Group member names: Aaron Henry, Max Phillips, Ulani Bouataveth, Dhara Patel, Ryan Bell, Michael Ly

**Exercise 7: Problem Solving** (10 points)

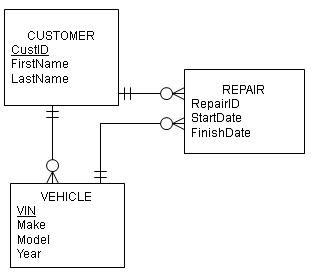
**Instructions:**

1. This is a group assignment. You only need to submit one solved exercise per group.
2. Provide your responses on this file, save it, and upload the file to the appropriate assignment response in blackboard.
3. You may upload as many attempts as you may like. Please note that only the most recent file uploaded file would be graded.
4. Do not handwrite any responses.
5. If you have any additional information you would like me to know about this assignment, you may provide that to me at the end of this document as a note.

**Use the data model shown below for this assignment:**

What to do:

1. This assignment must be completed in SQL Server. No other DBMS is allowed.
2. ***Even though this is a group assignment, every group member has to create the DB for this assignment in their assigned individual SQL Server account. Failure to do so would result in individual score of zero even if the group score is different.***
3. All SQL statements written by different team members should be collated in this one single document in its appropriate area. No exceptions.
4. Based on the data model shown above, write the CREATE TABLE commands to build these tables and their relationships.
5. Create these tables in your own database (CIS310xxx). Every team member should build the database in their own database.
6. Write the SQL commands to create the tables, their columns and the relationships shown in the data model. Copy paste these SQL statements in appropriate space for all team members for submission.
7. Be sure the CREATE TABLES statements are in the correct order. The primary key foreign key combination would dictate that.
8. Be sure that the INSERT INTO statements are also in the correct order.
9. All columns must be populated in these INSERT INTO statements.

Additional information about columns:

CustID – Integer, primary key, not null

FirstName – variable character, 12 characters, not null

LastName – variable character, 14 characters, not null

VIN – variable character, 10 characters, primary key

Make – variable character, 15 characters

Model – variable character, 20 characters

Year – integer

RepairID – Integer, primary key

StartDate – date

FinishDate – date

**NOTE: Foreign keys have not been listed but must be included in the SQL commands to create this database.**

1. Provide the SQL Statements for creating Customer, Vehicle, and Repair table.

Create Table CUSTOMER

(CustID Integer NOT NULL UNIQUE,

FirstName Varchar(12) NOT NULL,

LastName Varchar(14) NOT NULL,

Constraint CUSTOMER\_PK Primary Key (CustID));

Create Table VEHICLE

(VIN Varchar(10) NOT NULL UNIQUE,

Make Varchar(15),

Model Varchar(20),

Year Integer,

CustID Integer NOT NULL,

Constraint VEHICLE\_PK Primary Key (VIN),

Constraint VEHICLE\_FK1 Foreign Key (CustID) References CUSTOMER(CustID));

Create Table REPAIR

(RepairID Integer NOT NULL UNIQUE,

StartDate date,

FinishDate date,

CustID Integer NOT NULL,

VIN Varchar(10) NOT NULL,

Constraint REPAIR\_PK Primary Key (RepairID),

Constraint REPAIR\_FK1 Foreign Key (CustID) References CUSTOMER(CustID),

Constraint REPAIR\_FK2 Foreign Key (VIN) References VEHICLE(VIN));

1. Each member of the team would populate these tables with the following information:

Customer: Populate this table with last five digits of your student ID as a CustID, your first name, and your last name. Every team member is responsible for their own info. You are not responsible for adding info of other team members. If other team members do not enter their info into the database, this has no bearing on your individual grade.

|  |  |  |
| --- | --- | --- |
| CustID | FirstName | LastName |
| 58354 | Kevin | McCallister |

Vehicle: Pick at least one vintage car per team member of your choice and add this to your Vehicle database. Make up a somewhat realistic looking VIN number.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| VIN | Make | Model | Year | CustID |
| 6354355 | Chevrolet | Camaro | 1967 | 358354 |

Repair: Vintage cars need a lot of repairs. Unfortunately, your car needs repair as well. Create a realistic looking repair ID, a current date (today’s date) as a start date, a finish date which is a week from current date.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| RepairID | StartDate | FinishDate | CustID | VIN |
| 63573 | MM/DD/YYYY | MM/DD/YYYY | 58354 | 6354355 |

-- Populate Table

--

--

-- Aaron Henry

INSERT INTO CUSTOMER

VALUES (14007, 'Aaron', 'Henry');

INSERT INTO VEHICLE

VALUES (0000000001, 'Ford', 'T', 1908, 14007);

INSERT INTO REPAIR

VALUES (1, '10/26/2020', '11/02/2020', 14007, 0000000001);

--

--

-- Dhara Patel

Insert into CUSTOMER (CustID, FirstName, LastName)

Values ('5333033', 'Dhara', 'Patel');

Insert into VEHICLE (VIN, Make, Model, Year, CustID)

Values ('583018', 'Honda', 'Civic', '1956', '5333033');

Insert into REPAIR (RepairID, StartDate, FinishDate, CustID, VIN)

Values ('63947', '11/04/2020', '11/30/2020', '5333033', '583018');

--

--

-- Michael Ly

INSERT INTO CUSTOMER

VALUES ('5186135', 'Michael', 'Ly');

INSERT INTO VEHICLE

VALUES ('34523464', 'Toyota', 'Tundra', '1965', '5186135');

INSERT INTO REPAIR

VALUES ('45634291', '2020-10-28', '2020-11-13', '5186135', '828212314');

--

--

-- Max Phillips

-- Ulani Bouataveth

-- Ryan Bell