DDL

CREATE TABLE table_name	ALTER TABLE table_name
column_name1 data_type(size), column_name2 data_type(size), column_name3 data_type(size),	ADD column_name datatype
RENAME TABLENAME	ALTER TABLE Student
TO NEWTABLENAME;	RENAME COLUMN NAME TO FIRST_NAME;

DML

INSERT INTO table_name (column1, column2, column3, ...)	DELETE FROM table_name
VALUES (value1, value2, value3, ...);	WHERE condition;
UPDATE table_name SET column1 = value1	, column2 = value2, ... WHERE condition;

DQL

SELECT column1, column2,	'a%' Finds any values that start with "a".	'%or%' Finds any values that have "or" in any position.
... FROM table_name WHERE columnN LIKE pattern;	'_r%' Finds any values that have "r" in the second position 'a%o'	Finds any values that start with "a" and ends with "o"
SELECT * FROM Customers WHERE Country IN ('Germany', 'France', 'UK');	SELECT * FROM Customers WHERE Country NOT IN ('Germany', 'France', 'UK');	
A database cursor is an object that enables traversal over the rows of a result set. It allows you to process individual row returned by a query	Allows us to update one row at a time or perform an administrative process such as SQL Server database backups in a sequential manner.	

DQL (cont)

DECLARE all variables you'll need		DECLARE @product_name VARCHAR(MAX), @list_price DECIMAL;
DECLARE … CURSOR FOR SELECT query, where you'll declare a cursor and also define the query related to (populating) that cursor	DECLARE cursor_name CURSOR FOR select_statement;	DECLARE cursor_product CURSOR FOR SELECT product_name, list_price FROM production.products;
OPEN the cursor and FETCH NEXT from the cursor	OPEN cursor_name;....FETCH NEXT FROM cursor INTO variable_list;	OPEN cursor_product; FETCH NEXT FROM cursor_product INTO @product_name, @list_price;
In the WHILE loop you'll test the @@FETCH_STATUS variable (WHILE @@FETCH_STATUS = 0). If the condition holds, you'll enter the loop BEGIN … END block and perform statements inside that block	WHILE @@FETCH_STATUS = 0 BEGIN FETCH NEXT FROM cursor_name; END;	WHILE @@FETCH_STATUS = 0 BEGIN PRINT @product_name + CAST(@list_price AS varchar); FETCH NEXT FROM cursor_product INTO @product_name, @list_price; END;
CLOSE the cursor and DEALLOCATE it.	CLOSE cursor_name; DEALLOCATE cursor_name;	CLOSE cursor_product; DEALLOCATE cursor_product;