MotorsCertification Database Management Project

By: Deshni Stanley

Data Science and Machine Learning Internship

Introduction:

HerculesMotoCorp, a leading automobile retailer and garage handler in the United States, specializes in a diverse range of vehicles, from standard automobiles to custom-built models. Given the scale and complexity of their operations, an efficient and well-structured database is essential for seamless record-keeping, compliance with federal regulations, and transparent data access for shareholders.

To address this need, our company has been entrusted with designing a comprehensive database solution that streamlines day-to

-day operations, enhances data accessibility, and ensures regulatory adherence. As the Database Administrator, my objective is to develop a system that empowers HerculesMotoCorp employees with simplified workflows while enabling external stakeholders to analyze key business insights effortlessly.

This project aims to deliver a robust, scalable, and secure database, catering to operational efficiency and strategic decision-making within the organization.

As a part of the MotorsCertification Database Certification Project, the following documents are enclosed

- Design.sql – Table Design Scripts.
- Insertion.sql – Data/Values Insertion
- Task.sql – All Tasks with comments (from Question 4)
- MotorsCertification.bak – Database Backup.

Step 1 Database Creation:

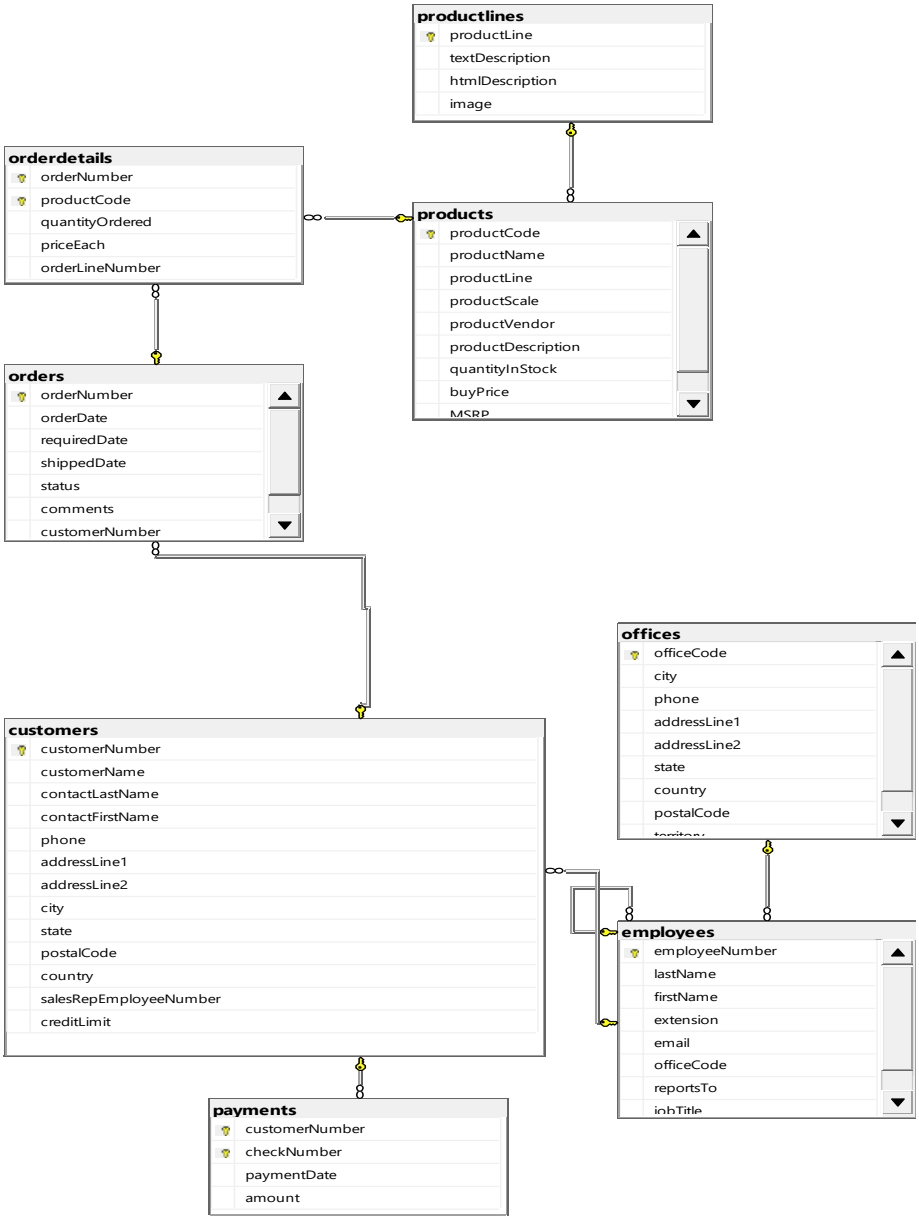
Create the MotorsCertification database in SQL Server. This initializes our workspace

Step 2: Build and Setup an ER Model

Defining the entity relationships based on project parameters:

- orderdetails references orders (orderNumber) & products (productCode).
- customers references employees (salesRepEmployeeNumber).
- employees references offices (officeCode) and itself (reportsTo).
- orders references customers (customerNumber).
- products references productlines (productLine).

The generated ER diagram in SSMS with all above relationships is presented below as a picture



Step 3: Table Creation

The table is now systematically designed according to the specification provided and found in the design.sql script attached.

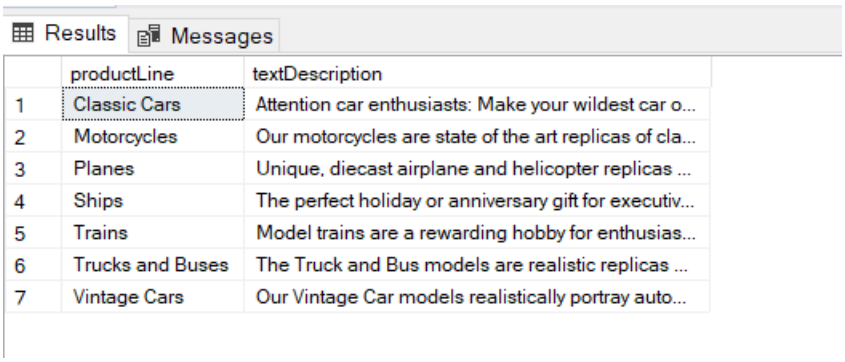
Step 4: Insert the Values

The values have been inserted in the following order and can be found in insert.sql file
Insert into offices
Insert into employees (linked to offices)
Insert into customers (linked to employees)
Insert into orders (linked to customers)
Insert into payments (linked to customers)
Insert into productlines
Insert into products (linked to productlines)
Insert into orderdetails (linked to orders and products)

Task (tasks.sql)

1. **Delete the columns in productlines which are useless that do not infer anything.**

   Used alter table to perform the function.
   ALTER TABLE productlines DROP COLUMN htmlDescription;
   ALTER TABLE productlines DROP COLUMN image;


Results Messages

	productLine	textDescription
1	Classic Cars	Attention car enthusiasts: Make your wildest car o...
2	Motorcycles	Our motorcycles are state of the art replicas of cla...
3	Planes	Unique, diecast airplane and helicopter replicas ...
4	Ships	The perfect holiday or anniversary gift for executiv...
5	Trains	Model trains are a rewarding hobby for enthusias...
6	Trucks and Buses	The Truck and Bus models are realistic replicas ...
7	Vintage Cars	Our Vintage Car models realistically portray auto...

Since htmlDescription and image are marked as NULL and do not contribute meaningfully to the project requirements, it has been deleted

2. Use a select statement to verify all insertions as well as updates.

Used the select option to verify the updates of all the insertion.

```
SELECT TOP 3 * FROM offices;
SELECT TOP 3 * FROM employees;
SELECT TOP 3 * FROM customers;
SELECT TOP 3 * FROM orders;
SELECT TOP 3 * FROM payments;
SELECT TOP 3* FROM products;
SELECT TOP 3 * FROM orderdetails;
```

Results Messages

	officeCode	city	phone	addressLine1	addressLine2	state	country	postalCode	territory
1	1	San Francisco	+1 650 219 4782	100 Market Street	Suite 300	CA	USA	94080	NA
2	2	Boston	+1 215 837 0825	1550 Court Place	Suite 102	MA	USA	02107	NA
3	3	NYC	+1 212 555 3000	523 East 53rd Street	apt. 5A	NY	USA	10022	NA

	employeeNumber	lastName	firstName	extension	email	officeCode	reportsTo	jobTitle
1	1002	Murphy	Diane	x5800	dmurphy@classicmodelcars.com	1	NULL	President
2	1056	Patterson	Mary	x4611	mpatterso@classicmodelcars.com	1	1002	VP Sales
3	1076	Firrelli	Jeff	x9273	jfirrelli@classicmodelcars.com	1	1002	VP Marketing

	customerNumber	customerName	contactLastName	contactFirstName	phone	addressLine1	addressLine2	city	state	postalCode	country	salesRepEmployeeNumber	creditLimit
1	103	Atelier graphique	Schmitt	Carine	40.32.2555	54, rue Royale	NULL	Nantes	NULL	44000	France	1370	21000
2	112	Signal Gift Stores	King	Jean	7025551838	8489 Strong St.	NULL	Las Vegas	NV	83030	USA	1166	71800
3	114	Australian Collectors, Co.	Ferguson	Peter	03 9520 4555	636 St Kilda Road	Level 3	Melbourne	Victoria	3004	Australia	1611	117300

	orderNumber	orderDate	requiredDate	shippedDate	status	comments	customerNumber
1	10100	2003-01-06	2003-01-13	2003-01-10	Shipped	NULL	103
2	10101	2003-01-09	2003-01-18	2003-01-11	Shipped	Check on availability.	128
3	10102	2003-01-10	2003-01-18	2003-01-14	Shipped	NULL	112

	customerNumber	checkNumber	paymentDate	amount
1	103	HQ336336	2004-10-19	6066.78
2	103	JM555205	2003-06-05	14571.44
3	103	OM314933	2004-12-18	1676.14

	productCode	productName	productLine	productScale	productVendor	productDescription	quantityInStock	buyPrice	MSRP
1	S10_1678	1969 Harley Davidson Ultimate Chopper	Motorcycles	01:10	Min Lin Diecast	This replica features working kickstand, front su...	7933	48.81	95.7
2	S10_1949	1952 Alpine Renault 1300	Classic Cars	01:10	Classic Metal Creations	Turnable front wheels; steering function; detail...	7305	98.58	214.3
3	S10_2016	1996 Moto Guzzi 1100i	Motorcycles	01:10	Highway 66 Mini Clas...	Official Moto Guzzi logos and insignias, saddle...	6625	68.99	118....

	orderNumber	productCode	quantityOrdered	priceEach	orderLineNumber
1	10100	S10_1949	30	136	3
2	10101	S10_4962	50	55.09	2
3	10102	S12_1666	22	75.46	4

3. Find out the highest and the lowest amount.

```
--Find out the highest and the lowest amount:
SELECT MAX(amount) AS HighestAmount, MIN(amount) AS LowestAmount FROM payments;
```

100 %

Results Messages

	HighestAmount	LowestAmount
1	101244.59	1491.38

Used max and min to get the desired output and result. It is shown in the above image.

4. Give the unique count of customerName from customers.

```
SELECT COUNT(DISTINCT customerName) AS UniqueCustomerCount
FROM customers;
```

100 %

Results Messages

	UniqueCustomerCount
1	20

Used distinct function to get the desired output and result is shown in the above image.

5. Create a view from customers and payments named cust_payment and select customerName, amount, contactLastName, contactFirstName who have paid.

```
SELECT * FROM cust_payment;
```

100 %

Results | Messages

	customerName	amount	contactLastName	contactFirstName
1	Atelier graphique	6066.78	Schmitt	Carine
2	Atelier graphique	14571.44	Schmitt	Carine
3	Atelier graphique	1676.14	Schmitt	Carine
4	Signal Gift Stores	14191.12	King	Jean
5	Signal Gift Stores	32641.98	King	Jean
6	Signal Gift Stores	33347.88	King	Jean
7	Australian Collectors, Co.	45864.03	Ferguson	Peter
8	Australian Collectors, Co.	82261.22	Ferguson	Peter
9	Australian Collectors, Co.	7565.08	Ferguson	Peter
10	Australian Collectors, Co.	44894.74	Ferguson	Peter
11	La Rochelle Gifts	19501.82	Labrune	Janine
12	La Rochelle Gifts	47924.19	Labrune	Janine
13	La Rochelle Gifts	49523.67	Labrune	Janine
14	Baane Mini Imports	50218.95	Bergulfsen	Jonas
15	Baane Mini Imports	1491.38	Bergulfsen	Jonas
16	Baane Mini Imports	17876.32	Bergulfsen	Jonas
17	Baane Mini Imports	34638.14	Bergulfsen	Jonas

Drop View

views don’t store actual data, so don't have to truncate lets only Drop

```
DROP VIEW cust_payment;
```

100 %

Messages

```
Commands completed successfully.

Completion time: 2025-05-23T02:48:22.8739180+05:30
```

6. Create a stored procedure on products which displays productLines for Classic Cars.

```
EXEC GetClassicCars;
```

100 %

Results | Messages

	productName	productLine
1	1952 Alpine Renault 1300	Classic Cars
2	1972 Alfa Romeo GTA	Classic Cars
3	1962 LanciaA Delta 16V	Classic Cars
4	1968 Ford Mustang	Classic Cars
5	2001 Ferrari Enzo	Classic Cars
6	1969 Corvair Monza	Classic Cars
7	1968 Dodge Charger	Classic Cars
8	1969 Ford Falcon	Classic Cars

7. Create a function to get the creditLimit of customers less than 96800.

```
SELECT * FROM GetCreditLimitBelow96800();
```

100 %

Results | Messages

	customerNumber	customerName	creditLimit
1	103	Atelier graphique	21000
2	112	Signal Gift Stores	71800
3	121	Baane Mini Imports	81700
4	125	Havel & Zbyszek Co	0
5	128	Blauer See Auto, Co.	59700
6	129	Mini Wheels Co.	64600
7	144	Volvo Model Replicas, Co	53100
8	145	Danish Wholesale Imports	83400
9	161	Technics Stores Inc.	84600

8. Create Trigger to store transaction record for employee table which displays employeeNumber, lastName, FirstName and office code upon insertion.

```
INSERT INTO employees (employeeNumber, lastName, firstName, extension, email, officeCode, reportsTo, jobTitle)
VALUES
(2025, 'Lilly', 'Pritchet', '4325', 'lilly@classicmodelcars.com', 3, 1143, 'Sales Coordinator');
```

100 %

Results | Messages

	ID	Surname	GivenName	Office_Location
1	2025	Lilly	Pritchet	3

9. Create a Trigger to display customer number if the amount is greater than10,000

```
INSERT INTO payments (customerNumber, checkNumber, paymentDate, amount)
VALUES (144, 'DD123456', '2025-09-04', 80000.00);
```

100 %

Results | Messages

	customerNumber
1	144

10.Create Users, Roles and Logins according to 3 Roles: Admin, HR, and Employee. Admin can view full database and has full access, HR can view and access only employee and offices table. Employee can view all table only.

```
-- 1. Create Logins for Users
CREATE LOGIN MasterAdmin WITH PASSWORD = 'SecurePassAdmin@99';
CREATE LOGIN HRManager WITH PASSWORD = 'HR_SafeKey_456';
CREATE LOGIN StaffMember WITH PASSWORD = 'EmpAccess_789!';

--2. Create Database Users
USE MotorsCertification;
CREATE USER AdminControl FOR LOGIN MasterAdmin;
CREATE USER HRAccess FOR LOGIN HRManager;
CREATE USER GeneralViewer FOR LOGIN StaffMember;

--3.Define Roles and Assign Permissions
CREATE ROLE SystemManager;
CREATE ROLE HRUnit;
CREATE ROLE DataViewer;

--4.Grant Permissions Based on Roles
GRANT CONTROL ON DATABASE::MotorsCertification TO SystemManager; --Full Access for SystemManager
GRANT SELECT, INSERT, UPDATE, DELETE ON employees TO HRUnit; --HR Can Only Access employees and offices
GRANT SELECT, INSERT, UPDATE, DELETE ON offices TO HRUnit;
GRANT SELECT ON SCHEMA::dbo TO DataViewer; --Employees Can Only View All Tables

--5.Assign Users to Roles
ALTER ROLE SystemManager ADD MEMBER AdminControl;
ALTER ROLE HRUnit ADD MEMBER HRAccess;
ALTER ROLE DataViewer ADD MEMBER GeneralViewer;

SELECT * FROM sys.database_role_members

-- STEP 13
```

Results | Messages

role_principal_id	member_principal_id
16384	1

11. Schedule a Job which backups and schedule it according to developer preference.

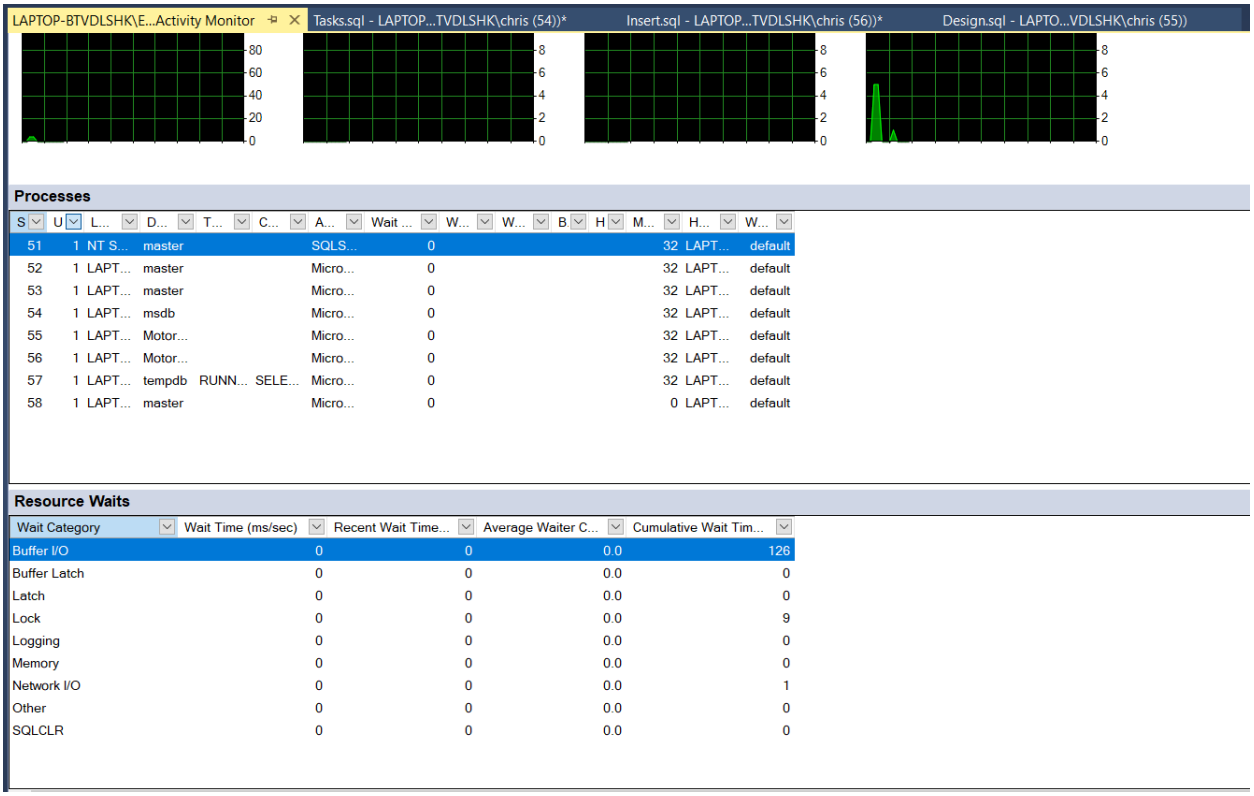
```
BACKUP DATABASE MotorsCertification TO DISK = 'C:\Program Files\Microsoft SQL Server\MSSQL16.EDUREKA\MSSQL\Backup\MotorsCertification1.bak';
```

Messages

```
Processed 768 pages for database 'MotorsCertification', file 'MotorsCertification' on file 1.
Processed 2 pages for database 'MotorsCertification', file 'MotorsCertification_log' on file 1.
BACKUP DATABASE successfully processed 770 pages in 0.850 seconds (7.072 MB/sec).

Completion time: 2025-05-23T20:36:46.0403417+05:30
```

12.Open Activity Monitor and list down some minor observations including Processes, Resource Waits, and Active Expensive Queries.



Azure Migration

12. Migrate the following SQL server workload to azure.

Steps to Migrate SQL Server to Azure:

1. Export the .BAK file.
    - Use SSMS to create a full backup of the database.
2. Use Azure Data Migration Assistant.
    - Analyze and migrate the database schema and data to Azure SQL Server.
3. Restore the database in Azure SQL Server.
    - Use the .BAK file to restore the database in Azure.