

WEEK-1

(1BM21CS059)

Q)Create the above tables by properly specifying the primary keys and the foreign keys as done in previous week's lab and Enter at least five tuples for each relationEnter at least five tuples for each relation

```
create database 1bm21cs045_insurance;
use 1bm21cs045_insurance;
create table person (
driver_id varchar(10),
name varchar(30),
address varchar(30),
primary key(driver_id)
);
desc person;
```

```
create table car(
reg_num varchar(10),
model varchar(10),
year int,
primary key(reg_num)
);
```

```
create table accident(
report_num int,
accident_date date,
location varchar(20),
primary key(report_num)
);
create table owns(
driver_id varchar(10),
reg_num varchar(10),
primary key(driver_id,reg_num),
foreign key(driver_id)references person(driver_id),
foreign key(reg_num)references car(reg_num)
);
create table participated(
driver_id varchar(10),
reg_num varchar(10),
report_num int,
damage_amount int,
primary key(driver_id,reg_num,report_num),
foreign key(driver_id) references person(driver_id),
```

```

foreign key(reg_num)references car(reg_num),
foreign key(report_num) references accident(report_num)
);
insert into accident values(11,'2003-01-01','Mysore road' );
insert into accident values(12,'2004-02-02','South end circle' );
insert into accident values(13,'2003-01-21','Bull temple road' );
insert into accident values(14,'2008-02-17','Mysore road' );
insert into accident values(15,'2004-03-05','Kanakpura road' );
insert into person values('A01','Richard','Srinivas nagar');
insert into person values('A02','Pradeep','Rajaji nagar');
insert into person values('A03','Smith','Ashok nagar');
insert into person values('A04','Venu','N R Colony');
insert into person values('A05','Jhon','Hanumanth nagar');
insert into car values('KA052250','Indica',1990);
insert into car values('KA031181','Lancer',1957);
insert into car values('KA095477','Toyota',1998);
insert into car values('KA053408','Honda',2008);
insert into car values('KA041702','Audi',2005);
insert into owns values('A01','KA052250');
insert into owns values('A02','KA053408');
insert into owns values('A03','KA095477');
insert into owns values('A04','KA031181');
insert into owns values('A05','KA041702');
insert into participated values('A01','KA052250',11,10000);
insert into participated values('A02','KA053408',12,50000);
insert into participated values('A03','KA095477',13,25000);
insert into participated values('A04','KA031181',14,3000);
insert into participated values('A05','KA041702',15,5000);

```

Display the entire CAR relation in the ascending order of manufacturing year.

SQL> select *from car

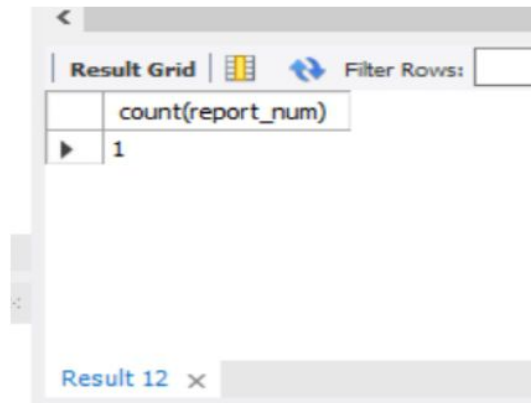
order by year asc;

reg_num	model	year
KA031181	Lancer	1957
KA052250	Indica	1990
KA095477	Toyota	1998
KA041702	Audi	2005
KA053408	Honda	2008
NULL	NULL	NULL

car 11 x

Find the number of accidents in which cars belonging to a specific model (example 'Lancer') were involved.

```
SQL> select count(report_num)
from car c, participated p
where c.reg_num=p.reg_num and c.model='Lancer';
```

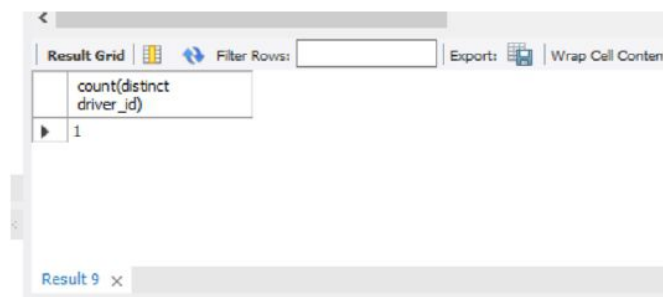


The screenshot shows a database query result grid. The grid has two columns: the first column is empty, and the second column is labeled 'count(report_num)'. There is one row with the value '1' in the second column. The interface includes a 'Filter Rows' button and a 'Result 12' tab at the bottom.

	count(report_num)
▶	1

Find the total number of people who owned cars that were involved in accidents in 2008.

```
SQL> select count(distinct driver_id)
from participated a, accident b
where a.report_num=b.report_num and b.accident_date like '2008%';
```



The screenshot shows a database query result grid. The grid has two columns: the first column is empty, and the second column is labeled 'count(distinct driver_id)'. There is one row with the value '1' in the second column. The interface includes a 'Filter Rows' button, an 'Exports' button, and a 'Wrap Cell Content' button at the bottom.

	count(distinct driver_id)
▶	1

TO DO

List the entire participated relation in descending order of damage amount.

```
SQL> select *from participated
order by damage_amount desc;
```

Result Grid | Filter Rows: | Edit:

	driver_id	reg_num	report_num	damage_amount
▶	A02	KA053408	12	50000
	A03	KA095477	13	25000
	A01	KA052250	11	10000
	A05	KA041702	15	5000
	A04	KA031181	14	3000
✱	NULL	NULL	NULL	NULL

participated 13 x

Find the average damage amount.

```
SQL> select avg(damage_amount)
from participated;
```

Result Grid | Filter Rows: | Export:

	avg(damage_amount)
▶	18600.0000

Result 14 x

Delete the tuple whose damage amount is below the average damage amount.

```
SQL> delete from participated
where damage_amount < (select t.avg1 from (select avg(damage_amount) as avg1 from
participated) t);
select *from participated;
```

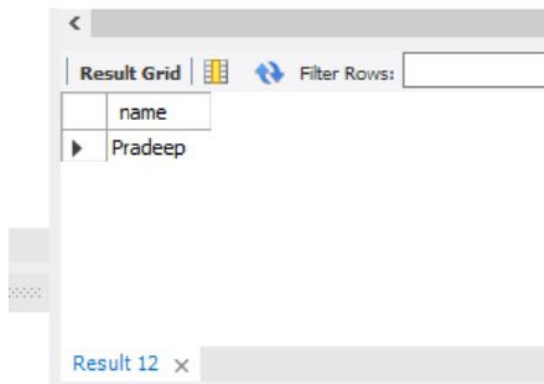
Result Grid | Filter Rows: | Edit:

	driver_id	reg_num	report_num	damage_amount
▶	A02	KA053408	12	50000
	A03	KA095477	13	25000
✱	NULL	NULL	NULL	NULL

participated 7 x

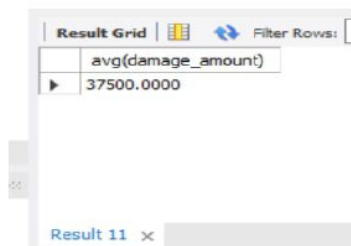
List the name of drivers whose damage is greater than the average damage amount.

```
SQL> select name
from person p, participated q
where p.driver_id=q.driver_id and damage_amount >
(select avg(damage_amount)
from participated
);
```



The screenshot shows a database interface with a 'Result Grid' tab. The grid contains one row with the name 'Pradeep'. Above the grid is a 'Filter Rows' input field. Below the grid is a tab labeled 'Result 12'.

name
Pradeep

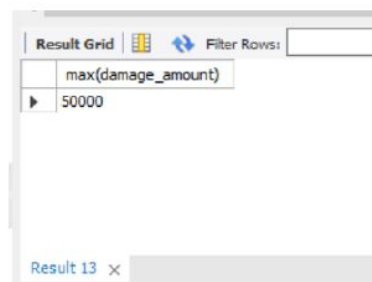


The screenshot shows a database interface with a 'Result Grid' tab. The grid contains one row with the value '37500.0000'. Above the grid is a 'Filter Rows' input field. Below the grid is a tab labeled 'Result 11'.

avg(damage_amount)
37500.0000

Find the maximum damage amount

```
SQL>select max(damage_amount)
from participated;
```



The screenshot shows a database interface with a 'Result Grid' tab. The grid contains one row with the value '50000'. Above the grid is a 'Filter Rows' input field. Below the grid is a tab labeled 'Result 13'.

max(damage_amount)
50000

