Coding Challenge SQL
Crime Management Shema DDL and DML

```
create database Crime_Management;
use Crime_Management;
```

```
mysql> use Crime_Management;
Database changed
mysql>
```

```
CREATE TABLE Crime (
 CrimeID INT PRIMARY KEY,
 IncidentType VARCHAR(255),
 IncidentDate DATE,
 Location VARCHAR(255),
 Description TEXT,
 Status VARCHAR(20)
);
```

```
mysql> desc Crime;
+--------------+--------------+------+-----+---------+-------+
| Field        | Type         | Null | Key | Default | Extra |
+--------------+--------------+------+-----+---------+-------+
| CrimeID      | int          | NO   | PRI | NULL    |       |
| IncidentType | varchar(255) | YES  |     | NULL    |       |
| IncidentDate | date         | YES  |     | NULL    |       |
| Location     | varchar(255) | YES  |     | NULL    |       |
| Description  | text         | YES  |     | NULL    |       |
| Status       | varchar(20)  | YES  |     | NULL    |       |
+--------------+--------------+------+-----+---------+-------+
6 rows in set (0.01 sec)
```

```
CREATE TABLE Victim (
 VictimID INT PRIMARY KEY,
 CrimeID INT,
 Name VARCHAR(255),
 ContactInfo VARCHAR(255),
 Injuries VARCHAR(255),
 FOREIGN KEY (CrimeID) REFERENCES Crime(CrimeID)
);
```

```
mysql> desc Victim;
+-------------+--------------+------+-----+---------+-------+
| Field       | Type         | Null | Key | Default | Extra |
+-------------+--------------+------+-----+---------+-------+
| VictimID    | int          | NO   | PRI | NULL    |       |
| CrimeID     | int          | YES  | MUL | NULL    |       |
| Name        | varchar(255) | YES  |     | NULL    |       |
| ContactInfo | varchar(255) | YES  |     | NULL    |       |
| Injuries    | varchar(255) | YES  |     | NULL    |       |
| age         | int          | YES  |     | NULL    |       |
+-------------+--------------+------+-----+---------+-------+
6 rows in set (0.00 sec)
```

```
CREATE TABLE Suspect (
 SuspectID INT PRIMARY KEY,
 CrimeID INT,
 Name VARCHAR(255),
 Description TEXT,
 CriminalHistory TEXT,
 FOREIGN KEY (CrimeID) REFERENCES Crime(CrimeID)
);
```

```
mysql> desc Suspect;
+-----------------+--------------+------+-----+---------+-------+
| Field           | Type         | Null | Key | Default | Extra |
+-----------------+--------------+------+-----+---------+-------+
| SuspectID       | int          | NO   | PRI | NULL    |       |
| CrimeID         | int          | YES  | MUL | NULL    |       |
| Name            | varchar(255) | YES  |     | NULL    |       |
| Description     | text         | YES  |     | NULL    |       |
| CriminalHistory | text         | YES  |     | NULL    |       |
| age             | int          | YES  |     | NULL    |       |
+-----------------+--------------+------+-----+---------+-------+
6 rows in set (0.01 sec)
```

```
INSERT INTO Crime (CrimeID, IncidentType, IncidentDate, Location, Description, Status)
VALUES
 (1, 'Robbery', '2023-09-15', '123 Main St, Cityville', 'Armed robbery at a convenience store',
'Open'),
 (2, 'Homicide', '2023-09-20', '456 Elm St, Townsville', 'Investigation into a murder case', 'Under
Investigation'),
 (3, 'Theft', '2023-09-10', '789 Oak St, Villagetown', 'Shoplifting incident at a mall', 'Closed');
```

```
mysql> select * from Crime;
+---------+--------------+--------------+-----------------------+-----------------------------------+--------------------+
| CrimeID | IncidentType | IncidentDate | Location              | Description                       | Status             |
+---------+--------------+--------------+-----------------------+-----------------------------------+--------------------+
|       1 | Robbery      | 2023-09-15   | 123 Main St, Cityville | Armed robbery at a convenience store | Open               |
|       2 | Homicide     | 2023-09-20   | 456 Elm St, Townsville | Investigation into a murder case     | Under
Investigation |
|       3 | Theft        | 2023-09-10   | 789 Oak St, Villagetown | Shoplifting incident at a mall       | Closed             |
+---------+--------------+--------------+-----------------------+-----------------------------------+--------------------+
3 rows in set (0.00 sec)
```

INSERT INTO Victim (VictimID, CrimeID, Name, ContactInfo, Injuries)
VALUES
(1, 1, 'John Doe', 'johndoe@example.com', 'Minor injuries'),
(2, 2, 'Jane Smith', 'janesmith@example.com', 'Deceased'),
(3, 3, 'Alice Johnson', 'alicejohnson@example.com', 'None');

```
mysql> select * from Victim;
+----------+---------+---------------+--------------------------+----------------+-------+
| VictimID | CrimeID | Name          | ContactInfo              | Injuries       | age   |
+----------+---------+---------------+--------------------------+----------------+-------+
|        1 |       1 | John Doe      | johndoe@example.com      | Minor injuries |    30 |
|        2 |       2 | Jane Smith    | janesmith@example.com    | Deceased       |    40 |
|        3 |       3 | Alice Johnson | alicejohnson@example.com | None           |    38 |
+----------+---------+---------------+--------------------------+----------------+-------+
3 rows in set (0.00 sec)
```

INSERT INTO Suspect (SuspectID, CrimeID, Name, Description, CriminalHistory)
VALUES
(1, 1, 'Robber 1', 'Armed and masked robber', 'Previous robbery convictions'),
(2, 2, 'Unknown', 'Investigation ongoing', NULL),
(3, 3, 'Suspect 1', 'Shoplifting suspect', 'Prior shoplifting arrests');

```
mysql> select * from Suspect;
+-----------+---------+-----------+-----------------------+-----------------------------+------+
| SuspectID | CrimeID | Name      | Description           | CriminalHistory             | age  |
+-----------+---------+-----------+-----------------------+-----------------------------+------+
|         1 |       1 | Robber 1  | Armed and masked robber | Previous robbery convictions |   37 |
|         2 |       2 | Unknown   | Investigation ongoing | NULL                        |   46 |
|         3 |       3 | Suspect 1 | Shoplifting suspect   | Prior shoplifting arrests   |   35 |
+-----------+---------+-----------+-----------------------+-----------------------------+------+
3 rows in set (0.00 sec)
```

1. Select all open incidents

select * from Crime where Status='Open';

```
mysql> select * from Crime where Status='Open';
+---------+--------------+--------------+-----------------------+-----------------------------------+--------+
| CrimeID | IncidentType | IncidentDate | Location              | Description                       | Status |
+---------+--------------+--------------+-----------------------+-----------------------------------+--------+
|       1 | Robbery      | 2023-09-15   | 123 Main St, Cityville | Armed robbery at a convenience store | Open   |
+---------+--------------+--------------+-----------------------+-----------------------------------+--------+
1 row in set (0.00 sec)
```

2. Find the total number of incidents

select count(*) as Total_Incidents from Crime;

```
mysql>  select count(*) as Total_Incidents from Crime;
+-----------------+
| Total_Incidents |
+-----------------+
|               3 |
+-----------------+
1 row in set (0.01 sec)
```

3. List all unique incident types

select distinct IncidentType from crime;

```
mysql> select distinct IncidentType from crime;
+--------------+
| IncidentType |
+--------------+
| Robbery      |
| Homicide     |
| Theft        |
+--------------+
3 rows in set (0.00 sec)
```

4. Retrieve incidents that occurred between '2023-09-01' and '2023-09-10'

select * from Crime where IncidentDate between '2023-09-01' and '2023-09-10';

```
mysql>   select * from Crime where IncidentDate between '2023-09-01' and '2023-09-10';
+---------+--------------+--------------+------------------------+-------------------------------+--------+
| CrimeID | IncidentType | IncidentDate | Location               | Description                   | Status |
+---------+--------------+--------------+------------------------+-------------------------------+--------+
|       3 | Theft        | 2023-09-10   | 789 Oak St, Villagetown | Shoplifting incident at a mall | Closed |
+---------+--------------+--------------+------------------------+-------------------------------+--------+
1 row in set (0.00 sec)
```

5. List persons involved in incidents in descending order of age

alter table Victim add age int;

desc Victim;

update Victim set age='30' where VictimID='1';

update Victim set age='40' where VictimID='2';

update Victim set age='38' where VictimID='3';

select * from Victim;

select * from Victim order by age desc;

```
mysql>    select * from Victim order by age desc;
+----------+---------+---------------+--------------------------+----------------+------+
| VictimID | CrimeID | Name          | ContactInfo              | Injuries       | age  |
+----------+---------+---------------+--------------------------+----------------+------+
|        2 |       2 | Jane Smith    | janesmith@example.com    | Deceased       |   40 |
|        3 |       3 | Alice Johnson | alicejohnson@example.com | None           |   38 |
|        1 |       1 | John Doe      | johndoe@example.com      | Minor injuries |   30 |
+----------+---------+---------------+--------------------------+----------------+------+
3 rows in set (0.00 sec)
```

6. Find the average age of persons involved in incidents.

select avg(age) as averag_age from Victim;

```
mysql>    select avg(age) as averag_age from Victim;
+-------------+
| averag_age |
+-------------+
|    36.0000 |
+-------------+
1 row in set (0.00 sec)
```

7. List incident types and their counts, only for open cases.

select IncidentType,count(*)
from Crime
where status='Open'
group by IncidentType;

```
mysql>    select IncidentType,count(*)
    ->    from Crime
    ->    where status='Open'
    ->    group by IncidentType;
+--------------+----------+
| IncidentType | count(*) |
+--------------+----------+
| Robbery      |        1 |
+--------------+----------+
1 row in set (0.00 sec)
```

8. Find persons with names containing 'Doe'

```
select * from Victim
where name like '%Doe';
```

```
mysql> select * from Victim
    ->     where name like '%Doe';
+----------+---------+----------+--------------------+---------------+------+
| VictimID | CrimeID | Name     | ContactInfo        | Injuries      | age  |
+----------+---------+----------+--------------------+---------------+------+
|        1 |       1 | John Doe | johndoe@example.com | Minor injuries |   30 |
+----------+---------+----------+--------------------+---------------+------+
1 row in set (0.00 sec)
```

9. Retrieve the names of persons involved in open cases and closed cases

```
select Victim.name
from Victim join Crime
on Crime.CrimeID=Victim.CrimeID
where Crime.status in ('open','closed');
```

```
mysql> select Victim.name
    ->     from Victim join Crime
    ->     on Crime.CrimeID=Victim.CrimeID
    ->     where Crime.status in ('open','closed');
+---------------+
| name          |
+---------------+
| John Doe      |
| Alice Johnson |
+---------------+
2 rows in set (0.00 sec)
```

10. List incident types where there are persons aged 30 or 35 involved.

```
select IncidentType
from Victim join Crime
on Victim.CrimeID=Crime.CrimeID
where age in(30,35);
```

```
mysql>     select IncidentType
    ->     from Victim join Crime
    ->     on Victim.CrimeID=Crime.CrimeID
    ->     where age in(30,35);
+--------------+
| IncidentType |
+--------------+
| Robbery      |
+--------------+
1 row in set (0.00 sec)
```

11. Find persons involved in incidents of the same type as 'Robbery'.

```
select name
from Crime join Victim on Victim.CrimeID = Crime.CrimeID
where IncidentType='Robbery';
```

```
mysql> select name
    ->          from Crime join Victim on Victim.CrimeID = Crime.CrimeID
    ->          where IncidentType='Robbery';
+----------+
| name     |
+----------+
| John Doe |
+----------+
1 row in set (0.00 sec)
```

12. List incident types with more than one open case.

```
select IncidentType
from Crime
where Status='open'
group by IncidentType
having count(*) >1;
```

```
mysql>     select IncidentType
    ->     from Crime
    ->     where Status='open'
    ->     group by IncidentType
    ->     having count(*) >1;
Empty set (0.00 sec)
```

13. List all incidents with suspects whose names also appear as victims in other incidents.

select * from Crime
join (select CrimeID
from Suspect where exists (select 1
from Victim where Victim.CrimeID=Suspect.CrimeID
and Victim.name=Suspect.name)
)as Subquery on Crime.CrimeID = Subquery.CrimeID;

```
mysql>     select * from Crime
    ->     join (select CrimeID
    ->     from Suspect where exists (select 1
    ->     from Victim where Victim.CrimeID=Suspect.CrimeID
    ->     and Victim.name=Suspect.name)
    ->     )as Subquery on Crime.CrimeID = Subquery.CrimeID;
Empty set (0.00 sec)
```

14. Retrieve all incidents along with victim and suspect details.

SELECT crime.*,
Victim.Name AS VictimName,Victim.VictimID, Victim.ContactInfo, Victim.Injuries,
Suspect.Name AS SuspectName,Suspect.SuspectID, Suspect.Description,
Suspect.CriminalHistory
from Crime
join Victim on Victim.CrimeID=Crime.CrimeID
join Suspect on Suspect.CrimeID=Crime.CrimeID;

```
mysql> SELECT crime.*,
    ->        Victim.Name AS VictimName,Victim.VictimID, Victim.ContactInfo, Victim.Injuries,
    ->        Suspect.Name AS SuspectName,Suspect.SuspectID, Suspect.Description, Suspect.CriminalHistory
    ->        from Crime
    ->        join Victim on Victim.CrimeID=Crime.CrimeID
    ->        join Suspect on Suspect.CrimeID=Crime.CrimeID
    -> ;
```

CrimeID	IncidentType	IncidentDate	Location	Description	Status	VictimName	VictimID	ContactInfo	Injuries	SuspectName	SuspectID	Description	CriminalHistory
1	Robbery	2023-09-15	123 Main St, Cityville	Armed robbery at a convenience store	Open	John Doe	1	johndoe@example.com	Minor injuries	Robber 1	1	Armed and masked robber	Previous robbery convictions
2	Homicide	2023-09-20	456 Elm St, Townsville	Investigation into a murder case	Under Investigation	Jane Smith	2	janesmith@example.com	Deceased	Unknown	2	Investigation ongoing	NULL
3	Theft	2023-09-10	789 Oak St, Villagetown	Shoplifting incident at a mall	Closed	Alice Johnson	3	alicejohnson@example.com	None	Suspect 1	3	Shoplifting suspect	Prior shoplifting arrests

3 rows in set (0.00 sec)

15. Find incidents where the suspect is older than any victim.

```
alter table Suspect add age int;
desc Suspect;
update Suspect set age='37' where SuspectID='1';
update Suspect set age='46' where SuspectID='2';
update Suspect set age='35' where SuspectID='3';
select * from Suspect

select * from Crime
join Victim on Crime.CrimeID=Victim.CrimeID
join Suspect on Crime.CrimeID=Suspect.CrimeID
where Suspect.age > Victim.age;
```

16. Find suspects involved in multiple incidents.

```
select Name, count(distinct CrimeID) AS IncidentCount
from Suspect
group by  Name
having count(distinct CrimeID) > 1;
```

```
mysql> select Name, count(distinct CrimeID) AS IncidentCount
    ->  from Suspect
    ->  group by  Name
    ->  having count(distinct CrimeID) > 1;
Empty set (0.00 sec)
```

17. List incidents with no suspects involved

```
select * from crime
left join Suspect on Crime.CrimeID = Suspect.CrimeID
WHERE Suspect.SuspectID IS NULL;
```

```
mysql>     select * from crime
    ->     left join Suspect on Crime.CrimeID = Suspect.CrimeID
    ->  WHERE Suspect.SuspectID IS NULL;
Empty set (0.00 sec)
```

18.List all cases where at least one incident is of type 'Homicide' and all other incidents are of type 'Robbery'

```
   select Crime.*
from Crime
where exists (
 select 1
 from Crime
 where CrimeID = Crime.CrimeID and IncidentType = 'Homicide'
)
and not exists (
 select 1
 from Crime
 where CrimeID = Crime.CrimeID and IncidentType != 'Robbery'
 );
```

```
mysql> select Crime.*
    -> from Crime
    -> where exists (
    ->  select 1
    ->  from Crime
    ->  where CrimeID = Crime.CrimeID and IncidentType = 'Homicide'
    -> )
    -> and not exists (
    ->  select 1
    ->  from Crime
    ->  where CrimeID = Crime.CrimeID and IncidentType != 'Robbery'
    ->  );
Empty set (0.00 sec)
```

19. Retrieve a list of all incidents and the associated suspects, showing suspects for each incident, or 'No Suspect' if there are none.

```
SELECT Crime.*, COALESCE(Suspect.Name, 'No Suspect') AS SuspectName
FROM Crime
LEFT JOIN Suspect  ON Crime.CrimeID = Suspect.CrimeID;
```

```
mysql> SELECT Crime.*, COALESCE(Suspect.Name, 'No Suspect') AS SuspectName
    -> FROM Crime
    -> LEFT JOIN Suspect  ON Crime.CrimeID = Suspect.CrimeID;
+---------+--------------+--------------+-----------------------+-------------------------------------+----------------------+-------------+
| CrimeID | IncidentType | IncidentDate | Location              | Description                         | Status               | SuspectName |
+---------+--------------+--------------+-----------------------+-------------------------------------+----------------------+-------------+
|       1 | Robbery      | 2023-09-15   | 123 Main St, Cityville | Armed robbery at a convenience store | Open                 | Robber 1    |
|       2 | Homicide     | 2023-09-20   | 456 Elm St, Townsville | Investigation into a murder case     | Under
Investigation | Unknown     |
|       3 | Theft        | 2023-09-10   | 789 Oak St, Villagetown | Shoplifting incident at a mall       | Closed               | Suspect 1   |
+---------+--------------+--------------+-----------------------+-------------------------------------+----------------------+-------------+
3 rows in set (0.00 sec)
```

20. List all suspects who have been involved in incidents with incident types 'Robbery' or 'Assault

```
SELECT DISTINCT Name
FROM Suspect
WHERE CrimeID IN (
 SELECT CrimeID
 FROM Crime
 WHERE IncidentType IN ('Robbery', 'Assault')
 );
```