**Structured Query Language (SQL)**

**1. Create, Use and Drop Database**

1.1 Create a Database

1.2 Use a Database

1.3 Drop a Database

**2. Constraints in SQL Database**

2.1 Not Null Constraints

2.2 Default Constraints

2.3 Unique Constraints

2.4 Primary Key Constraints

**3. Create a Stored Recorded in Table**

3.1 Create a Table

3.2 Stored Record in Table

**4. Select and Distinct Statement**

4.1 Select a Single Column from Table

4.2 Select a Multiple Column from Table

4.3 Select Entire Columns from Table

4.4 Select Distinct

**5. Where Clause**

5.1 Simple Conditions

5.2 And Conditions

5.3 OR Conditions

5.4 Not Conditions

5.5 Like Conditions

5.6 Between Condition

**6. Function**

6.1 Min()

6.2 Max()

6.3 Count()

6.4 Sum()

6.5 Avg()

6.6 String LTRIM()

6.7 String Lower()

6.8 String Upper()

6.9 Reverse()

6.10 String Substring()

**7. Order By**

7.1 Ascending Order

7.2 Descending Order

**8. Top Clause**

**9. Group By**

**10. Having Clause**

**11. Update Statement**

**12. Delete Statement**

**13. Truncate Statement**

**14. Join Statement**

14.1 Inner Join

14.2 Left Join

14.3 Right Join

14.4 Full Join

14.5 Update using Join

**15. Union and Intersection Operator**

15.1 Union Operator

15.2 Union All Operator

15.3 Except Operator

15.4 Intersect Operator

**16. Create View and Drop View**

16.1 Create View

16.2 Drop View

**17. Alter Table**

17.1 Add Column

17.2 Drop Column

**18. Merge Statement**

**19. Table Valued Function**

**20. Temporary Table**

**21. Case Statement**

**22. IIF() Function**

**23. Stored Procedure**

1. **Create, Use and Drop the Database**
2. **Create a Database**

**Syntax:**

create database [databasename];

**Example:**



Here we create a database with the name of the company

1. **Use a Database**

**Syntax:**

Use [databasename];

**Example:**



Here we use a company database.

1. **Drop a Database**

**Syntax:**

Drop database [databasename];

**Example:**



Here we delete a company database.

1. **Constraints in SQL Database**

Constraints are used to specify rules for data in table

1. **Not Null Constraints**

Not Null constraints ensure that a column cannot have a Null value

**Example:**



For example, if we specify no null constraints on *e\_salary* columns then it’s meant no null value should be in that column.

1. **Default Constraints**

Default constraints set a default value for cols when no value is specified

**Example:**



For example, if we specify default value 25 in *e\_age* columns then it’s meant all the values in that column will be 25

1. **Unique Constraints**

Unique constraints ensure that all values in cols are different.

**Example:**



For example, if we specify unique constraints in the *e\_name* column then it’s meant all the values in that column will be unique

1. **Primary Key Constraints**

* Primary key constraint uniquely identifiers each record in a table.
* It is also a combination of *not null + unique*
* **Note:** in table no more than one primary key

**Example:**



For example, if we assign a primary key on the e\_id column then it means all the values in that column are unique and not null.

1. **Create and Store Record in Table**
2. **Create a Table**

Graphical user interface, text, application

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated**Syntax:**

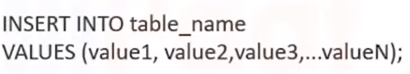
Text, letter

Description automatically generated

1. **Stored Record in a Table**

Text

Description automatically generated with medium confidence**Syntax:**



1. **Select and Distinct Statement Syntax**

**Syntax:**

Text

Description automatically generated

1. **Select a Single Column from Table**

Here we select the **name** column from the company table that we create and store the value inside that.



1. **Select a Multiple Column from Table**

Here we select the **name, age, department** columns from the company table that we create and store the value inside that.



Table

Description automatically generated

1. **Select Entire Columns from Table**

Here we select the **all** columns from the company table that we create and store the value inside that.



Table

Description automatically generated

1. **Select Distinct Syntax**

Select Distinct (unique value) is used to select only distinct values from our column.

**For example,** in the gender column, we have multiple male and female records, and we want only unique



Graphical user interface, application

Description automatically generated

1. **Where Clause**

Let’s filter records with where clause

Where Clause is used to extract records that satisfy a condition

For example: age >60 , Occupation = Data Scientist

1. **Simple Conditions**

**Syntax:**

Text

Description automatically generated

**Example:**

1. Age >= 22



Table

Description automatically generated

1. Gender = male



Table

Description automatically generated

1. salary >= 7200



Table

Description automatically generated

1. **And Conditions**

AND operator displays records if all the conditions separated by AND are TRUE

For example: age >60 **AND** Occupation = Data Scientist

Let’s impose multiple conditions with AND

**Syntax:**

Text

Description automatically generated

1. Age >= 22 AND salary >= 7200



Table

Description automatically generated

1. Age <= 20 AND salary >= 5000 AND Gender = female



Table

Description automatically generated

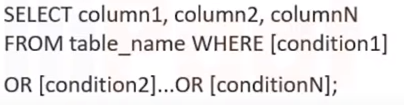
1. **OR Conditions**

OR operator displays records if any of the conditions separated by OR are TRUE

For example: age >60 **OR** Occupation = Data Scientist

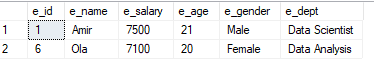
Let’s impose multiple conditions with OR

**Syntax:**



1. e\_dept = 'Data Scientist' or e\_dept = 'Data Analysis'





1. **Not Conditions**

Nor operator displays records if the condition is **NOT** True

For example: Occupation = Data Scientist >> then display except Data Scientist

Let’s impose records where the condition is NOT TRUE

**Syntax:**

Text

Description automatically generated

1. Not e\_dept = 'Data Scientist' (show all except this)



Table

Description automatically generated

1. **Like Conditions**

Like operator is used to extract records where a particular pattern is present

For example: pattern John will display Johnathon and Johny etc

**Wild Card Character**





**Syntax:**

Text

Description automatically generated with medium confidence

1. Extract name with K letter



Table

Description automatically generated

1. Extract department only who have data in pattern



Table

Description automatically generated

1. Select all employee who age in 20



Table

Description automatically generated with medium confidence

1. **Between Conditions**

Between operator is used to select values within in given age

For example, salary range between 5000 and 6000

Note: here 5000 and 6000 are inclusive

**Syntax:**

Text

Description automatically generated

1. Age between 19 to 22



Table

Description automatically generated

1. **Functions**

Number Function and String Function

1. **Min()**

Give the smallest value

**Syntax:**

A picture containing diagram

Description automatically generated

**Example**

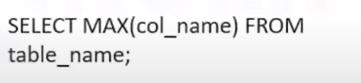




1. **Max()**

Give the largest value

**Syntax:**



**Example**

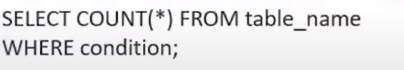




1. **Count()**

Return the number of rows that match specific criteria.

**Syntax:**



**Example**





1. **sum()**

return the sum of selected numeric column

**Syntax:**



**Example**

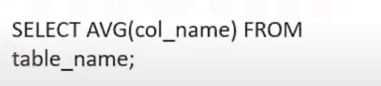




1. **Avg()**

return the Average of selected numeric column

**Syntax:**



**Example**





1. **String LTRIM()**

Removes blank on the left side of the character expression

**Example**



Graphical user interface, text, application

Description automatically generated

1. **String LOWER()**

Convert all characters to lower case letters

**Syntax**

A picture containing company name

Description automatically generated

**Example**



Table

Description automatically generated

1. **lower ()**

Convert all characters to lower case letters

**Syntax**

A picture containing company name

Description automatically generated

**Example**



Table

Description automatically generated

1. **Upper ()**

Convert all characters to upper case letters

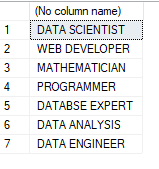
**Syntax**

Logo

Description automatically generated with medium confidence

**Example**





1. **Reverse ()**

Reverse all case letter into reverse

**Syntax**

A picture containing logo

Description automatically generated

**Example**



Table

Description automatically generated

1. **String SUBSTRING ()**

Given a substring from the original strings

**Syntax**

Logo

Description automatically generated with medium confidence

**Example:**



Table

Description automatically generated

1. **Order By**

Order by is used to sort the data in ascending or descending order.

A picture containing text

Description automatically generated

**Example of Ascending order (By Default)**



Table

Description automatically generated with medium confidence

**Example of Descending order**



Table

Description automatically generated with medium confidence

1. **Top Clause**

Top is used to fetch the TOP N records

A picture containing graphical user interface

Description automatically generated

**Example**



Table

Description automatically generated

1. **Group By**

Group by is used to get aggregate result with respect to a group

Text

Description automatically generated

**Example**



Table

Description automatically generated

**10. Having Clause**

Having clause is used in combination with group by to impose conditions on groups

Text

Description automatically generated

**Example:**

A picture containing text

Description automatically generated



**11. Update Statement**

The update is used to modify the existing records in a table

Text

Description automatically generated

**Example:** (update the age of amir)



Table

Description automatically generated with low confidence

**12. Delete Statement**

The Delete statement is used to delete existing records in the table

Text

Description automatically generated with medium confidence

**Example:** (delete the row of Nader)



Table

Description automatically generated

**13. Truncate Statement**

Truncate statement deletes all the data inside the table

Logo

Description automatically generated with medium confidence

**Example** (delete all records of employee)





1. **Join**

By using join we can join multiple tables on the basis of different types which we will discuss now.

1. **Inner Join**

Inner Join returns records that have matching values in both tables. It is also known as simple join.

A picture containing text, electronics

Description automatically generated

**Text, letter

Description automatically generated**

**Example**

Table

Description automatically generated Table

Description automatically generated

Text

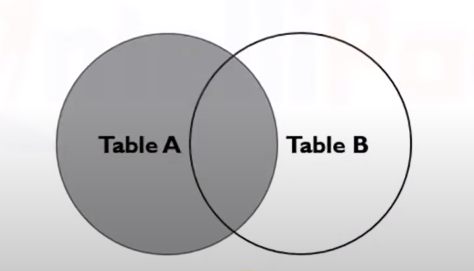
Description automatically generated with medium confidence

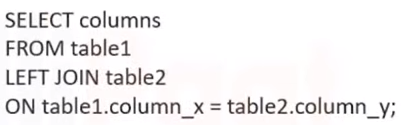
Table

Description automatically generated

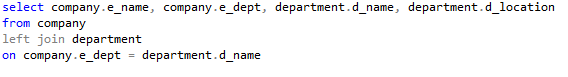
1. **Left Join**

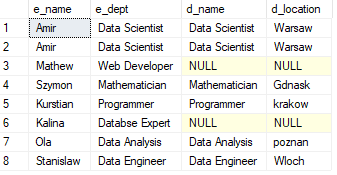
Left Join returns records all the records from the left table, and the matched records from the right table





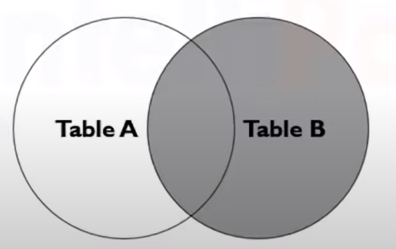
**Example**

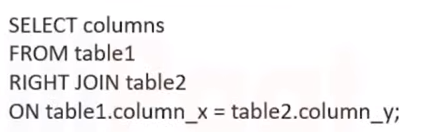




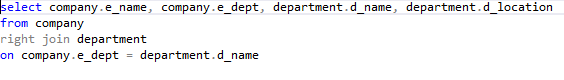
1. **Right Join**

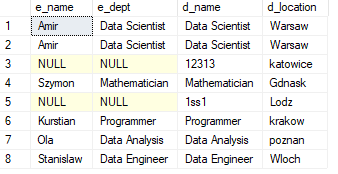
Right join return s all the records from the right table, and the matched records from the left table.





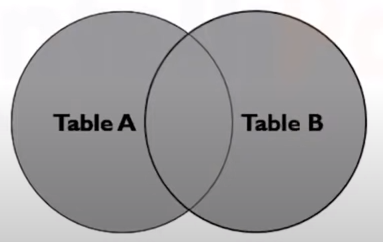
**Example**

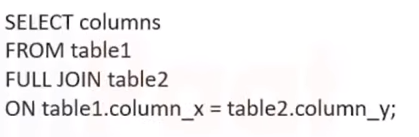
****

****

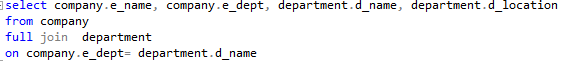
1. **Full Join**

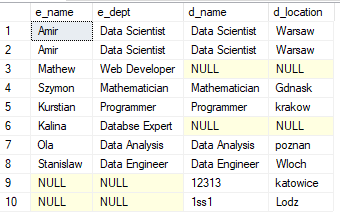
It returns all rows from the left table and the right table with Null values in place where join condition is not met



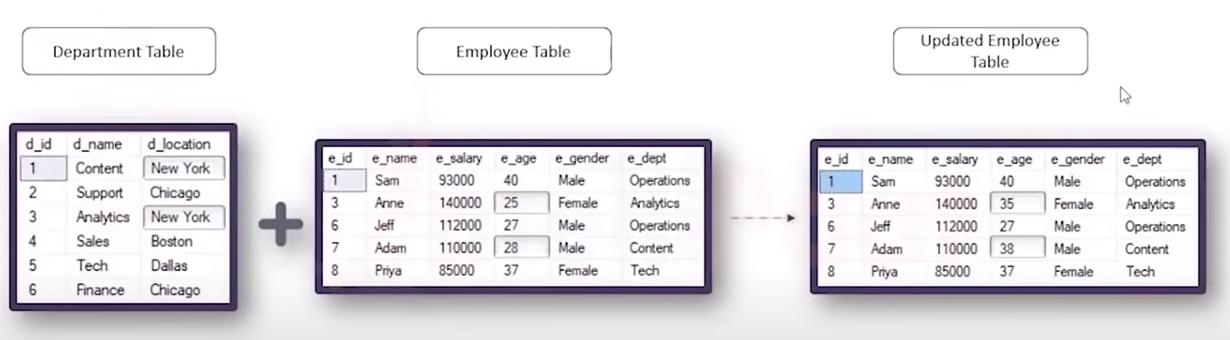


**Example**

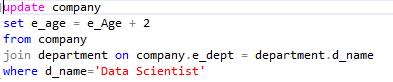




1. **Update using join**

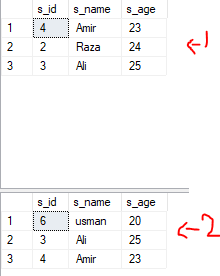


**Example:** update the company age where department location is warsaw



**15. Union and Intersection operator**

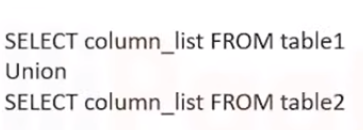
We will use these two tables for union and intersection operation

****

1. **Union operator**

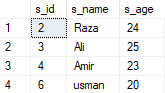
Union operator is used to combine the result-set of two or more SELECT statement

****

****

**Example:**

****

****

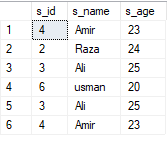
1. **Union All operator**

Union operator gives all the rows from both the tables including the duplicate



**Example:**

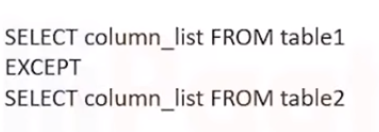
****

****

1. **Except Operator**

Except Operator combines two select statements and returns unique records from the left query which is not part of the right query.





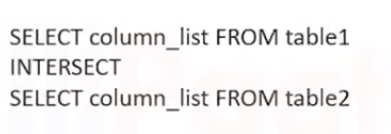
**Example:**

****

****

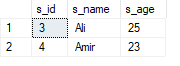
1. **Intersect Operator**

Intersect Operator helps to combine two select statements and returns the records which are common to both the select statements.



**Example:**

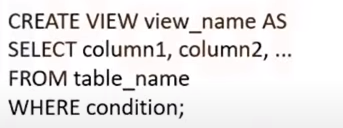
****

****

**16. Create View & Drop View**

1. **Create View**

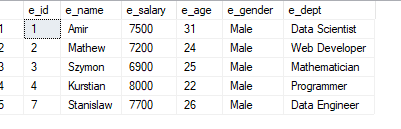
View is a virtual table based on the result of an sql statement.



**Example:**







1. **Drop View**



**Example**

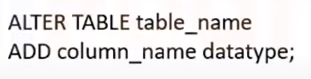


We drop male student view that we created

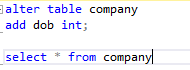
**15. Alter Table**

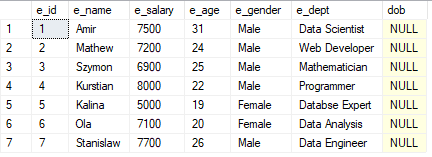
Alter table statement is used to add, delete or modify columns in a table

1. **Add Column**

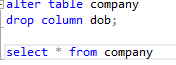


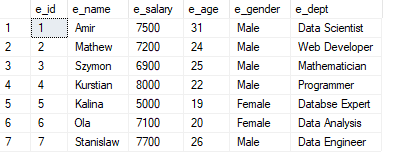
**Example**





1. **Drop Column**





**18. Merge**

Merge is the combination of Insert, delete and update statements.

Graphical user interface, text

Description automatically generated

**Example**

Graphical user interface, text, application

Description automatically generated

**19. Table Valued Function**

Table valued function returns a table

Graphical user interface, text, application, chat or text message, website

Description automatically generated  
**Example**

Graphical user interface, text, email, website

Description automatically generated



A screenshot of a computer

Description automatically generated with medium confidence

**20. Temporary Table**

Temporary Table are created in tempDB and delete as the session is terminated.

Logo

Description automatically generated with medium confidence

**Example**

Graphical user interface, text, application

Description automatically generated

Table

Description automatically generated with medium confidence

**21. Case Statement**

Case Statement helps in multi way decision making.

Text, letter

Description automatically generated

**Example**

Text

Description automatically generated

Table

Description automatically generated with medium confidence

**22. IIF() Function**

IIF() function is an alternative for the case statement.



**Example**

A picture containing text

Description automatically generated

Table

Description automatically generated

**23. Stored Procedure**

Stored procedure is a prepared sql code which can be saved and reused.

Text

Description automatically generated with low confidence



**END😊**