**SQL Exercise**

**Customer table**



**Picture 1**

**CustomerClass table**



**Picture 2**

**CustomerDetail view**  


**Picture 3**

Write SQL command and capture result.

* Write the correct SQL statement to create a new database called myDB.

CREATE DATABASE myDB;

* Write the correct SQL statement to create a new table called Customers.
* CustomerID - INT (PK, not null)
* FirstName - VARCHAR
* LastName - VARCHAR
* Email - VARCHAR
* PhoneNumber - VARCHAR
* ClassCode - INT

CREATE TABLE Customers (

CustomerID int NOT NULL PRIMARY KEY,

LastName varchar(255),

FirstName varchar(255),

Email varchar(255),

PhoneNumber varchar(255),

ClassCode int,

);

* Add a new column of type INT called Age.

ALTER TABLE Customers

ADD Age int;

* Insert 4 new records in the Customers table. (Picture 1)

INSERT INTO Customers (CustomerID, FirstName, LastName,

Email,PhoneNumber, ClassCode,Age)

VALUES (1,'FirstA','LastA','FirstA@gmail.com','0911234567',1,20),

(2,'FirstB','LastB','FirstB@hotmail.com','0929876543',2,21),

(3,'FirstC','LastC','FirstC@hotmail.com','0937894561',3,22),

(4,'FirstD','LastD','FirstD@gmail.com','0941253654',2,22);

* Write the correct SQL statement to create a new table called CustomerClass.
* ClassCode - INT (PK, not null)
* ClassDescription - VARCHAR

CREATE TABLE CustomerClass (

ClassCode int NOT NULL PRIMARY KEY,

ClassDescription varchar(255),

);

Insert 3 new records in the CustomerClass table. (Picture 2)  
  
 INSERT INTO CustomerClass(ClassCode, ClassDescription)

VALUES (1,'Classic'),

(2,'Gold'),

(3,'Platinum');

Create a foreign key linking the Customer and CustomerClass tables.  
ALTER TABLE Customers

ADD FOREIGN KEY (ClassCode) REFERENCES CustomerClass(ClassCode);

* Select all records of Customer and CustomerClass tables (Picture 1 and 2).

SELECT \* FROM Customers;

SELECT \* FROM CustomerClass;

Select record from Customer table where the FirstName value is ‘FirstD’.  
SELECT FirstName

FROM Customers

WHERE FirstName = 'FirstD';

* Select all records of FirstName, LastName, Age, ClassCode, ClassDescription of Customer and CustomerClass table.  
  SELECT \* FROM Customers LEFT JOIN CustomerClass ON Customers.ClassCode = CustomerClass.ClassCode;

Select all the different values from Age column in the Customer table.   
SELECT DISTINCT Age

FROM Customers;

* Select all records from the Customers table, sort the result alphabetically by the column PhoneNumber.  
  SELECT \* FROM Customers ORDER BY PhoneNumber ASC;
* Select all records where the value of an Email column contains the word "hotmail".  
  SELECT \* FROM Customers WHERE Email LIKE '%hotmail%';
* Select the record with the highest value of the ClassCode column.  
  SELECT MAX(ClassCode) FROM Customers ;
* Update the Email column to new email of CustomerID value is 1.  
  UPDATE Customers SET Email = 'newFirstA@gmail.com' WHERE CustomerID =1 ;
* Select the last record of the Customer table.  
  SELECT TOP 1 \* FROM Customers ORDER BY CustomerID DESC;
* Select the class description of the customer phone number included ‘7894561’.  
  SELECT CustomerClass.ClassDescription

FROM CustomerClass RIGHT JOIN Customers ON CustomerClass.ClassCode = Customers.ClassCode   
 WHERE PhoneNumber = '7894561' ;

Select all customers that their class = ‘Gold’  
SELECT \*

FROM CustomerClass RIGHT JOIN Customers ON CustomerClass.ClassCode = Customers.ClassCode

WHERE CustomerClass.ClassDescription = 'Gold';

* Delete the record from the Customers table where the FirstName value is ‘FirstD’.  
  DELETE FROM Customers WHERE FirstName='FirstD';

Create view called CustomerDetail (Picture 3).  
CREATE VIEW [CustomerDetail] AS

SELECT Customers.FirstName, Customers.LastName , Customers.Age , Customers.Email , Customers.PhoneNumber , CustomerClass.ClassCode

FROM Customers

INNER JOIN CustomerClass on CustomerClass.ClassCode = Customers.ClassCode;

* Select all records of CustomerDetail view.  
  SELECT \* FROM [CustomerDetail];