INVESTIGATION OFFICE

# Investigation Office Database

Authors:

Reem Saleh Saeed Al Malki
Aasyel Mashhour Al Amoudi
Haifa Messnad Al Thobaiti
Sara Mustafa Al Shaikh

# Contents

# PART I: Analysis

## 1. Problem Definition and Data Requirements

### 1.1 Problem Definition

Every successful establishment must have a database to manage the system, and control and maintain data. Therefore, the database is the cornerstone of every establishment specifically in the Investigation Office. Investigation Office demands a management system to keep all investigators' data, including their rank, the cases they are working on, and everything relative to their cases such as case type, witnesses, accused, and lawyer of the accused. In addition, all the case requirements data is included in the database and used to link related data sets together.

As mentioned above, the primary purpose of this database is to obtain information about detectives and case information and show the average number of successful and unsuccessful cases.

### 1.2 Data Requirements

**1. Accused**

1. Each accused has a name (first, middle, last), unique ID number, gender, address, communication(Email and phone number) and arrest status(arrest or not).

**2. Branch**

1. Each branch has a unique number and phone number for communication.
2. Each branch is located in a city.

**3. Case**

1. Each case has a unique number, and accusation (ex: steal, killing, injustice).
2. Each case has actions taken by the detective (reservation, questioning, hearing the witness, secondment of experts, release, arrest).
3. Each case has name that's determined the department name.
4. Each case has number of evidence.

**4. Detective**

1. Each detective has a name (first, middle, last), unique id number, gender, address, commination (phone number and email),salary.
2. Each detective takes action for a case .

**5. Department**

1. Each department has a unique name(ex: criminal, civil investigation), a unique number, a location),

**6. Witnesse**

1. Each witness has a name (first, middle, last), unique id number, address, gender, and communication (Email and phone number )
2. It is required a witness testimony.

**7. Lawyer**

1. Each lawyer has a name (first, middle, last), unique id number, address, gender, and communication (Email and phone number ) specialization.

**8. Suspect**

1. Each suspect has a name (first, middle, last), unique id number, address, gender, and communication (Email and phone number), and indication of suspicion.

**9. Clerk**

1. Each clerk has a name (first, middle, last), unique id number, address, gender, and communication (Email and phone number ) and salary .
2. It is required to write the accused and victim statement and track the current number of statements per week that a clerk collecting on each case.

**10. Victim**

1. Each Victim has a name (first, middle, last), unique id number, address, gender, and communication (Email and phone number ).

## 1.3 Business Rules

Our system follows some of the rules to facilitate Investigation Office and avoid mistakes in all its parts and services:

1- Each supervisor supervises many Supervisee in the detective office.
2- Each Branch contains more than one department, and each department founds in all branches.
3- A department controls several cases, and each case has one department control it.
4- A detective can manage one department, and each department has one detective manager.
5- Each detective work in one department, and each department has many detectives work on it.
6- A detective work on several cases and a case has only one detective.
7- Each clerk has more than one case and the case has only one clerk.
8- A case can have more than one witness and vice versa.
9- A case can have more than one suspect and vice versa.
10- A case can have more than one victim and vice versa.
11- A case can have more than one accused and vice versa.
12- A lawyer can have more than one defendant, and every defendant has a lawyer. (Defendant means suspect and victim and accused)

## 1.4 Intended Output of the system

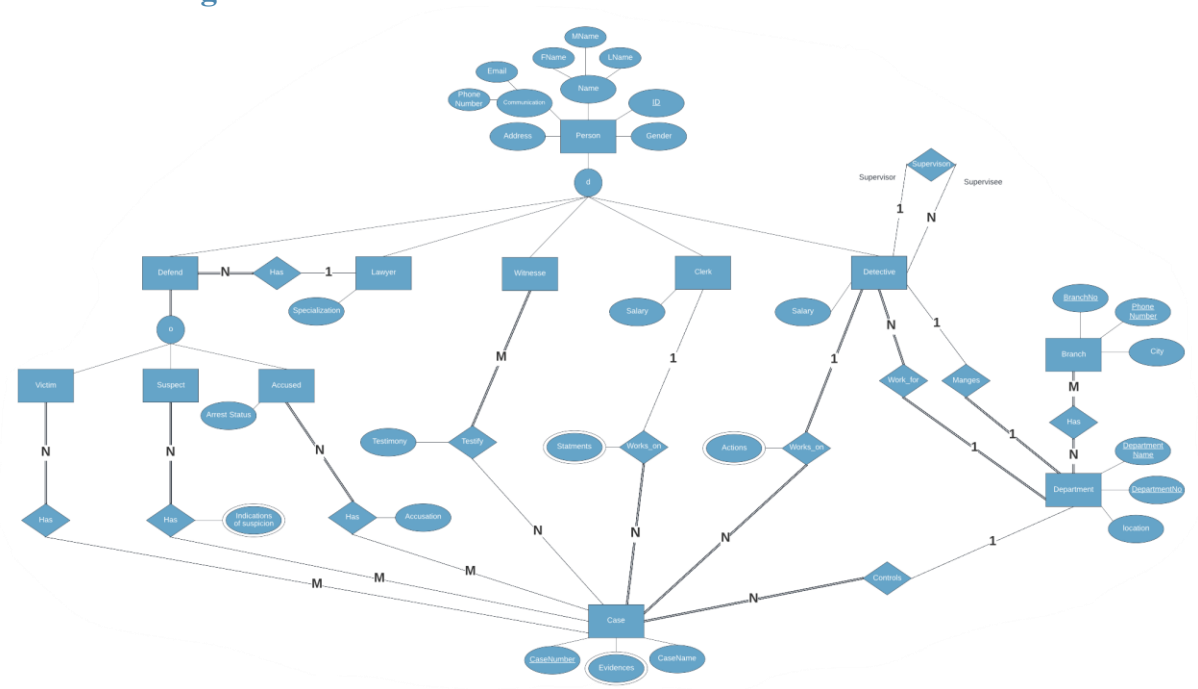
**We can query the investigation office DB as follows:**

1. List the cases with maximum number of witnesses.
2. List the detectives who are responsible of cases with the highest number of witnesses.
3. List number of victims, accused, suspects in the system.
4. List the department with the highest number of detectives.
5. List the cases that have at least 3 evidences.

# PART II: DB DEISGN

## 2 ER Diagram Design

### 1.1 ER diagram



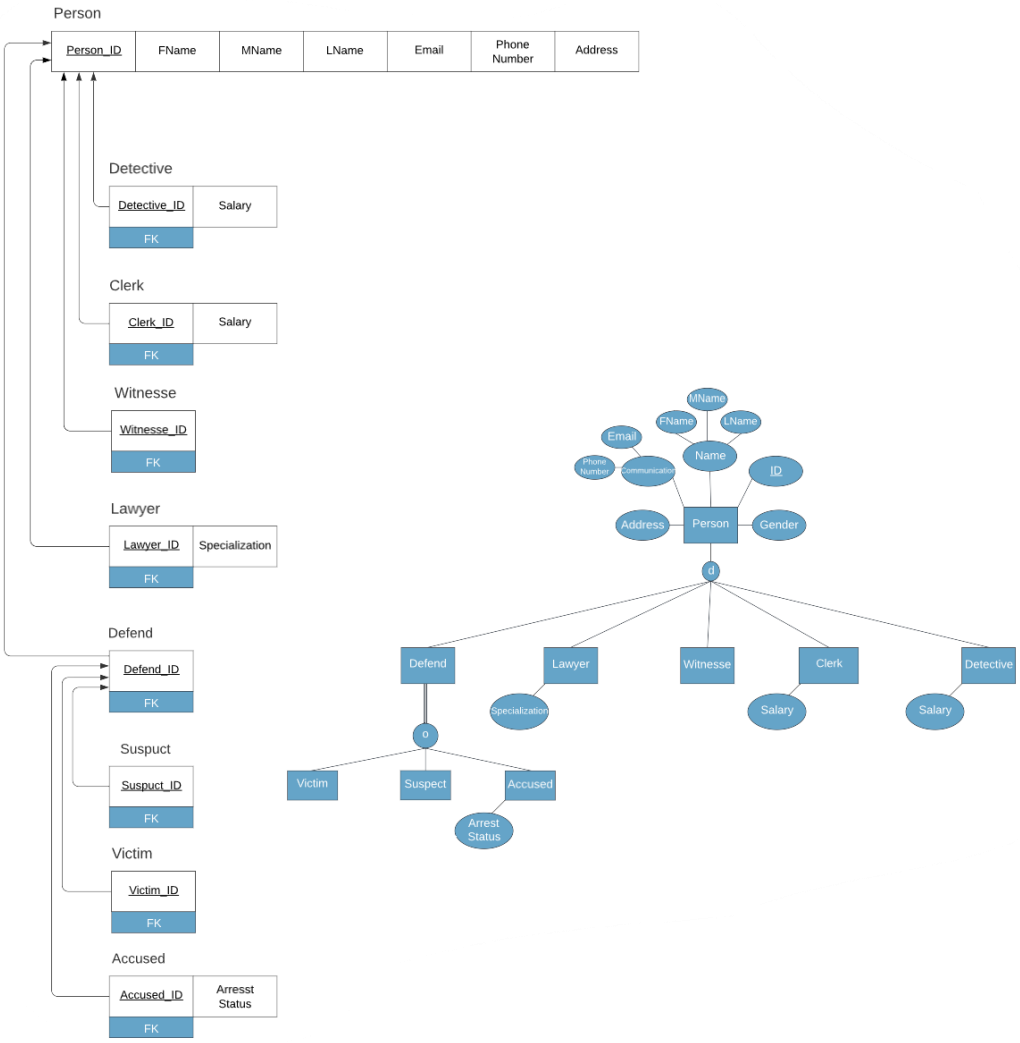

### 2.2 Design of Business Rules

| Business Rule | Design Decisions | Justification (if any) |
|---|---|---|
| Each supervisor supervises many supervisee in the detective office. | 1: N binary recursive relationship | |
| Each person must have a specialization. | Superclass/subclass relationship with disjoint Ness and completeness constraints | |
| Each department control many cases. | 1: N binary relationship between DEPARTMENT and CASE | One Department control many case (partially participation). The case must be controlled by department (total participation). |
| Each defendants must have lawyer. | 1: N binary relationship between DEFEND and LAWYER | A lawyer has many defendants (Partially participation). defendants has only one lawyer. (Total participation) |
| A detective work in one department. | N:1 binary relationship between DETECTIVE and DEPARTMENT | A detective must work in one department AND each department must has many |

| | | |
|---|---|---|
| | | detectives. (Full participation on both side) |
| A detective must work for many case. | 1: N binary relationship between DETECTIVE and CASE | A detective work on many cases.<br>The case has one detective works on it.<br>(Total participation on both side) |
| Each clerk work on at least one case. | 1: N binary relationship between CLERK and CASE | A clerk work on many case (partially participation).<br>The case must write by single clerk (total participation). |
| A case can have more than one witness. | N:M binary relationship between WITNESS and CASE | A case testifying by many witness (partially participation).<br>Each witness testify a case (total participation). |
| Each branch contains more than one department. | M: N binary relationship between BRANCH and DEPARTMENT | Each branch must have more than one department.<br>Each Department must be found in all branches.<br>(Total participation on both sides) |
| Each defend must have a specialization. | Superclass/subclass relationship with overlapping constraint. | |
| Each case can have more than one accused. | M: N binary relationship between ACCUSED and CASE | Each case can have more than one accused (partially participation).<br>Accused may have more than one case. (Total participation). |
| Each case can have more than one suspect. | M: N binary relationship between SUSPECT and CASE | Each case can have more than one Suspect (partially participation).<br>A suspect may have more than one case. (Total participation). |
| Each case can have more than one victim. | M: N binary relationship between VICTIM and CASE | Each case can have more than one victim (Partially participation).<br>A victim may have more than one case. (Total participation). |

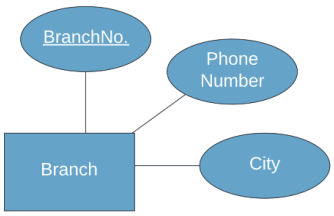
# 3 ER-to-logical schema mapping

## 3.1 Mapping of Regular Entity Types



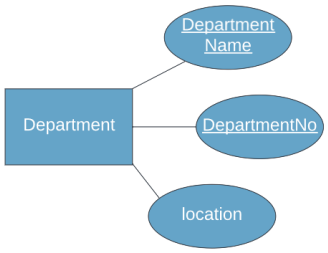
Branch

| BranchNo | ‣ Phone Number | City |
|---|---|---|



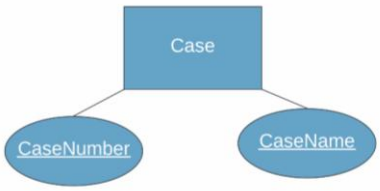

Department

| DepartmentNo | ‣ Department Name | location |
|---|---|---|




Case

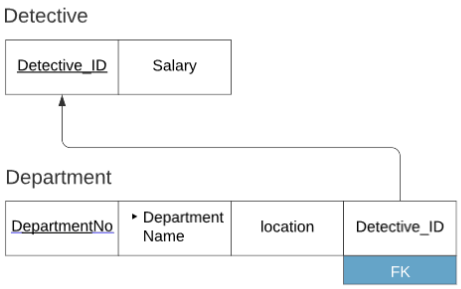
| CaseNumber | CaseName |
|---|---|



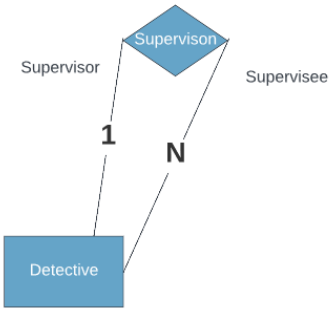
## 3.2 Mapping of Weak Entity Types

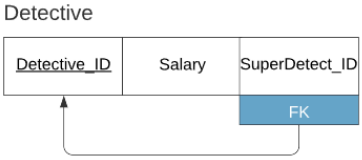
The system dose not have weak entity types.

## 3.3 Mapping of binary 1-1 relationship types


## 3.4 Mapping of binary 1-N relationship types




Detective

| Detective_ID | Salary | SuperDetect_ID | DepartmentNo |
|---|---|---|---|
| FK | | FK | FK |

Department

| DepartmentNo | ‣ Department Name | location | Detective_ID |
|---|---|---|---|
| | | | FK |


Department

| DepartmentNo | ‣ Department Name | location | Detective_ID |
|---|---|---|---|
| | | | FK |

Case

| CaseNumber | CaseName | DepartmentNo |
|---|---|---|
| | | FK |


Case

| CaseNumber | CaseName | DepartmentNo | Clerk_ID |
|---|---|---|---|
| | | FK | FK |

Clerk

| Clerk_ID | Salary |
|---|---|


Detective

| Detective_ID | Salary | SuperDetect_ID | DepartmentNo |
|---|---|---|---|
| PK | | FK | FK |

Case

| CaseNumber | CaseName | DepartmentNo | Clerk_ID | Detective_ID |
|---|---|---|---|---|
| | | FK | FK | FK |

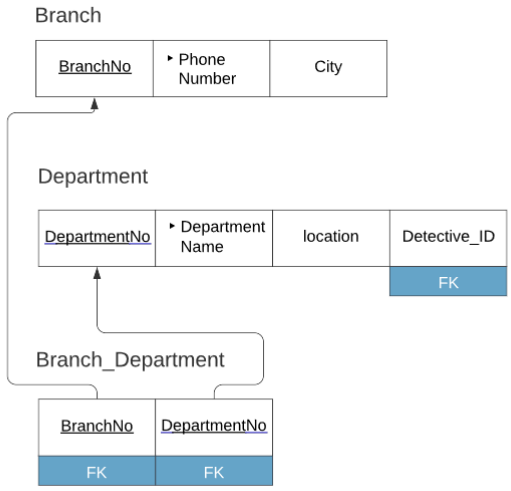

Defend

| Defend_ID | Lawyer_ID |
|---|---|
| | FK |

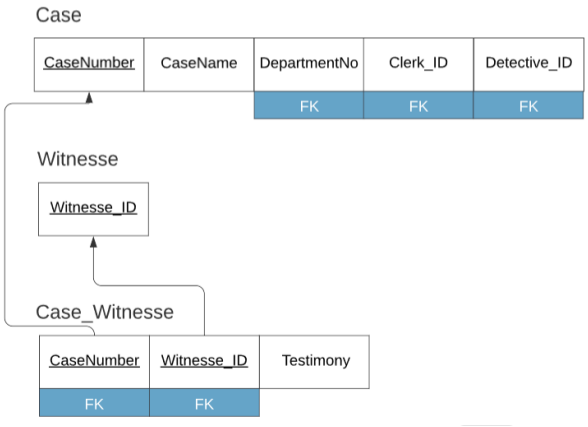
Lawyer

| Lawyer_ID | Specialization |
|---|---|

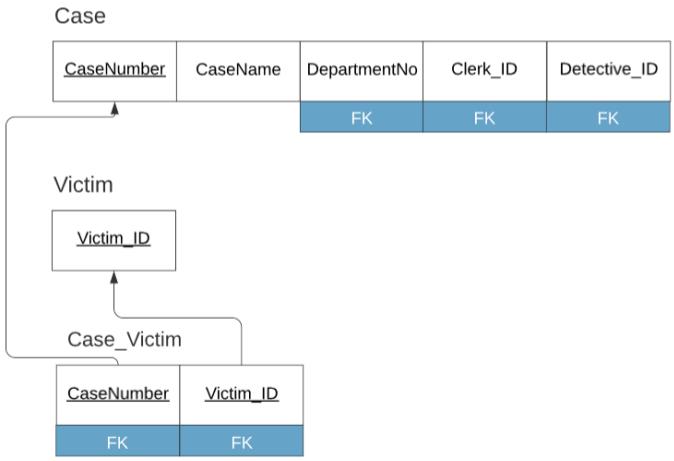
## 3.5 Mapping of binary M-N relationship types

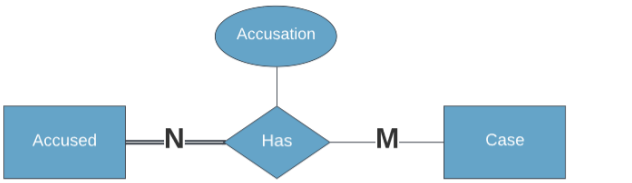



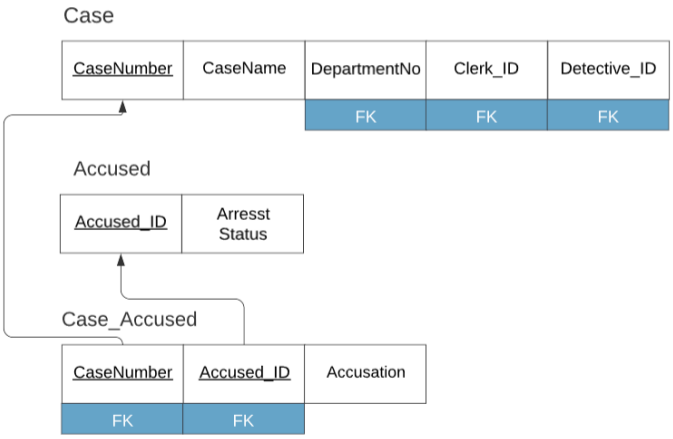


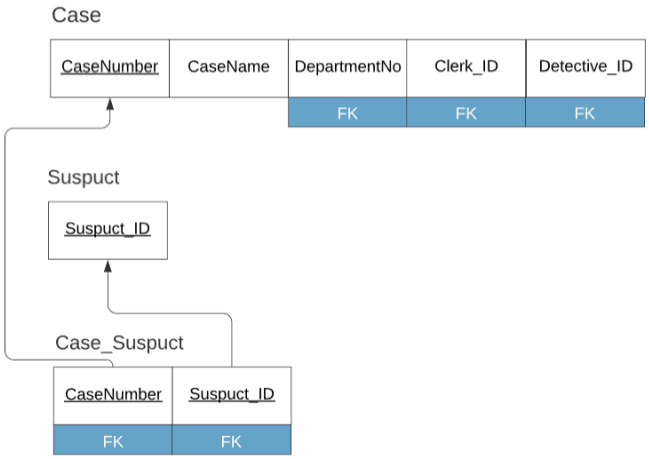



Victim
N
Has
M
Case

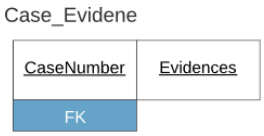

Case
CaseNumber
CaseName
DepartmentNo
Clerk_ID
Detective_ID
FK
FK
FK
Victim
Victim_ID
Case_Victim
CaseNumber
Victim_ID
FK
FK

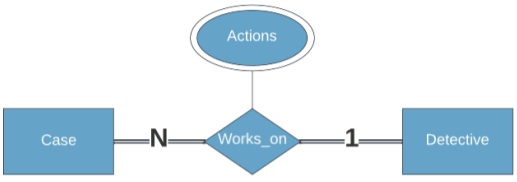

Accusation
Accused
N
Has
M
Case

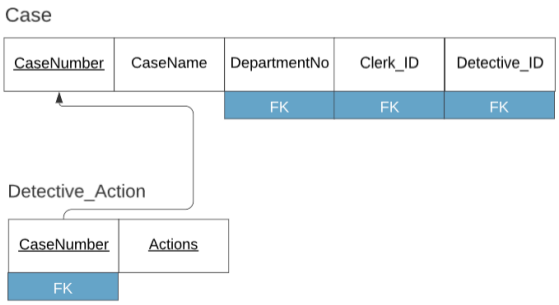

Case
CaseNumber
CaseName
DepartmentNo
Clerk_ID
Detective_ID
FK
FK
FK
Accused
Accused_ID
Arresst Status
Case_Accused
CaseNumber
Accused_ID
Accusation
FK
FK

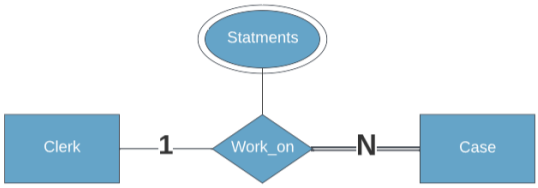



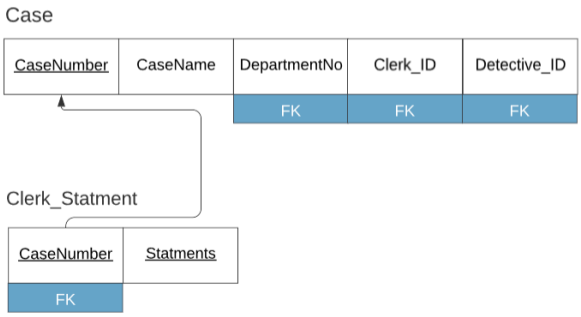

## 3.6 Mapping of multivalued attributes

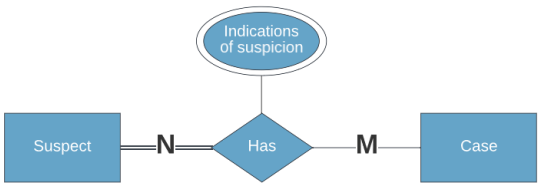



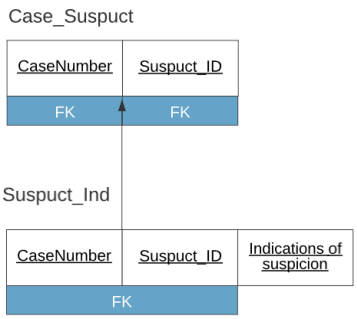





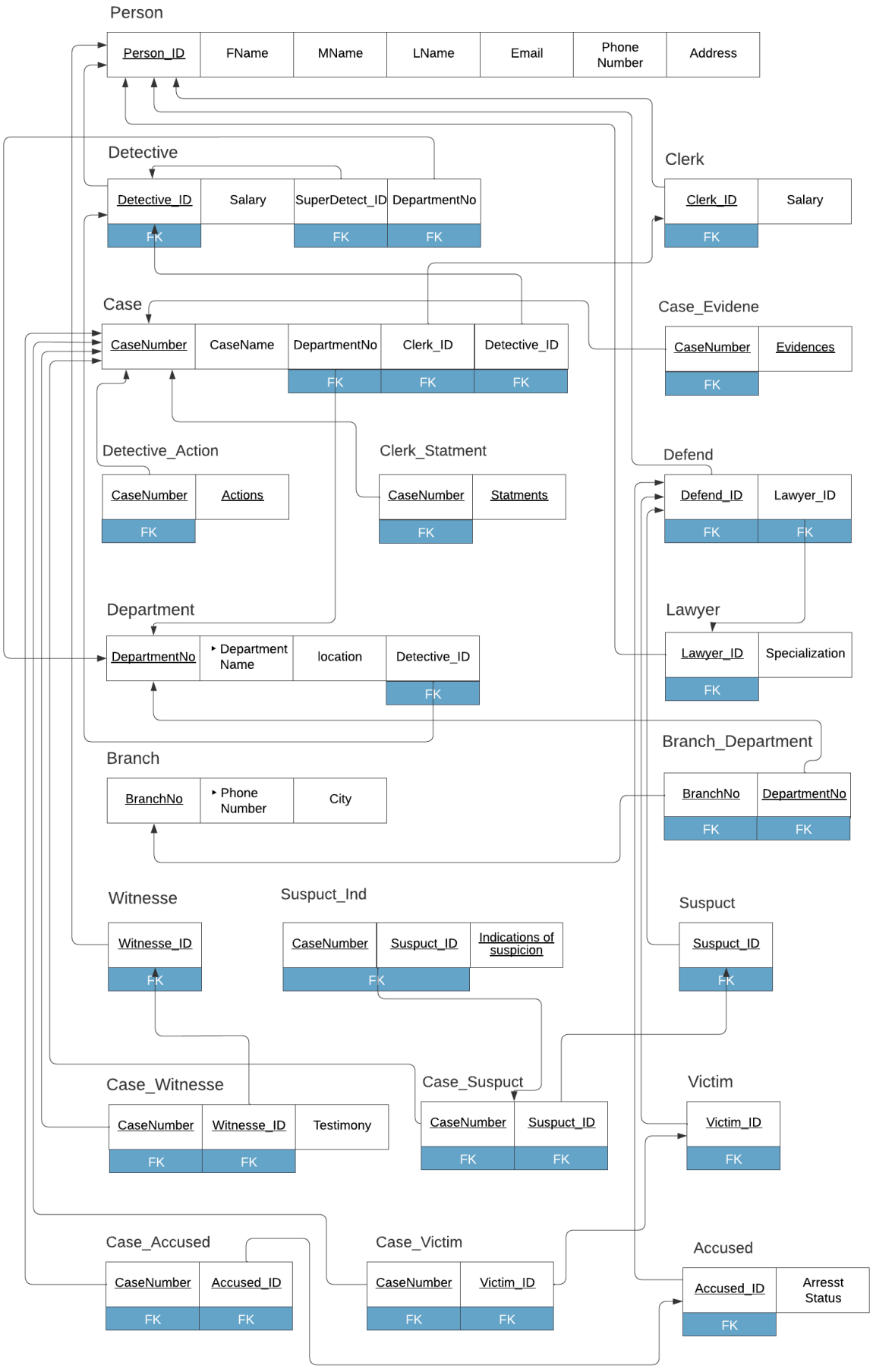




## 3.7 Mapping of n-ary relationship types

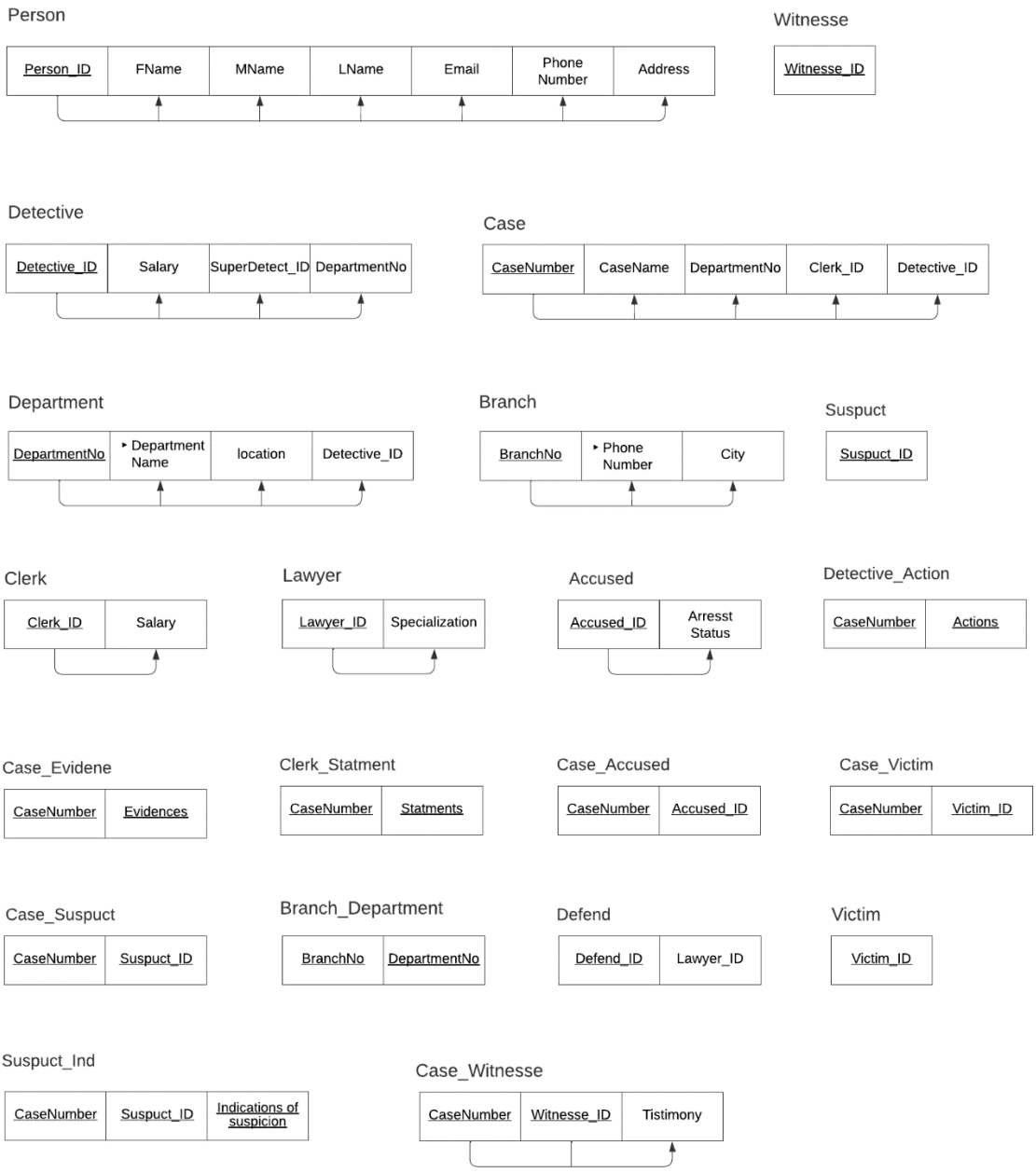
The system does not have n-ary relationship types.

## 3.8 Schema Diagram



# 4 Normalization

## Functional dependence



## 4.1 First Normal Form

Person

| Person_ID | FName | MName | LName | Email | Phone Number | Address |
|---|---|---|---|---|---|---|

Table on first normal form

**Reason**: Because there is no multivalued attribute nested relation.

Detective

| Detective_ID | Salary | SuperDetect_ID | DepartmentNo |
|---|---|---|---|

Table on first normal form

**Reason**: Because there is no multivalued attribute nested relation.

Case

| CaseNumber | CaseName | DepartmentNo | Clerk_ID | Detective_ID |
|---|---|---|---|---|

Table on first normal form

**Reason**: Because there is no multivalued attribute nested relation.

Department

| DepartmentNo | ▸ Department Name | location | Detective_ID |
|---|---|---|---|

Table on first normal form

**Reason**: Because there is no multivalued attribute nested relation.

Branch

| BranchNo | ▸ Phone Number | City |
|---|---|---|

Table on first normal form

**Reason**: Because there is no multivalued attribute nested relation.

Witnesse

| Witnesse_ID |
|---|

Table on first normal form

**Reason**: Because there is no multivalued attribute nested relation.

Clerk

| Clerk_ID | Salary |
|---|---|

Table on first normal form

**Reason**: Because there are no multivalued attribute nested relation .

Lawyer

| Lawyer_ID | Specialization |
|---|---|

Table on first normal form

**Reason**: Because there are no multivalued attribute nested relation.

Defend

| Defend_ID | Lawyer_ID |
|---|---|

Table on first normal form

**Reason**: Because there is no multivalued attribute nested relation.

Accused

| Accused_ID | Arresst Status |
|---|---|

Table on first normal form

**Reason**: Because there are no multivalued attribute nested relation .

Victim

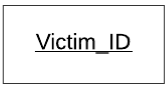



Table on first normal form

**Reason**: Because there is no multivalued attribute nested relation.

Suspuct

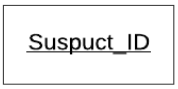



Table on first normal form

**Reason**: Because there is no multivalued attribute nested relation**.**

Case_Evidene

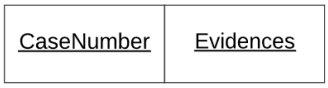



Table on first normal form

**Reason**: Because there is no multivalued attribute nested relation.

Clerk_Statment

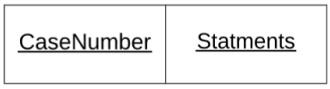



Table on first normal form

**Reason**: Because there is no multivalued attribute nested relation.

Detective_Action

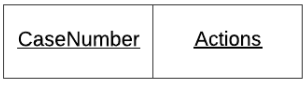



Table on first normal form

**Reason**: Because there is no multivalued attribute nested relation.

Branch_Department

| BranchNo | DepartmentNo |
|---|---|

Table on first normal form

**Reason**: Because there is no multivalued attribute nested relation.

Suspuct_Ind

| CaseNumber | Suspuct_ID | Indications of suspicion |
|---|---|---|

Table on first normal form

**Reason**: Because there is no multivalued attribute nested relation.

Case_Witnesse

| CaseNumber | Witnesse_ID | Testimony |
|---|---|---|

Table on first normal form

**Reason**: Because there is no multivalued attribute nested relation.

Case_Accused

| CaseNumber | Accused_ID |
|---|---|

Table on first normal form

**Reason**: Because there is no multivalued attribute nested relation.

Case_Victim

| CaseNumber | Victim_ID |
|---|---|

Table on first normal form

**Reason**: Because there is no multivalued attribute nested relation.

Case_Suspuct

| CaseNumber | Susspuct_ID |
|---|---|

Table on first normal form

**Reason**: Because there is no multivalued attribute nested relation.

## 4.2 Second Normal Form

Person

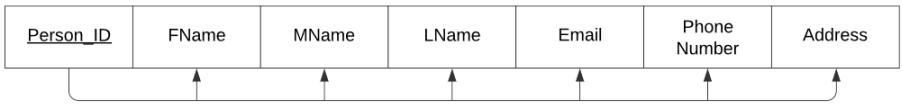



Table on second normal form

**Reason**: because functional dependence have not a partial functional dependency on PK .

Case

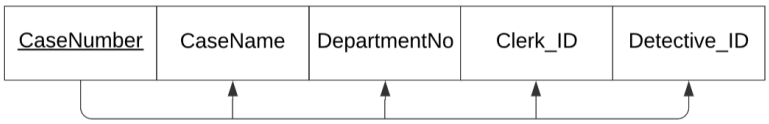



Table on second normal form

**Reason**: because functional dependence have not a partial functional dependency on PK .

Detective

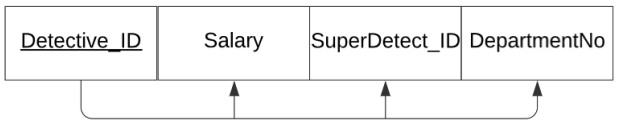



Table on second normal form

**Reason**: because functional dependence have not a partial functional dependency on PK .

Detective_Action

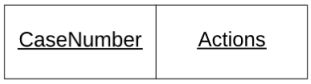



Table on second normal form

**Reason**: because functional dependence have not a partial functional dependency on PK .

Clerk_Statment

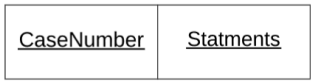



Table on second normal form

**Reason**: because functional dependence have not a partial functional dependency on PK .

Department

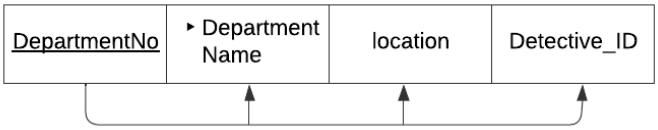



Table on second normal form

**Reason**: because functional dependence have not a partial functional dependency on PK .

Branch

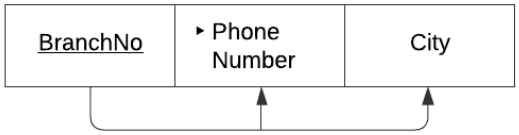



Table on second normal form

**Reason**: because functional dependence have not a partial functional dependency on PK .

Witnesse

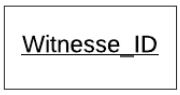



Table on second normal form

**Reason**: because functional dependence have not a partial functional dependency on PK .

Suspuct_Ind

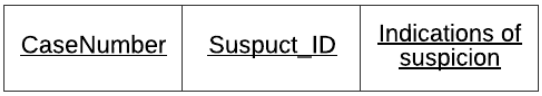



Table on second normal form

**Reason**: because functional dependence have not a partial functional dependency on PK .

Case_Witnesse

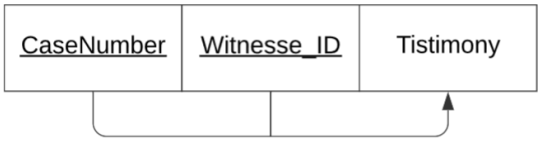



Table on second normal form

**Reason**: because functional dependence have not a partial functional dependency on PK

Case_Suspuct

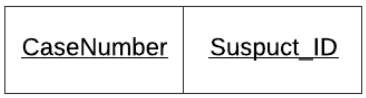



Table on second normal form

**Reason**: because functional dependence have not a partial functional dependency on PK .

Case_Accused

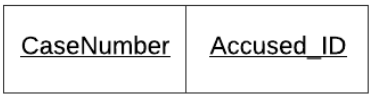



Table on second normal form

**Reason**: because functional dependence have not a partial functional dependency on PK .

Case_Victim

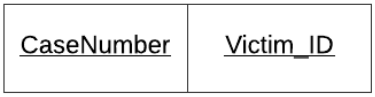



Table on second normal form

**Reason**: because functional dependence have not a partial functional dependency on PK .

Clerk

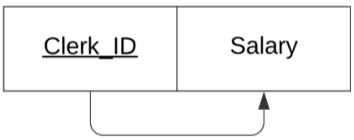



Table on second normal form

**Reason**: because functional dependence have not a partial functional dependency on PK .

Case_Evidene

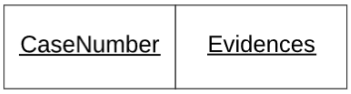



Table on second normal form

**Reason**: because functional dependence have not a partial functional dependency on PK .

Defend

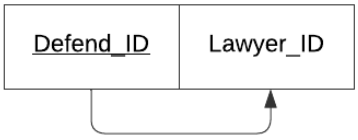



Table on second normal form

**Reason**: because functional dependence have not a partial functional dependency on PK .

Lawyer

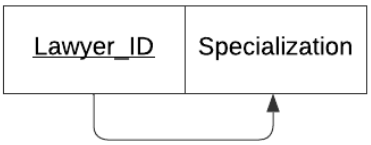



Table on second normal form

**Reason**: because functional dependence have not a partial functional dependency on PK .

Branch_Department

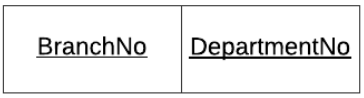



Table on second normal form

**Reason**: because functional dependence have not a partial functional dependency on PK .

Suspuct

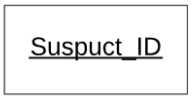



Table on second normal form

**Reason**: because functional dependence have not a partial functional dependency on PK .

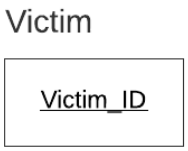



Table on second normal form

**Reason**: because functional dependence have not a partial functional dependency on PK .

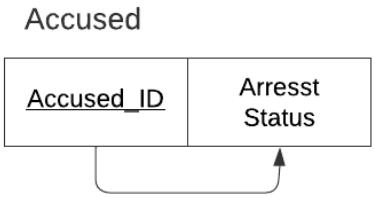



Table on second normal form

**Reason**: because functional dependence have not a partial functional dependency on PK .

## 4.3 Third Normal Form

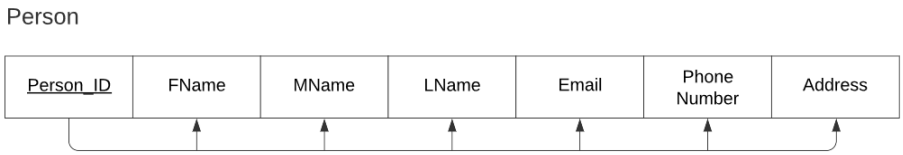



Table on third normal form

**Reason**: because functional dependence have not transitive functional dependency on PK .

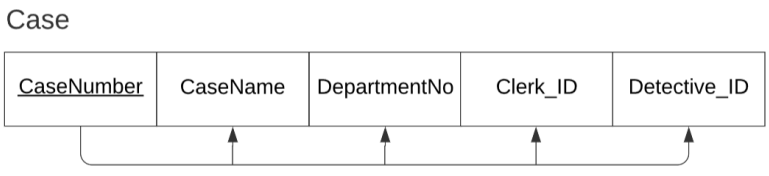



Table on third normal form

**Reason**: because functional dependence have not transitive functional dependency on PK .

Detective

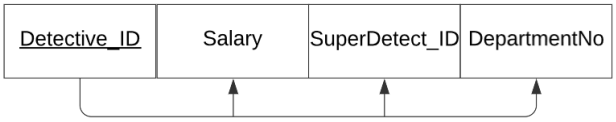



Table on third normal form

**Reason**: because functional dependence have not transitive functional dependency on PK .

Detective_Action

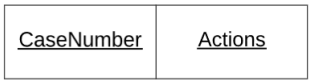



Table on third normal form

**Reason**: because no functional dependence on PK .

Clerk_Statment

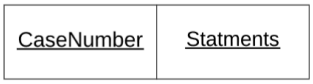



Table on third normal form

**Reason**: because no functional dependence on PK .

Department

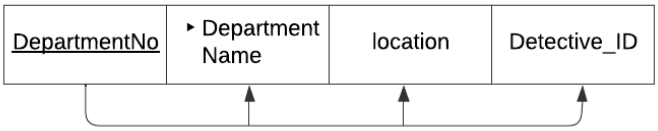



Table on third normal form

**Reason**: because functional dependence have not transitive functional dependency on PK .

Branch

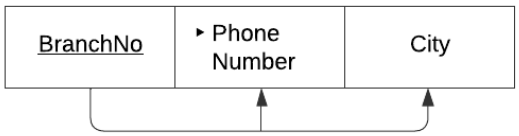



Table on third normal form

**Reason**: because functional dependence have not transitive functional dependency on PK .

Witnesse

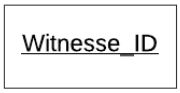



Table on third normal form

**Reason**: because no functional dependence on PK .

Suspuct_Ind

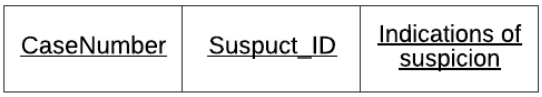



Table on third normal form

**Reason**: because no functional dependence on PK .

Case_Witnesse

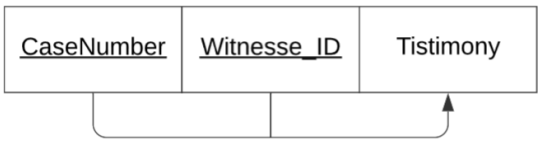



Table on third normal form

**Reason**: because functional dependence have not transitive functional dependency on PK.

Case_Suspuct

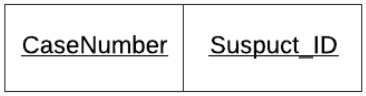



Table on third normal form

**Reason**: because no functional dependence on PK .

Case_Accused

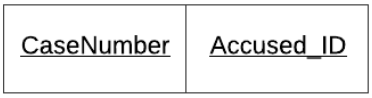



Table on third normal form

**Reason**: because no functional dependence on PK .

Case_Victim

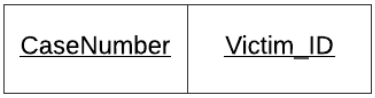



Table on third normal form

**Reason**: because no functional dependence on PK .

Clerk

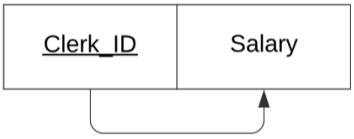



Table on third normal form

**Reason**: because functional dependence have not transitive functional dependency on PK .

Case_Evidene

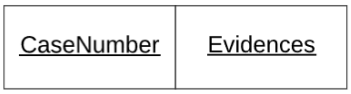



Table on third normal form

**Reason**: because no functional dependence on PK .

Defend

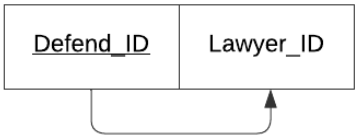



Table on third normal form

**Reason**: because functional dependence have not transitive functional dependency on PK .

Lawyer

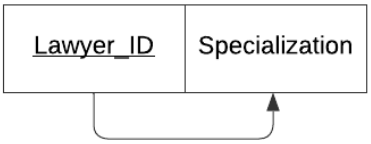



Table on third normal form

**Reason**: because functional dependence have not transitive functional dependency on PK .

Branch_Department

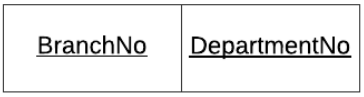



Table on third normal form

**Reason**: because no functional dependence on PK .

Suspuct

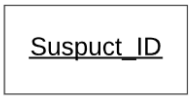



Table on third normal form

**Reason**: because no functional dependence on PK .

Victim

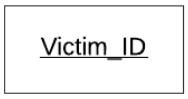



Table on third normal form

**Reason**: because no functional dependence on PK .

Accused

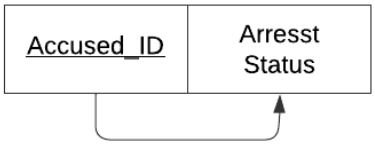
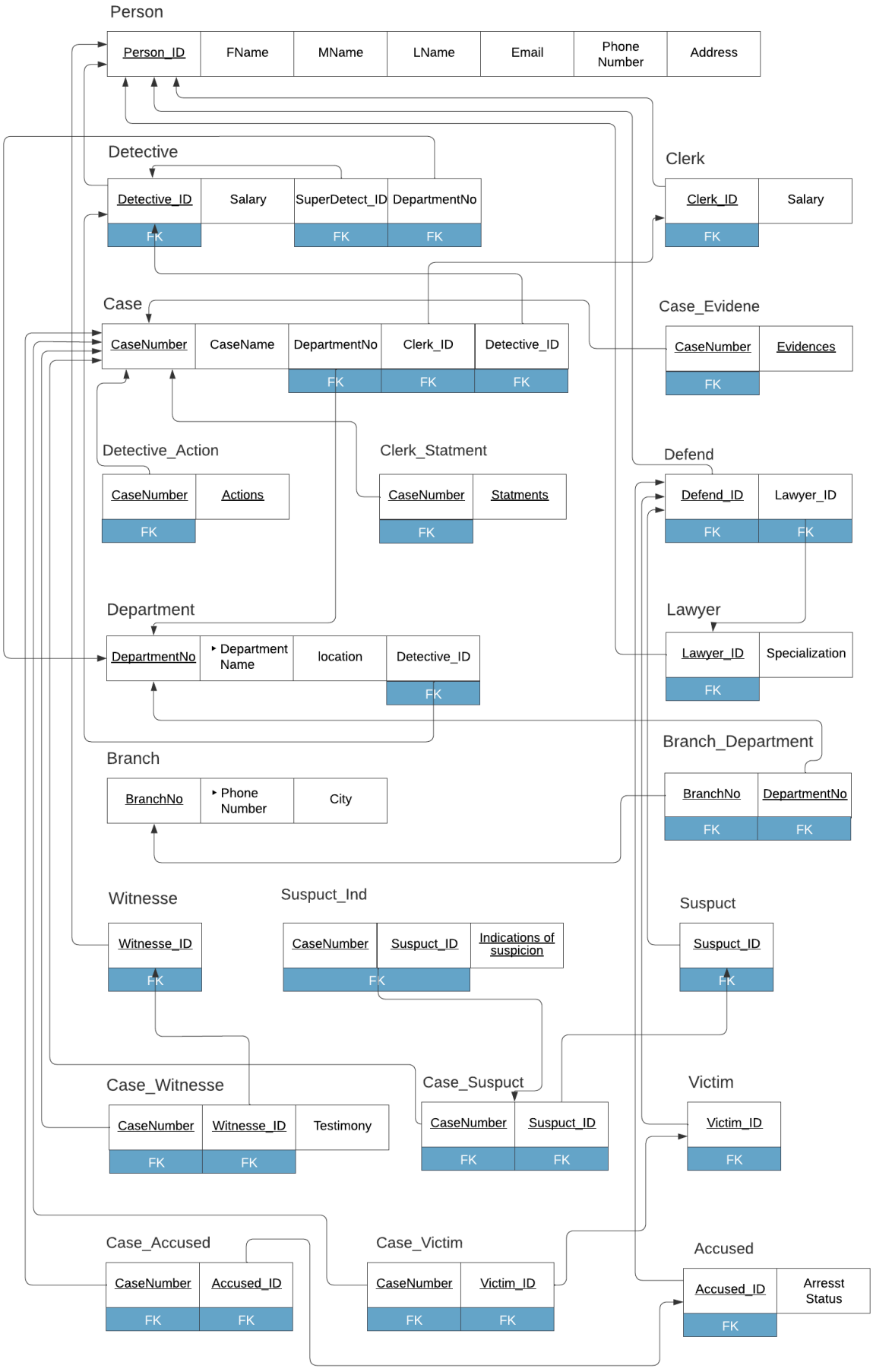



Table on third normal form

**Reason**: because functional dependence have not transitive functional dependency on PK .

# 5 Final DB Schema Diagram



# PART III: IMPLEMENTATION

## 6 Table Creation Script

### 6.1 Person Table

```
Create table Person( Person_ID number(30) ,FName varchar2(255) not null,
MName varchar2(255)not null,
Lname varchar2(255) not null,Gender varchar2(255) not null,Address
varchar2(255) not null,
PhoneNumber number(20)not null,Email varchar2(255) not null,
constraint PK_Person PRIMARY KEY(Person_ID));
```

### 6.2 Case Table

```
Create table case(CaseNumber number(30)  ,CaseName varchar2(255),
DepartmentNo number(30)not null ,Clerk_id number(30)not null,detective_id
number(30)not null,
constraint PK_Case  primary key (CaseNumber),
constraint FK_6_2_1 foreign key (DepartmentNo) references
Department(DepartmentNo)ON DELETE CASCADE,
constraint FK_6_2_2  foreign key (Clerk_id) references Clerk(Clerk_id)ON
DELETE CASCADE,
constraint FK_6_2_3 foreign key (detective_id ) references detective
(detective_id) ON DELETE CASCADE);
```

### 6.3 Detective Table

```
Create table Detective(Detective_ID number(30), Salary number(30),
SuperDetective number(30), DepartmentNo Number (30),
constraint PK_Detective primary key(Detective_ID),
constraint FK_6_3_1 foreign key(Detective_ID) references Person(Person_ID)
ON DELETE CASCADE,
constraint FK_6_3_2 foreign key(SuperDetective) references
Detective(Detective_ID) ON DELETE CASCADE,
constraint FK_6_3_3 foreign key(DepartmentNo) references
Department(DepartmentNo) ON DELETE CASCADE);
```

### 6.4 Department Table

```
create Table Department (DepartmentNo number (30), DepartmentName
varchar2(255) not null, Location Varchar2(255),DetectiveID Number (30));

Alter Table Department Add Constraint PK_Department  primary key
(DepartmentNo);

Alter Table Department Add Constraint FK_6_4_1 foreign Key(DetectiveID)
references Detective(Detective_ID) ON DELETE CASCADE;
```

## 6.5 Branch Table

```
create Table Branch ( BranchNum Number(30) ,phonNumber Number(30) not null,
city Varchar2(255) ,constraint FK_Branch primary key (BranchNum));
```

## 6.6 Suspect Table

```
create table Suspect(Suspect_ID number(30),
constraint PK_Suspect primary key(Suspect_ID),
constraint FK_6_6_1 foreign key(Suspect_ID)references Defend(Defend_ID)ON
DELETE CASCADE);
```

## 6.7 Clerk Table

```
create Table Clerk( Clerk_ID number(30) ,Salary number (30),
Constraint Pk_Clerk primary key(Clerk_ID),
Constraint Fk_6_7_1 foreign key(Clerk_ID)references Person(Person_ID) ON
DELETE CASCADE);
```

## 6.8 Victim Table

```
Create table Victim(Victim_ID number(30),
constraint PK_Victim primary key(Victim_ID),
constraint FK_6_8_1 foreign key(Victim_ID)references Defend(Defend_ID) ON
DELETE CASCADE);
```

## 6.9 Lawyer Table

```
create table Lawyer(Lawyer_ID number(30) , Specialization varchar2(255) not
null,
constraint PK_Lawyer primary key(Lawyer_ID),
constraint FK_6_9_1 foreign key(Lawyer_ID)references Person(Person_ID)  ON
DELETE CASCADE);
```

## 6.10 Accused Table

```
Create table Accused (Accused_ID number(10) not null, Arresst_Status
varchar2(255),
constraint PK_Accused primary key(Accused_ID),
constraint FK_6_10_1  foreign key (Accused_ID) references Defend
(Defend_ID)ON DELETE CASCADE);
```

### 6.11 Witness Table

```
create table Witnesse(Witnesse_ID number(30),Tistimony varchar2(255) not
null,
constraint PK_Witnesse  primary key(Witnesse_ID),
constraint FK_6_11_1 foreign key(Witnesse_ID)references Person(Person_ID)
ON DELETE CASCADE);
```

### 6.12 Defend Table

```
create table Defend( Defend_ID number(30) ,Lawyer_ID number(10) not null,
constraint PK_Defend primary key(Defend_ID),
constraint FK_6_12_1 foreign key(Defend_ID) references Person(Person_ID)
ON DELETE CASCADE,
constraint FK_6_12_2 foreign key(Lawyer_ID) references Lawyer(Lawyer_ID) ON
DELETE CASCADE);
```

### 6.13 Detective_Action Table

```
Create table Detective_Action(CaseNumber number(30), Actions varchar2(255),
constraint PK_Detective_Action primary key(CaseNumber, Actions),
constraint FK_6_13_1 foreign key(CaseNumber) references Case(CaseNumber) ON
DELETE CASCADE);
```

### 6.14 Case_Evidence Table

```
Create table Case_Evidence(CaseNumber number(30), Evidence varchar2(255),
constraint PK_Case_Evidence primary key(CaseNumber,Evidence),
constraint FK_6_14_1 foreign key(CaseNumber) references Case(CaseNumber) ON
DELETE CASCADE);
```

### 6.15 Clerk_Statement Table

```
create table Clerk_Statment (CaseNumber Number(30),Statment varchar2(255),
Constraint  PkClerk_Statment primary key (CaseNumber,Statment),
Constraint FK_Clerk_Statment foreign key (CaseNumber)references
case(CaseNumber)On Delete CasCade);
```

### 6.16 Case_Accused Table

```
Create table case_Accused(CaseNumber number(10) not null, Accused_ID
number(10) not null,
constraint PK_Case_Accused primary key(Accused_ID, CaseNumber),
constraint FK_6_16_1 foreign key (Accused_ID) references Accused
(Accused_ID)ON DELETE CASCADE,
constraint FK_6_16_2 foreign key (CaseNumber) references case
(CaseNumber)ON DELETE CASCADE);
```

## 6.17 Case_Victim Table

```
create table Case_Victim(CaseNumber number(10) not null, Victim_ID
number(10) not null,
constraint PK_Case_Victim primary key(Victim_ID, CaseNumber),
constraint FK_6_17_1 foreign key(Victim_ID ) references Victim(Victim_ID)
ON DELETE CASCADE,
constraint FK_6_17_2 foreign key(CaseNumber ) references case(CaseNumber)ON
DELETE CASCADE);
```

## 6.18 Case_Suspect Table

```
Create table Case_Suspect(CaseNumber number(10) not null , Suspect_ID
number(10),
constraint FKPK_Case_Suspuct  primary key(Suspect_ID, CaseNumber),
constraint FK_6_18_1 foreign key(CaseNumber) references case(CaseNumber)ON
DELETE CASCADE,
constraint FK_6_18_2  foreign key(Suspect_ID) references
Suspect(Suspect_ID) ON DELETE CASCADE);
```

## 6.19 Branch_Department Table

```
create Table Branch_Department (DepartmentNo Number (30),BranchNumber
Number (30));

Alter Table Branch_Department Add Constraint PK_Branch_Department primary
key (DepartmentNo,BranchNumber);

Alter Table Branch_Department Add Constraint FK_6_19_1 foreign Key
(DepartmentNo)references  Department (DepartmentNo) ON delete CasCade;

Alter Table Branch_Department Add Constraint FK_6_19_2 foreign Key
(BranchNumber)references  Branch(BranchNum) ON delete CasCade;
```

## 6.20 Suspect_Ind Table

```
Create table Suspect_Ind(CaseNumber number(30), Suspect_ID number(30),
IndicationOfSuspicion varchar2(255),
constraint PK_Suspect_Ind primary key(CaseNumber, Suspect_ID,
IndicationOfSuspicion),
constraint FK_6_20_1 foreign key(CaseNumber, Suspect_ID) references
Case_Suspect (CaseNumber, Suspect_ID) ON DELETE CASCADE);
```

## 6.21 Case_Witnesse Table

```
Create table Case_Witnesse(CaseNumber number(10) not null, Witnesse_ID
number(10) not null,
testimony varchar2(255),
```

```
constraint FKPK_Case_Witnesse  primary key(Witnesse_ID,CaseNumber),
constraint FK_6_21_1 foreign key (Witnesse_ID) references Witnesse
(Witnesse_ID)ON DELETE CASCADE,
constraint FK_6_21_2 foreign key (CaseNumber) references case(CaseNumber)ON
DELETE CASCADE);
```

## 7 Constraints Script

| Business Rule | SQL Script | Table |
|---|---|---|
| Each person must have a specialization. | Alter table person add constraint PK_Person PRIMARY KEY(Person_ID);<br><br>Alter Table Clerk Add Constraint Fk_6_7_1 foreign key(Clerk_ID)references Person(Person_ID) ON DELETE CASCADE;<br><br>Alter Table Detective  Add constraint FK_6_3_1 foreign key(Detective_ID) references Person(Person_ID) ON DELETE CASCADE;<br><br>Alter Table Lawyer Add constraint FK_6_9_1 foreign key(Lawyer_ID)references Person(Person_ID)  ON DELETE CASCADE;<br><br>Alter Table Defend Add constraint FK_6_12_1 foreign key(Defend_ID) references Person(Person_ID)  ON DELETE CASCADE;<br><br>Alter Table  Witnesse Add constraint FK_6_11_1 foreign key(Witnesse_ID)references Person(Person_ID)  ON DELETE CASCADE; | 1-Person<br>2- Clerk<br>3- Detective<br>4-Lawyer<br>5- Defend<br>6- Witnesse |
| Each supervisor supervises many | Alter table Detective add constraint FK_6_3_2 foreign | 1- Detective |

| | | |
|---|---|---|
| supervisee in the detective office. | key(SuperDetective) references Detective(Detective_ID) ON DELETE CASCADE; | |
| Each department control many cases. | Alter Table Department Add Constraint PK_Department primary key (DepartmentNo);<br><br>Alter table Case add constraint FK_6_2_1 foreign key (DepartmentNo) references Department(DepartmentNo)ON DELETE CASCADE; | 1- Department<br>2-case |
| Each defendants must have lawyer. | Alter table Lawyer add constraint PK_Lawyer PRIMARY KEY (Lawyer_ID);<br><br>ALTER TABLE DEFEND ADD CONSTRAINT FK_6_12_2 Foreign Key references Lawyer(Lawyer_ID); | 1-Lawyer<br>2- DEFEND |
| A detective work in one department. | Alter table Department Add Constraint PK_Department primary key (DepartmentNo);<br><br>Alter table Detective add constraint FK_6_3_3 foreign key(DepartmentNo) references Department(DepartmentNo) ON DELETE CASCADE; | 1- Department<br>2- Detective |
| A detective must work for many case. | Alter table Detective add constraint PK_Detective primary key(Detective_ID);<br><br>Alter table Case add constraint FK_6_2_3 foreign key (detective_id ) references detective (detective_id) ON DELETE CASCADE); | 1- Detective<br>2- Case |

| | | |
|---|---|---|
| Each clerk work on at least one case. | Alter Table Clerk Add Constraint Pk_ Clerk primary key ( Clerk_ID);<br><br>Alter Table Case Add constraint FK_6_2_2 foreign key ( Clerk_id) references Clerk(Clerk_id)ON DELETE CASCADE); | 1- Clerk<br>2- Case |
| Each case can have more than one witness. | Alter table Case add constraint PK_ Case primary key (CaseNumber );<br><br>Alter table Witnesse add constraint PK_Witnesse primary key(Witnesse_ID);<br><br>Alter table Case_Witnesse add constraint FK_6_21_1 foreign key (Witnesse_ID ) references Witnesse (Witnesse_ID)ON DELETE CASCADE ;<br><br>Alter table Case_Witnesse add constraint FK_6_21_2 foreign key (CaseNumber ) references case (CaseNumber)ON DELETE CASCADE; | 1- Case<br>2- Witnesse<br>3- Case_Witnesse |
| Each branch contains more than one department. | Alter Table Department Add Constraint PK_Department primary key (DepartmentNo);<br><br>Alter Table Branch Add Constraint PK_ Branch primary key (BranchNumber);<br><br>Alter Table Branch_Department Add Constraint PK_Branch_Department primary key (DepartmentNo,BranchNumber));<br><br>Alter Table Branch_Department Add Constraint FK_6_19_1 foreign Key ( DepartmentNo)references Department (DepartmentNo) ON delete CasCade); | 1- Department<br>2- Branch<br>3- Branch_Department |

| | | |
|---|---|---|
| | Alter Table Branch_Department Add Constraint FK_6_19_2 foreign Key ( BranchNumber)references Branch ( BranchNumber) ON delete CasCade); | |
| Each defend must have a specialization. | Alter table Defend Add Constraint PK_Defend primary key(Defend_ID);<br><br>Alter Table Defend Add Constraint FK_6_12_1 foreign key(Defend_ID) references Person(Person_ID) ON DELETE CASCADE;<br><br>Alter Table Defend Add constraint FK_6_12_2 foreign key(Lawyer_ID) references Lawyer(Lawyer_ID) ON DELETE CASCADE; | 1- Defend<br>2- Person<br>3- Lawyer |
| Each case can have more than one accused. | Alter table Case add constraint PK_ Case primary key (CaseNumber );<br><br>Alter table accused add constraint PK_Accused primary key(Accused_ID);<br><br>Alter table Case_ accused add constraint FK_6_16_1 foreign key (Accused_ID ) references Accused (Accused_ID)ON DELETE CASCADE;<br><br>Alter table Case_ accused add constraint FK_6_17_1 foreign key(CaseNumber ) references case(CaseNumber)ON DELETE CASCADE**;** | 1- Case<br>2- accused.<br>3- Case_ accused |
| Each case can have more than one suspect. | Alter table Case add constraint PK_ Case primary key (CaseNumber ); | 1- Case<br>2- suspect<br>3- Case_suspect |

| | | |
|---|---|---|
| | Alter table suspect add constarint PK_Suspect primary key(Suspect_ID);<br><br>Alter table Case_suspect add constraint FK_6_18_2 foreign key(Suspuct_ID ) references Suspuct(Suspuct_ID) ON DELETE CASCADE;<br><br>Alter table Case_suspect add constraint FK_6_17_1 foreign key(CaseNumber ) references case(CaseNumber)ON DELETE CASCADE; | |
| Each case can have more than one victim. | Alter table Case add constraint PK_ Case primary key (CaseNumber );<br><br>Alter table Victim add constraint PK_Victim primary key(Victim_ID);<br><br>Alter table Case_Victim add constraint FK_6_17_1 foreign key(Victim_ID ) references Victim(Victim_ID) ON DELETE CASCADE;<br><br>Alter table Case_Victim add constraint FK_6_17_1 foreign key(CaseNumber ) references case(CaseNumber)ON DELETE CASCADE; | 1- Case<br>2- Case_Victim<br>3- Victim |

# 8 Queries

## 8.1 *Cases with maximum number of witnesses*

**Query in natural language (ENGLISH)**

List the cases with maximum number of witnesses.

**SQL script**

```
SELECT CaseNumber AS Case_Number, count(*) AS Number_Of_Witnesses FROM
Case_Witnesse
GROUP BY CaseNumber HAVING count(*) IN (SELECT MAX (mycount) FROM (SELECT
COUNT(*) mycount FROM Case_Witnesse  GROUP BY CaseNumber));
```

**Caption of the first five rows of the output**

| CASE_NUMBER | NUMBER_OF_WITNESSES |
|---|---|
| 1 | 2 |

## 8.2 *Responsible detectives of cases with the highest number of witnesses*

**Query in natural language (ENGLISH)**

List the detectives who are responsible of cases with the highest number of witnesses.

**SQL script**

```
SELECT * from detective where Detective_ID IN (select Detective_ID from
Case where CaseNumber IN(select CaseNumber from Case_Witnesse group by
CaseNumber having count(*) IN(select Max(mycount)from(select count(*)
mycount from Case_Witnesse group by CaseNumber))));
```

**Caption of the first five rows of the output**

| DETECTIVE_ID | SALARY | SUPERDETECTIVE | DEPARTMENTNO |
|---|---|---|---|
| 2015 | 31000 | 2003 | 5005 |

## 8.3 *Number of victims, accused, suspects in the system*

**Query in natural language (ENGLISH)**

List number of victims, accused, suspects in the system.

**SQL script**

```
select  sum(count(V.Victim_ID)) as Victim,sum(count(A.Accused_ID)) as
Accused ,sum(count(S.Suspect_ID)) as Suspect
from Defend D Join Suspect S on d.Defend_ID =S.Suspect_ID join Accused A on
d.Defend_ID= A.Accused_ID join Victim V on d.Defend_ID=V.Victim_ID
group by V.Victim_ID,S.Suspect_ID ,A.Accused_ID;
```

**Caption of the first five rows of the output**

| VICTIM | ACCUSED | SUSPECT |
|---|---|---|
| 7 | 7 | 7 |

## 8.4 *Department with the highest number of detectives*

**Query in natural language (ENGLISH)**

List the department with the highest number of detectives.

**SQL script**

```
SELECT DEPARTMENTNO,DEPARTMENTNAME FROM Department
WHERE  DEPARTMENTNO IN (SELECT DEPARTMENTNO FROM Detective  GROUP BY
DEPARTMENTNO
HAVING count(*) IN (SELECT MAX (mycount) FROM (SELECT COUNT(*) mycount FROM
Detective  GROUP BY DEPARTMENTNO))) ;
```

**Caption of the first five rows of the output**

| DEPARTMENTNO | DEPARTMENTNAME |
|---|---|
| 5000 | Guns and Drugs Section |

## 8.5 *Cases that have at least 3 evidences*

**Query in natural language (ENGLISH)**

List the cases that have at least 3 evidences.

**SQL script**

```
Select count(*) as Number_of_Evidence ,CASENUMBER from Case_Evidence
Group BY (CASENUMBER)
having count(CASENUMBER) >=3 ;
```

**Caption of the first five rows of the output**

| NUMBER_OF_EVIDENCE | CASENUMBER |
|---|---|
| 3 | 2 |
| 4 | 4 |
| 3 | 14 |

# APPENDIX

### Person Table

SQL Script:

```
Select * from person;
```

Caption of the result:

| PERSON_ID | FNAME | MNAME | LNAME | GENDER | ADDRESS | PHONENUMBER | EMAIL |
|---|---|---|---|---|---|---|---|
| 2000 | Rana | Khaled | Basodan | Female | 1661,Quise Bin Obadh Street | 578437576 | Rana_3524@gamil.com |
| 2003 | Omar | Salem | Motairi | Male | 1551,Alhamra Street | 566779676 | OmarSlem787@gmail.com |
| 2006 | Emad | Talal | Soidi | Male | 1543,Alrawdah Street | 543560998 | Emad_Soidi70@gmail.com |
| 2009 | Yosra | Omar | Remmi | Female | 7712,Laban Street | 510214674 | YomarRemmi_0@gmail.com |
| 2012 | Anode | Rakan | Omari | Female | 1435,Ba Kashab Street | 55432109 | AnodeRakan657@gmail.com |
| 2015 | Osamh | Abdullah | Almotairi | Male | 4423,Sharafiah | 512438953 | Osama_676@randatmail.com |
| 2018 | Khaled | Rami | Adani | Male | 1667,Alkaldiah | 5709325400 | KH_Adani@gmail.com |
| 2021 | Shuog | Salama | Jezani | Female | 3435,Alanode | 552160091 | Shaoge_Jezani@hotmail.com |
| 2024 | Reema | Adel | Khaldi | Female | 1467,Alwahaa | 512542896 | Reema_901@gmail.com |
| 2027 | Saad | Mohamed | Sofiani | Male | 4548,Alwrood | 528699086 | SaadM4490@gmail..com |
| 2030 | Rami | Abdul | Farrell | Female | 2365,Alnakheel Street | 598981076 | RamiAB1998@gmail.com |
| 2033 | Wedad | Mohamed | Altobaiti | Female | 3425,Alsafa | 552615701 | Wedad_aLOT_0@gmail.com |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **2036** | Sami | Mohamed | Alotaibi | Male | 1223,Al-Rawnaq | 544231080 | SAmiM_987@gmail.com |
| **2039** | Renad | Thamer | Almalki | Female | 1667,Alkal | 504312098 | Renad_Thamer@gmail.com |
| **2042** | Amjad | Ali | Al-khidadi | Female | 1989,Kaldia | 534354545 | Amjad0908@hotmail.com |
| **2045** | Ahmed | Salem | Al-solami | Male | 1090,Barq Arhagama | 500909086 | Ahmed_SAle-@gmail.com |
| **2048** | Talal | Anas | Alzahrani | Male | 1989,Kaldia | 575689043 | Talal.Anas9@hotmail.com |
| **100** | Asayel | Mashhour | Alamoudi | Female | 4615,quis Street | 555439098 | asayel@gmail.com |
| **101** | Ahmed | Mohammed | Salesh | Male | 1323,Alsafa | 55543934 | Ahmed@gmail.com |
| **102** | Mohammed | Salem | Salesh | Male | 3413,Om Alarad | 65543564 | Mohammed@gmail.com |
| **103** | Dareen | Salem | Ahmed | Female | 1232,Alkhaldia | 565523564 | Dareen@gmail.com |
| **104** | Hana | Salem | Ahmed | Female | 9898,Safwa | 565513564 | Hana@gmail.com |
| **105** | Ruba | Ahmed | Saleh | Female | 1213,Alanode | 5784675222 | Ruba@randatmail.com |
| **106** | Adel | Sadeq | Hawsawi | Male | 1879,Alrane | 578467576 | AdelH88@gmail.com |
| **107** | Selena | Thomas | Farrell | Female | 1557,King Majed | 578467576 | m.robinson@randatmail.com |
| **108** | Salem | Ahmed | Mohammed | Male | 2343,Sari street | 5784675878 | m.robinson@randatmail.com |
| **109** | Khaled | Talal | Ahmed | Male | 1565,Daren | 578455576 | Khaled@gmail.com |
| **110** | Reem | Saleh | Ahmed | Female | 1243,Alshafa | 57845666 | Reem@gmail.com |
| **111** | Wed | Ahmed | Saleh | Female | 1989,al-seteen street | 57665666 | Wed@gmail.com |
| **112** | Musab | Ahmed | Saleem | Male | 1121,Alajaweed | 57345666 | Musab@gmail.com |
| **3000** | Rami | Mashhour | Alqahtani | Male | 4615,King Khaled Street | 554889098 | Rami_66@gmail.com |
| **3002** | Ahmed | Emad | Alotaibi | Male | 1323,Alsafa | 587310931 | Ahmed@gmail.com |
| **3004** | Mohammed | Salem | Alamri | Male | 3413,Ali Street | 56768961 | Mohammed@gmail.com |
| **3006** | Hani | Salem | Aldahri | Male | 9898,Safwa | 523109345 | Hani@gmail.com |
| **3008** | Raed | Khaled | Saleh | Male | 1213,Alanode | 574509123 | Raead443@randatmail.com |
| **3010** | Adel | Khaled | Hawsawi | Male | 1879,Alrane | 554120911 | Adel0098@gmail.com |
| **700** | Sara | Mustafa | Alshaikh | Female | AL-tyseer | 559879059 | saraqw1233@gmail.com |
| **701** | Salman | Mohammed | Alshaikh | Male | AL-ajwad | 538749044 | salaman@gmail.com |
| **702** | Araw | Khaled | Algamdi | Female | Al-samer | 559879051 | Araw123@gmail.com |
| **703** | Layan | abdullmoen | Alshaikh | Female | Al-samer | 559879052 | layan@gmail.com |
| **704** | Jana | Salem | Ahmed | Female | -Alsameim | 559879054 | Janaa@gmail.com |
| **705** | Joury | Ahmed | Saleh | Female | -Alsameah | 55987939 | JOURYa@randatmail.com |
| **706** | Lana | Thomas | Farrell | Female | Al-samer1 | 559976059 | Lana.robinson@randatmail.com |
| **707** | Reman | Thomas | Farrell | Female | AL-tyseerm | 559878987 | Reman.obinson@randatmail.com |
| **708** | Esra | Talal | Alshaikh | Female | AL-tyseer | 559879058 | Esra@gmail.com |
| **709** | Ghala | Saleh | Ahmed | Female | AL-hamra | 55987909 | Ghalam@gmail.com |
| **710** | Asayal | Ahmed | Saleh | Femal | AL-tyseer-1444 | 559876459 | ASALALf@gmail.com |
| **711** | Merah | Ahmed | Saleh | Female | Al-naseem | 556759787 | Wed@gmail.com |
| **712** | Fasial | Ahmed | Saleem | Male | Al-naseeah | 559870009 | FASIALab@gmail.com |
| **713** | Maha | Mashhour | Alamoudi | Female | AL-tyseer-1444 | | |

## Case Table

SQL Script:

```
Select * from Case;
```

Caption of the result:

| CASENUMBER | CASENAME | DEPARTMENTNO | CLERK_ID | DETECTIVE_ID |
|---|---|---|---|---|
| **1** | Violence | 5005 | 3002 | 2015 |
| **2** | Killing | 5000 | 3002 | 2039 |
| **3** | Drugs Promotion | 5000 | 3000 | 2003 |
| **4** | Hooliganism | 5002 | 3008 | 2000 |
| **5** | Take Drugs | 5000 | 3016 | 2024 |
| **6** | Stealing | 5004 | 3000 | 2006 |
| **7** | Division of inheritance | 5005 | 3018 | 2015 |
| **8** | Stealing | 5004 | 3004 | 2027 |
| **9** | Vandalism of public property | 5002 | 3020 | 2000 |
| **10** | Hooliganism on the street | 5001 | 3004 | 2045 |
| **11** | Violence | 5005 | 3014 | 2042 |
| **12** | Rape | 5002 | 3002 | 2000 |
| **13** | Illicit drugs use | 5000 | 3020 | 2003 |
| **14** | Terrorism | 5001 | 3008 | 2021 |
| **15** | Take Drugs | 5000 | 3012 | 2024 |
| **16** | Use of an unauthorized weapon | 5000 | 3004 | 2030 |
| **17** | Division of inheritance | 5005 | 3020 | 2048 |
| **18** | Stealing | 5004 | 3014 | 2018 |
| **19** | Vandalism of public property | 5001 | 3020 | 2009 |
| **20** | Hooliganism on the street | 5001 | 3010 | 2012 |

## Detective Table

SQL Script:

```
select * from Detective;
```

Caption of the result:

| DETECTIVE_ID | SALARY | SUPERDETECTIVE | DEPARTMENTNO |
|---|---|---|---|
| **2000** | 70000 | - | 5002 |
| **2003** | 39000 | - | 5000 |
| **2006** | 37000 | - | 5004 |
| **2009** | 50000 | 2000 | 5001 |
| **2012** | 33000 | 2009 | 5001 |
| **2015** | 31000 | 2003 | 5005 |
| **2018** | 40000 | 2003 | 5004 |
| **2021** | 49000 | 2009 | 5001 |
| **2024** | 47000 | 2000 | 5000 |
| **2027** | 50000 | 2000 | 5004 |
| **2030** | 37000 | 2009 | 5000 |
| **2033** | 32000 | 2000 | 5000 |
| **2036** | 50000 | 2000 | 5005 |
| **2039** | 50000 | 2003 | 5000 |
| **2042** | 39000 | 2036 | 5005 |
| **2045** | 43000 | 2009 | 5001 |
| **2048** | 33000 | 2036 | 5005 |

## Department Table

SQL Script:

```
Select * from Department;
```

Caption of the result:

| DEPARTMENTNO | DEPARTMENTNAME | LOCATION | DETECTIVEID |
|---|---|---|---|
| **5000** | Guns and Drugs Section | Jeddah | 2000 |
| **5001** | Organized Crime Section | Makkah | 2003 |
| **5002** | General Investigation Section | Jeddah | 2006 |
| **5003** | Major Crime Section | Jeddah | 2009 |
| **5004** | Criminal Investigation Section | Makkah | 2012 |
| **5005** | Major Crime Section | Makkah | 2015 |

## Branch Table

SQL Script:

```
Select * from Branch;
```

Caption of the result:

| BRANCHNUM | PHONNUMBER | CITY |
|---|---|---|
| **1** | 129244311 | Makkah |
| **2** | 128724109 | Jeddah |

## Suspect Table

SQL Script:

```
Select * from suspect;
```

Caption of the result:

| SUSPECT_ID |
|---|
| **100** |
| **101** |
| **102** |
| **103** |
| **104** |
| **105** |
| **106** |

### Clerk Table

SQL Script:

```
Select * from Clerk;
```

Caption of the result:

| CLERK_ID | SALARY |
|---|---|
| **3000** | 20000 |
| **3002** | 25000 |
| **3004** | 22500 |
| **3006** | 39000 |
| **3008** | 50000 |
| **3010** | 20000 |
| **3012** | 30000 |
| **3014** | 10000 |
| **3016** | 9000 |
| **3018** | 10000 |
| **3020** | 10000 |

### Victim Table

SQL Script:

```
Select * from victim;
```

Caption of the result:

| VICTIM_ID |
|---|
| **100** |
| **101** |
| **102** |
| **103** |
| **104** |
| **105** |
| **106** |

## Lawyer Table

SQL Script:

```
Select * from Lawyer;
```

Caption of the result:

| LAWYER_ID | SPECIALIZATION |
|---|---|
| 700 | Civil Specialty |
| 701 | General Specialty |
| 702 | Criminal Specialty |
| 703 | Consultant specialty |
| 704 | General Specialty |
| 705 | Consultant specialty |
| 706 | Criminal Specialty |
| 707 | Consultant specialty |
| 708 | General Specialty |
| 709 | Consultant specialty |
| 710 | General Specialty |
| 711 | Criminal Specialty |
| 712 | Consultant specialty |
| 713 | Civil Specialty |

## Accused Table

SQL Script:

```
Select * from Accused;
```

Caption of the result:

| ACCUSED_ID | ARRESST_STATUS |
|---|---|
| **100** | Under Arrest |
| **101** | Under investigation |
| **102** | Realease |
| **103** | The Accused has not been arrested |
| **104** | Under investigation |
| **105** | Under Arrest |
| **106** | Under Arrest |
| **107** | Realease |
| **108** | The Accused has not been arrested |
| **109** | Under investigation |
| **110** | Realease |
| **111** | Under Arrest |
| **112** | Under investigation |
| **113** | Realease |
| **114** | The Accused has not been arrested |
| **115** | Under investigation |
| **116** | Under investigation |

### Witness Table

SQL Script:

```
Select * from Witnesse;
```

Caption of the result:

| WITNESSE_ID |
|---|
| 1 |
| 2 |
| 3 |
| 4 |
| 5 |

### Defend Table

SQL Script:

```
Select * from Defend;
```

Caption of the result:

| DEFEND_ID | LAWYER_ID |
|---|---|
| 100 | 700 |
| 101 | 704 |
| 102 | 710 |
| 104 | 705 |
| 105 | 701 |
| 107 | 712 |
| 108 | 711 |
| 109 | 713 |
| 112 | 707 |
| 113 | 711 |
| 115 | 710 |
| 116 | 703 |
| 117 | 708 |
| 119 | 711 |
| 103 | 710 |
| 106 | 711 |
| 110 | 702 |
| 114 | 703 |
| 118 | 705 |
| 111 | 700 |
| 120 | 700 |
| 121 | 700 |
| 122 | 700 |
| 123 | 700 |

## Detective_Action Table

SQL Script:

```
Select * from Detective_Action;
```

Caption of the result:

| CASENUMBER | ACTIONS |
|---|---|
| 1 | Primary Hearing |
| 2 | Searching |
| 3 | Surveillance of Mail Correspondence |
| 4 | Search of Persons |
| 5 | conducting Inquiries |
| 6 | Detention Warrant |
| 7 | Closing of the Case |
| 8 | Surveillance of Mail Correspondence |
| 9 | Placing in Custody |
| 10 | Closing of the Case |
| 11 | Release Order |
| 12 | Request for Scrutiny |
| 13 | Statement Taking |
| 14 | Release Order |
| 15 | Primary Hearing |
| 16 | Closing of the Case |
| 17 | Release Order |
| 18 | Statement Taking |
| 19 | Inspection |
| 20 | Closing of the Case |

### Case_Evidence Table

SQL Script:

```
Select * from Case_Evidence;
```

Caption of the result:

| CASENUMBER | EVIDENCE |
|---|---|
| **1** | Victim"s complaint |
| **2** | The presence of fingerprints on the weapon |
| **2** | Vedio recordings |
| **2** | Victim"s complaint |
| **3** | Presence of drugs in the possession of the accused |
| **3** | drug test |
| **4** | Being at the crime scene |
| **4** | Damage to public property |
| **4** | Having a witness to the case |
| **4** | Vedio recording |
| **5** | drug test |
| **6** | Presence of fingerprints |
| **7** | The heirs"complaint |
| **8** | Presence of fingerprints |
| **9** | Being at the crime scene |
| **10** | Being at the crime scene |
| **11** | Victim"s complaint |
| **12** | Victim"s complaint, DNA Test |
| **13** | Being at the crime scene |
| **13** | drug test |
| **14** | Audio recordings |
| **14** | Being at the crime scene |
| **14** | Having a witness to the case |
| **15** | drug test |
| **16** | The weapon is in his possession |
| **17** | The heirs"complaint |
| **18** | Presence of fingerprints,Victim"s complaint |
| **18** | Vedio recordings |
| **19** | Being at the crime scene,Vedio recordings |

## Clerk_Statement Table

SQL Script:

```
Select * from Clerk_Statment;
```

Caption of the result:

| CASENUMBER | STATMENT |
|---|---|
| **10** | I was with a group of guys in the street and they told me if I vandalized they would reward me, I was foolish and sorry for what I did |
| **15** | Yes, I was taking Drugs |
| **18** | You ask me if we closed that boy in. Yes that's right. You ask me if it is rigt that I had my hand in his coat pocket and took 500 Euros. Yes that is right. I saw no keys. I took 5 Euros |

## Case_Accused Table

SQL Script:

```
Select * from Case_Accused;
```

Caption of the result:

| CASENUMBER | ACCUSED_ID |
|---|---|
| **9** | 100 |
| **11** | 100 |
| **3** | 101 |
| **5** | 102 |
| **2** | 103 |
| **14** | 103 |
| **6** | 104 |
| **4** | 105 |
| **15** | 106 |
| **1** | 107 |
| **13** | 107 |
| **16** | 107 |
| **17** | 107 |
| **12** | 108 |
| **18** | 109 |
| **7** | 110 |
| **5** | 111 |
| **10** | 112 |
| **9** | 113 |
| **5** | 114 |
| **8** | 115 |
| **20** | 115 |
| **19** | 116 |

### Case_Victim Table

SQL Script:

```
Select * from Case_VICTIM;
```

Caption of the result:

| CASENUMBER | VICTIM_ID |
|---|---|
| **1** | 100 |
| **2** | 101 |
| **3** | 102 |
| **4** | 103 |
| **5** | 104 |
| **6** | 105 |
| **7** | 106 |

### Case_Suspect Table

SQL Script:

```
Select * from Case_Suspect;
```

Caption of the result:

| CASENUMBER | SUSPECT_ID |
|---|---|
| **1** | 100 |
| **2** | 101 |
| **3** | 102 |
| **4** | 103 |
| **5** | 104 |
| **6** | 105 |
| **7** | 106 |

### Branch_Department Table

SQL Script:

```
Select * from Branch_Department;
```

Caption of the result:

| DEPARTMENTNO | BRANCHNUMBER |
|---|---|
| **5000** | 2 |
| **5001** | 1 |
| **5002** | 2 |
| **5003** | 1 |
| **5004** | 2 |
| **5005** | 1 |

## Suspect_Ind Table

SQL Script:

```
Select * from Suspect_Ind;
```

Caption of the result:

| CASENUMBER | SUSPECT_ID | INDICATIONOFSUSPICION |
|---|---|---|
| **1** | 100 | Aggression |
| **1** | 100 | Marks of beatings |
| **2** | 101 | Aggression |
| **3** | 102 | Close to the site of the accident |
| **3** | 102 | Suspicious behavior |
| **4** | 103 | Consanguinity with persons involved in the case |
| **5** | 104 | Weird behavior |
| **6** | 105 | Close to the site of the accident |
| **7** | 106 | Seizure of funds of unknown source |

## Case_Witness Table

SQL Script:

```
Select * from Case_Witnesse;
```

Caption of the result:

| CASENUMBER | WITNESSE_ID | TESTIMONY |
|---|---|---|
| **1** | 1 | the person was acting as though she were drunk. |
| **2** | 2 | I observed a black truck approach a red light and continue driving through it without stopping. |
| **3** | 3 | the car was driving too fast also the person was acting as though she were drunk. |
| **6** | 4 | Audio and 3 images |
| **12** | 5 | I observed a blue truck approach a red light and continue driving through it without stopping and the person inside car was acting as though she were drunk |
| **1** | 2 | I saw him doing it |