

WEEK 6– QUERIES

1. Using Scheme diagram, Create tables by properly specifying the primary keys and the foreign keys.

TABLE DEPARTMENT

```
create table dept(
```

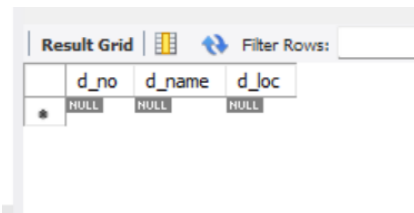
```
  d_no int,
```

```
  d_name varchar (10),
```

```
  d_loc varchar (30),
```

```
  primary key(d_no)
```

```
);
```



A screenshot of a database management system's 'Result Grid' window. The window has a title bar with 'Result Grid' and a 'Filter Rows' input field. Below the title bar is a table with three columns: 'd_no', 'd_name', and 'd_loc'. The first row of data shows three 'NULL' values. A small asterisk icon is visible in the first column of the first row.

| | d_no | d_name | d_loc |
|---|------|--------|-------|
| * | NULL | NULL | NULL |

TABLE PROJECT

```
create table project(
```

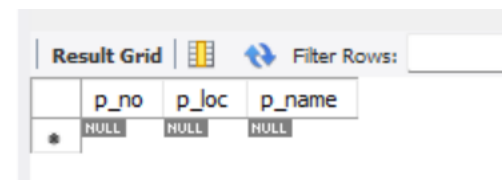
```
  p_no int,
```

```
  p_loc varchar(20),
```

```
  p_name varchar(15),
```

```
  PRIMARY KEY(p_no)
```

```
);
```



A screenshot of a database management system's 'Result Grid' window. The window has a title bar with 'Result Grid' and a 'Filter Rows' input field. Below the title bar is a table with three columns: 'p_no', 'p_loc', and 'p_name'. The first row of data shows three 'NULL' values. A small asterisk icon is visible in the first column of the first row.

| | p_no | p_loc | p_name |
|---|------|-------|--------|
| * | NULL | NULL | NULL |

TABLE EMPLOYEE

```
create table employee(
```

```
  emp_no int,
```

```
  emp_name varchar(10),
```

```
  mgr_no int,
```

```
  hiredate date,
```

```
  sal real,
```

```
  d_no int,
```

```
  primary key(emp_no),
```

```
  foreign key(d_no) references dept(d_no)
```

on update cascade on delete cascade

);



| | emp_no | emp_name | mgr_no | hiredate | sal | d_no |
|---|--------|----------|--------|----------|------|------|
| * | NULL | NULL | NULL | NULL | NULL | NULL |

TABLE INCENTIVES

create table incentives(

emp_no int,

incentive_date date,

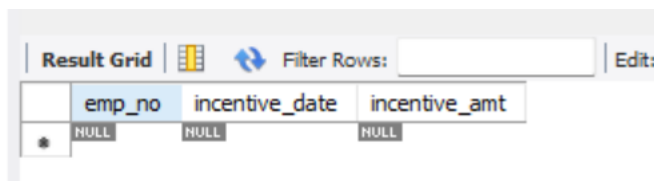
incentive_amt real,

primary key(incentive_date),

foreign key(emp_no) references employee(emp_no)

on update cascade on delete cascade

);



| | emp_no | incentive_date | incentive_amt |
|---|--------|----------------|---------------|
| * | NULL | NULL | NULL |

TABLE ASSIGNED

create table assigned(

emp_no int,

p_no int,

job_role varchar(10),

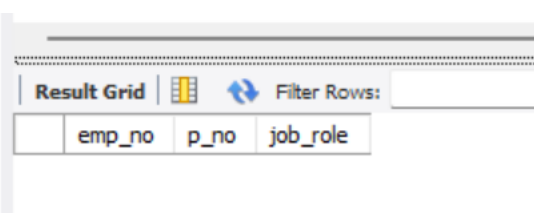
foreign key(emp_no) references employee(emp_no)

on update cascade on delete cascade,

foreign key(p_no) references project(p_no)

on update cascade on delete cascade

);



| | emp_no | p_no | job_role |
|--|--------|------|----------|
|--|--------|------|----------|

2. Enter greater than five tuples for each table.

SQL>

```
insert into dept values(10,'IT','mysore');
```

```
insert into dept values(20,'Marketing','patna');
```

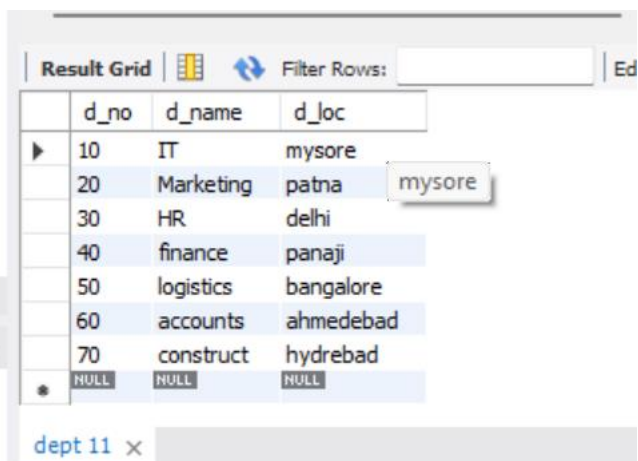
```
insert into dept values(30,'HR','delhi');
```

```
insert into dept values(40,'finance','panaji');
```

```
insert into dept values(50,'logistics','bangalore');
```

```
insert into dept values(60,'accounts','ahmedabad');
```

```
insert into dept values(70,'construct','hydrebad');
```



| | d_no | d_name | d_loc |
|---|------|-----------|-----------|
| ▶ | 10 | IT | mysore |
| | 20 | Marketing | patna |
| | 30 | HR | delhi |
| | 40 | finance | panaji |
| | 50 | logistics | bangalore |
| | 60 | accounts | ahmedabad |
| | 70 | construct | hydrebad |
| * | NULL | NULL | NULL |

dept 11 x

```
insert into project values(1,'mysore','oA1B1');
```

```
insert into project values(2,'patna','oA2B2');
```

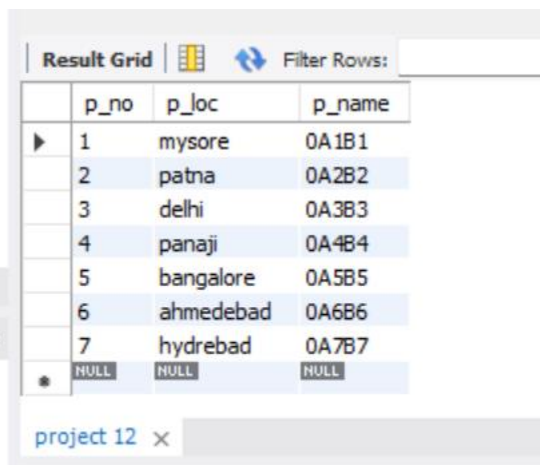
```
insert into project values(3,'delhi','oA3B3');
```

```
insert into project values(4,'panaji','oA4B4');
```

```
insert into project values(5,'bangalore','oA5B5');
```

```
insert into project values(6,'ahmedabad','oA6B6');
```

```
insert into project values(7,'hydrebad','oA7B7');
```



| | p_no | p_loc | p_name |
|---|------|-----------|--------|
| ▶ | 1 | mysore | 0A1B1 |
| | 2 | patna | 0A2B2 |
| | 3 | delhi | 0A3B3 |
| | 4 | panaji | 0A4B4 |
| | 5 | bangalore | 0A5B5 |
| | 6 | ahmedabad | 0A6B6 |
| | 7 | hydrebad | 0A7B7 |
| * | NULL | NULL | NULL |

project 12 x

```
insert into employee values(101,'sony',null,'2010-01-01',140000,10);
```

```
insert into employee values(102,'toni',104,'2009-07-31',28000,20);
```

```

insert into employee values(103,'rishi',104,'2015-02-24',30000,30);
insert into employee values(104,'santhosh',101,'2018-09-08',94000,10);
insert into employee values(105,'vineeth',108,'2009-05-18',11000,40);
insert into employee values(106,'twinkle',104,'2002-12-25',30000,50);
insert into employee values(107,'riddhi',108,'2010-03-01',10000,60);
insert into employee values(108,'dhruv',104,'2012-03-05',70000,70);
insert into employee values(109,'anirudh',101,'2016-06-06',20000,30);
insert into employee values(110,'ansh',108,'2015-07-23',17000,70);
insert into employee values(111,'kanan',101,'2018-08-11',29000,70);

```

| emp_no | emp_name | mgr_no | hiredate | sal | d_no |
|--------|----------|--------|------------|--------|------|
| 101 | sony | NULL | 2010-01-01 | 140000 | 10 |
| 102 | toni | 104 | 2009-07-31 | 28000 | 20 |
| 103 | rishi | 104 | 2015-02-24 | 30000 | 30 |
| 104 | santhosh | 101 | 2018-09-08 | 94000 | 10 |
| 105 | vineeth | 108 | 2009-05-18 | 11000 | 40 |
| 106 | twinkle | 104 | 2002-12-25 | 30000 | 50 |
| 107 | riddhi | 108 | 2010-03-01 | 10000 | 60 |
| 108 | dhruv | 104 | 2012-03-05 | 70000 | 70 |
| 109 | anirudh | 101 | 2016-06-06 | 20000 | 30 |
| 110 | ansh | 108 | 2015-07-23 | 17000 | 70 |
| 111 | kanan | 101 | 2018-08-11 | 29000 | 70 |

```

insert into incentives values(103,'2022-07-12',2500);
insert into incentives values(104,'2021-01-29',3500);
insert into incentives values(109,'2022-02-28',2000);
insert into incentives values(110,'2021-01-21',6000);
insert into incentives values(106,'2022-04-18',5000);
insert into incentives values(107,'2022-06-11',2300);

```

| emp_no | incentive_date | incentive_amt |
|--------|----------------|---------------|
| 110 | 2021-01-21 | 6000 |
| 104 | 2021-01-29 | 3500 |
| 109 | 2022-02-28 | 2000 |
| 106 | 2022-04-18 | 5000 |
| 107 | 2022-06-11 | 2300 |
| 103 | 2022-07-12 | 2500 |
| NULL | NULL | NULL |

```

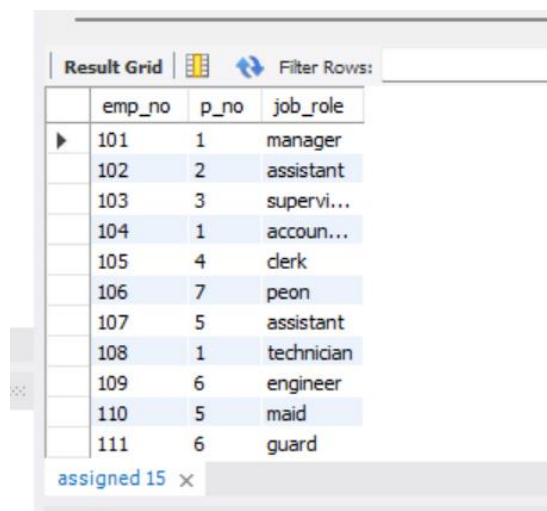
insert into assigned values(101,1,'manager');
insert into assigned values(102,2,'assistant');

```

```

insert into assigned values(103,3,'supervisor');
insert into assigned values(104,1,'accountant');
insert into assigned values(105,4,'clerk');
insert into assigned values(106,7,'peon');
insert into assigned values(107,5,'assistant');
insert into assigned values(108,1,'technician');
insert into assigned values(109,6,'engineer');
insert into assigned values(110,5,'maid');
insert into assigned values(111,6,'guard');

```



The screenshot shows a 'Result Grid' with columns 'emp_no', 'p_no', and 'job_role'. It contains 11 rows of data, representing employees and their supervisors. The interface includes a 'Filter Rows' button and a status bar at the bottom indicating 'assigned 15'.

| emp_no | p_no | job_role |
|--------|------|------------|
| 101 | 1 | manager |
| 102 | 2 | assistant |
| 103 | 3 | supervi... |
| 104 | 1 | accoun... |
| 105 | 4 | clerk |
| 106 | 7 | peon |
| 107 | 5 | assistant |
| 108 | 1 | technician |
| 109 | 6 | engineer |
| 110 | 5 | maid |
| 111 | 6 | guard |

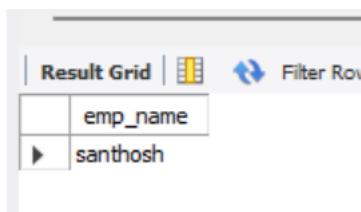
3. List the name of the managers with the maximum employees

SQL>

```

select emp_name from employee where emp_no =(select mgr_no from employee group by mgr_no having
count(emp_no)=(
select count(emp_no) from employee group by mgr_no order by count(emp_no) desc limit 1));

```



The screenshot shows a 'Result Grid' with a single column 'emp_name' and one row containing the name 'santhosh'.

| emp_name |
|----------|
| santhosh |

4. Display those managers name whose salary is more than average salary of his employee.

SQL>

```

select * from employee where emp_no =( select emp_no from incentives where incentive_date
between '2021-01-01' and '2021-01-31'

and incentive_amt!=(select max(incentive_amt) from incentives where incentive_date between
'2021-01-01' and '2021-01-31'));

```

| | |
|-------------|--------------|
| Result Grid | Filter Rows: |
| emp_name | |
| sony | |
| santhosh | |
| dhruv | |

5. Find the name of the second top level managers of each department.

SQL>

```
select emp_name from employee where emp_no in(select distinct mgr_no
from employee where emp_no in (select distinct mgr_no
from employee where emp_no in(select distinct mgr_no from employee)));
```

| | |
|-------------|--------------|
| Result Grid | Filter Rows: |
| emp_name | |
| sony | |

6. Find the employee details who got second maximum incentive in January 2019.

SQL> select * from employee where emp_no =(select emp_no from incentives where incentive_date between '2019-01-01' and '2019-01-31' and incentive_amt!=(select max(incentive_amt) from incentives where incentive_date between '2019-01-01' and '2019-01-31'));

Result Grid

Filter Rows:

Edit:

Export/Import:

| | emp_no | emp_name | mgr_no | hiredate | sal | d_no |
|---|--------|----------|--------|------------|-------|------|
| ▶ | 104 | santhosh | 101 | 2018-09-08 | 94000 | 10 |
| * | NULL | NULL | NULL | NULL | NULL | NULL |

7. Display those employees who are working in the same department where his manager is working.

SQL>

```
select e.emp_name from employee e where e.d_no=(select d_no from employee where
e.mgr_no=emp_no);
```

| | |
|-------------|--------------|
| Result Grid | Filter Rows: |
| emp_name | |
| santhosh | |
| ansh | |

