

DATA ENGINEERING 101: SQL



Shwetank Singh
GritSetGrow - GSGLearn.com



1

SELECT

Used to select data from a database.

EXAMPLE

*SELECT * FROM Employees;*

Shwetank Singh
GritSetGrow - GSGLearn.com



2

FROM

Specifies the table to select or delete data from.

EXAMPLE

SELECT Name FROM Employees;

3

WHERE

Filters records.

EXAMPLE

*SELECT * FROM Employees WHERE Age > 30;*

4

INSERT

Inserts new data into a table.

EXAMPLE

```
INSERT INTO Employees (Name, Age)  
VALUES ('John', 28);
```

 Shwetank Singh
GritSetGrow - GSGLearn.com

5

UPDATE

Modifies existing data in a table.

EXAMPLE

```
UPDATE Employees SET Age = 30  
WHERE Name = 'John';
```

 Shwetank Singh
GritSetGrow - GSGLearn.com

6

DELETE

Deletes data from a table.

EXAMPLE

```
DELETE FROM Employees  
WHERE Name = 'John';
```

 Shwetank Singh
GritSetGrow - GSGLearn.com

7

CREATE TABLE

Creates a new table.

EXAMPLE

```
CREATE TABLE Employees  
(ID int,  
Name varchar(255));
```

8

DROP TABLE

Deletes a table.

EXAMPLE

DROP TABLE Employees;

9

ALTER TABLE

Modifies an existing table.

EXAMPLE

ALTER TABLE Employees ADD Salary int;

10

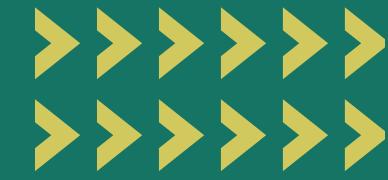
INNER JOIN

Returns records with matching values in both tables.

EXAMPLE

```
SELECT * FROM Employees  
INNER JOIN Departments  
ON Employees.DeptID = Departments.ID;
```

11



LEFT JOIN

Returns all records from the left table, and matched records from the right table.

EXAMPLE

```
SELECT * FROM Employees  
LEFT JOIN Departments  
ON Employees.DeptID = Departments.ID;
```



Shwetank Singh
GritSetGrow - GSGLearn.com



12

RIGHT JOIN

Returns all records from the right table,
and matched records from the left table.

EXAMPLE

```
SELECT * FROM Employees  
RIGHT JOIN Departments  
ON Employees.DeptID = Departments.ID;
```

13

FULL JOIN

Returns all records when there is a match in either left or right table.

EXAMPLE

```
SELECT * FROM Employees  
FULL JOIN Departments  
ON Employees.DeptID = Departments.ID;
```

14

CROSS JOIN

Returns the Cartesian product of the two tables.

EXAMPLE

```
SELECT * FROM Employees  
CROSS JOIN Departments;
```

15

GROUP BY

Groups rows that have the same values into summary rows.

EXAMPLE

```
SELECT COUNT(*), Department  
FROM Employees  
GROUP BY Department;
```

16

HAVING

Filters records that work on summarized GROUP BY results.

EXAMPLE

```
SELECT COUNT(*), Department  
FROM Employees  
GROUP BY Department HAVING COUNT(*) > 5;
```

17

ORDER BY

Sorts the result set in ascending or descending order.

EXAMPLE

```
SELECT * FROM Employees  
ORDER BY Age DESC;
```

18

DISTINCT

Selects only distinct (different) values.

EXAMPLE

```
SELECT DISTINCT Department  
FROM Employees;
```

 Shwetank Singh
GritSetGrow - GSGLearn.com

19

LIMIT

Specifies the number of records to return.

EXAMPLE

```
SELECT * FROM Employees LIMIT 10;
```

 Shwetank Singh
GritSetGrow - GSGLearn.com

20

OFFSET

Specifies the offset of the first row to return.

EXAMPLE

```
SELECT * FROM Employees LIMIT 5 OFFSET 10;
```

Shwetank Singh
GritSetGrow - GSGLearn.com



21

UNION

Combines the result set of two or more SELECT statements.

EXAMPLE

```
SELECT City FROM Customers  
UNION  
SELECT City FROM Suppliers;
```

22

UNION ALL

Combines the result set of two or more
SELECT statements, including
duplicates.

EXAMPLE

```
SELECT City FROM Customers  
UNION ALL  
SELECT City FROM Suppliers;
```

23

INTERSECT

Returns the intersection of two or more SELECT statements.

EXAMPLE

```
SELECT City FROM Customers  
INTERSECT  
SELECT City FROM Suppliers;
```

Shwetank Singh
GritSetGrow - GSGLearn.com

24

EXCEPT

Returns the difference between two SELECT statements.

EXAMPLE

```
SELECT City FROM Customers  
EXCEPT  
SELECT City FROM Suppliers;
```

25

IN

Checks for values within a set.

EXAMPLE

```
SELECT * FROM Employees  
WHERE Department IN ('HR', 'Finance');
```

Shwetank Singh
GritSetGrow - GSGLearn.com

26

BETWEEN

Selects values within a given range.

EXAMPLE

```
SELECT * FROM Employees  
WHERE Age BETWEEN 25 AND 30;
```

 Shwetank Singh
GritSetGrow - GSGLearn.com

27

LIKE

Searches for a specified pattern in a column.

EXAMPLE

```
SELECT * FROM Employees  
WHERE Name LIKE 'J%';
```

Shwetank Singh
GritSetGrow - GSGLearn.com

28

IS NULL

Tests for empty (NULL) values.

EXAMPLE

```
SELECT * FROM Employees  
WHERE Age IS NULL;
```

 Shwetank Singh
GritSetGrow - GSGLearn.com

29

IS NOT NULL

Tests for non-empty (NOT NULL) values.

EXAMPLE

```
SELECT * FROM Employees  
WHERE Age IS NOT NULL;
```

30

CASE

Returns value based on a condition.

EXAMPLE

```
SELECT Name, Age, CASE  
WHEN Age > 30 THEN 'Senior' ELSE 'Junior' END  
FROM Employees;
```

Shwetank Singh
GritSetGrow - GSGLearn.com



31

COALESCE

Returns the first non-null value in a list.

EXAMPLE

```
SELECT COALESCE(Address, 'No Address')  
FROM Employees;
```

 Shwetank Singh
[GritSetGrow - GSGLearn.com](http://GritSetGrow-GSGLearn.com)

32

NULLIF

Returns NULL if two expressions are equal.

EXAMPLE

```
SELECT NULLIF(Salary, 0) FROM Employees;
```

33

CAST

Converts a data type into another data type.

EXAMPLE

```
SELECT CAST(Age AS varchar)  
FROM Employees;
```

 Shwetank Singh
GritSetGrow - GSGLearn.com

34

CONVERT

Converts a data type into another data type with style options.

EXAMPLE

```
SELECT CONVERT(varchar, Age, 1)  
FROM Employees;
```

 Shwetank Singh
GritSetGrow - GSGLearn.com

35

SUBSTRING

Extracts characters from a string.

EXAMPLE

```
SELECT SUBSTRING(Name, 1, 2)  
FROM Employees;
```

Shwetank Singh
GritSetGrow - GSGLearn.com



36

LENGTH

Returns the length of a string.

EXAMPLE

```
SELECT LENGTH(Name)  
FROM Employees;
```

Shwetank Singh
GritSetGrow - GSGLearn.com



37

TRIM

Removes spaces or specified characters from both ends of a string.

EXAMPLE

```
SELECT TRIM(Name) FROM Employees;
```

38

UPPER

Converts a string to uppercase.

EXAMPLE

SELECT UPPER(Name) FROM Employees;

Shwetank Singh
GritSetGrow - GSGLearn.com



39

LOWER

Converts a string to lowercase.

EXAMPLE

SELECT LOWER(Name) FROM Employees;

Shwetank Singh
GritSetGrow - GSGLearn.com

40

REPLACE

Replaces occurrences of a specified string.

EXAMPLE

```
SELECT REPLACE(Name, 'John', 'Jon')  
FROM Employees;
```

Shwetank Singh
GritSetGrow - GSGLearn.com

41

CHARINDEX

Returns the position of a substring in a string.

EXAMPLE

```
SELECT CHARINDEX('a', Name)  
FROM Employees;
```

Shwetank Singh
GritSetGrow - GSGLearn.com

42

ROUND

Rounds a number to a specified number of decimal places.

EXAMPLE

```
SELECT ROUND(Salary, 2) FROM Employees;
```

Shwetank Singh
GritSetGrow - GSGLearn.com



43



AVG

Returns the average value of a numeric column.

EXAMPLE

SELECT AVG(Salary) FROM Employees;



Shwetank Singh
GritSetGrow - GSGLearn.com



44

COUNT

Returns the number of rows that matches a specified criterion.

EXAMPLE

```
SELECT COUNT(*) FROM Employees;
```

Shwetank Singh
[GritSetGrow - GSGLearn.com](http://GritSetGrow-GSGLearn.com)



45

SUM

Returns the total sum of a numeric column.

EXAMPLE

```
SELECT SUM(Salary) FROM Employees;
```

 Shwetank Singh
GritSetGrow - GSGLearn.com

46

MAX

Returns the maximum value in a set.

EXAMPLE

```
SELECT MAX(Salary) FROM Employees;
```

47

MIN

Returns the minimum value in a set.

EXAMPLE

SELECT MIN(Salary) FROM Employees;

Shwetank Singh
GritSetGrow - GSGLearn.com

48

NOW

Returns the current date and time.

EXAMPLE

SELECT NOW();

Shwetank Singh
GritSetGrow - GSGLearn.com



49



CURDATE

Returns the current date.

EXAMPLE

SELECT CURDATE();



Shwetank Singh
GritSetGrow - GSGLearn.com

50

CURTIME

Returns the current time.

EXAMPLE

```
SELECT CURTIME();
```

 Shwetank Singh
GritSetGrow - GSGLearn.com

51

DATE_FORMAT

Formats a date.

EXAMPLE

```
SELECT DATE_FORMAT(NOW(), '%Y-%m-%d');
```

 Shwetank Singh
GritSetGrow - GSGLearn.com

52

DATE_ADD

Adds a time/date interval to a date.

EXAMPLE

```
SELECT DATE_ADD(NOW(), INTERVAL 1 DAY);
```

 Shwetank Singh
[GritSetGrow - GSGLearn.com](http://GritSetGrow-GSGLearn.com)

53

DATE_SUB

Subtracts a time/date interval from a date.

EXAMPLE

```
SELECT DATE_SUB(NOW(), INTERVAL 1 DAY);
```

Shwetank Singh
GritSetGrow - GSGLearn.com

54

DATEDIFF

Returns the number of days between two dates.

EXAMPLE

```
SELECT DATEDIFF('2024-12-31', '2024-01-01');
```

 Shwetank Singh
GritSetGrow - GSGLearn.com

55

YEAR

Extracts the year part of a date.

EXAMPLE

```
SELECT YEAR(NOW());
```

 Shwetank Singh
GritSetGrow - GSGLearn.com

56

MONTH

Extracts the month part of a date.

EXAMPLE

```
SELECT MONTH(NOW());
```

Shwetank Singh
GritSetGrow - GSGLearn.com

57

DAY

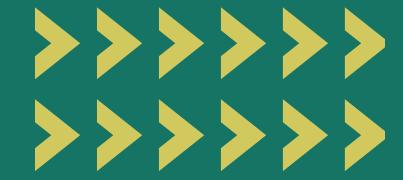
Extracts the day part of a date.

EXAMPLE

```
SELECT DAY(NOW());
```

Shwetank Singh
[GritSetGrow - GSGLearn.com](http://GritSetGrow-GSGLearn.com)





58

HOUR

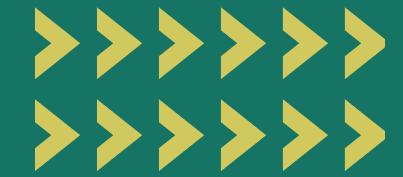
Extracts the hour part of a time.

EXAMPLE

```
SELECT HOUR(NOW());
```



Shwetank Singh
GritSetGrow - GSGLearn.com



59

MINUTE

Extracts the minute part of a time.

EXAMPLE

```
SELECT MINUTE(NOW());
```



Shwetank Singh
GritSetGrow - GSGLearn.com

60

SECOND

Extracts the second part of a time.

EXAMPLE

```
SELECT SECOND(NOW());
```

 Shwetank Singh
GritSetGrow - GSGLearn.com

61

TIMESTAMPDIFF

Returns the difference between two timestamps.

EXAMPLE

```
SELECT TIMESTAMPDIFF(MINUTE, '2024-01-01  
00:00:00', NOW());
```

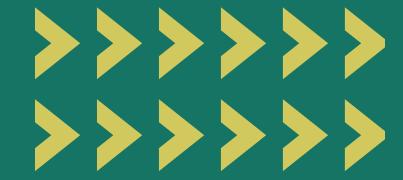
62

IFNULL

Returns a specified value if the expression is NULL.

EXAMPLE

```
SELECT IFNULL(Salary, 0)  
FROM Employees;
```



63

CONCAT

Concatenates two or more strings.

EXAMPLE

```
SELECT CONCAT(FirstName, ' ', LastName)  
FROM Employees;
```



Shwetank Singh
GritSetGrow - GSGLearn.com

64

IF

Returns a value based on a condition.

EXAMPLE

```
SELECT Name, IF(Age > 30, 'Senior', 'Junior')  
FROM Employees;
```

Shwetank Singh
GritSetGrow - GSGLearn.com

65

EXISTS

Checks for the existence of any record in a subquery.

EXAMPLE

```
SELECT * FROM Employees  
WHERE EXISTS  
(SELECT 1 FROM Departments WHERE  
Employees.DeptID = Departments.ID);
```

66

ALL

Compares a value to all values in another value set.

EXAMPLE

```
SELECT * FROM Employees  
WHERE Salary > ALL (SELECT Salary FROM  
Employees WHERE Department = 'HR');
```

Shwetank Singh
GritSetGrow - GSGLearn.com

67

ANY

Compares a value to any value in another value set.

EXAMPLE

```
SELECT * FROM Employees  
WHERE Salary > ANY (SELECT Salary  
FROM Employees WHERE Department = 'HR');
```

68

SOME

Synonym for ANY.

EXAMPLE

```
SELECT * FROM Employees  
WHERE Salary > SOME (SELECT Salary FROM  
Employees WHERE Department = 'HR');
```

69

SUBQUERY

A query nested inside another query.

EXAMPLE

```
SELECT * FROM Employees  
WHERE DeptID = (SELECT ID FROM  
Departments WHERE Name = 'HR');
```

Shwetank Singh
GritSetGrow - GSGLearn.com

70

CORRELATED SUBQUERY

A subquery that references columns from the outer query.

EXAMPLE

```
SELECT Name FROM Employees E1  
WHERE Salary > (SELECT AVG(Salary) FROM  
Employees E2 WHERE E1.DeptID = E2.DeptID);
```

71

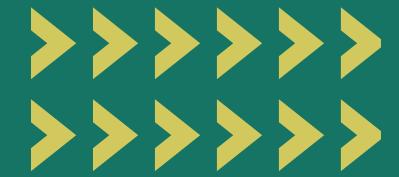
VIEW

A virtual table based on the result-set of an SQL statement.

EXAMPLE

```
CREATE VIEW EmployeeView  
AS  
SELECT Name, Age FROM Employees;
```

 Shwetank Singh
GritSetGrow - GSGLearn.com



72

INDEX

Used to speed up the performance of queries.

EXAMPLE

CREATE INDEX idx_name ON Employees (Name);



Shwetank Singh
GritSetGrow - GSGLearn.com

73

TRIGGER

Executes a batch of SQL code when an insert, update or delete command is run against a specific table.

EXAMPLE

```
CREATE TRIGGER trg_after_insert  
ON Employees AFTER INSERT  
AS  
BEGIN PRINT 'New Employee Inserted'; END;
```

74

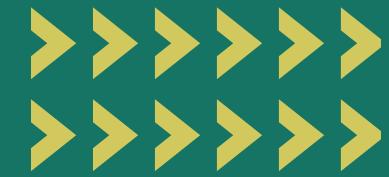
PROCEDURE

A stored subroutine available to applications accessing a relational database system.

EXAMPLE

```
CREATE PROCEDURE GetEmployee @ID int  
AS  
BEGIN  
SELECT * FROM Employees WHERE ID = @ID;  
END;
```

 Shwetank Singh
GritSetGrow - GSGLearn.com



75

FUNCTION

A subroutine available to applications accessing a relational database system that returns a single value.

EXAMPLE

```
CREATE FUNCTION GetEmployeeName (@ID int)
RETURNS varchar(255)
AS
BEGIN
DECLARE @Name varchar(255);
SELECT @Name = Name FROM Employees
WHERE ID = @ID; RETURN @Name;
END;
```



Shwetank Singh
GritSetGrow - GSGLearn.com

76

CURSOR

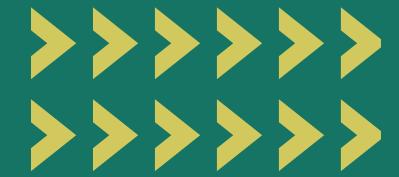
A database object used to retrieve data row-by-row.

EXAMPLE

```
DECLARE cursor_name  
CURSOR FOR SELECT Name FROM Employees;
```



Shwetank Singh
GritSetGrow - GSGLearn.com



77

FETCH

Retrieves rows one at a time, or in a block, from the result set of a multi-row query.

EXAMPLE

FETCH NEXT FROM cursor_name INTO @Name;



Shwetank Singh
GritSetGrow - GSGLearn.com

78

CLOSE

Closes the cursor and releases the current result set.

EXAMPLE

CLOSE cursor_name;

Shwetank Singh
GritSetGrow - GSGLearn.com

79

DEALLOCATE

Removes a cursor reference and releases resources.

EXAMPLE

DEALLOCATE cursor_name;

Shwetank Singh
GritSetGrow - GSGLearn.com

80

DECLARE

Declares a variable or cursor.

EXAMPLE

DECLARE @Age int;

 Shwetank Singh
GritSetGrow - GSGLearn.com

81

SET

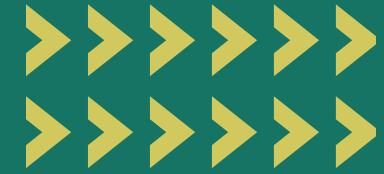
Initializes or assigns a value to a variable.

EXAMPLE

SET @Age = 30;

Shwetank Singh
GritSetGrow - GSGLearn.com

82



PRINT

Displays a user-defined message.

EXAMPLE

PRINT 'Hello SQL';



Shwetank Singh
GritSetGrow - GSGLearn.com

83

RAISERROR

Returns a user-defined error message.

EXAMPLE

RAISERROR('This is an error message', 16, 1);

84

TRY...CATCH

Handles exceptions in T-SQL code.

EXAMPLE

```
BEGIN TRY;  
    SELECT 1/0;  
END TRY  
BEGIN CATCH;  
    PRINT 'Error';  
END CATCH;
```

Shwetank Singh
GritSetGrow - GSGLearn.com



85

TRANSACTION

A sequence of operations performed as a single logical unit of work.

EXAMPLE

```
BEGIN TRANSACTION;  
UPDATE Employees SET Age = 30  
WHERE Name = 'John';  
COMMIT;
```

Shwetank Singh
GritSetGrow - GSGLearn.com

86

COMMIT

Saves the changes made in the transaction.

EXAMPLE

COMMIT TRANSACTION;

Shwetank Singh
GritSetGrow - GSGLearn.com



87

ROLLBACK

Undoes the changes made in the transaction.

EXAMPLE

ROLLBACK TRANSACTION;

Shwetank Singh
GritSetGrow - GSGLearn.com

88

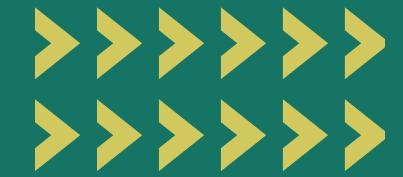
SAVEPOINT

Sets a point within a transaction to which a rollback can occur.

EXAMPLE

SAVE TRANSACTION savepoint_name;

Shwetank Singh
GritSetGrow - GSGLearn.com



89

SET TRANSACTION ISOLATION LEVEL

Sets the isolation level for the current session.

EXAMPLE

SET TRANSACTION ISOLATION LEVEL READ COMMITTED;



Shwetank Singh
GritSetGrow - GSGLearn.com

90

BULK INSERT

Imports a large amount of data into a table.

EXAMPLE

```
BULK INSERT Employees  
FROM 'datafile.txt'  
WITH  
(FIELDTERMINATOR = ',', ROWTERMINATOR = '\n');
```

Shwetank Singh
GritSetGrow - GSGLearn.com



91

TEMPORARY TABLE

A table that is created and can be automatically deleted when no longer used.

EXAMPLE

```
CREATE TABLE #TempTable  
(ID int, Name varchar(255));
```

 Shwetank Singh
GritSetGrow - GSGLearn.com

RECURSIVE CTE

Common Table Expressions that refer to themselves.

EXAMPLE

WITH RECURSIVE CTE

AS (

SELECT 1 AS n

UNION ALL SELECT n+1

FROM CTF

WHERE n < 10)

*SELECT * FROM CTE;*



Shwetank Singh

GritSetGrow - GSGLearn.com

93

WINDOW FUNCTION

Performs a calculation across a set of table rows related to the current row.

EXAMPLE

```
SELECT Name, Salary,  
AVG(Salary) OVER (PARTITION BY Department)  
FROM Employees;
```

Shwetank Singh
GritSetGrow - GSGLearn.com

94

RANK

Assigns a rank to each row within the partition of a result set.

EXAMPLE

```
SELECT Name, Salary,  
RANK() OVER (ORDER BY Salary DESC)  
FROM Employees;
```

95

DENSE_RANK

Assigns ranks to rows in an ordered partition without gaps in rank values.

EXAMPLE

```
SELECT Name, Salary,  
DENSE_RANK() OVER (ORDER BY Salary DESC)  
FROM Employees;
```

Shwetank Singh
GritSetGrow - GSGLearn.com

96

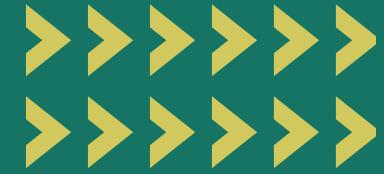
ROW_NUMBER

Assigns a unique sequential integer to rows within a partition.

EXAMPLE

```
SELECT Name, Salary,  
ROW_NUMBER() OVER (ORDER BY Salary DESC)  
FROM Employees;
```

97



NTILE

Distributes rows of an ordered partition into a specified number of groups.

EXAMPLE

```
SELECT Name, Salary,  
NTILE(4) OVER (ORDER BY Salary DESC)  
FROM Employees;
```



Shwetank Singh
GritSetGrow - GSGLearn.com

98

LAG

Accesses data from a previous row in the same result set.

EXAMPLE

```
SELECT Name, Salary,  
LAG(Salary, 1) OVER (ORDER BY Salary)  
FROM Employees;
```

99

LEAD

Accesses data from a subsequent row in the same result set.

EXAMPLE

```
SELECT Name, Salary,  
LEAD(Salary, 1) OVER (ORDER BY Salary)  
FROM Employees;
```

102

CROSS APPLY

Applies a table-valued function to each row of an outer table.

EXAMPLE

```
SELECT * FROM Employees  
CROSS APPLY  
GetEmployeeDetails(Employees.ID);
```

103

OUTER APPLY

Similar to CROSS APPLY, but returns all rows from the outer table.

EXAMPLE

```
SELECT * FROM Employees  
OUTER APPLY  
GetEmployeeDetails(Employees.ID);
```

 Shwetank Singh
[GritSetGrow - GSGLearn.com](http://GritSetGrow-GSGLearn.com)

THANK YOU