Warm-up Project

Comp353-CD

Dr. Khaled Jababo

July 20th 2025

Team ID: ntc353_1

Angel Asencios (40176253)
Haaris Muhammad Khawaja (40261525)
Ali Sher (40255236)
Farhan Ali Khan (40287970)

AITS login:

Username: ntc353_1

Password: Draty@01

Faculty of Engineering and Computer Science
Expectations of Originality

This form sets out the requirements for originality for work submitted by students in the Faculty of Engineering and Computer Science. Submissions such as assignments, lab reports, project reports, computer programs and take-home exams must conform to the requirements stated on this form and to the Academic Code of Conduct. The course outline may stipulate additional requirements for the course.

1. Your submissions must be your own original work. Group submissions must be the original work of the students in the group.
2. Direct quotations must not exceed 5% of the content of a report, must be enclosed in quotation marks, and must be attributed to the source by a numerical reference citation[1]. Note that engineering reports rarely contain direct quotations.
3. Material paraphrased or taken from a source must be attributed to the source by a numerical reference citation.
4. Text that is inserted from a web site must be enclosed in quotation marks and attributed to the web site by numerical reference citation.
5. Drawings, diagrams, photos, maps or other visual material taken from a source must be attributed to that source by a numerical reference citation.
6. No part of any assignment, lab report or project report submitted for this course can be submitted for any other course.
7. In preparing your submissions, the work of other past or present students cannot be consulted, used, copied, paraphrased or relied upon in any manner whatsoever.
8. Your submissions must consist entirely of your own or your group's ideas, observations, calculations, information and conclusions, except for statements attributed to sources by numerical citation.
9. Your submissions cannot be edited or revised by any other student.
10. For lab reports, the data must be obtained from your own or your lab group's experimental work.
11. For software, the code must be composed by you or by the group submitting the work, except for code that is attributed to its sources by numerical reference.

You must write one of the following statements on each piece of work that you submit:
For individual work: **"I certify that this submission is my original work and meets the Faculty's Expectations of Originality",** with your signature, I.D. #, and the date.
For group work: **"We certify that this submission is the original work of members of the group and meets the Faculty's Expectations of Originality",** with the signatures and I.D. #s of all the team members and the date.

A signed copy of this form must be submitted to the instructor at the beginning of the semester in each course.

I certify that I have read the requirements set out on this form, and that I am aware of these requirements. I certify that all the work I will submit for this course will comply with these requirements and with additional requirements stated in the course outline.

Course Number: 353 Instructor: Khaled Jababo
Name: Angel Asencios I.D. # 40176253
Signature: Angel Asencios Date: 20/07/2025

[1] Rules for reference citation can be found in "Form and Style" by Patrich MacDonagh and Jack Bordan, fourth edition, May, 2000, available at http://www.encs.concordia.ca/scs/Forms/Form&Style.pdf.
Approved by the ENCS Faculty Council February 10, 2012

Faculty of Engineering and Computer Science
Expectations of Originality

This form sets out the requirements for originality for work submitted by students in the Faculty of Engineering and Computer Science. Submissions such as assignments, lab reports, project reports, computer programs and take-home exams must conform to the requirements stated on this form and to the Academic Code of Conduct. The course outline may stipulate additional requirements for the course.

1. Your submissions must be your own original work. Group submissions must be the original work of the students in the group.
2. Direct quotations must not exceed 5% of the content of a report, must be enclosed in quotation marks, and must be attributed to the source by a numerical reference citation[1]. Note that engineering reports rarely contain direct quotations.
3. Material paraphrased or taken from a source must be attributed to the source by a numerical reference citation.
4. Text that is inserted from a web site must be enclosed in quotation marks and attributed to the web site by numerical reference citation.
5. Drawings, diagrams, photos, maps or other visual material taken from a source must be attributed to that source by a numerical reference citation.
6. No part of any assignment, lab report or project report submitted for this course can be submitted for any other course.
7. In preparing your submissions, the work of other past or present students cannot be consulted, used, copied, paraphrased or relied upon in any manner whatsoever.
8. Your submissions must consist entirely of your own or your group's ideas, observations, calculations, information and conclusions, except for statements attributed to sources by numerical citation.
9. Your submissions cannot be edited or revised by any other student.
10. For lab reports, the data must be obtained from your own or your lab group's experimental work.
11. For software, the code must be composed by you or by the group submitting the work, except for code that is attributed to its sources by numerical reference.

You must write one of the following statements on each piece of work that you submit:
For individual work: **"I certify that this submission is my original work and meets the Faculty's Expectations of Originality",** with your signature, I.D. #, and the date.
For group work: **"We certify that this submission is the original work of members of the group and meets the Faculty's Expectations of Originality",** with the signatures and I.D. #s of all the team members and the date.

A signed copy of this form must be submitted to the instructor at the beginning of the semester in each course.

Haaris
Khawaja

[1] Rules for reference citation can be found in "Form and Style" by Patrich MacDonagh and Jack Bordan, fourth edition, May, 2000, available at http://www.encs.concordia.ca/scs/Forms/Form&Style.pdf. Approved by the ENCS Faculty Council February 10, 2012

I certify that I have read the requirements set out on this form, and that I am aware of these requirements. I certify that all the work I will submit for this course will comply with these requirements and with additional requirements stated in the course outline.

Course Number: COMP 353 Instructor: Khaled Jababo
Name: Haaris Muhammad Khawaja I.D. # 40261525

Signature: Haaris Khawaja Date: 20 July 2025

Faculty of Engineering and Computer Science
Expectations of Originality

This form sets out the requirements for originality for work submitted by students in the Faculty of Engineering and Computer Science. Submissions such as assignments, lab reports, project reports, computer programs and take-home exams must conform to the requirements stated on this form and to the Academic Code of Conduct. The course outline may stipulate additional requirements for the course.

1. Your submissions must be your own original work. Group submissions must be the original work of the students in the group.
2. Direct quotations must not exceed 5% of the content of a report, must be enclosed in quotation marks, and must be attributed to the source by a numerical reference citation[1]. Note that engineering reports rarely contain direct quotations.
3. Material paraphrased or taken from a source must be attributed to the source by a numerical reference citation.
4. Text that is inserted from a web site must be enclosed in quotation marks and attributed to the web site by numerical reference citation.
5. Drawings, diagrams, photos, maps or other visual material taken from a source must be attributed to that source by a numerical reference citation.
6. No part of any assignment, lab report or project report submitted for this course can be submitted for any other course.
7. In preparing your submissions, the work of other past or present students cannot be consulted, used, copied, paraphrased or relied upon in any manner whatsoever.
8. Your submissions must consist entirely of your own or your group's ideas, observations, calculations, information and conclusions, except for statements attributed to sources by numerical citation.
9. Your submissions cannot be edited or revised by any other student.
10. For lab reports, the data must be obtained from your own or your lab group's experimental work.
11. For software, the code must be composed by you or by the group submitting the work, except for code that is attributed to its sources by numerical reference.

You must write one of the following statements on each piece of work that you submit:
For individual work: **"I certify that this submission is my original work and meets the Faculty's Expectations of Originality",** with your signature, I.D. #, and the date.
For group work: **"We certify that this submission is the original work of members of the group and meets the Faculty's Expectations of Originality",** with the signatures and I.D. #s of all the team members and the date.

A signed copy of this form must be submitted to the instructor at the beginning of the semester in each course.

I certify that I have read the requirements set out on this form, and that I am aware of these requirements. I certify that all the work I will submit for this course will comply with these requirements and with additional requirements stated in the course outline.

Course Number: Comp 353 CD 2251 Instructor: Khaled Jababo
Name: Ali Sher I.D. # 40255236
Signature: Date: 2025-07-20

[1] Rules for reference citation can be found in "Form and Style" by Patrich MacDonagh and Jack Bordan, fourth edition, May, 2000, available at http://www.encs.concordia.ca/scs/Forms/Form&Style.pdf.
Approved by the ENCS Faculty Council February 10, 2012

Faculty of Engineering and Computer Science
Expectations of Originality

This form sets out the requirements for originality for work submitted by students in the Faculty of Engineering and Computer Science. Submissions such as assignments, lab reports, project reports, computer programs and take-home exams must conform to the requirements stated on this form and to the Academic Code of Conduct. The course outline may stipulate additional requirements for the course.

1. Your submissions must be your own original work. Group submissions must be the original work of the students in the group.
2. Direct quotations must not exceed 5% of the content of a report, must be enclosed in quotation marks, and must be attributed to the source by a numerical reference citation[1]. Note that engineering reports rarely contain direct quotations.
3. Material paraphrased or taken from a source must be attributed to the source by a numerical reference citation.
4. Text that is inserted from a web site must be enclosed in quotation marks and attributed to the web site by numerical reference citation.
5. Drawings, diagrams, photos, maps or other visual material taken from a source must be attributed to that source by a numerical reference citation.
6. No part of any assignment, lab report or project report submitted for this course can be submitted for any other course.
7. In preparing your submissions, the work of other past or present students cannot be consulted, used, copied, paraphrased or relied upon in any manner whatsoever.
8. Your submissions must consist entirely of your own or your group's ideas, observations, calculations, information and conclusions, except for statements attributed to sources by numerical citation.
9. Your submissions cannot be edited or revised by any other student.
10. For lab reports, the data must be obtained from your own or your lab group's experimental work.
11. For software, the code must be composed by you or by the group submitting the work, except for code that is attributed to its sources by numerical reference.

You must write one of the following statements on each piece of work that you submit:
For individual work: **"I certify that this submission is my original work and meets the Faculty's Expectations of Originality"**, with your signature, I.D. #, and the date.
For group work: **"We certify that this submission is the original work of members of the group and meets the Faculty's Expectations of Originality"**, with the signatures and I.D. #s of all the team members and the date.

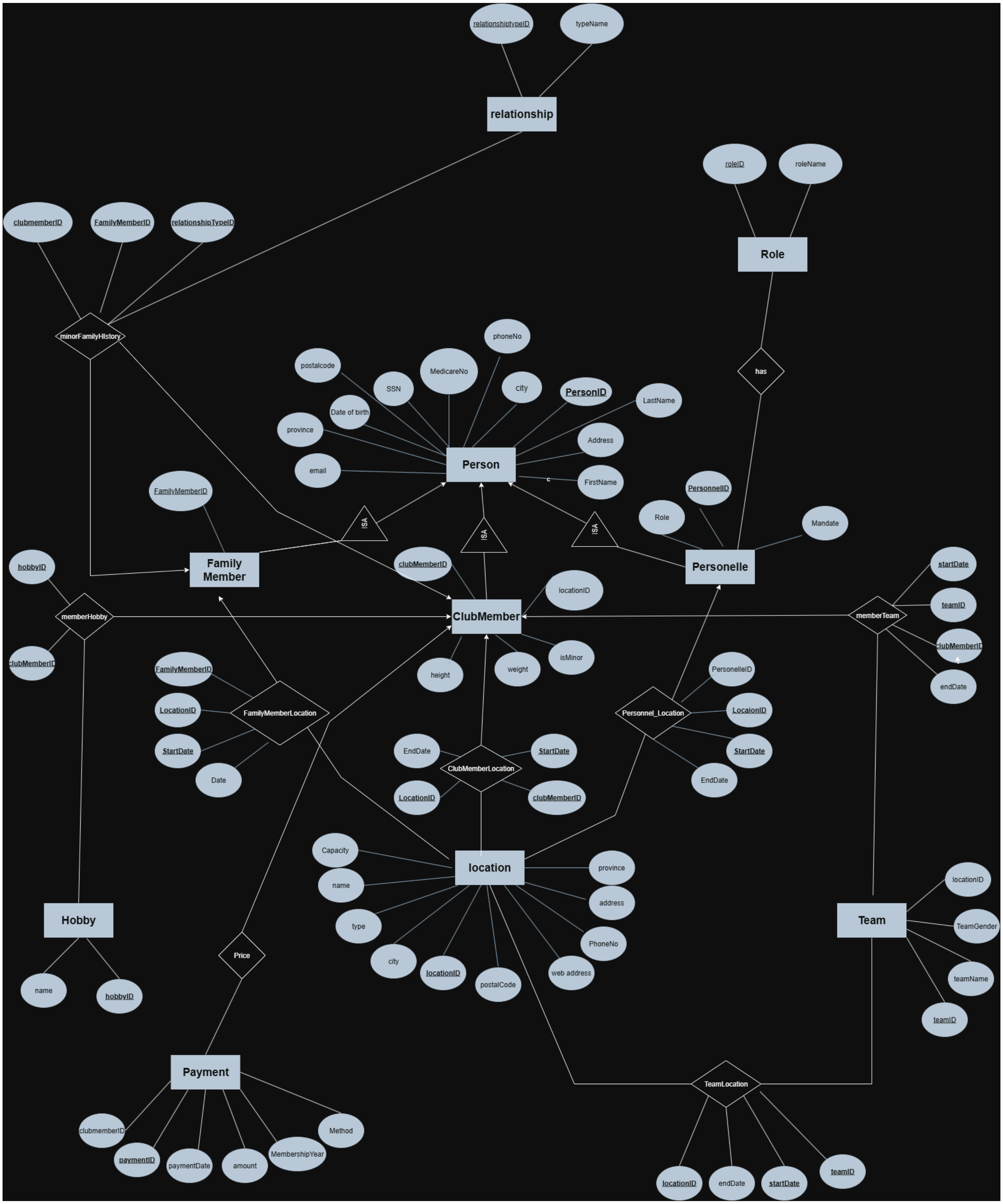
A signed copy of this form must be submitted to the instructor at the beginning of the semester in each course.

I certify that I have read the requirements set out on this form, and that I am aware of these requirements. I certify that all the work I will submit for this course will comply with these requirements and with additional requirements stated in the course outline.

Course Number: Comp 353 CD 2251
Instructor: Khaled Jababo
Name: Farhan Ali Khan
I.D. # 40287970
Signature:
Date: 2025-07-20

[1] Rules for reference citation can be found in "Form and Style" by Patrich MacDonagh and Jack Bordan, fourth edition, May, 2000, available at http://www.encs.concordia.ca/scs/Forms/Form&Style.pdf.
Approved by the ENCS Faculty Council February 10, 2012

Conceptual DB Design (E/R)



E/R to relations conversion

Entity sets

- **Person**(PersonID, FirstName, LastName, DateOfBirth, SSN, MedicareNo, PhoneNo, Address, City, Province, PostalCode, Email)
- **FamilyMember**(FamilyMemberID)
- **Personnel**(PersonnelID
- **ClubMember**(ClubMemberID, Height, Weight, isMinor, locationID)
- **Location**(locationID, name, type, Address, City, Province, PostalCode, PhoneNo, WebAddress, Capacity)
- **Team**(teamID, teamName, team_gender, LocationID)
- **Hobby**(hobbyID, name)
- **Payment**(PaymentID,ClubMemberID, PaymentDate, Amount, Method, MembershipYear)
- **Relationship**(relationshipTypeID, typeName)

Relationships as tables

- **ClubMemberLocation**(ClubMemberID, LocationID, StartDate, EndDate)
- **Personnel_Location**(PersonnelID, LocationID, StartDate, EndDate)
- **FamilyMemberLocation**(FamilyMemberID, LocationID, StartDate, EndDate)
- **memberHobby**(ClubMemberID, hobbyID)
- **teamLocation**(teamID, LocationID, startDate, endDate)
- **memberTeam**(startDate, teamID, ClubMemberID, endDate)
- **minorFamilyHistory**(relationshipTypeID, FamilyMemberID, ClubMemberID)

DDL Statements

```
USE ntc353_1;

SHOW TABLES;

CREATE TABLE Person(
PersonID INT AUTO_INCREMENT NOT NULL PRIMARY KEY,
FirstName VARCHAR(50) NOT NULL,
    LastName VARCHAR(50),
    DateOfBirth DATE NOT NULL,
    SSN VARCHAR(50) UNIQUE NOT NULL,
    MedicareNo VARCHAR(20) UNIQUE,
    PhoneNo VARCHAR(20),
    Address VARCHAR(200),
    City VARCHAR(50),
    Province VARCHAR(50),
    PostalCode VARCHAR(10),
    Email VARCHAR(100)
);

CREATE TABLE FamilyMember(
FamilyMemberID INT NOT NULL PRIMARY KEY,
FOREIGN KEY(FamilyMemberID) REFERENCES Person(PersonID)
ON DELETE CASCADE
ON UPDATE CASCADE
);

CREATE TABLE Personnel(
PersonnelID INT NOT NULL PRIMARY KEY,
    `Role` ENUM('Administrator','Captain','Coach','Assistant Coach','Other') NOT NULL,
    Mandate ENUM('Volunteer','Salaried') NOT NULL,
    FOREIGN KEY(PersonnelID) REFERENCES Person(PersonID)
      ON DELETE CASCADE
      ON UPDATE CASCADE
);

CREATE TABLE ClubMember(
ClubMemberID INT NOT NULL PRIMARY KEY,
    Height DECIMAL(5,2),
    Weight DECIMAL(5,2),
    FamilyID INT,
    FOREIGN KEY(ClubMemberID) REFERENCES Person(PersonID)
      ON DELETE CASCADE
      ON UPDATE CASCADE,
FOREIGN KEY(FamilyID) REFERENCES FamilyMember(FamilyMemberID)
      ON DELETE SET NULL
      ON UPDATE CASCADE
```

```
);

CREATE TABLE Location(
LocationID INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
    `name` VARCHAR(100) NOT NULL,
    `type` ENUM('Head','Branch') NOT NULL,
    Address VARCHAR(200) NOT NULL,
    City VARCHAR(50) NOT NULL,
    Province VARCHAR(50) NOT NULL,
    PostalCode VARCHAR(10) NOT NULL,
    PhoneNo VARCHAR(20) NOT NULL,
    WebAddress VARCHAR(100),
    Capacity INT NOT NULL
);

CREATE TABLE ClubMemberLocation(
ClubMemberID INT NOT NULL,
    LocationID INT NOT NULL,
    StartDate DATE NOT NULL,
    EndDate DATE,
    PRIMARY KEY(ClubMemberID,LocationID,StartDate),
    FOREIGN KEY(ClubMemberID) REFERENCES ClubMember(ClubMemberID)
      ON DELETE CASCADE
      ON UPDATE CASCADE,
FOREIGN KEY(LocationID) REFERENCES Location(LocationID)
      On DELETE CASCADE
      ON UPDATE CASCADE
);

CREATE TABLE Personnel_Location(
PersonnelID INT NOT NULL,
    LocationID INT NOT NULL,
    StartDate DATE NOT NULL,
    EndDate DATE,
    PRIMARY KEY(PersonnelID, LocationID,StartDate),
    FOREIGN KEY(PersonnelID) REFERENCES Personnel(PersonnelID)
      ON DELETE CASCADE
      ON UPDATE CASCADE,
FOREIGN KEY(LocationID) REFERENCES Location(LocationID)
      ON DELETE CASCADE
      ON UPDATE CASCADE
);

CREATE TABLE FamilyMemberLocation(
FamilyMemberID INT NOT NULL,
    LocationID INT NOT NULL,
    StartDate DATE NOT NULL,
    EndDate Date,
    PRIMARY KEY(FamilyMemberID,LocationID,StartDate),
    FOREIGN KEY(FamilyMemberID) REFERENCES FamilyMember(FamilyMemberID)
      ON DELETE CASCADE
      ON UPDATE CASCADE,
FOREIGN KEY(LocationID) REFERENCES Location(LocationID)
      ON DELETE CASCADE
      ON UPDATE CASCADE
);


CREATE TABLE Payment(
PaymentID INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
ClubMemberID INT NOT NULL,
    PaymentDate DATE NOT NULL,
    Amount DECIMAL(10,2) NOT NULL,
    Method ENUM('Cash','Debit','Credit') NOT NULL,
    MembershipYear INT NOT NULL,
    FOREIGN KEY(ClubMemberID) REFERENCES ClubMember(ClubMemberID)
      ON DELETE CASCADE
      ON UPDATE CASCADE
);

CREATE TABLE Hobby(
    hobbyID INT NOT NULL,
    name varchar(50) NOT NULL,
    PRIMARY KEY(hobbyID)
    );

CREATE TABLE memberHobby(
    hobbyID INT NOT NULL,
    clubMemberID INT NOT NULL,
    PRIMARY KEY(ClubMemberID, hobbyID),
    FOREIGN KEY(ClubMemberID) REFERENCES ClubMember(ClubMemberID)
      ON DELETE CASCADE
      ON UPDATE CASCADE,
    FOREIGN KEY(hobbyID) REFERENCES Hobby(hobbyID)
      ON DELETE CASCADE
      ON UPDATE CASCADE
    );

    create table Team(

teamID int not null Auto_Increment primary key,
teamName varchar(50) unique,
team_gender enum ('Male','Female','other') not null,
LocationID INT NOT NULL,
Foreign key (LocationID) references Location(LocationID)
ON DELETE CASCADE
ON UPDATE CASCADE
);

create table teamLocation(
teamID int not null,
LocationID INT NOT NULL,
startDate date not null,
endDate date,
 PRIMARY KEY (teamID, LocationID, startDate),
Foreign key (LocationID) references Location(LocationID)
ON DELETE CASCADE
ON UPDATE CASCADE,
Foreign key (teamID) references Team(teamID)
ON DELETE CASCADE
ON UPDATE CASCADE
);

create table memberTeam(
startDate date not null,
teamID int not null,
ClubMemberID INT NOT NULL,
primary key (startDate, teamID, ClubMemberID),
Foreign key (teamID) references Team(teamID)
ON DELETE CASCADE
ON UPDATE CASCADE,
FOREIGN KEY(ClubMemberID) REFERENCES ClubMember(ClubMemberID)
ON DELETE CASCADE
ON UPDATE CASCADE
);

ALTER TABLE memberTeam
ADD COLUMN EndDate DATE;

CREATE TABLE Role (
   roleID INT AUTO_INCREMENT PRIMARY KEY,
   roleName VARCHAR(50) NOT NULL UNIQUE
);

CREATE TABLE Relationship(
   relationshipTypeID INT NOT NULL,
   typeName VARCHAR(50) NOT NULL,
   PRIMARY KEY(relationshipTypeID)
   );

CREATE TABLE minorFamilyHistory(
   relationshipTypeID INT NOT NULL,
   FamilyMemberID INT NOT NULL,
   ClubmemberID INT NOT NULL,
   PRIMARY KEY(relationshipTypeID, FamilyMemberID, ClubmemberID),
   FOREIGN KEY(ClubMemberID) REFERENCES ClubMember(ClubMemberID)
     ON DELETE CASCADE
     ON UPDATE CASCADE,
   FOREIGN KEY(FamilyMemberID) REFERENCES FamilyMember(FamilyMemberID)
     ON DELETE CASCADE
     ON UPDATE CASCADE,
                                                     FOREIGN KEY(relationshipTypeID)
REFERENCES Relationship(relationshipTypeID)
     ON DELETE CASCADE
     ON UPDATE CASCADE
);
```

DML Statements (Transactions)

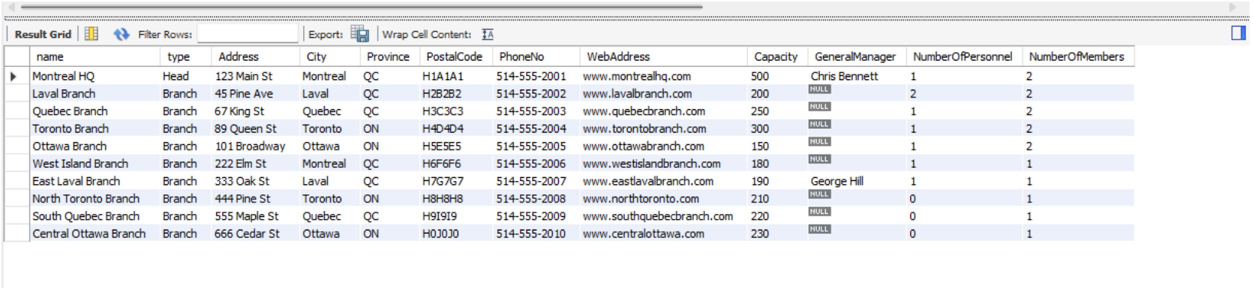
```
INSERT INTO Person (FirstName, LastName, DateOfBirth, SSN, MedicareNo, PhoneNo,
Address, City, Province, PostalCode, Email) VALUES
('Anna','Clark','1995-03-21','SSN001','MC001','514-555-1001','101 King
St','Toronto','ON','H1A1A1','anna.clark@gmail.com'),
('John','Martinez','1990-04-14','SSN002','MC002','514-555-1002','102 Pine
Ave','Laval','QC','H1A2A2','john.martinez@yahoo.com'),
('Eve','Johnson','1998-05-12','SSN003','MC003','514-555-1003','103 Queen
St','Quebec','QC','H1A3A3','eve.johnson@outlook.com'),
('Sam','Lee','1992-06-18','SSN004','MC004','514-555-1004','104 Main
St','Montreal','QC','H1A4A4','sam.lee@gmail.com'),
('Lily','Brown','1996-07-22','SSN005','MC005','514-555-1005','105 Elm
St','Ottawa','ON','H1A5A5','lily.brown@gmail.com'),
('Tom','Taylor','1997-08-30','SSN006','MC006','514-555-1006','106 Cedar
St','Toronto','ON','H1A6A6','tom.taylor@gmail.com'),
('Jane','Walker','1993-09-05','SSN007','MC007','514-555-1007','107 Oak
St','Laval','QC','H1A7A7','jane.walker@gmail.com'),
('Max','Anderson','1994-10-12','SSN008','MC008','514-555-1008','108 Maple
St','Montreal','QC','H1A8A8','max.anderson@gmail.com'),
('Grace','Lewis','1999-11-23','SSN009','MC009','514-555-1009','109 Birch
St','Toronto','ON','H1A9A9','grace.lewis@gmail.com'),
('Ryan','Smith','1989-12-01','SSN010','MC010','514-555-1010','110 Spruce
St','Ottawa','ON','H1B1B1','ryan.smith@gmail.com'),
('Olivia','Hall','1991-01-11','SSN011','MC011','514-555-1011','111 Pine
St','Montreal','QC','H1B2B2','olivia.hall@gmail.com'),
('Daniel','Young','1994-02-22','SSN012','MC012','514-555-1012','112 Cedar
St','Toronto','ON','H1B3B3','daniel.young@gmail.com'),
('Chloe','Adams','2010-03-15','SSN013','MC013','514-555-1013','113 King
St','Quebec','QC','H1B4B4','chloe.adams@gmail.com'),
('Lucas','Hughes','2011-04-10','SSN014','MC014','514-555-1014','114 Queen
St','Ottawa','ON','H1B5B5','lucas.hughes@gmail.com'),
('Emma','Price','2012-05-25','SSN015','MC015','514-555-1015','115 Oak
St','Toronto','ON','H1B6B6','emma.price@gmail.com'),
('Chris','Bennett','1980-06-20','SSN016','MC016','514-555-1016','201 Main
St','Montreal','QC','H1B7B7','chris.bennett@gmail.com'),
('Sophie','Cook','1985-07-12','SSN017','MC017','514-555-1017','202 Pine
St','Toronto','ON','H1B8B8','sophie.cook@gmail.com'),
('Mark','Green','1978-08-08','SSN018','MC018','514-555-1018','203 Cedar
St','Ottawa','ON','H1B9B9','mark.green@gmail.com'),
('Hannah','Ward','1983-09-14','SSN019','MC019','514-555-1019','204 Elm
St','Quebec','QC','H1C1C1','hannah.ward@gmail.com'),
('Peter','Wright','1987-10-09','SSN020','MC020','514-555-1020','205 Maple
St','Montreal','QC','H1C2C2','peter.wright@gmail.com'),
('Laura','Scott','1990-11-03','SSN021','MC021','514-555-1021','206 Birch
St','Laval','QC','H1C3C3','laura.scott@gmail.com'),
('George','Hill','1982-12-30','SSN022','MC022','514-555-1022','207 King
St','Toronto','ON','H1C4C4','george.hill@gmail.com'),
('Maria','Cooper','1975-01-19','SSN023','MC023','514-555-1023','208 Oak
St','Montreal','QC','H1C5C5','maria.cooper@gmail.com'),
('James','Reed','1976-02-28','SSN024','MC024','514-555-1024','209 Pine
St','Ottawa','ON','H1C6C6','james.reed@gmail.com'),
('Ella','Bell','1974-03-17','SSN025','MC025','514-555-1025','210 Cedar
St','Quebec','QC','H1C7C7','ella.bell@gmail.com'),
('David','Ward','1979-04-25','SSN026','MC026','514-555-1026','211 Elm
St','Montreal','QC','H1C8C8','david.ward@gmail.com'),
('Samantha','Brooks','1981-05-19','SSN027','MC027','514-555-1027','212 Maple
St','Toronto','ON','H1C9C9','samantha.brooks@gmail.com'),
('Adam','Kelly','1977-06-13','SSN028','MC028','514-555-1028','213 Birch
St','Laval','QC','H1D1D1','adam.kelly@gmail.com'),
('Victoria','James','1984-07-22','SSN029','MC029','514-555-1029','214 Spruce
St','Ottawa','ON','H1D2D2','victoria.james@gmail.com'),
('Ethan','Perry','1973-08-30','SSN030','MC030','514-555-1030','215 King
St','Montreal','QC','H1D3D3','ethan.perry@gmail.com');

INSERT INTO FamilyMember (FamilyMemberID) VALUES
(23),(24),(25),(26),(27),(28),(29),(30);

INSERT INTO ClubMember (ClubMemberID, Height, Weight, FamilyID) VALUES
(1,170.0,65.0,NULL),
(2,180.0,72.0,NULL),
(3,175.0,70.0,NULL),
(4,168.0,60.0,NULL),
(5,182.0,75.0,NULL),
(6,176.0,68.0,NULL),
(7,165.0,55.0,NULL),
(8,190.0,85.0,NULL),
(9,178.0,74.0,NULL),
(10,185.0,80.0,NULL),
(11,172.0,67.0,NULL),
(12,180.0,78.0,NULL),
(13,160.0,50.0,23),
(14,158.0,48.0,24),
(15,162.0,52.0,25);

INSERT INTO Personnel (PersonnelID, Role, Mandate) VALUES
(16,'Administrator','Salaried'),
(17,'Coach','Salaried'),
(18,'Assistant Coach','Volunteer'),
(19,'Captain','Salaried'),
(20,'Other','Volunteer'),
(21,'Coach','Volunteer'),
(22,'Administrator','Salaried');

INSERT INTO Location
(name,type,Address,City,Province,PostalCode,PhoneNo,WebAddress,Capacity) VALUES
('Montreal HQ','Head','123 Main
St','Montreal','QC','H1A1A1','514-555-2001','www.montrealhq.com',500),
('Laval Branch','Branch','45 Pine
Ave','Laval','QC','H2B2B2','514-555-2002','www.lavalbranch.com',200),
('Quebec Branch','Branch','67 King
St','Quebec','QC','H3C3C3','514-555-2003','www.quebecbranch.com',250),
('Toronto Branch','Branch','89 Queen
St','Toronto','ON','H4D4D4','514-555-2004','www.torontobranch.com',300),
('Ottawa Branch','Branch','101
Broadway','Ottawa','ON','H5E5E5','514-555-2005','www.ottawabranch.com',150),
('West Island Branch','Branch','222 Elm
St','Montreal','QC','H6F6F6','514-555-2006','www.westislandbranch.com',180),
('East Laval Branch','Branch','333 Oak
St','Laval','QC','H7G7G7','514-555-2007','www.eastlavalbranch.com',190),
('North Toronto Branch','Branch','444 Pine
St','Toronto','ON','H8H8H8','514-555-2008','www.northtoronto.com',210),
('South Quebec Branch','Branch','555 Maple
St','Quebec','QC','H9I9I9','514-555-2009','www.southquebecbranch.com',220),
('Central Ottawa Branch','Branch','666 Cedar
St','Ottawa','ON','H0J0J0','514-555-2010','www.centralottawa.com',230);

INSERT INTO ClubMemberLocation (ClubMemberID, LocationID, StartDate, EndDate)
VALUES
(1,1,'2023-01-10',NULL),
(2,2,'2023-02-05',NULL),
(3,3,'2023-03-15',NULL),
(4,4,'2023-04-12',NULL),
(5,5,'2023-05-20',NULL),
(6,6,'2023-06-25',NULL),
(7,7,'2023-07-05',NULL),
(8,8,'2023-08-18',NULL),
(9,9,'2023-09-30',NULL),
(10,10,'2023-10-11',NULL),
(11,1,'2023-11-20',NULL),
(12,2,'2023-12-01',NULL),
(13,3,'2024-01-15',NULL),
(14,4,'2024-02-10',NULL),
(15,5,'2024-03-05',NULL);

INSERT INTO Personnel_Location (PersonnelID, LocationID, StartDate, EndDate) VALUES
(16,1,'2023-01-01',NULL),
(17,2,'2023-02-01',NULL),
(18,3,'2023-03-01',NULL),
(19,4,'2023-04-01',NULL),
(20,5,'2023-05-01',NULL),
(21,6,'2023-06-01',NULL),
(22,7,'2023-07-01',NULL);

INSERT INTO FamilyMemberLocation (FamilyMemberID, LocationID, StartDate, EndDate)
VALUES
(23,1,'2023-01-15',NULL),
(24,2,'2023-02-20',NULL),
(25,3,'2023-03-25',NULL),
(26,4,'2023-04-18',NULL),
(27,5,'2023-05-22',NULL),
(28,6,'2023-06-10',NULL),
(29,7,'2023-07-05',NULL),
(30,8,'2023-08-12',NULL);

INSERT INTO Hobby (hobbyID, name) VALUES
(1,'Soccer'),
(2,'Basketball'),
(3,'Tennis'),
(4,'Swimming'),
(5,'Cycling'),
(6,'Running'),
(7,'Hockey'),
(8,'Volleyball'),
(9,'Table Tennis'),
(10,'Badminton');

INSERT INTO memberHobby (hobbyID, clubMemberID) VALUES
(1,1),
(2,2),
(3,3),
(4,4),
(5,5),
(6,6),
(7,7),
(8,8),
(9,9),
(10,10),
(1,11),
(2,12),
(3,13),
(4,14),
(5,15);

INSERT INTO Team (teamName, team_gender, LocationID) VALUES
('Tigers','Male',1),
('Lions','Female',2),
('Eagles','Male',3),
('Panthers','Female',4),
('Sharks','Male',5);

INSERT INTO teamLocation (teamID, LocationID, startDate, endDate) VALUES
(1,1,'2023-01-01',NULL),
(2,2,'2023-02-01',NULL),
(3,3,'2023-03-01',NULL),
(4,4,'2023-04-01',NULL),
(5,5,'2023-05-01',NULL);

INSERT INTO memberTeam (startDate, teamID, ClubMemberID) VALUES
('2023-01-05',1,1),
('2023-01-05',1,2),
('2023-02-10',2,3),
('2023-02-10',2,4),
('2023-03-12',3,5),
('2023-03-12',3,6),
('2023-04-20',4,7),
('2023-04-20',4,8),
('2023-05-15',5,9),
('2023-05-15',5,10);

UPDATE memberTeam
SET EndDate = '2024-06-30'
WHERE teamID = 1 AND ClubMemberID = 1;

UPDATE memberTeam
SET EndDate = '2024-05-10'
WHERE teamID = 2 AND ClubMemberID = 4;

UPDATE memberTeam
SET EndDate = '2024-07-01'
WHERE teamID = 4 AND ClubMemberID = 7;


INSERT INTO Payment (ClubMemberID, PaymentDate, Amount, Method, MembershipYear)
VALUES
(1,'2024-01-10',100.00,'Cash',2024),
(2,'2024-02-15',120.00,'Debit',2024),
(3,'2024-03-12',90.00,'Credit',2024),
(4,'2024-04-18',110.00,'Cash',2024),
(5,'2024-05-22',130.00,'Debit',2024),
(6,'2024-06-05',150.00,'Credit',2024),
(7,'2024-07-09',80.00,'Cash',2024),
(8,'2024-08-14',95.00,'Debit',2024),
(9,'2024-09-30',105.00,'Credit',2024),
(10,'2024-10-12',115.00,'Cash',2024);

INSERT INTO Relationship (relationshipTypeID, typeName) VALUES
(1,'Parent'),
(2,'Guardian'),
(3,'Other');
```

```
INSERT INTO minorFamilyHistory (relationshipTypeID, FamilyMemberID, ClubmemberID)
VALUES
(1,23,13),
(1,24,14),
(1,25,15);

#i inserted some more data so i can get the donation information
INSERT INTO Payment (ClubMemberID, PaymentDate, Amount, Method, MembershipYear)
VALUES (3, '2024-11-10', 210.00, 'Credit', 2024);
INSERT INTO Payment (ClubMemberID, PaymentDate, Amount, Method, MembershipYear)
VALUES (4, '2024-11-12', 200.00, 'Cash', 2024);
INSERT INTO Payment (ClubMemberID, PaymentDate, Amount, Method, MembershipYear)
VALUES (5, '2024-11-15', 200.00, 'Debit', 2024);
INSERT INTO Payment (ClubMemberID, PaymentDate, Amount, Method, MembershipYear)
VALUES (6, '2024-11-18', 180.00, 'Credit', 2024);
INSERT INTO Payment (ClubMemberID, PaymentDate, Amount, Method, MembershipYear)
VALUES (10, '2024-11-20', 200.00, 'Cash', 2024);
INSERT INTO Payment (ClubMemberID, PaymentDate, Amount, Method, MembershipYear)
VALUES (7, '2024-11-25', 180.00, 'Debit', 2024);
INSERT INTO Payment (ClubMemberID, PaymentDate, Amount, Method, MembershipYear)
VALUES (8, '2024-11-26',370.00, 'Credit', 2024);
INSERT INTO Payment (ClubMemberID, PaymentDate, Amount, Method, MembershipYear)
VALUES (9, '2024-11-27', 400.00, 'Cash', 2024);


#more hobby information
INSERT INTO memberHobby (hobbyID, clubMemberID) VALUES
(5,2),
(6,2),
(7,2),
(8,3),
(9,3),
(10,3);

DELETE FROM memberHobby
WHERE clubMemberID = 4;


-- Make Anna Clark also a Personnel (e.g., Coach)
```

```
INSERT INTO Personnel (PersonnelID, Role, Mandate)
VALUES (1, 'Coach', 'Volunteer');

-- Assign Anna Clark to a Personnel_Location (e.g., working at LocationID 2)
INSERT INTO Personnel_Location (PersonnelID, LocationID, StartDate, EndDate)
VALUES (1, 2, '2023-09-01', NULL);
```

DML Statements (Queries)

```
-- Query no 1

SELECT
    l.name,
    l.type,
    l.Address,
    l.City,
    l.Province,
    l.PostalCode,
    l.PhoneNo,
    l.WebAddress,
    l.Capacity,
    CONCAT(gm.FirstName, ' ', gm.LastName) AS GeneralManager,
    IFNULL(p.num_personnel, 0) AS NumberOfPersonnel,
    IFNULL(c.num_members, 0) AS NumberOfMembers
FROM
    Location l


LEFT JOIN (
    SELECT LocationID, COUNT(*) AS num_personnel
    FROM Personnel_Location
    GROUP BY LocationID
) p ON l.LocationID = p.LocationID


LEFT JOIN (
    SELECT LocationID, COUNT(*) AS num_members
    FROM ClubMemberLocation
    GROUP BY LocationID
) c ON l.LocationID = c.LocationID

LEFT JOIN (
    SELECT
        pl.LocationID,
        per.FirstName,
        per.LastName
    FROM Personnel_Location pl
    JOIN Personnel pe ON pl.PersonnelID = pe.PersonnelID
    JOIN Person per ON pe.PersonnelID = per.PersonID
    WHERE pe.Role = 'Administrator'
    AND pl.StartDate = (
        SELECT MIN(StartDate)
        FROM Personnel_Location pl2
        JOIN Personnel pe2 ON pl2.PersonnelID = pe2.PersonnelID
        WHERE pe2.Role = 'Administrator' AND pl2.LocationID = pl.LocationID
    )
) gm ON l.LocationID = gm.LocationID

ORDER BY NumberOfMembers DESC;
```

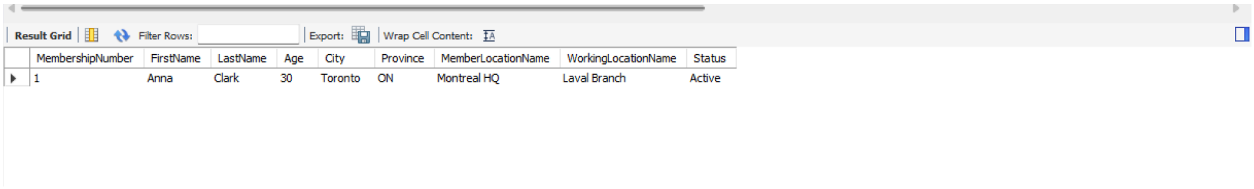


name	type	Address	City	Province	PostalCode	PhoneNo	WebAddress	Capacity	GeneralManager	NumberOfPersonnel	NumberOfMembers
Montreal HQ	Head	123 Main St	Montreal	QC	H1A1A1	514-555-2001	www.montrealhq.com	500	Chris Bennett	1	2
Laval Branch	Branch	45 Pine Ave	Laval	QC	H2B2B2	514-555-2002	www.lavalbranch.com	200	NULL	2	2
Quebec Branch	Branch	67 King St	Quebec	QC	H3C3C3	514-555-2003	www.quebecbranch.com	250	NULL	1	2
Toronto Branch	Branch	89 Queen St	Toronto	ON	H4D4D4	514-555-2004	www.torontobranch.com	300	NULL	1	2
Ottawa Branch	Branch	101 Broadway	Ottawa	ON	H5E5E5	514-555-2005	www.ottawabranch.com	150	NULL	1	2
West Island Branch	Branch	222 Elm St	Montreal	QC	H6F6F6	514-555-2006	www.westislandbranch.com	180	NULL	1	1
East Laval Branch	Branch	333 Oak St	Laval	QC	H7G7G7	514-555-2007	www.eastlavalbranch.com	190	George Hill	1	1
North Toronto Branch	Branch	444 Pine St	Toronto	ON	H8H8H8	514-555-2008	www.northtoronto.com	210	NULL	0	1
South Quebec Branch	Branch	555 Maple St	Quebec	QC	H9I9I9	514-555-2009	www.southquebecbranch.com	220	NULL	0	1
Central Ottawa Branch	Branch	666 Cedar St	Ottawa	ON	H0J0J0	514-555-2010	www.centralottawa.com	230	NULL	0	1

-- Query no 2

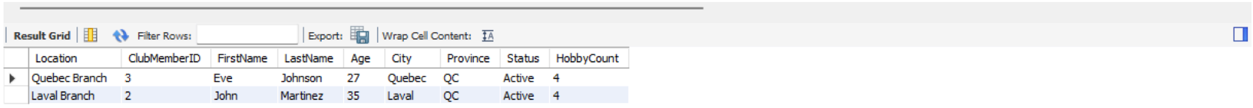
```
SELECT
    cm.ClubMemberID AS MembershipNumber,
    p.FirstName,
    p.LastName,
    TIMESTAMPDIFF(YEAR, p.DateOfBirth, CURDATE()) AS Age,
    p.City,
    p.Province,
    l1.name AS MemberLocationName,
    l2.name AS WorkingLocationName,
    CASE
        WHEN plo.EndDate IS NULL THEN 'Active'
        ELSE 'Inactive'
    END AS Status
FROM
    Person p
```

```
JOIN
   ClubMember cm ON cm.ClubMemberID = p.PersonID
JOIN
   ClubMemberLocation cml ON cm.ClubMemberID = cml.ClubMemberID
JOIN
   Location l1 ON cml.LocationID = l1.LocationID
JOIN
   Personnel pl ON pl.PersonnelID = p.PersonID
JOIN
   Personnel_Location plo ON plo.PersonnelID = pl.PersonnelID
JOIN
   Location l2 ON plo.LocationID = l2.LocationID
WHERE
   TIMESTAMPDIFF(YEAR, p.DateOfBirth, CURDATE()) >= 18
ORDER BY l2.`name` , Age;
```



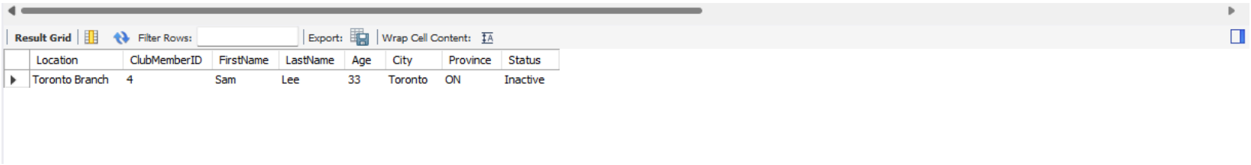
	MembershipNumber	FirstName	LastName	Age	City	Province	MemberLocationName	WorkingLocationName	Status
▶	1	Anna	Clark	30	Toronto	ON	Montreal HQ	Laval Branch	Active

```
#Query no 3
SELECT L.`name` AS Location, CM.ClubMemberID, P.FirstName, P.LastName,
TIMESTAMPDIFF(YEAR, P.DateOfBirth, CURDATE()) AS Age, L.City, L.Province,
CASE
  WHEN MT.EndDate IS NULL THEN 'Active'
  Else 'Inactive'
End as `Status`,
COUNT(DISTINCT MH.hobbyID) AS HobbyCount
FROM Location L
JOIN ClubMemberLocation CML ON CML.LocationID = L.LocationID
JOIN ClubMember CM ON CM.ClubMemberID = CML.ClubMemberID
JOIN Person P ON CM.ClubMemberID = P.PersonID
JOIN memberTeam MT ON CM.ClubMemberID = MT.ClubMemberID
JOIN memberHobby MH ON MH.ClubMemberID = MT.ClubMemberID
GROUP BY
   L.name,
   CM.ClubMemberID,
   P.FirstName,
   P.LastName,
   P.DateOfBirth,
   L.City,
   L.Province,
   MT.EndDate
HAVING COUNT(DISTINCT MH.hobbyID) >= 3
ORDER BY P.DateOfBirth DESC, L.`name`;
```



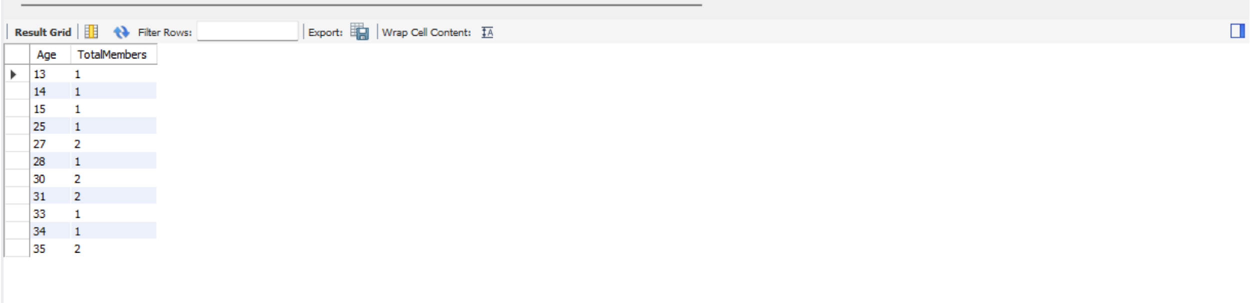
Location	ClubMemberID	FirstName	LastName	Age	City	Province	Status	HobbyCount
Quebec Branch	3	Eve	Johnson	27	Quebec	QC	Active	4
Laval Branch	2	John	Martinez	35	Laval	QC	Active	4

```
#Query no 4
SELECT L.`name` as Location, CM.ClubMemberID, P.FirstName, P.LastName,
TIMESTAMPDIFF(YEAR, P.DateOfBirth, CURDATE()) AS Age, L.City, L.Province,
CASE
  WHEN MT.EndDate IS NULL THEN 'Active'
  Else 'Inactive'
End as `Status`
FROM Location L
JOIN ClubMemberLocation CML ON CML.LocationID = L.LocationID
JOIN ClubMember CM ON CM.ClubMemberID = CML.ClubMemberID
JOIN Person P ON CM.ClubMemberID = P.PersonID
JOIN memberTeam MT ON CM.ClubMemberID = MT.ClubMemberID
LEFT JOIN memberHobby MH ON MH.ClubMemberID = MT.ClubMemberID
GROUP BY
    L.name,
    CM.ClubMemberID,
    P.FirstName,
    P.LastName,
    P.DateOfBirth,
    L.City,
    L.Province,
    MT.EndDate
HAVING COUNT(DISTINCT MH.hobbyID) = 0
ORDER BY L.`name`,P.DateOfBirth;
```



Location	ClubMemberID	FirstName	LastName	Age	City	Province	Status
Toronto Branch	4	Sam	Lee	33	Toronto	ON	Inactive

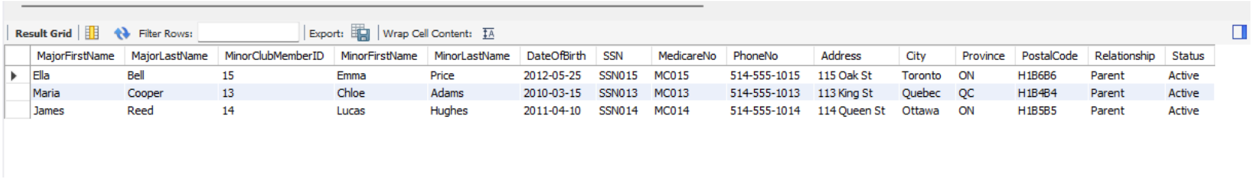
```
#Query no 5
SELECT TIMESTAMPDIFF(YEAR, p.DateOfBirth, CURDATE()) AS Age,
   COUNT(*) AS TotalMembers
FROM ClubMember cm
JOIN Person p ON cm.ClubMemberID = p.PersonID
GROUP BY Age
ORDER BY Age ASC;
```



Age	TotalMembers
13	1
14	1
15	1
25	1
27	2
28	1
30	2
31	2
33	1
34	1
35	2

```
#Query no 6
SELECT
   -- Major family member info
   pf.FirstName AS MajorFirstName,
   pf.LastName AS MajorLastName,

   -- Associated minor club member info
   cm.ClubMemberID AS MinorClubMemberID,
   pc.FirstName AS MinorFirstName,
   pc.LastName AS MinorLastName,
   pc.DateOfBirth,
   pc.SSN,
   pc.MedicareNo,
   pc.PhoneNo,
   pc.Address,
   pc.City,
   pc.Province,
   pc.PostalCode,
   rel.typeName AS Relationship,

   -- Active/Inactive Status
```

```
    CASE
        WHEN mt.EndDate IS NULL THEN 'Active'
        ELSE 'Inactive'
    END AS Status

FROM minorFamilyHistory mf
JOIN FamilyMember fm ON mf.FamilyMemberID = fm.FamilyMemberID
JOIN Person pf ON fm.FamilyMemberID = pf.PersonID
JOIN ClubMember cm ON mf.ClubMemberID = cm.ClubMemberID
JOIN Person pc ON cm.ClubMemberID = pc.PersonID
JOIN Relationship rel ON mf.relationshipTypeID = rel.relationshipTypeID
LEFT JOIN memberTeam mt ON cm.ClubMemberID = mt.ClubMemberID

WHERE
    TIMESTAMPDIFF(YEAR, pf.DateOfBirth, CURDATE()) >= 18

ORDER BY pf.LastName, pf.FirstName, cm.ClubMemberID;
```

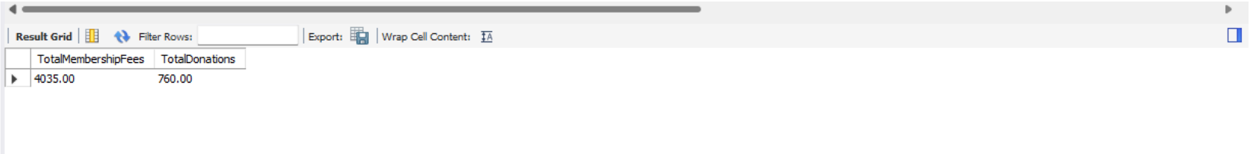


MajorFirstName	MajorLastName	MinorClubMemberID	MinorFirstName	MinorLastName	DateOfBirth	SSN	MedicareNo	PhoneNo	Address	City	Province	PostalCode	Relationship	Status
Ella	Bell	15	Emma	Price	2012-05-25	SSN015	MC015	514-555-1015	115 Oak St	Toronto	ON	H1B6B6	Parent	Active
Maria	Cooper	13	Chloe	Adams	2010-03-15	SSN013	MC013	514-555-1013	113 King St	Quebec	QC	H1B4B4	Parent	Active
James	Reed	14	Lucas	Hughes	2011-04-10	SSN014	MC014	514-555-1014	114 Queen St	Ottawa	ON	H1B5B5	Parent	Active

#Query no 7

```
SELECT
    SUM(CASE
            WHEN TIMESTAMPDIFF(YEAR, p.DateOfBirth, CONCAT(pay.MembershipYear,
'-01-01')) >= 18
            THEN LEAST(200, pay.Amount)
            ELSE 0
        END) AS TotalMembershipFees,

    SUM(CASE
            WHEN TIMESTAMPDIFF(YEAR, p.DateOfBirth, CONCAT(pay.MembershipYear,
'-01-01')) >= 18
            THEN GREATEST(0, pay.Amount - 200)
            ELSE 0
        END) AS TotalDonations
```

```
FROM
   Payment pay
JOIN ClubMember cm ON pay.ClubMemberID = cm.ClubMemberID
JOIN Person p ON cm.ClubMemberID = p.PersonID
WHERE
   pay.MembershipYear BETWEEN 2020 AND 2024;
```

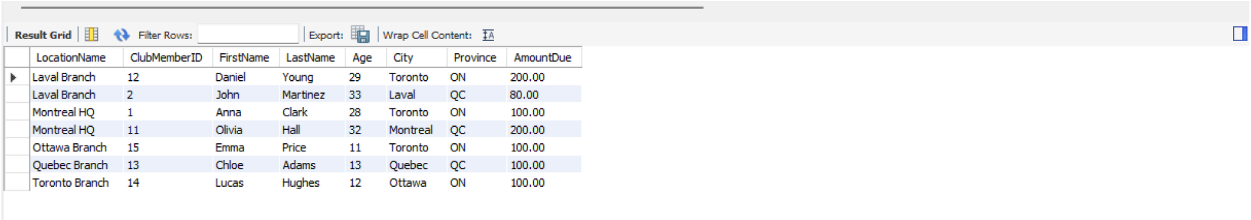


TotalMembershipFees	TotalDonations
4035.00	760.00

<u>#Query no 8</u>

```
SELECT
   l.name AS LocationName,
   cm.ClubMemberID,
   p.FirstName,
   p.LastName,
   TIMESTAMPDIFF(YEAR, p.DateOfBirth, '2024-01-01') AS Age,
   p.City,
   p.Province,
   CASE
      WHEN TIMESTAMPDIFF(YEAR, p.DateOfBirth, '2024-01-01') < 18
         THEN 100 - COALESCE(SUM(pay.Amount), 0)
      ELSE 200 - COALESCE(SUM(pay.Amount), 0)
   END AS AmountDue
FROM
   ClubMember cm
JOIN Person p ON cm.ClubMemberID = p.PersonID
JOIN ClubMemberLocation cml ON cm.ClubMemberID = cml.ClubMemberID
JOIN Location l ON cml.LocationID = l.LocationID
LEFT JOIN Payment pay ON cm.ClubMemberID = pay.ClubMemberID AND
pay.MembershipYear = 2024
GROUP BY
   cm.ClubMemberID, l.name, p.FirstName, p.LastName, p.DateOfBirth, p.City, p.Province
HAVING
   (TIMESTAMPDIFF(YEAR, p.DateOfBirth, '2024-01-01') < 18 AND
COALESCE(SUM(pay.Amount), 0) < 100)
```

```
 OR (TIMESTAMPDIFF(YEAR, p.DateOfBirth, '2024-01-01') >= 18 AND
COALESCE(SUM(pay.Amount), 0) < 200)
ORDER BY
    LocationName ASC,
    Age ASC;
```

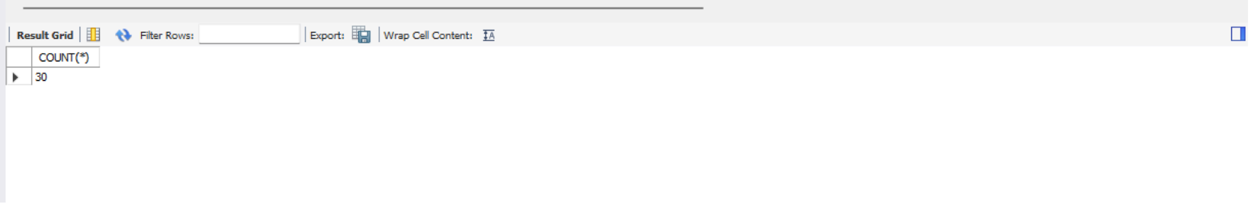


LocationName	ClubMemberID	FirstName	LastName	Age	City	Province	AmountDue
Laval Branch	12	Daniel	Young	29	Toronto	ON	200.00
Laval Branch	2	John	Martinez	33	Laval	QC	80.00
Montreal HQ	1	Anna	Clark	28	Toronto	ON	100.00
Montreal HQ	11	Olivia	Hall	32	Montreal	QC	200.00
Ottawa Branch	15	Emma	Price	11	Toronto	ON	100.00
Quebec Branch	13	Chloe	Adams	13	Quebec	QC	100.00
Toronto Branch	14	Lucas	Hughes	12	Ottawa	ON	100.00

*Count * Statements*

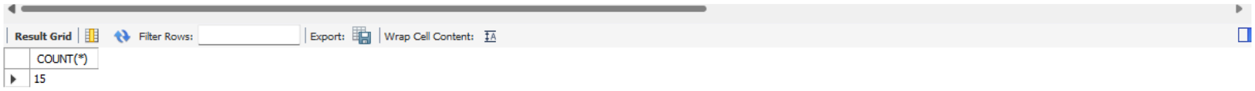
#TEST

```
SELECT COUNT(*) FROM Person;
```



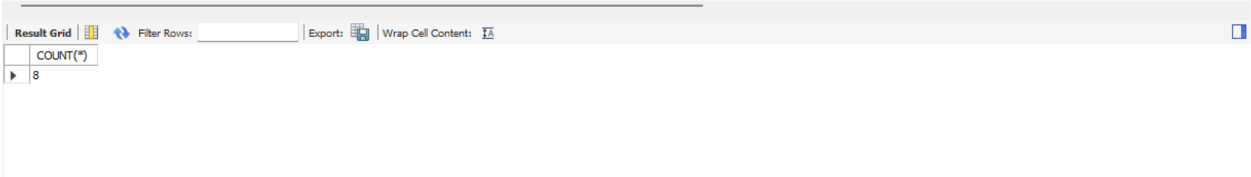
```
COUNT(*)
30
```

```
SELECT COUNT(*) FROM ClubMember;
```



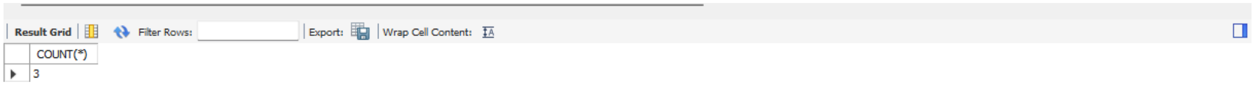
```
COUNT(*)
15
```

```
SELECT COUNT(*) FROM FamilyMember;
```



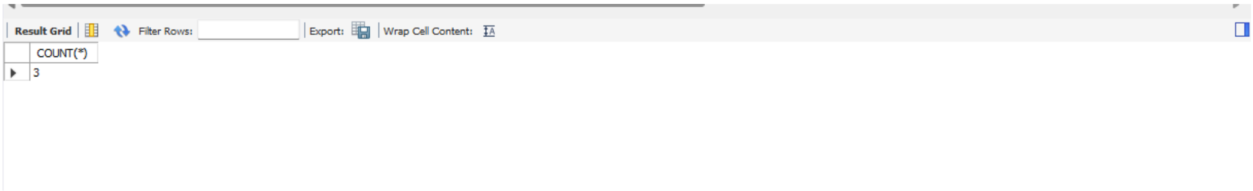
```
COUNT(*)
8
```

```
SELECT COUNT(*) FROM Relationship;
```



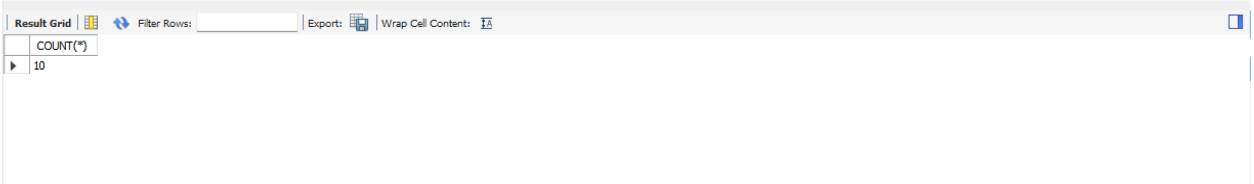
```
COUNT(*)
3
```

```
SELECT COUNT(*) FROM minorFamilyHistory;
```



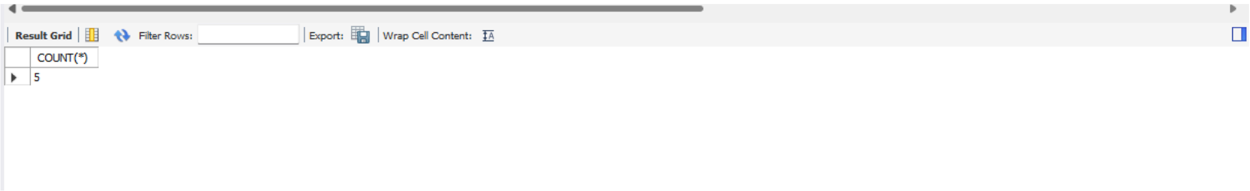
```
COUNT(*)
3
```

```
SELECT COUNT(*) FROM memberTeam;
```



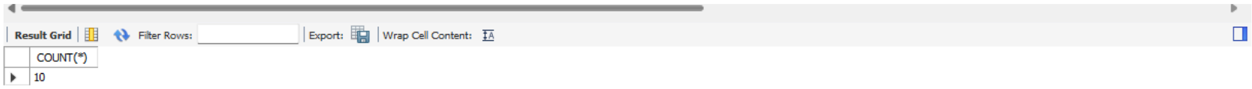
```
COUNT(*)
10
```

SELECT COUNT(*) FROM Team;



```
COUNT(*)
5
```

SELECT COUNT(*) FROM Location;



```
COUNT(*)
10
```

Contributions:

Angel Asencios (40176253)	● E/R database design ● DDL Statements ● E/R to Relations Conversions ● Count* Statements ● Queries (5 & 6) ● Project report
Farhan Ali Khan (40287970)	● I contributed in E/R diagram ● Helped with all the entities and relations ● Creation of tables ● Inserted data into the Tables ● Solved queries 1&2 ● Helped with the project prsentation
Ali Sher	● Help in the design of the E/R diagram ● Contributed to defining entity realtionships and logic table design ● Assisted in structuring data model and ensuring integrity constraints ● Queries (7 & 8) ● Participated in team discussions to refine the SQL logic and optimize the structure ● Help with presention and summarizing the projects objectives,E/R diagram and table design
Haaris Muhammad Khawaja (40261525)	● E/R database Design (Diagram) ● DDL Statements (Tables) ● Inserting Data ● Managing the database ● Queries (3 & 4) ● Contributed in Project Presentation(Queries and count statements)