Samuel Greenlee

SamsCarsSGG

Database Description

My database project has allowed me to track the dimensions of my favorite cars with SQL Server. Those dimensions include the tables Car, Engine, Fuel, Tires and Transmission. The break down of each table starting with the Car table has allowed me to assign a car identification number to each entered car incrementally. The Car table also has allowed me to enter credentials about each car that I have chosen. Those include columns such as year made, make, model, drivetrain, and the date added.

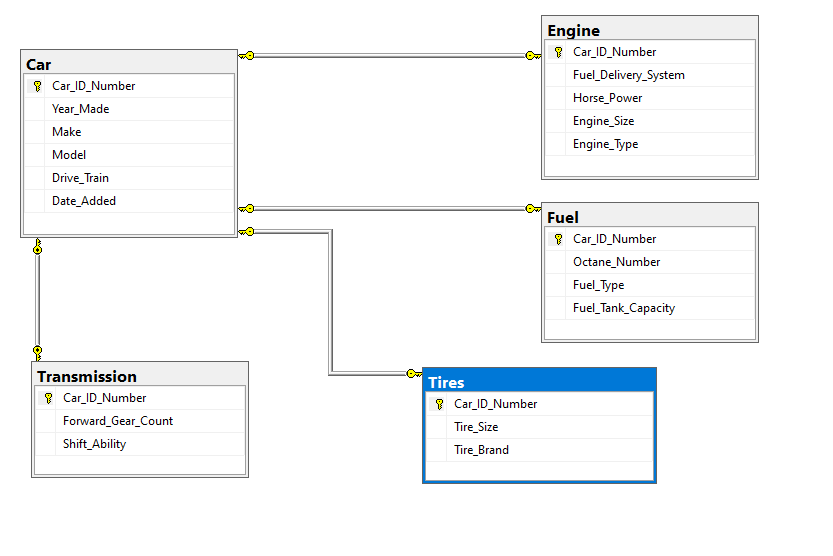
Moving on to my Engine table, it has allowed me to track other things about the cars that I have chosen for my database project. Allowing me to connect to the Car table, the Engine table brings down the car identification numbers. Then, sets it as a primary key. It also includes the fuel delivery system, horsepower, engine size, and the engine type.

Next up would be the Fuel table. Same as the Engine table, the Fuel table has allowed me to track other things about the cars that I have chosen for my database project. Allowing me to connect to the car table, the Fuel table brings down the car identification numbers. Then, sets it as a primary key. The Fuel table also includes the octane number, fuel type, and fuel tank capacity.

Second to last would be the Tires table. Like the other two tables, the Tires table has allowed me to track other things about the cars that I have chosen for my database project. Allowing me to connect to the Car table, the Tires table brings down the car identification numbers. Then, sets it as a primary key. The Tires table also includes the tire size, and the tire brand.

Lastly, would be the Transmission table. Like the other two tables, the Transmission table has allowed me to track other things about the cars that I have chosen for my database project. Allowing me to connect to the Car table, the Transmission table brings down the car identification numbers. Then, sets it as a primary key. The Tires table also includes the forward gear count, and the shift ability.

Database Diagram



Data Dictionary

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Table Name | Field Name | Data Type | Required  (Y/N) | Primary Key  (Y/N) | Foreign Key  (Y/N) | References |
| Car | Car\_ID\_Number | Int | Yes | Yes | No |  |
| Car | Year\_Made | Int | Yes | No | No |  |
| Car | Make | Varchar(50) | Yes | No | No |  |
| Car | Model | Varchar(50) | Yes | No | No |  |
| Car | Drive\_Train | Varchar(50) | Yes | No | No |  |
| Car | Date\_Added | Date | Yes | No | No |  |
| Engine | Car\_ID\_Number | Int | Yes | Yes | Yes | Car |
| Engine | Fuel\_Delivery\_System | Varchar(50) | Yes | No | No |  |
| Engine | Horse\_Power | Varchar(50) | Yes | No | No |  |
| Engine | Engine\_Size | Decimal(3,1) | Yes | No | No |  |
| Engine | Engine\_Type | Varchar(50) | Yes | No | No |  |
| Transmission | Car\_ID\_Number | Int | Yes | Yes | Yes | Car |
| Transmission | Forward\_Gear\_Count | Int | Yes | No | No |  |
| Transmission | Shift\_Ability | Varchar(50) | Yes | No | No |  |
| Fuel | Car\_ID\_Number | Int | Yes | Yes | Yes | Car |
| Fuel | Octane\_Number | Int | Yes | No | No |  |
| Fuel | Fuel\_Type | Varchar(50) | Yes | No | No |  |
| Fuel | Fuel\_Tank\_Capacity | Varchar(50) | Yes | No | No |  |
| Tires | Car\_ID\_Number | Int | Yes | Yes | Yes | Car |
| Tires | Tire\_Size | Varchar(50) | Yes | No | No |  |
| Tires | Tire\_Brand | Varchar(50) | Yes | No | No |  |

DDL Script

/\* SQL Server \*/

/\* Create the Database \*/

Create Database SamsCarsSGG;

GO

/\* Use statement to use the newly created database \*/

Use SamsCarsSGG;

/\* Create the Car table \*/

Create table dbo.Car (

Car\_ID\_Number int NOT NULL IDENTITY(1,1),

Year\_Made int NOT NULL,

Make varchar(50) NOT NULL,

Model varchar(50) NOT NULL,

Drive\_Train varchar(50) NOT NULL,

Date\_Added date NOT NULL,

Constraint pk\_Car primary Key(car\_ID\_Number));

/\* Create the Engine table \*/

Create table dbo.Engine (

Car\_ID\_Number int NOT NULL,

Fuel\_Delivery\_System varchar(50) NOT NULL,

Horse\_Power varchar(50) NOT NULL,

Engine\_Size decimal(3,1) NOT NULL,

Engine\_Type varchar(50) NOT NULL,

Constraint pk\_Engine primary Key(Car\_ID\_Number),

constraint fk\_Engine\_Car foreign key(Car\_ID\_Number) references Car(Car\_ID\_Number));

/\* Create the Transmission table \*/

Create table dbo.Transmission (

Car\_ID\_Number int NOT NULL,

Forward\_Gear\_Count int NOT NULL,

Shift\_Ability varchar(50) NOT NULL,

Constraint pk\_Transmission primary Key(Car\_ID\_Number),

constraint fk\_Transmission\_Car foreign key(Car\_ID\_Number) references Car(Car\_ID\_Number));

/\* Create the Fuel table \*/

Create table dbo.Fuel (

Car\_ID\_Number int NOT NULL,

Octane\_Number int NOT NULL,

Fuel\_Type varchar(50) NOT NULL,

Fuel\_Tank\_Capacity varchar(50) NOT NULL,

Constraint pk\_Fuel primary Key(Car\_ID\_Number),

constraint fk\_Fuel\_Car foreign key(Car\_ID\_Number) references Car(Car\_ID\_Number));

/\* Create the Tires table \*/

Create table dbo.Tires (

Car\_ID\_Number int NOT NULL,

Tire\_Size varchar(50) NOT NULL,

Tire\_Brand varchar(50) NOT NULL,

Constraint pk\_Tires primary Key(Car\_ID\_Number),

constraint fk\_Tires\_Car foreign key(Car\_ID\_Number) references Car(Car\_ID\_Number));

/\* Insert Data into the Car table \*/

insert into Car

(Year\_Made, Make, Model, Drive\_Train, Date\_Added)

values

(2008, 'Ford', 'Fusion SE', 'Front-Wheel-Drive','2021-03-26'),

(2008, 'Ford', 'Escape XLT', 'All-Wheel-Drive', '2021-03-26'),

(2002, 'Ford', 'Pontiac', 'Front-Wheel-Drive', '2021-03-26'),

(2003, 'BMW', 'M5', 'Rear-Wheel-Drive', '2021-03-27'),

(2001, 'Toyota', 'Camry LE', 'Front-Wheel-Drive', '2021-03-27'),

(2021, 'Chevrolet', 'Malibu', 'Front-Wheel-Drive', '2021-03-27'),

(2020, 'Buick', 'Enclave', 'All-Wheel-Drive', '2021-03-28'),

(2021, 'BMW', 'M2', 'Rear-Wheel-Drive', '2021-03-28'),

(2008, 'Nissan', 'Altima', 'Front-Wheel-Drive', '2021-03-28'),

(2021, 'Honda', 'Accord SE', 'Front-Wheel-Drive', '2021-03-29'),

(2021, 'Jeep', 'Wrangler', 'All-Wheel-Drive', '2021-03-30'),

(2008, 'Cadillac', 'CTS', 'Rear-Wheel-Drive', '2021-03-30'),

(2012, 'Toyota', 'Camry SE', 'Front-Wheel-Drive', '2021-03-30'),

(2013, 'Chevrolet', 'Sonic', 'Front-Wheel-Drive', '2021-03-31'),

(2002, 'Pontiac', 'Grand Am', 'Front-Wheel-Drive', '2021-03-31');

/\* Insert Data into the Engine table \*/

insert into Engine

(Car\_ID\_Number, Fuel\_delivery\_System, Horse\_Power, Engine\_Size, Engine\_Type)

values

(1, 'Injection', 221, '3.0', 'V6'),

(2, 'Injection', 200, '3.0', 'V6'),

(3, 'Injection', 240, '3.1', 'V6'),

(4, 'Injection', 394, '4.9', 'V8'),

(5, 'Injection', 194, '3.0', 'V6'),

(6, 'Injection', 250, '2.0', 'I4'),

(7, 'Injection', 310, '3.0', 'V6'),

(8, 'Injection', 405, '3.0', 'V6'),

(9, 'Injection', 270, '3.5', 'V6'),

(10, 'Injection', 192, '1.5', 'I4'),

(11, 'Injection', 470, '6.4', 'V8'),

(12, 'Injection', 304, '3.6', 'V6'),

(13, 'Injection', 178, '2.5', 'I4'),

(14, 'Injection', 138, '1.8', 'I4'),

(15, 'Injection', 175, '3.4', 'V6');

/\* Insert Data into the Transmission table \*/

insert into Transmission

(Car\_ID\_Number, Forward\_Gear\_Count, Shift\_Ability)

values

(1, 5,'Automatic'),

(2, 5,'Automatic'),

(3, 4,'Automatic'),

(4, 6,'Manual'),

(5, 5,'Manual'),

(6, 9,'Automatic'),

(7, 9,'Automatic'),

(8, 6,'Manual'),

(9, 6,'Manual'),

(10, 10,'Automatic'),

(11, 8,'Automatic'),

(12, 6,'Automatic'),

(13, 6,'Automatic'),

(14, 6,'Automatic'),

(15, 5,'Manual');

/\* Insert Data into the Tires table \*/

insert into Tires

(Car\_ID\_Number, Tire\_Size, Tire\_Brand)

values

(1, 'P205/60R16','Pirellie'),

(2, 'P235/70R16','Cooper'),

(3, 'P225/60R16','Goodyear'),

(4, 'P275/35R18','Firestone'),

(5, 'P205/65R15','Hancook'),

(6, 'P205/65R16','Goodyear'),

(7, 'P255/65R16','Cooper'),

(8, 'P265/35R16','Hancook'),

(9, 'P215/55R17','Pirellie'),

(10, 'P235/40R19','Michelin'),

(11, 'P255/70R18','Goodyear'),

(12, 'P235/55R17','Firestone'),

(13, 'P225/45R18','Hancook'),

(14, 'P195/65R15','Cooper'),

(15, 'P215/60R15','Cooper');

/\* Insert Data into the Fuel table \*/

insert into Fuel

(Car\_ID\_Number, Octane\_Number, Fuel\_Type, Fuel\_Tank\_Capacity)

values

(1, 87, '10% Ethenal', '17.5 gallons'),

(2, 87, 'Regular', '16.5 gallons'),

(3, 87, 'Regular', '17.5 gallons'),

(4, 91, 'Regular', '18.5 gallons'),

(5, 87, 'Regular', '18.5 gallons'),

(6, 87, 'Regular', '15.8 gallons'),

(7, 87, 'Regular', '22.0 gallons'),

(8, 91, 'Regular', '17.2 gallons'),

(9, 87, '10% Ethanol', '20.0 gallons'),

(10, 87, '10% Ethanol', '14.8 gallons'),

(11, 87, 'Regular', '21.5 gallons'),

(12, 87, 'Regular', '18.0 gallons'),

(13, 87, '10% Ethanol', '17.0 gallons'),

(14, 87, '10% Ethanol', '12.2 gallons'),

(15, 87, 'Regular', '14.3 gallons');

Queries

/\* Samuel Greenlee \*/

/\* SQL Server \*/

/\* Use Statement for SamsCarsSGG Database \*/

Use SamsCarsSGG;

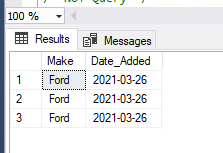
/\* AND Query \*/

/\* What are the cars who were added on 2021-03-26, and were Fords? \*/

Select Make, Date\_Added

From Car

Where Make = 'Ford' AND Date\_Added = '2021-03-26';



/\* OR Query \*/

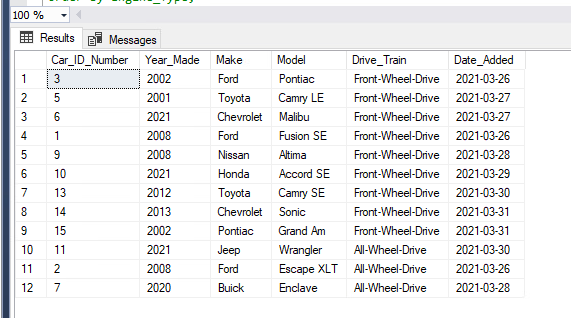
/\* What are the listings of cars that either contain Front-Wheel-Drive drive trains, or All-Wheel-Drive drive train? \*/

Select \*

From Car

Where Drive\_Train = 'Front-Wheel-Drive' OR Drive\_Train = 'All-Wheel-Drive'

Order by Drive\_Train Desc;



/\* NOT Query \*/

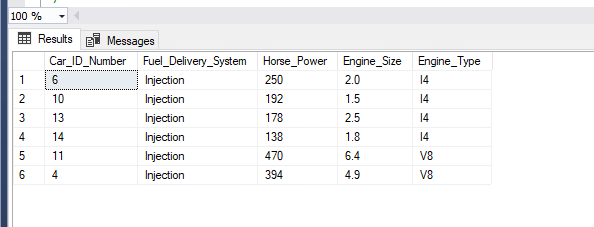
/\* What are the listings of engines that are not of the engine type V6? \*/

Select \*

From Engine

Where Not Engine\_Type = 'V6'

Order by Engine\_Type;



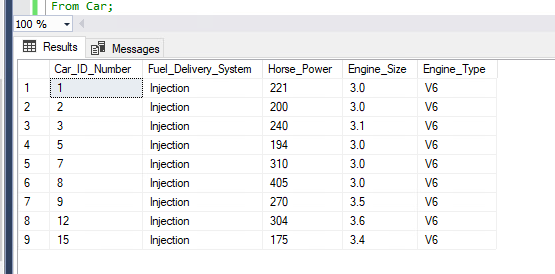
/\* BETWEEN, AND Query \*/

/\* What are the listings of engines that are between 3.0 and 4.0 in engine size? \*/

Select \*

From Engine

Where Engine\_Size Between 3.0 And 4.0;

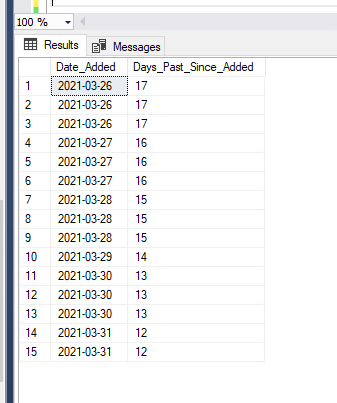


/\* Calculated Field Query\*/

/\* How long has past since the addition of each car? \*/

Select Date\_Added, DATEDIFF(Day, Date\_Added, GetDate()) As 'Days\_Past\_Since\_Added'

From Car;



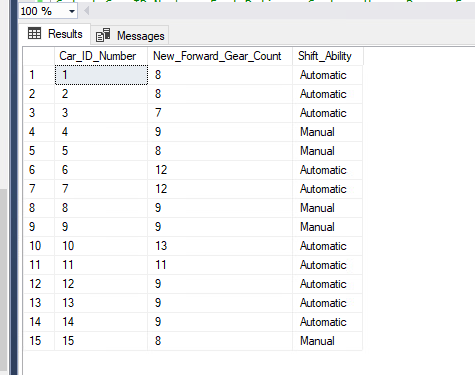
/\* Aggregate Function, Group By Query \*/

/\* What would my forward gear count be if I want to add three more to each transmission? \*/

Select Car\_ID\_Number, SUM(Forward\_Gear\_Count + 3) as 'New\_Forward\_Gear\_Count', Shift\_Ability

From Transmission

Group By Car\_ID\_Number, Shift\_Ability;



/\* Aggregate Function, Group By and Having Query\*/

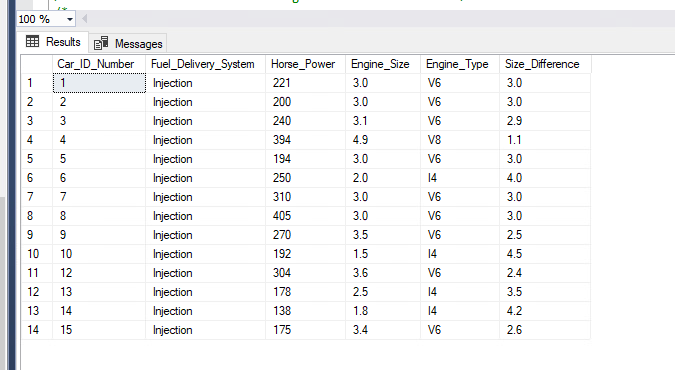
/\* How much bigger would each engine listing be if it was under the engine size of 6.0?\*/

Select Car\_ID\_Number, Fuel\_Delivery\_System, Horse\_Power, Engine\_Size, Engine\_Type, SUM(6.0 - Engine\_Size) as 'Size\_Difference'

From Engine

Group By Car\_ID\_Number, Fuel\_Delivery\_System, Horse\_Power, Engine\_Size, Engine\_Type

Having Engine\_Size < 6.0;

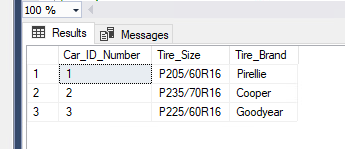


/\* Top Values Query \*/

/\* What are the first three listings of the tires table? \*/

Select Top 3 \*

From Tires;



/\* Inner Join Query \*/

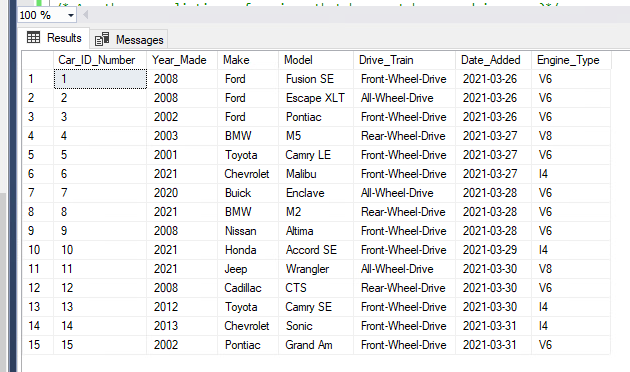
/\* What car goes with what engine type?\*/

Select c.Car\_ID\_Number, Year\_Made, Make, Model, Drive\_Train, Date\_Added, Engine\_Type

From Car as c

Inner Join Engine as e

On c.Car\_ID\_Number = e.Car\_ID\_Number;



/\* Outer Join, AND, WHERE Query \*/

/\* Are there any listings of engines that have not been used in a car?\*/

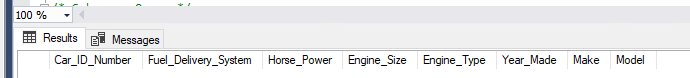
Select e.Car\_ID\_Number, Fuel\_Delivery\_System, Horse\_Power, Engine\_Size, Engine\_Type, Year\_Made, Make, Model

From Engine as e

Left Outer Join Car as c

On e.Car\_ID\_Number = c.Car\_ID\_Number

Where Year\_Made = Null AND Make Is Null AND Model Is Null;



/\* Subquery Query \*/

/\* What are the car listings for cars who have the fuel tank capacity of 17.5 gallons? \*/

Select \*

From Car

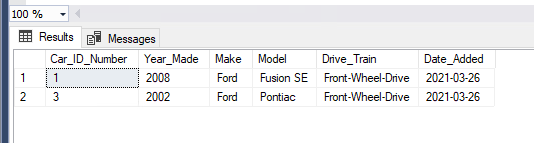
Where Car\_ID\_Number IN (

Select Car\_ID\_Number

From Fuel

Where Fuel\_Tank\_Capacity = '17.5 gallons'

);



Views

/\* Samuel Greenlee \*/

/\* SQL Server \*/

/\* Use Statement for SamsCarsSGG Database \*/

Use SamsCarsSGG;

/\* Not View \*/

Create View EnginesNotV6

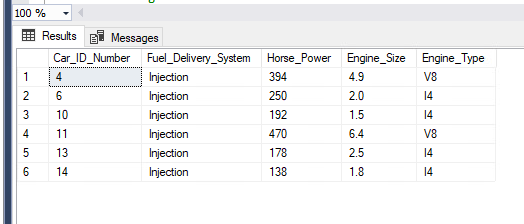
As

Select \*

From Engine

Where Not Engine\_Type = 'V6';

Select \* From EnginesNotV6;



/\* Aggregate Function, Group By and Having View \*/

Create View EngineSizeDifference

As

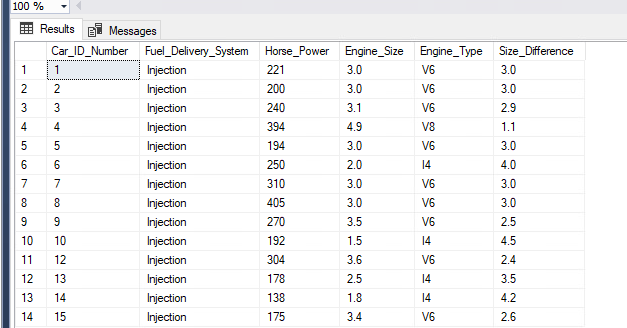
Select Car\_ID\_Number, Fuel\_Delivery\_System, Horse\_Power, Engine\_Size, Engine\_Type, SUM(6.0 - Engine\_Size) as 'Size\_Difference'

From Engine

Group By Car\_ID\_Number, Fuel\_Delivery\_System, Horse\_Power, Engine\_Size, Engine\_Type

Having Engine\_Size < 6.0;

Select \* From EngineSizeDifference;



/\* Inner Join View \*/

Create View CarEngineTypeMatch

As

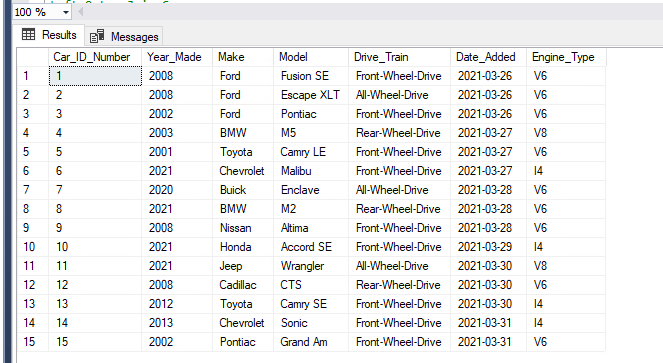
Select c.Car\_ID\_Number, Year\_Made, Make, Model, Drive\_Train, Date\_Added, Engine\_Type

From Car as c

Inner Join Engine as e

On c.Car\_ID\_Number = e.Car\_ID\_Number;

Select \* From CarEngineTypeMatch;



/\* Outer Join, AND, WHERE View \*/

Create View UnusedEngines

As

Select e.Car\_ID\_Number, Fuel\_Delivery\_System, Horse\_Power, Engine\_Size, Engine\_Type, Year\_Made, Make, Model

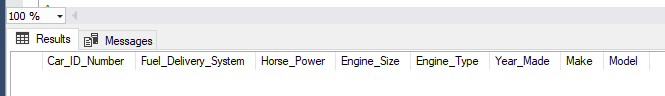
From Engine as e

Left Outer Join Car as c

On e.Car\_ID\_Number = c.Car\_ID\_Number

Where Year\_Made Is Null AND Make Is Null AND Model = Null;

Select \* From UnusedEngines;



/\* Subquery View \*/

Create View CarsWithI4Engines

As

Select \*

From Car

Where Car\_ID\_Number IN (

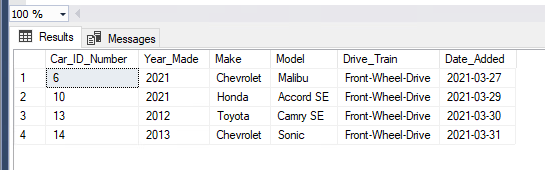
Select Car\_ID\_Number

From Engine

Where Engine\_Type = ‘I4’

);

Select \* From CarsWithI4Engines;



Stored Procedures

/\* Samuel Greenlee \*/

/\* SQL Server \*/

/\* Use Statement for SamsCarsSGG Database \*/

Use SamsCarsSGG;

/\* Stored Procedure One\*/

CREATE PROC daysPast

AS

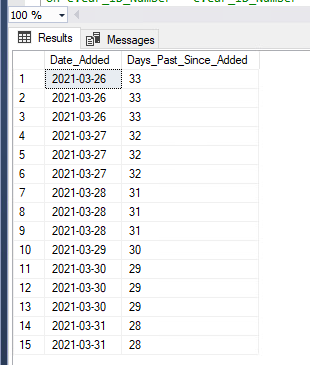
Begin

Select Date\_Added, DATEDIFF(Day, Date\_Added, GetDate()) As 'Days\_Past\_Since\_Added'

From Car

End;

exec daysPast;



/\* Stored Procedure Two \*/

CREATE PROC EngineMatch

@Engine varchar(30) = ''

AS

Begin

Select Engine\_Type, c.Car\_ID\_Number, Year\_Made, UPPER(Make) AS 'Make', UPPER(Model) AS 'Model', UPPER(Drive\_Train) AS 'Drive\_Train', Date\_Added

From Engine as e

Inner Join Car as c

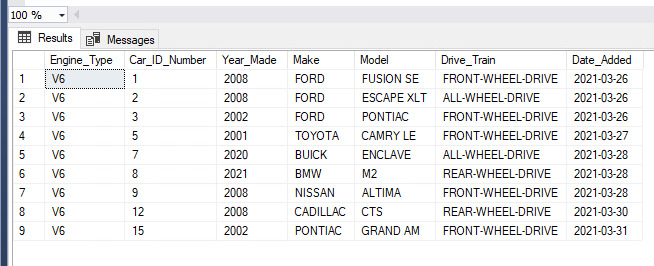
On e.Car\_ID\_Number = c.Car\_ID\_Number

Group By Engine\_Type, c.Car\_ID\_Number, Year\_Made, Make, Model, Drive\_Train, Date\_Added

Having Engine\_Type Like @Engine

End;

exec EngineMatch 'V6';



Logins and Users

/\* Samuel Greenlee \*/

/\* I used SQL Server \*/

/\* Use Statement \*/

Use SamsCarsSGG;

/\*Login Creation \*/

CREATE LOGIN User\_SGG WITH PASSWORD = 'CIT%2411',  
DEFAULT\_DATABASE = SamsCarsSGG,  
CHECK\_EXPIRATION=OFF,  
CHECK\_POLICY=OFF;

/\* User Creation \*/

CREATE USER User\_SGG

FOR LOGIN User\_SGG

WITH DEFAULT\_SCHEMA = dbo

/\* Grant Permission Statements \*/

GRANT INSERT, UPDATE, DELETE

ON Car

TO User\_SGG;

GRANT INSERT, UPDATE, DELETE

ON CarsWithI4Engines

TO User\_SGG;