sql assignment

1. Create a Function that Calls Another Function

✅ What is this process called?

> This process is called Nesting Functions or Nested Function Calls.

✅ Example in SQL Server:

```
sql
CopyEdit
-- Inner Function
CREATE FUNCTION dbo.GetDiscount (@Amount DECIMAL)
RETURNS DECIMAL
AS
BEGIN
    RETURN @Amount * 0.1; -- 10% discount
END;

-- Outer Function that calls inner function
CREATE FUNCTION dbo.GetFinalAmount (@Amount DECIMAL)
RETURNS DECIMAL
AS
BEGIN
    DECLARE @Discount DECIMAL;
    SET @Discount = dbo.GetDiscount(@Amount);
    RETURN @Amount - @Discount;
END;

-- Usage:
SELECT dbo.GetFinalAmount(1000) AS FinalAmount;
```

2. How to Inspect the Query's Execution Plan

✅ Purpose:

Helps identify performance issues by analyzing how SQL Server executes a query.

✅ Methods:

- **In SQL Server Management Studio (SSMS):**
  - Click **"Display Estimated Execution Plan"** (Ctrl + L)
  - Or click **"Include Actual Execution Plan"** (Ctrl + M), then run your query
- **Using SQL command:**

```
sql
CopyEdit
SET SHOWPLAN_ALL ON;
GO
-- Write your query here
SELECT * FROM Employees WHERE DepartmentID = 1;
GO
SET SHOWPLAN_ALL OFF;
```

3. Purpose of MAXDOP and Recompile Keywords

✅ MAXDOP (Maximum Degree of Parallelism):

Controls how many CPU threads can be used for a single query.

```
sql
CopyEdit
-- Use 1 CPU thread only
SELECT * FROM Orders
```

```
OPTION (MAXDOP 1);
```

> Use Case: Prevent parallelism when it may degrade performance or in workloads with high concurrency.

✅ RECOMPILE:

Forces SQL Server to **recompile** the query every time it's run, generating a fresh execution plan.

```
sql
CopyEdit
SELECT * FROM Orders WHERE CustomerID = 5
OPTION (RECOMPILE);
```

> Use Case: Useful when parameter sniffing causes performance issues.

---

4. How to Build DDL Statements from Existing Tables

> DDL (Data Definition Language) includes CREATE, ALTER, DROP, etc.

✅ Steps to Generate DDL:

In SQL Server Management Studio (SSMS):

1. Right-click on the table.
2. Select **Script Table as** → **CREATE To** → **New Query Editor Window**.

Using SQL:

```
sql
CopyEdit
-- No direct SQL command. Use SSMS or tools like:
-- sp_help or sp_columns to get metadata
EXEC sp_help 'YourTableName';
```

In MySQL:

```
sql
CopyEdit
SHOW CREATE TABLE your_table_name;
```

---

5. Update Data Using INNER JOIN

✅ Example:

```
sql
CopyEdit
-- Two tables
-- Employees (EmpID, Name, DeptID)
-- Departments (DeptID, DeptName)

-- Update Employee table with new department names
UPDATE E
SET E.DeptID = D.DeptID
FROM Employees E
INNER JOIN Departments D ON D.DeptName = 'Sales'
WHERE E.Name = 'John';
```

> Note: SQL Server syntax shown above. MySQL syntax differs slightly.

---

6. Difference Between TRUNCATE, DELETE, and DROP

Operation	Description	Can Rollback?	Removes Table Structure?
`DELETE`	Deletes rows based on condition	✅ Yes	❌ No
`TRUNCATE`	Deletes **all** rows quickly (no logging)	⚠️ Depends*	❌ No
`DROP`	Deletes table **and** structure	❌ No	✅ Yes

✅ Example:

```
sql
CopyEdit
-- DELETE example
DELETE FROM Employees WHERE Department = 'HR';

-- TRUNCATE example
TRUNCATE TABLE Employees;

-- DROP example
DROP TABLE Employees;
```