In Class Exercise for Advanced SQL

Create the following table. Ensure that id is a primary key

```
mysql> select * from crime;
+------+------------+---------------+------------+
| id   | date       | crime         | suspect_id |
+------+------------+---------------+------------+
|    1 | 2009-06-28 | armed robbery |          1 |
|    2 | 2008-05-05 | car theft     |       NULL |
|    3 | 2007-11-03 | burglary      |          3 |
|    4 | 2006-11-03 | burglary      |          3 |
|    5 | 2006-08-04 | vandalism     |       NULL |
|    6 | 2006-02-03 | armed robbery |       NULL |
+------+------------+---------------+------------+
```

Create the following suspects table. Ensure that id is a primary key

```
+------------+-------------+------+
| given_name | family_name | id   |
+------------+-------------+------+
| Rowan      | Seymour     |    1 |
| Bob        | Jones       |    2 |
| Amy        | Tang        |    3 |
+------------+-------------+------+
```

1. Write a query to show the number of records with null values in the suspect_id column

```
select count(*) from crime where suspect_id is null;
```

2. Show only the records in the crime table which have a matching id in the suspect table

```
select * from crime inner join suspect on
crime.suspect_id = suspect.id;
```

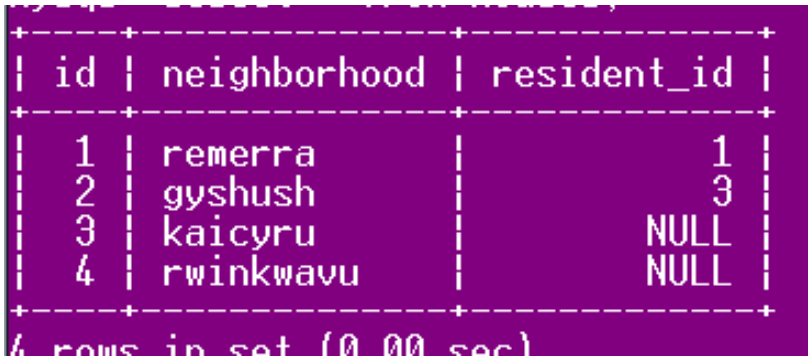
3. Show all the records in the crime table and only those from the suspect table having a matching id

```
select * from crime left join suspect on
crime.suspect_id = suspect.id
```

4. Show all the records from the suspect table and show only the crime records with suspect_ids matching the id field in the crime table

```
select * from crime right join suspect on
crime.suspect_id = suspect.id
```

Create the following houses table. Ensure that id is a primary key



```
+----+--------------+-------------+
| id | neighborhood | resident_id |
+----+--------------+-------------+
|  1 | remerra      |           1 |
|  2 | gyshush      |           3 |
|  3 | kaicyru      |        NULL |
|  4 | rwinkwavu    |        NULL |
+----+--------------+-------------+
4 rows in set (0.00 sec)
```

`

5. Show only the neighborhood that have residents in the suspect table, the suspect's given name, and only the crimes committed by those residents

```
select n.neighborhood,s.given_name,c.crime from
(neighborhood as n inner join suspects as s on
n.id=s.id) inner join crime as c on c.id = s.id;
```

6. Show all existing neighborhoods, the residents that live in those houses and all existing crimes

```
select * from (houses left join suspects on
house.resident_id = suspect.id) right join crime on
crime.suspect_id = suspect.id
```

7. Modify the suspects table to ensure the family name will never have a null entry

```
alter table suspects modify family_name varchar(50) not
null;
```

8. Alter table houses and change the neighborhood column name to village

```
alter table house change neighborhood village
varchar(30);
```

9. Alter the houses table and add a column to for country

```
alter table house add column country varchar(30);
```

10. Add a constraint to ensure that id is unique

```
alter table test add constraint unique(id2);

alter table suspect modify id int unique;
```