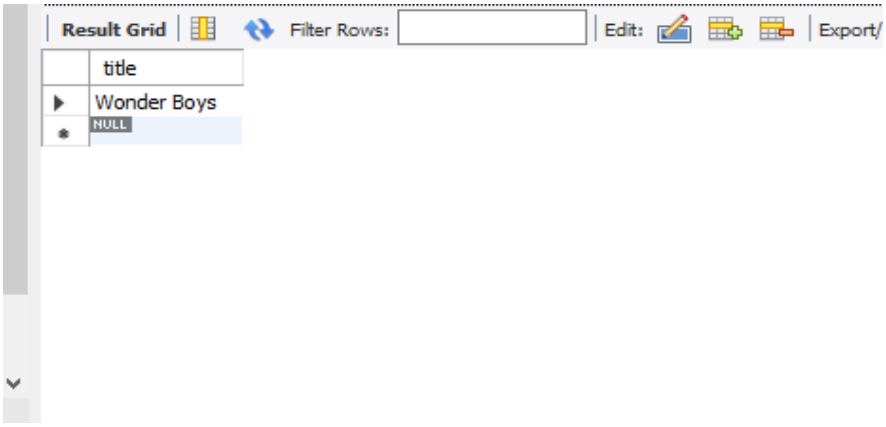
INSERT INTO acts(actor,title)values('mcdormand','Wonder Boys');

QUERY:

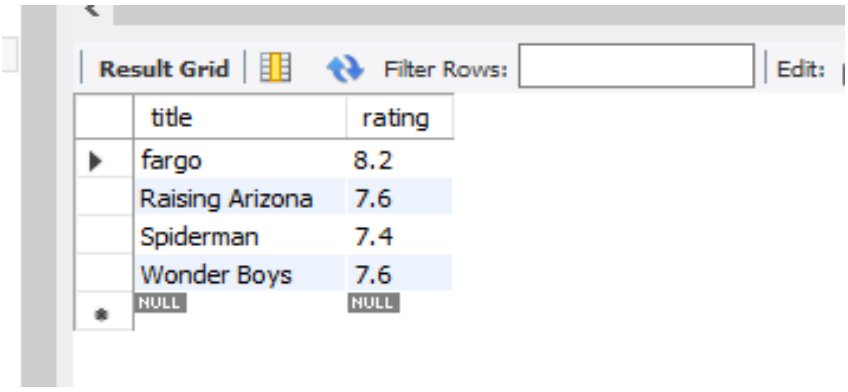
select title from movie where myear>=1997;



select title from movie where myear>=1997 and director='Hanson';

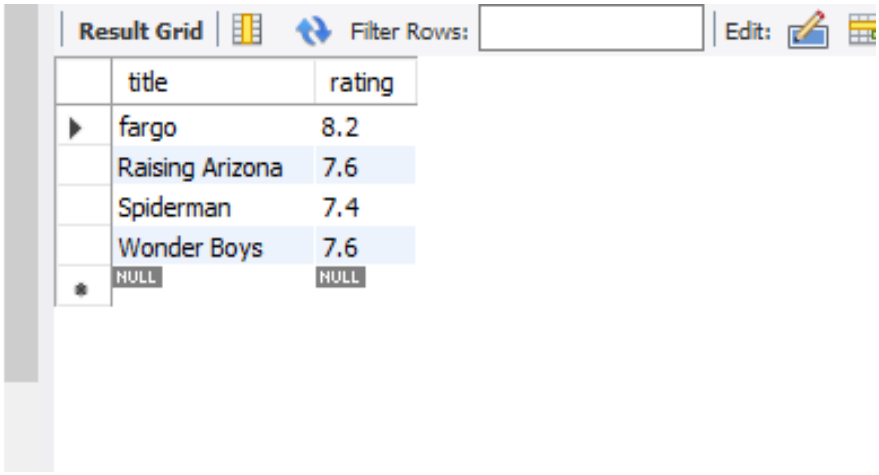


select title,rating from movie;



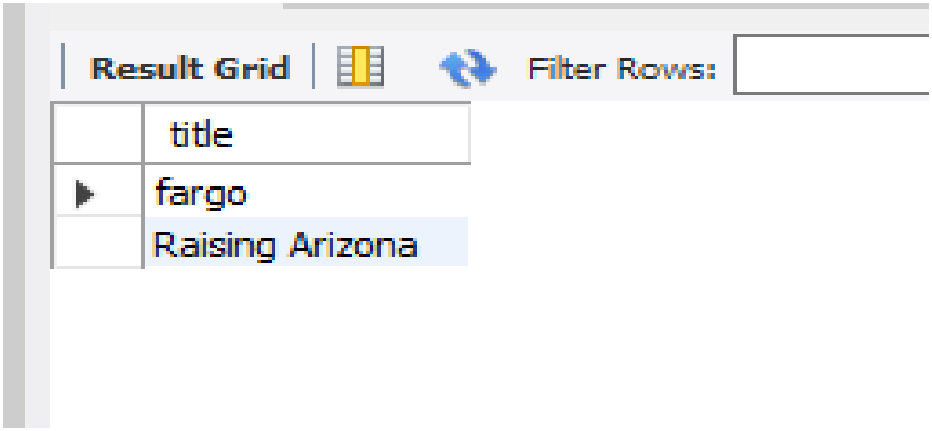
title	rating
fargo	8.2
Raising Arizona	7.6
Spiderman	7.4
Wonder Boys	7.6
NULL	NULL

select movie.director,acts.actor from movie inner join acts on movie.title=acts.title;



title	rating
fargo	8.2
Raising Arizona	7.6
Spiderman	7.4
Wonder Boys	7.6
NULL	NULL

select movie.title from movie left outer join acts on movie.title=acts.title where director='coen' and actor='mcdormand';



title
fargo
Raising Arizona

RESULT: Output obtained successfully.

PROGRAM NO 2:

Consider Dept table

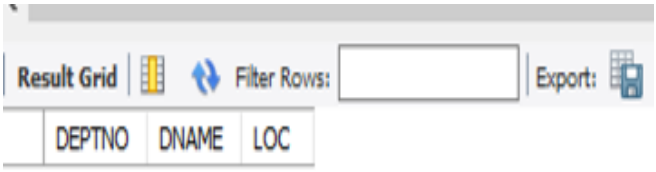
DEPTNO	DNAME	LOC

Perform the following:

1. Rename the table dept as department
2. Add a new column PINCODE with not null constraints to the existing table DEPT
3. All constraints and views that reference the column are dropped automatically, along with the column.
4. Rename the column DNAME to DEPT_NAME in dept table
5. Change the data type of column loc as CHAR with size 10
6. Delete table

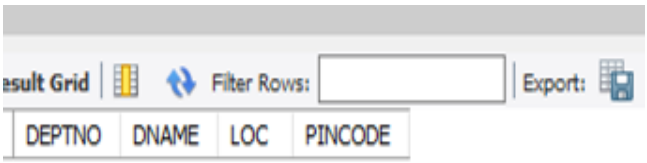
CODE:

```
create database q2;

use q2;

create table Dept(DEPTNO INT NOT NULL primary KEY,DNAME VARCHAR(20) NOT NULL,LOC VARCHAR(20) NOT NULL);

rename table Dept to department;

select * from department;
```

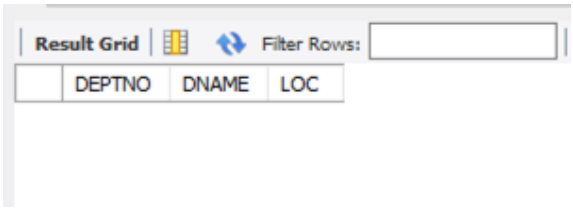


USE q2;

alter table department add column(PINCODE INT NOT NULL);

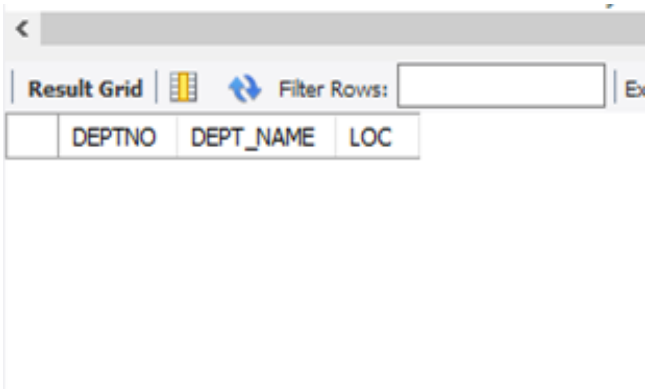


ALTER TABLE DEPT DROP COLUMN PINCODE;

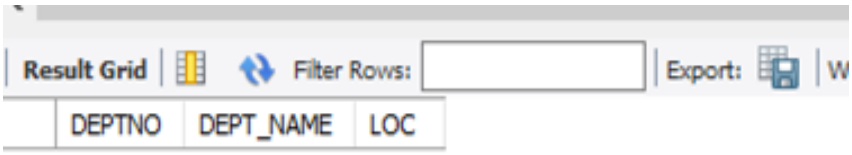


USE q2;

select * from department;alter table department change dname dept_name varchar(20);



ALTER TABLE DEPARTMENT MODIFY LOCATION CHAR(10);



DROP TABLE DEPARTMENT;

RESULT: Output obtained successfully.

PROGRAM NO 3

Consider Employee table:

EMPNO	EMP_NAME	DEPT	SALARY	DOJ	BRANCH
E101	Amit	oduction	45000	12-Mar-00	Bangalore
E102	Amit	HR	70000	03-Jul-02	Bangalore
E103	sunita	anagemer	120000	11-Jan-01	mysore
E105	sunita	IT	67000	01-Aug-01	mysore
E106	mahesh	Civil	145000	20-Sep-03	Mumbai

Perform the following :

1. Display all the fields of employee table
2. Retrieve employee number and their salary
3. Retrieve average salary of all employee
4. Retrieve number of employee
5. Retrieve distinct number of employee
6. Retrieve total salary of employee group by employee name and count similar names
7. Retrieve total salary of employee which is greater than >120000
8. Display name of employee in descending order
9. **Display details of employee whose name is AMIT and salary greater than 50000**

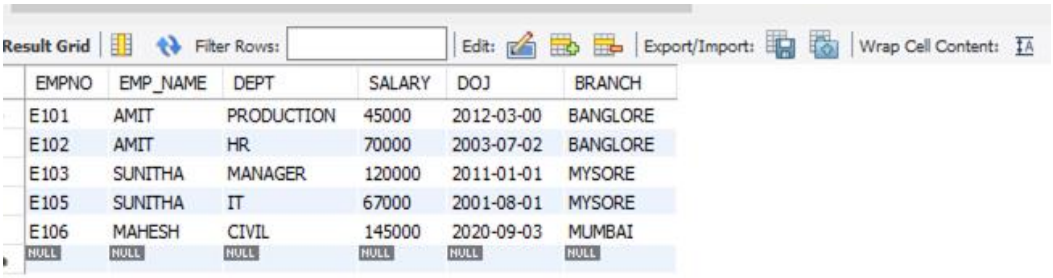
CODE:

```
CREATE DATABASE EMP;
USE EMP;
CREATE TABLE EMPLOYEE(EMPNO CHAR(4),EMP_NAME
VARCHAR(20),DEPT VARCHAR(20),SALARY INT,DOJ DATE NOT
NULL ,BRANCH VARCHAR(20) );
INSERT INTO
EMPLOYEE(EMPNO,EMP_NAME,DEPT,SALARY,DOJ,BRANCH)V
ALUES('E101','DON','PRODUCTION',45000,'2000-03-
12','BANGLORE');
INSERT INTO
EMPLOYEE(EMPNO,EMP_NAME,DEPT,SALARY,DOJ,BRANCH)V
ALUES('E102','DON','HR',70000,'2002-03-07','BANGLORE');
INSERT INTO
EMPLOYEE(EMPNO,EMP_NAME,DEPT,SALARY,DOJ,BRANCH)V
ALUES('E103','MANYA','MANAGER',120000,'2001-11-
01','MYSORE');
```

INSERT INTO EMPLOYEE(EMPNO,EMP_NAME,DEPT,SALARY,DOJ,BRANCH)VALUES('E105','MANYA','IT',67000,'2002-01-08','MYSORE');
INSERT INTO EMPLOYEE(EMPNO,EMP_NAME,DEPT,SALARY,DOJ,BRANCH)VALUES('E106','BINOY','CIVIL',145000,'2002-09-20','MUMBAI');

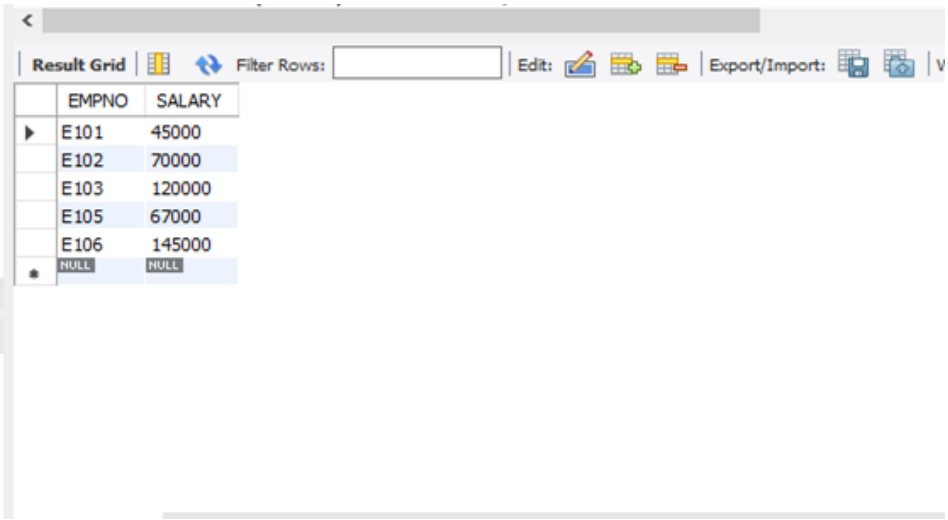
QUERY:

SELECT * FROM EMPLOYEE;



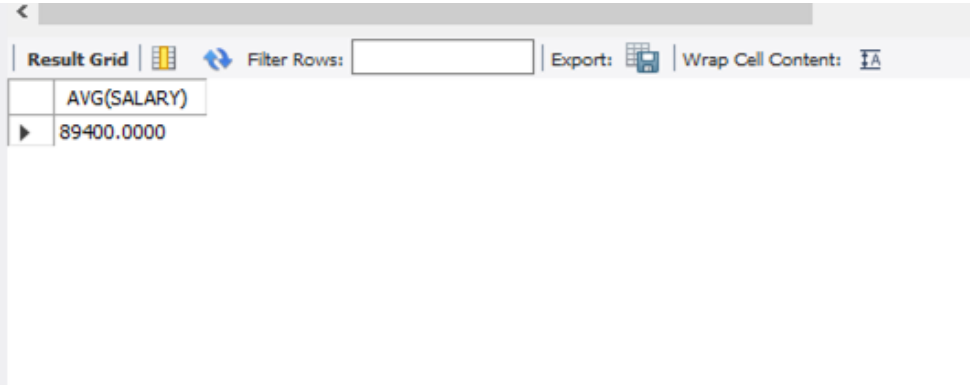
EMPNO	EMP_NAME	DEPT	SALARY	DOJ	BRANCH
E101	AMIT	PRODUCTION	45000	2012-03-00	BANGLORE
E102	AMIT	HR	70000	2003-07-02	BANGLORE
E103	SUNITHA	MANAGER	120000	2011-01-01	MYSORE
E105	SUNITHA	IT	67000	2001-08-01	MYSORE
E106	MAHESH	CIVIL	145000	2020-09-03	MUMBAI
NULL	NULL	NULL	NULL	NULL	NULL

SELECT EMPNO,SALARY FROM EMPLOYEE;

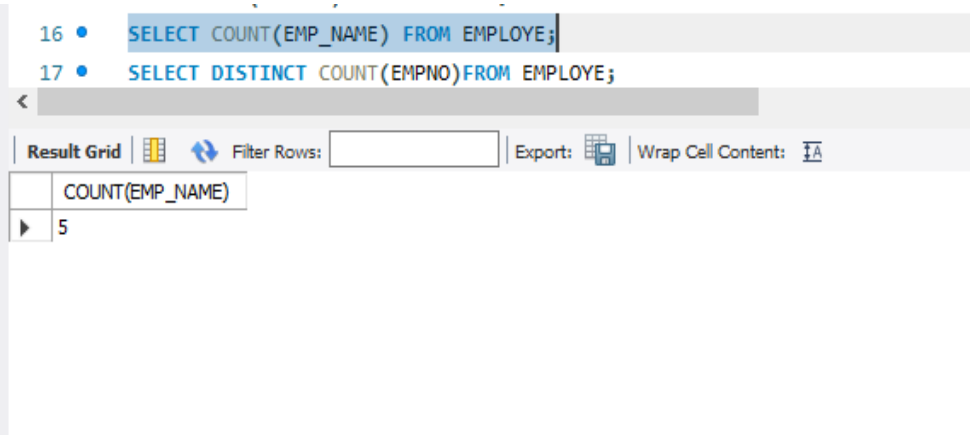


EMPNO	SALARY
E101	45000
E102	70000
E103	120000
E105	67000
E106	145000
NULL	NULL

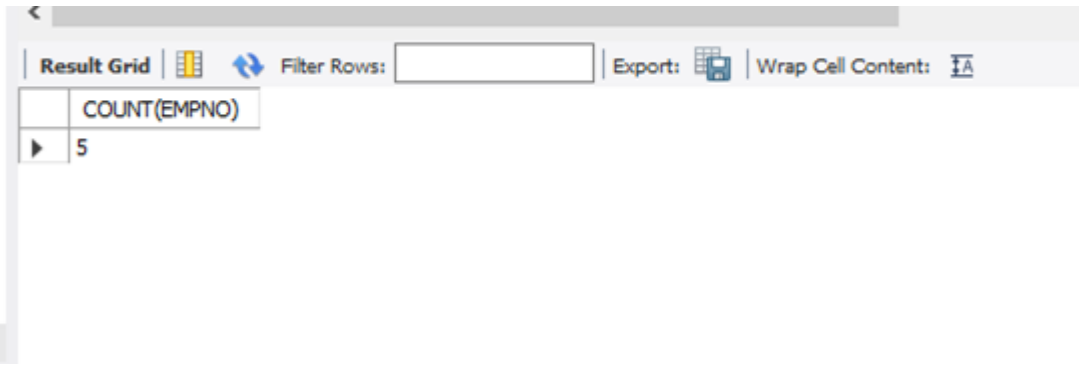
SELECT AVG(SALARY) FROM EMPLOYEE;



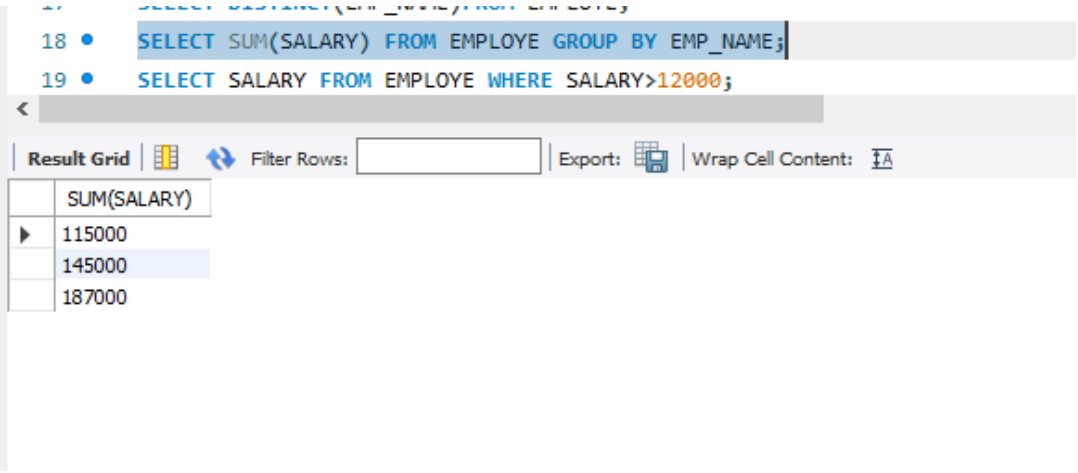
SELECT COUNT(*) FROM EMPLOYEE;



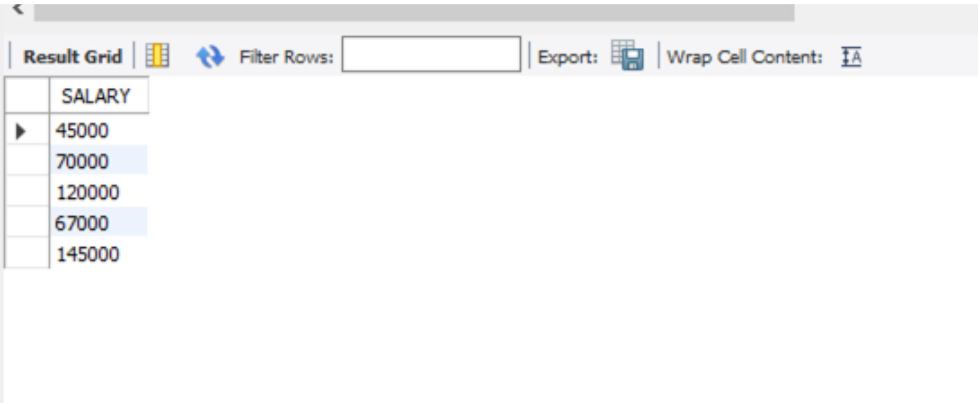
SELECT COUNT(DISTINCT EMP_NAME) FROM EMPLOYEE;



SELECT SUM(SALARY)FROM EMPLOYEE GROUP BY EMP_NAME;



SELECT * FROM EMPLOYEE WHERE SALARY>120000;



SELECT * FROM EMPLOYEE ORDER BY EMP_NAME DESC;

```
21 •  SELECT EMP_NAME FROM EMPLOYE ORDER BY EMP_NAME desc;
22 •  SELECT * FROM EMPLOYE WHERE EMP_NAME='AMIT' AND SALARY>50000;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

EMP_NAME
SUNITHA
SUNITHA
MAHESH
AMIT
AMIT

SELECT * FROM EMPLOYEE WHERE EMP_NAME='DON' AND SALARY>50000;

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

EMPNO	EMP_NAME	DEPT	SALARY	DOJ	BRANCH
E102	AMIT	HR	70000	2003-07-02	BANGLORE
NULL	NULL	NULL	NULL	NULL	NULL

RESULT: Output obtained successfully.