

Al-Maaref University

Database Suspect

Nour Merhi

Maryam Morsel

Database Course

Dr. Mubarak Mohammad

Wednesday, January 3, 2024

Table of Contents

I.	Introduction	3
II.	Entities	3
	❖ Face:.....	3
	❖ Body:.....	3
	❖ Hands.....	3
	❖ Feet.....	4
	❖ Hair	4
	❖ Location	4
	❖ Building.....	4
	❖ Apartment	4
	❖ Criminal record	4
	❖ Victim.....	5
	❖ Hospital Record.....	5
	❖ Doctor	5
	❖ Suspect	5
	❖ Mother Info.....	5
	❖ Wife Info	6
III.	Relationships:.....	6
IV.	Er diagram:.....	8
V.	Relational Model.....	9
VI.	Queries:	10

Introduction

This project aims the need up-to-date global data on criminals in order to carry out successful international investigation. It is a 2-student group work, Nour Merhi and Maryam Morsel, form Computer Science department in the Al-Maaref University. This project has developed a database system for searching for suspects that detectives or forensics might suspect for doing a crime. It will cover the entities, attributes, relationships using ER diagram, relational schema and their queries and description.

Entities

1. Face:

This entity describes the face of the suspect form the forehead, eyebrows, eyes, nose, mouth and chin.

FaceID (primary key)	int
Forehead	Varchar (10)
Eyebrows	Varchar (10)
EyeColor	Varchar (10)
Nose	Varchar (10)
Mouth	Varchar (10)
Chin	Varchar (10)
FaceShape	Varchar (20)

2. Body:

This entity shows the wight, height, body shape of the suspect.

BodyID (primary key)	Int
Height in cm	Decimal (3,2)
Weight in Kg	Decimal (3, 3)
BodyShape	Varchar (20)

3. Hands:

This entity contains the hand shape.

HandID (primary key)	Int
FingerNumbers	Int
HandShape	Varchar (10)
Fingerprints	Varchar (3)
HandNumber	int

4. Feet:

This entity contains the feet shape.

FeetID (primary key)	Int
FingerNumbers	Int
FootShape	Varchar (10)
FootSize	Int
FeetNumber	Int

5. Hair:

This entity shows the hair color and shape of the suspect.

HairID (primary key)	Int
Color	Varchar (15)
HairHeight	Varchar (10)
HairShape	Varchar (20)

6. Location:

This entity gives us information about where the suspect lives.

LocationID (primary key)	Int
City	Varchar (25)
Country	Varchar (25)
Addresss	Varchar (25)

7. Building:

This entity gives us more info about the building where the suspect lives in.

BuildingID (primary key)	Int
FloorNb	Int

8. Apartment:

This weak entity shows us the apartment number and house phone number.

ApartmentNb	int
ApartmentPhoneNb	Varchar (8)

9. Criminal record:

This entity informs us if the suspect had a criminal record or not, what type of crime and when happened.

CrimeID (primary key)	Int
Record (O)	Varchar (3)
CrimeType	Varchar (25)

DateOfCrime	Date
LocationOfCrime	Varchar (25)

10. Victim:

This entity tells us who was the victim.

VictimID (primary key)	Int
FirstName	Varchar (20)
LastName	Varchar (20)
Gender	Varchar (10)
Age	Int

11. Hospital Record:

This entity tells us if the suspect had any hospital records.

ID (primary key)	Int
Record (O)	Varchar (3)
TypeOfExamination	Varchar (25)

12. Doctor:

This entity informs us about the doctor who examined the suspect.

DoctorID (primary key)	Int
FisrtName	Varchar (25)
LastName	Varchar (25)
Specialty	Varchar (25)

13. Suspect:

This entity gives every information about the suspect himself.

SuspectID (primary key)	Int
FisrtName	Varchar (25)
MiddleName	Varchar (25)
LastName	Varchar (25)
Gender	Varchar (10)
DateOfBirth	Date
PersonalPhoneNb	Varchar (8)

14. Mother Info:

This entity gives us suspect's mother full name and her phone number.

MotherID (primary key)	Int
FirstName	Varchar (25)
MiddleName	Varchar (25)

LastName	Varchar (25)
PhoneNumber	Varchar (8)

15. Wife Info:

This entity gives us suspect's wife full name and phone number.

WifeID (primary key)	Int
FirstName	Varchar (25)
LastName	Varchar (25)
PhoneNumber	Varchar (8)

Relationships:

➤ Between Face and Hair:

Each face might have hair type or not and each hair type should be on one or many faces.

➤ Between Body and Hands:

Each Body could have one hand or not, and each hand should be connected to one and only one body.

➤ Between Body and Feet:

Each body could have one or might not foot and each foot should be connected to one and only one body.

➤ Between Body and Face

Each body must have one and only one face and each face should be connected to one and only one body.

➤ Between Location and Building

Each location contains one or many buildings and each building contains only one location.

➤ Between Building and Apartment

Each building has one or many apartments and each apartment is located in one and only one building.

➤ Between Criminal Record and Victim

Each suspect might kill one victim or not and each victim is killed by one and only one criminal.

➤ Between Hospital Record and Doctor

Each hospital has one or many doctors that works in and each doctor can work in one or many hospitals.

➤ Between Suspect and Mother Info

Each suspect might have one mother or not and each mother could have many suspects or one.

➤ Between Suspect and Wife Info

Each suspect might be married to one or not and each wife might be married to a suspect or not.

➤ Between Suspect and Body

Each suspect should have one and only one body and each body might be assigned to one or many suspects.

➤ Between Suspect and location

Each suspect could have one or many locations and each location might have one suspect or not.

➤ Between Suspect and Criminal Record

Each suspect might have criminal record or not and each criminal records should be associated to one or many suspects.

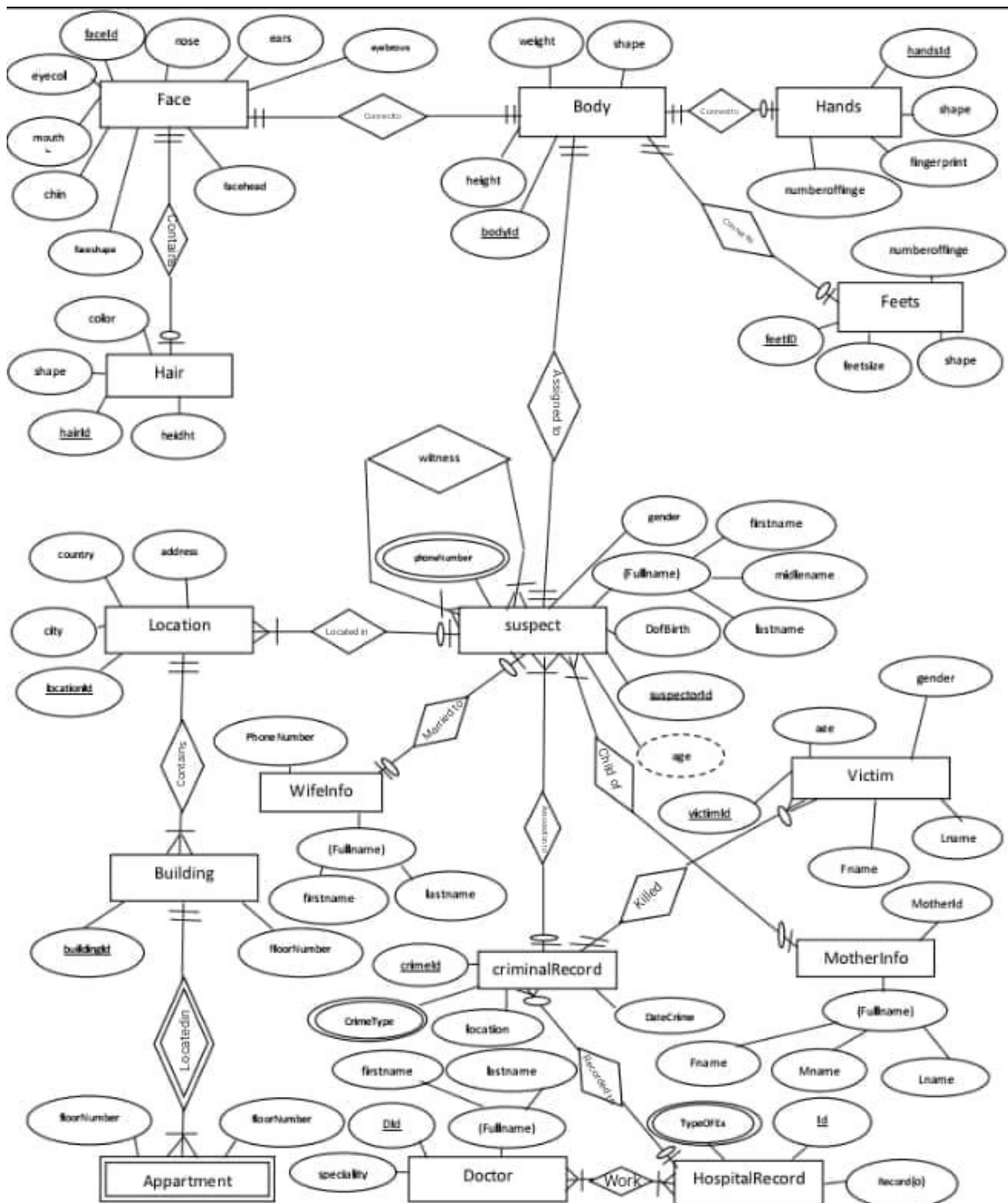
➤ Between Hospital Record and Criminal Record

Each hospital might have many records or not for a criminal and each criminal might have a record but not necessary.

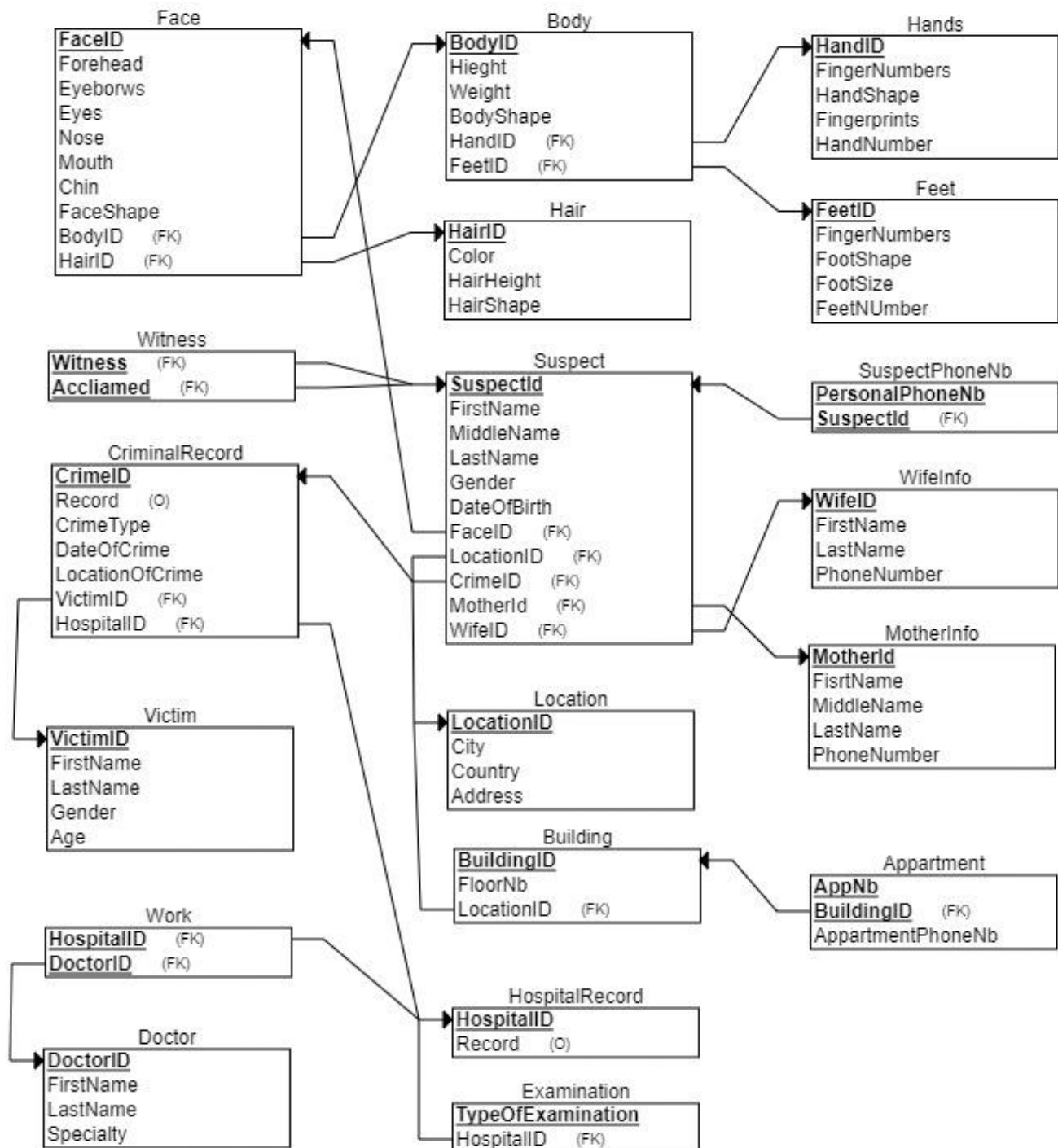
➤ Between Suspect and Suspect

Each suspect might be an actual witness or can be acclaimed by others.

Er diagram:



Relational Model



Queries:

1) Join Queries with 2 Tables:

- a. Retrieve suspect's full name and his mother's name.

Select s.firstName, s.MiddleName, s.LastName, m.FisrtName, m.LastNAme from suspect as s inner join mother as m on m.motherId = s.motherID;

- b. Rretrieve the full name of the suspect who made a crime at date 31-12-2003.

Select fisrtName, middleName lastName, c.dateOfCrime from suspect as s inner join criminalRecord as c on c.crimeld = s.CrimeID where c.dateOfCrime = "31-12-2003";

- c. suspect's fisrt name and last name whose eyes are green, face shape rectangular and mouth is small.

Select s.firstName, s.LastName from suspect as s inner join face as f on f.FaceID = s.FaceID where f.eyeColor = "Green" and f.faceShape = "Rectangular" and f.mouth = "small";

2) Join Queries with 3 tables or more:

- a. Retrieve victim's name of the criminal Nour Morsel.

Select v.firstName, v.lasName from victim as v inner join criminalRecord as c on v.victimID = c.victimID inner join suspect as s on s.crimeID = c.CrimeID where s.FirstName = "Nour" and s.lastName = "Morsel";

- b. Retrieve suspect's full name who lives on 10th floor.

Select s.firstName, s.midddleName, s.lastName from suspect as s inner join location as l on l.LocationID = s.locationID inner join building as b on b.LocstionId = l.locationId where b.floorNB = 10;

3) Subqueries:

- a. Retrieve doctor's name who wokrs in the hospital having id 3.

Select FisrtName, lastName from doctor where doctorId = (select HospitalID from work where hospitalId = 3);

- b. Retrieve suspect's all personal numbers who have a wife named Maryam Merhi.

Select * from suspectPhoneNb where suspectId = (select suspectID from suspect where Wifeld = (select wifeld from wifeInfo where FisrtName = "Maryam" and lastName = "Merhi");

4) Queries with aggregate calculations:

- a. Count how many criminal records are there.

Select count(record) from criminalRecord where record = "Yes";

- b. Retrieve how many suspects have eye color brown.

Select count (*) from suspect where faceld = (select faceld from face where eyeColor = "Brown");

5) Single table query:

- a. Retrieve all suspect from older to youngest.

Select * from suspect order by dateOfBirth desc;