```sql
/*
GROUP 1 Project
Vehicle Insurance Database

Data was ingested into the database using Import Wizard
For this, the excel sheets were converted to csv (comma delimited) and each table was ⇗
  created
into a separate delimted file. Each file was then imported one by one.
*/
-- Create Database
CREATE DATABASE [Vehicle Insurance Database];

/*########## Define Keys ########*/

USE [Vehicle Insurance Database]

/* Defining all the Primary Keys*/
--Coverage
ALTER TABLE Coverage
ALTER COLUMN CoverageID varchar(50) not null
ALTER TABLE Coverage
ADD PRIMARY KEY(Coverage ID)

--PolicyEditLog
ALTER TABLE [Policy Edit Log]
ALTER COLUMN PolicyEditLog_ID varchar(50) not null
ALTER TABLE [Policy Edit Log]
ADD PRIMARY KEY(PolicyEditLog_ID)

--Policy
ALTER TABLE [Policy]
ALTER COLUMN Policy_ID varchar(50) not null
ALTER TABLE [Policy]
ADD PRIMARY KEY(Policy_ID)

--Bill
ALTER TABLE [Bill]
ALTER COLUMN Bill_ID varchar(50) not null
ALTER TABLE [Bill]
ADD PRIMARY KEY(Bill_ID)

--Vehicle Coverage
ALTER TABLE [vehicle Cobverage]
ALTER COLUMN Vehicle_Coverage_ID varchar(50) not null
ALTER TABLE [vehicle Cobverage]
ADD PRIMARY KEY(Vehicle_Covergae_ID)

--Policy coverage
ALTER TABLE [Policy Coverage]
ALTER COLUMN Policy_CoverageID varchar(50) not null
ALTER TABLE [Policy Coverage]
ADD PRIMARY KEY(Policy_CoverageID)
```

```sql
--Driver
ALTER TABLE [Driver]
ALTER COLUMN Driver_ID varchar(50) not null
ALTER TABLE [Driver]
ADD PRIMARY KEY(Driver_ID)

--Claim
ALTER TABLE [Claim]
ALTER COLUMN Claim_ID varchar(50) not null
ALTER TABLE [Claim]
ADD PRIMARY KEY(Claim_ID)

--Bank Payment Detail
ALTER TABLE [Bank Payment Detail]
ALTER COLUMN BankPaymentDetail_ID varchar(50) not null
ALTER TABLE [Bank Payment Detail]
ADD PRIMARY KEY(BankPaymentDetail_ID)

--Credit Card Payment
ALTER TABLE [CreditCardPayment]
ALTER COLUMN CreditCardPayment_ID varchar(50) not null
ALTER TABLE [CreditCardPayment]
ADD PRIMARY KEY(CreditCardPayment_ID)

--Vehicle
ALTER TABLE [Vehicle]
ALTER COLUMN Vehicle_ID varchar(50) not null
ALTER TABLE [Vehicle]
ADD PRIMARY KEY(Vehicle_ID)


--DriverDriver Vehile Driver
ALTER TABLE [Vehicle_Driver]
ALTER COLUMN Vehicle_Driver_ID varchar(50) not null
ALTER TABLE [Vehicle_Driver]
ADD PRIMARY KEY(Vehicle_Driver_ID)

--Driver Address
ALTER TABLE [Driver Address]
ALTER COLUMN DriverAddress_ID varchar(50) not null
ALTER TABLE [Driver Address]
ADD PRIMARY KEY(DriverAddress_ID)

--Driver Traffic Violation Code
ALTER TABLE [Driver_TrafficViolation_Record]
ALTER COLUMN Driver_TrafficViolation_Record_ID varchar(50) not null
ALTER TABLE [Driver_TrafficViolation_Record]
ADD PRIMARY KEY(Driver_TrafficViolation_Record_ID)

--Traffic Violation Code
ALTER TABLE [Traffic Violation Code]
```

```sql
ALTER COLUMN TrafficViolationCode_ID varchar(50) not null
ALTER TABLE [Traffic Violation Code]
ADD PRIMARY KEY(TrafficViolationCode_ID)

/*############ Relationships ############*/

--Relation between Policy and Policy Coverage
ALTER TABLE [Policy Coverage] ADD CONSTRAINT PolicyCoverage_FKColumn_FK
FOREIGN KEY (Policy_ID) REFERENCES Policy(Policy_ID)

--Relation between Policy and Bill
ALTER TABLE [Bill] ADD CONSTRAINT Bill_FKColumn_FK
FOREIGN KEY (Policy_ID) REFERENCES Policy(Policy_ID)

--Relation between Policy and Driver
ALTER TABLE [Driver] ADD CONSTRAINT Driver_FKColumn_FK
FOREIGN KEY (Policy_ID) REFERENCES Policy(Policy_ID)

--Relation between Policy and PolicyEditLog(on hold)
ALTER TABLE [Policy Edit Log] ADD CONSTRAINT Policyeditlog_FKColumn_FK
FOREIGN KEY (Policy_ID) REFERENCES Policy(Policy_ID)

--Relation between Coverage and Vehicle Coverage
ALTER TABLE [vehicle Coverage] ADD CONSTRAINT vechicleCoverage_FKColumn_FK
FOREIGN KEY (Coverage_ID) REFERENCES Coverage(Policy_ID)

--Relation between Coverage and Policy Coverage
ALTER TABLE [Policy Coverage] ADD CONSTRAINT Coverage_FKColumn_FK
FOREIGN KEY (Coverage_ID) REFERENCES Coverage(CoverageID)

--Relation between Policy and Vehicle (on hold)
ALTER TABLE [Vehicle] ADD CONSTRAINT vehicle_FKColumn_FK
FOREIGN KEY (Policy_ID) REFERENCES Policy(Policy_ID)

-- Both Primary Keys in DriverDriver Address
ALTER TABLE [Driver Driver Address]
ADD PRIMARY KEY (Driver_ID, DriverAddress_ID)

--Relation between Driver and Driver Driver Address
ALTER TABLE [Driver Driver Address] ADD CONSTRAINT address_FKColumn_FK
FOREIGN KEY (Driver_ID) REFERENCES Driver(Driver_ID)

--Relation between Driver Address and Driver Driver Address
ALTER TABLE [Driver Driver Address] ADD CONSTRAINT daddress_FKColumn_FK
FOREIGN KEY (DriverAddress_ID) REFERENCES [Driver Address](DriverAddress_ID)

--Relation between Driver and Driver Traffic Violation Record

ALTER TABLE [Driver Traffic Violation Record] ADD CONSTRAINT
  driverviolation_FKColumn_FK
FOREIGN KEY (Driver_ID) REFERENCES [Driver](Driver_ID)
```

```sql
--Relation between TrafficViolationCode and Driver Traffic Violation Record

ALTER TABLE [Driver_TrafficViolation_Record] ADD CONSTRAINT
  trafficviolation_FKColumn_FK
FOREIGN KEY (TrafficViolationCode_ID) REFERENCES [Traffic Violation Code]
  (TrafficViolationCode_ID)

--Relation between Driver and Vehicle Driver
ALTER TABLE [Vehicle_Driver] ADD CONSTRAINT vehicledriver_FKColumn_FK
FOREIGN KEY (Driver_ID) REFERENCES [Driver](Driver_ID)

--Relation between Vehicle and Vehicle Driver
ALTER TABLE [Vehicle_Driver] ADD CONSTRAINT driverID_FKColumn_FK
FOREIGN KEY (Vehicle_ID) REFERENCES [Vehicle](Vehicle_ID)

--Relation between Claim and Driver
ALTER TABLE [Claim] ADD CONSTRAINT  driverID_claim_FKColumn_FK
FOREIGN KEY (Driver_ID) REFERENCES [Driver](Driver_ID)

--Relation between Vehicle and Vehicle Driver- DriverID
ALTER TABLE [Vehicle_Driver] ADD CONSTRAINT driverID_vehicle_FKColumn_FK
FOREIGN KEY (Driver_ID) REFERENCES [Driver](Driver_ID)

--Relation between Driver and Driver_TrafficViolation_Record-DriverID
ALTER TABLE [Driver_TrafficViolation_Record] ADD CONSTRAINT  driverID_tvc_FKColumn_FK
FOREIGN KEY (Driver_ID) REFERENCES [Driver](Driver_ID)

--Relation between Vehicle Coverage and Vehicle - VehicleID
ALTER TABLE [vehicle Coverage] ADD CONSTRAINT vehicleID_FKColumn_FK
FOREIGN KEY (Vehicle_ID) REFERENCES [Vehicle](Vehicle_ID)

  -- change the data type of certain columns
ALTER TABLE dbo.[vehicle Coverage]
ALTER COLUMN Vehicle_Coverage_ID varchar(50) NOT NULL;

ALTER TABLE dbo.Vehicle_Driver
ALTER COLUMN Vehicle_Driver_ID varchar(50) NOT NULL;

ALTER TABLE dbo.[Driver Address]
ALTER COLUMN Driver_ID varchar(50) NOT NULL;

ALTER TABLE dbo.[Driver_TrafficViolation_Record]
ALTER COLUMN Driver_TrafficViolation_Record_ID varchar(50) NOT NULL;

ALTER TABLE dbo.[Driver_TrafficViolation_Record]
ALTER COLUMN Driver_ID varchar(50) NOT NULL;

ALTER TABLE dbo.[Driver_TrafficViolation_Record]
ALTER COLUMN Driver_TrafficViolation_Record_ID varchar(50) NOT NULL;

ALTER TABLE dbo.Driver
ALTER COLUMN DoB AS CAST(DoB as DATE)
```

```sql
ALTER TABLE dbo.Driver
ALTER COLUMN Dob
ALTER COLUMN CreatedDate


ALTER TABLE dbo.DRIVER
ADD Dob2 DATETIME;


ALTER TABLE dbo.DRIVER
DROP COLUMN Dob2;

--Verification

SELECT *
FROM dbo.[vehicle Coverage]
ORDER BY Vehicle_Coverage_ID;

SELECT *
FROM dbo.Driver;




-- Verification
USE [Vehicle Insurance Database];

SELECT *
FROM dbo.[Driver Address];

-- Modification
DROP TABLE dbo.[Driver Address];




/*############# Views ############*/

USE [Vehicle Insurance Database];

-- Create View - View 1
create view Drivers_WithTrafficViolation AS
select FirstName,LastName,TrafficViolationCode,CodeDescription from Driver d
join Driver_TrafficViolation_Record dtv on
d.Driver_ID = dtv.Driver_ID join [Traffic Violation Code] tvc on
tvc.TrafficViolationCode_ID = dtv.TrafficViolationCode_ID;

--Create View 2
CREATE VIEW Vehciles_WithTrafficViolation AS
SELECT v.Make, v.Model, v.Color, v.VehicleNumberPlate, tvc.TrafficViolationCode,
  tvc.CodeDescription
```

```sql
FROM dbo.Vehicle v
JOIN dbo.Driver d on
d.Policy_ID = v.Policy_ID
JOIN Driver_TrafficViolation_Record dtv on
d.Driver_ID = dtv.Driver_ID
JOIN [Traffic Violation Code] tvc on
tvc.TrafficViolationCode_ID = dtv.TrafficViolationCode_ID;


--Verification
SELECT *
FROM Vehciles_WithTrafficViolation;


select * from dbo.Drivers_WithTrafficViolation;

/*############### Encryption ##########*/

/* ENCRYPTION OF COLUMN SSN in Driver Table*/

--Create DMK
CREATE MASTER KEY
ENCRYPTION BY PASSWORD = 'Test_P@ssword';

--Certificate to protect symmetric key
CREATE CERTIFICATE Certificate1
WITH SUBJECT = 'Auto_Insurance_Certificate',
EXPIRY_DATE = '2026-10-31';

--Symmetric Key
CREATE SYMMETRIC KEY SSNkey
WITH ALGORITHM = AES_128
ENCRYPTION BY CERTIFICATE Certificate1;

--Open symmetric key
OPEN SYMMETRIC KEY SSNkey
DECRYPTION BY CERTIFICATE Certificate1;

--Encrypt SSN in Driver table

UPDATE Driver
SET SSN = ENCRYPTBYKEY(KEY_GUID('SSnkey'), SSN);

--View encrypted data
SELECT Driver_ID, FirstName, LastName, SSN FROM Driver

--View the SSN by 'DECRYPTBYKEY' function'
SELECT Driver_ID, FirstName, LastName, CONVERT(varchar, DECRYPTBYKEY(SSN))


/*### table level check constraint based on a function ###*/
```

```sql
--Contraints based on a function

CREATE FUNCTION CheckViolation (@DName varchar(50))
RETURNS smallint
AS
BEGIN
    DECLARE @Count smallint=0;
    SELECT @Count = COUNT(LastName)
            FROM Driver d join Driver_TrafficViolation_Record tvr on d.Driver_ID =
              tvr.Driver_ID
            WHERE LastName = @DName
            AND TrafficViolationCode_ID IS NOT NULL;
    RETURN @Count;
END;

------- Add table-level CHECK constraint based on the new function for the Vehicle
  table-------------
ALTER TABLE Vehicle ADD CONSTRAINT DriverViolationCheck CHECK (dbo.CheckViolation
  (LastName) <= 3);

--Constraints in Table dbo.Driver

ALTER TABLE dbo.[Driver]
ADD CONSTRAINT CHK_PhoneNumber
CHECK (PhoneNumber not like '%[^0-9]%'); /* phone number only numeric */

ALTER TABLE dbo.[Driver]
ADD CONSTRAINT CHK_CellNumber
CHECK (CellNumber not like '%[^0-9]%'); /* cell number only numeric */

ALTER TABLE dbo.[Driver]
ADD CONSTRAINT CHK_MaritalStatus
CHECK (MaritalStatus IN ('married', 'single')); /* MaritalStatus can be only Married
  or Single */

--Constraints in dbo.Bank Payment Detail
ALTER TABLE dbo.[Bank Payment Detail]
ADD CONSTRAINT CHK_Amount
CHECK (Amount not like '%[^0-9]%'); /* Amount only numeric */

ALTER TABLE dbo.[Bank Payment Detail]
ADD CONSTRAINT CHK_ZIPCode
CHECK (ZipCode not like '%[^0-9]%'); /* ZipCode only numeric */


--Constraints in dbo.Bill
ALTER TABLE dbo.[Bill]
ADD CONSTRAINT CHK_MinimumPayment
CHECK (MinimumPayment not like '%[^0-9]%'); /* MinimumPayment only numeric */

ALTER TABLE dbo.[Bill]
ADD CONSTRAINT CHK_Balance
```

```sql
CHECK (Balance not like '%[^0-9]%'); /* Balance only numeric */

ALTER TABLE dbo.[Bill]
ADD CONSTRAINT CHK_Status
CHECK (Status IN ('Pending', 'Completed')); /* Status can be only Pending or Completed↵
    */

-- Constraints on dbo.Claim

ALTER TABLE dbo.Claim
ADD CONSTRAINT CHK_

-- Constraints on dbo.Coverage
ALTER TABLE dbo.Coverage
ADD CONSTRAINT CHK_IsPolicyCoverage
CHECK (IsPolicyCoverage IN ('Yes', 'No'));

ALTER TABLE dbo.Coverage
ADD CONSTRAINT CHK_IsVehicleCoverage
CHECK (IsVehicleCoverage IN ('Yes', 'No'));

-- Constraints on dbo.CreditCardPayment
ALTER TABLE dbo.CreditCardPayment
ADD CONSTRAINT CHK_CardNumber
CHECK (CardNumber not like '%[^0-9]%');

ALTER TABLE dbo.CreditCardPayment
ADD CONSTRAINT CHK_ZipCode1
CHECK (ZipCode not like '%[^0-9]%');

--Constraints on dbo.[Driver Address]
ALTER TABLE dbo.[Driver Address]
ADD CONSTRAINT CHK_ZipCode2
CHECK (ZipCode not like '%[^0-9]%');

ALTER TABLE dbo.[Driver Address]
ADD CONSTRAINT CHK_IsItGarageAddress
CHECK (IsItGarageAddress IN('Yes', 'No'));

--Constraints on dbo.[Driver_TrafficViolation_Record]

ALTER TABLE dbo.[Driver_TrafficViolation_Record]
ADD CONSTRAINT CHK_Active
CHECK (Active IN ('Active', 'Inactive'));
```