WEEK – 2

More Queries on Insurance Database

PERSON (driver_id: String, name: String, address: String)
CAR (reg_num: String, model: String, year: int)
ACCIDENT (report_num: int, accident_date: date, location: String)
OWNS (driver_id: String, reg_num: String)
PARTICIPATED (driver_id: String,reg_num: String, report_num: int, damage_amount: int)

Create the above tables by properly specifying the primary keys and the foreign keys as done in "Program-1"week's lab and Enter at least five tuples for each relation.

i. Display the entire CAR relation in the ascending order of manufacturing year.
ii. Find the number of accidents in which cars belonging to a specific model (example 'Lancer') were involved.
iii. Find the total number of people who owned cars that involved in accidents in 2008.
iv. List the Entire Participated Relation in the Descending Order Of Damage Amount.
Find the Average Damage Amount
v. Delete the Tuple Whose Damage Amount is below the Average Damage Amount
vi. List the Name Of Drivers Whose Damage is Greater than The Average Damage Amount.
vii. Find Maximum Damage Amount.

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

```
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

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

```
select count(*)
from participated p
join car c on p.reg_num= c.reg_num
where c.model = 'lancer';
```

	count(*)
▶	1

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

```
select count(distinct p.driver_id) as accident_in_2008
from person p
join participated pa on p.driver_id = pa.driver_id
join accident a on pa.report_num = a.report_num
where year(a.accident_date) = 2008;
```

	accident_in_2008
▶	2

List the Entire Participated Relation in the Descending Order Of Damage Amount.
Find the Average Damage Amount

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

select avg(damage_amount) as avg_amount from participated;
```

	avg_amount
▶	14166.6667

driver_id	reg_num	report_num	damage_amount
A02	KA053408	12	25000
A03	KA095477	13	25000
A02	KA053408	16	17000
A01	KA052250	11	10000
A05	KA041702	15	5000
A04	KA031181	14	3000
NULL	NULL	NULL	NULL

Delete the Tuple Whose Damage Amount is below the Average Damage Amount

```
delete from participated
where damage_amount <
(select avg_amt from
(select avg(damage_amount) as avg_amt from participated) as temp);
```

driver_id	reg_num	report_num	damage_amount
A02	KA053408	12	25000
A02	KA053408	16	17000
A03	KA095477	13	25000
NULL	NULL	NULL	NULL

List the Name Of Drivers Whose Damage is Greater than The Average Damage Amount.

```
select p.driver_name, pa.damage_amount
from person p
join participated pa  on pa.driver_id = p.driver_id
where damage_amount > ( select avg(damage_amount) from participated);
```

driver_name	damage_amount
Pradeep	25000
Pradeep	17000
Smith	25000

Find Maximum Damage Amount.

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

	max(damage_amount)
▶	25000