COSC265 Lab 3 - Solutions

Queries for the MOVIES database

1. Find names of stars who acted in movies directed by Sofia Coppola.
   ```
   select distinct star.fname, star.lname
   from star, stars, movie, director
   where star=snumber and movie=mnumber and director=dnumber
           and director.fname=’Sofia’ and director.lname=’Coppola’;
   ```
2. Find names of stars who acted in at least two movies directed by Sofia Coppola.
   ```
   select star.fname, star.lname, count(*)
   from star, stars, movie, director
   where star=snumber and movie=mnumber and director=dnumber
           and director.fname=’Sofia’ and director.lname=’Coppola’
   group by star.fname, star.lname, snumber
   having count(*)>=2;
   ```
3. Show types of movies for which there are more than 5 movies in the database. Order the results by decreasing number of movies.
   ```
   select type, count(*) as no_movies
   from movie
   group by type
   having count(*)>5
   order by no_movies desc;
   ```
4. Find the names of all directors who directed at least as many movies as the director number 15.
   ```
   select fname, lname
   from director join movie on director=dnumber
   group by director, fname, lname
   having count(*) >= (select count(*) from movie where director=15);
   ```
5. Find the director who has directed most dramas.
   ```
   select fname, lname
   from director join movie on director = dnumber
   where type=’drama’
   group by director, fname, lname
   having count(*) >= all (select count(*) from movie
                              where type=’drama’ group by director);
   ```

Tasks for the REGISTRATION database

1. Get full details of all vehicles which were registered during July 2011.
   ```
   select *
   from vehicle join  registration R on R.plates = vehicle.plates
   where  reg_date between ’01-jul-2011’ and ’31-jul-2011’;
   ```

2. Get the list of vehicles imported from Japan since 1985 which had been registered less than 3 times in New Zealand, listing their plates, makes and models.

```
select vehicle.plates, make, model, count(*) as No_regs
from vehicle, registration R1, registration R2
where R1.plates = vehicle.plates and R1.country = ’Japan’ and R1.reg_date>'01-jan-85'
      and R1.plates=R2.plates
group by vehicle.plates, make, model
having count(*)<3;
```

3. Find the names of people who own more than one vehicle.

```
select lname, init, fname
from owner join owns on ownerid=dr_lic
where datesold is null
group by dr_lic ,lname, init, fname
having 1<count(*);
```

4. Write SQL statements to update the REGISTRATION database in the following cases:
   a. Anna Simmons has had her VW golf painted in green.

```
        update color
        set color='green'
        where plates=(select owns.plates
        from owns, owner, vehicle
        where owns.plates = vehicle.plates and
        ownerid=dr_lic and
        lname='Simmons' and
        fname='Anna' and
        make='VW' and model='golf');
```

   b. Write the INSERT statement to add a new registration for the car with plates number TX9283. The car was registered on July 1, 2011 by employee 21321322 who works for registration organization 1352. The DRR reading on the day was 169654 kilometres, and the cost was $137.85.

```
   insert into registration
   values (’TX9283’, 21321322,1352,’01-jul-2011’, null,169654,137.85);
```

   c. Delete all registrations for vehicle TX9283.

```
   delete from registration
   where plates = ’TX9283’;
```

5. Create a view EMPS which contains the names and birthdates of all people employed in registration organizations. Try to change the birthdate of some employee.

```
   create view emps
   as select lname, fname, bdate
   from employee
   where reg_org is not null;

   update emps
   set bdate=’25-jul-2011’
   where fname=’Anna’ and lname=’Simmons’;
```

Oracle allows birthdates to be changes. We have seen that in the standard SQL such a view cannot be updated, as the primary key is not included in the view.

6. Create a view which contains the plates, make and model of each vehicle, and the number of times it has been registered. Using this view, find vehicles that have been registered at least three times. Try deleting information about a specific vehicle. Discuss what happens with DML statements when performed on views. Check what happens both to the view and the base table(s) on which the view is defined.

```
create view veh_info
as select vehicle.plates, make, model, count(*) as regno
        from vehicle join registration on registration.plates=vehicle.plates
        group by vehicle.plates, make, model;

select plates
from veh_info
where regno>3;

delete from veh_info
where plates='FO2341';
```

   This view cannot be updated, as it is defined using an aggregate function.

7. Create an index IYEAR on the YEAR attribute of the VEHICLE table. Consider other candidates for indexing.

```
create index IYEAR
on vehicle(year);
```