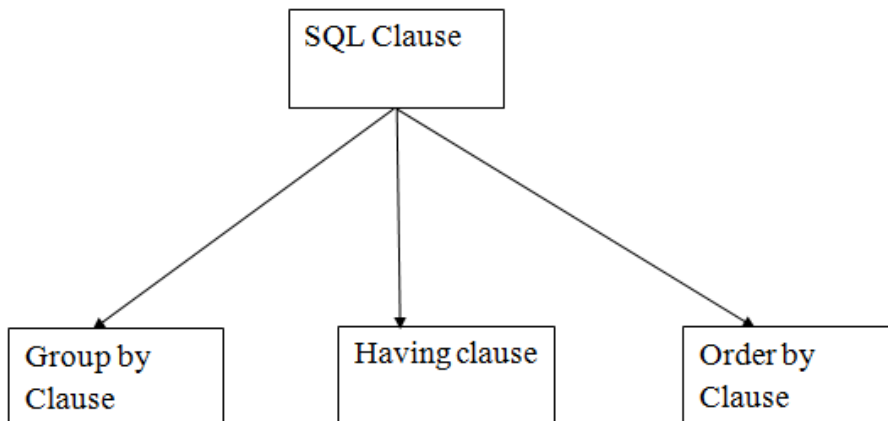


SQL Clauses

The following are the various SQL clauses:



1. GROUP BY

- SQL GROUP BY statement is used to arrange identical data into groups. The GROUP BY statement is used with the SQL SELECT statement.
- The GROUP BY statement follows the WHERE clause in a SELECT statement and precedes the ORDER BY clause.

- | | sick_leave | (No column name) |
|---|------------|------------------|
| 1 | 10 | 4 |
| 2 | 20 | 1 |

 with aggregation function.

2. HAVING

- HAVING clause is used to specify a search condition for a group or an aggregate.
- Having is used in a GROUP BY clause. If you are not using GROUP BY clause then you can use HAVING function like a WHERE clause.

5	E001	E005
---	------	------

3. ORDER BY

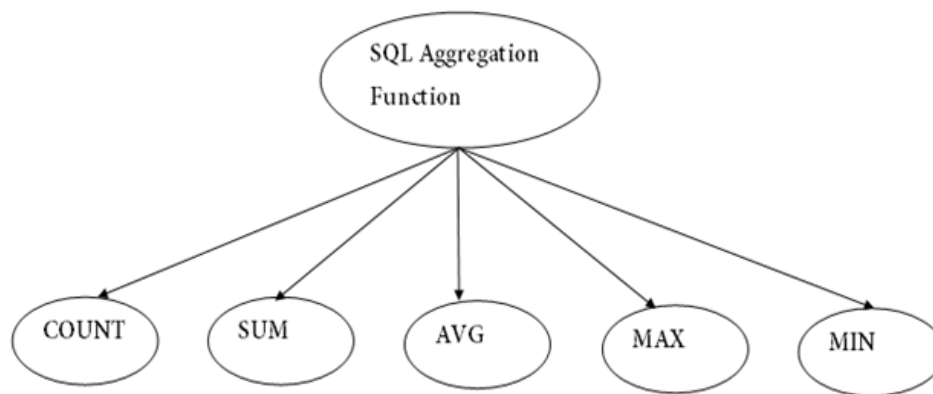
- The ORDER BY clause sorts the result-set in ascending or descending order.
- It sorts the records in ascending order by default. DESC keyword is used to sort the records in descending order.

/	P002	ERP	HTC	Raja
8	P003	Optical	Eagle Vision	Kumar

SQL Aggregate Functions

- SQL aggregation function is used to perform the calculations on multiple rows of a single column of a table. It returns a single value.
- It is also used to summarize the data.

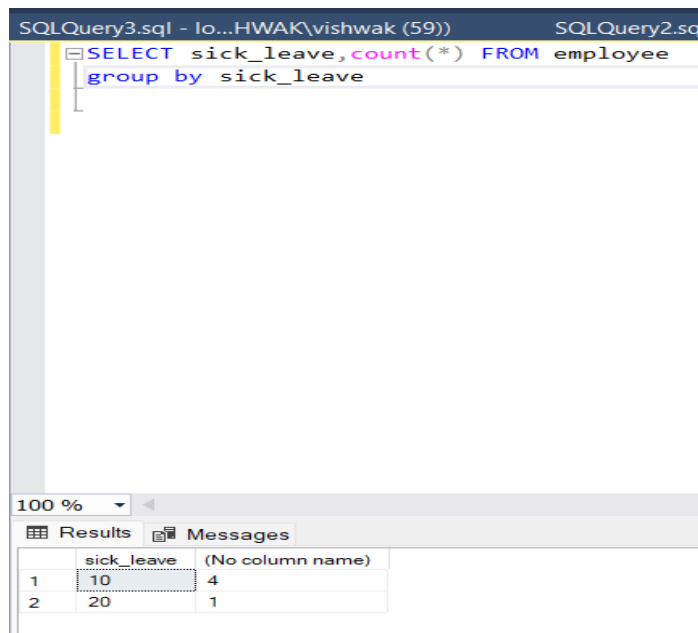
Types of SQL Aggregation Function



1. COUNT FUNCTION

- COUNT function is used to Count the number of rows in a database table. It can work on both numeric and non-numeric data types.

- COUNT function uses the COUNT(*) that returns the count of all the rows in a specified table. COUNT(*) considers duplicate and Null.



The screenshot shows a SQL Server Enterprise Manager window with a query editor and a results pane. The query editor contains the following SQL statement:

```
SELECT sick_leave, count(*) FROM employee
group by sick_leave
```

The results pane displays the output of the query. It shows a table with two columns: 'sick_leave' and '(No column name)'. The data is as follows:

	sick_leave	(No column name)
1	10	4
2	20	1

2. SUM Function

- Sum function is used to calculate the sum of all selected columns. It works on numeric fields only.

SQLQuery5.sql - Io...HWAK\vishwak (63)) SQLQ

```
select sum(salary) from employee  
where emp_id is not null
```

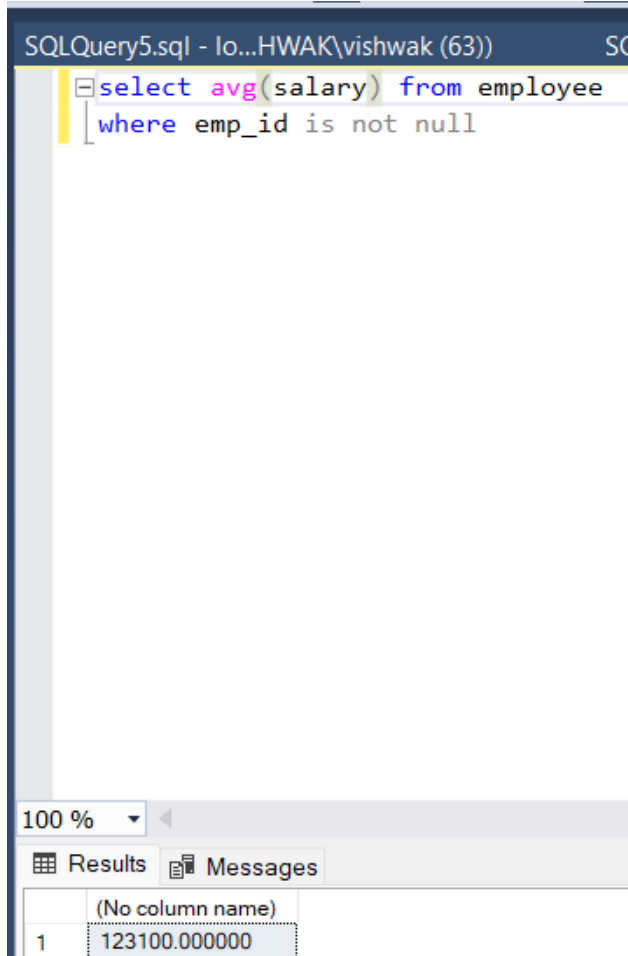
100 %

Results Messages

(No column name)	
1	615500.00

3. AVG function

- The AVG function is used to calculate the average value of the numeric type. AVG function returns the average of all non-Null values.



The screenshot shows a SQL Server Enterprise Manager window with a query editor and a results pane. The query editor contains the following SQL statement:

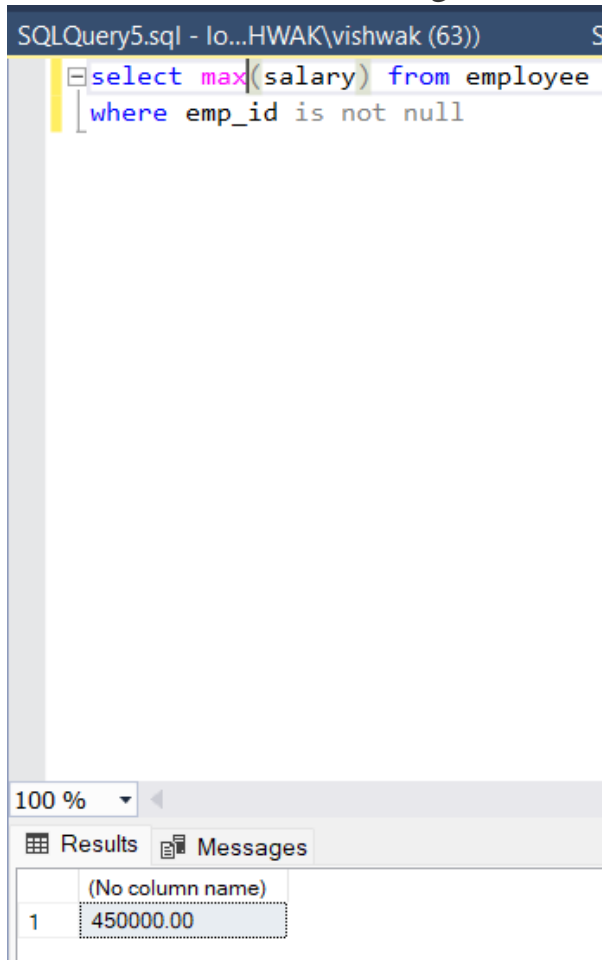
```
select avg(salary) from employee
where emp_id is not null
```

The results pane shows a single row with the following data:

	(No column name)
1	123100.000000

4. MAX Function

- MAX function is used to find the maximum value of a certain column. This function determines the largest value of all selected values of a column.



The screenshot shows a SQL query window titled "SQLQuery5.sql - lo...HWAK\vishwak (63)". The query text is:

```
select max(salary) from employee
where emp_id is not null
```

Below the query window, the "Results" tab is active, displaying a single row of data:

	(No column name)
1	450000.00

5. MIN Function

- MIN function is used to find the minimum value of a certain column. This function determines the smallest value of all selected values of a column.

```
SQLQuery5.sql - lo...HWAK\vishwak (63))
select min(salary) from employee
where emp_id is not null
```

100 %

Results Messages

	(No column name)
1	20500.00

SQL Server Functions

- SQL Server has many built-in functions.
- This reference contains string, numeric, date, conversion, and some advanced functions in SQL Server.

SQL Server String Functions

Function	Description
<u>ASCII</u>	Returns the ASCII value for the specific character
<u>CHAR</u>	Returns the character based on the ASCII code
<u>CHARINDEX</u>	Returns the position of a substring in a string
<u>CONCAT</u>	Adds two or more strings together
<u>Concat with +</u>	Adds two or more strings together
<u>CONCAT_WS</u>	Adds two or more strings together with a separator
<u>DATALength</u>	Returns the number of bytes used to represent an expression
<u>DIFFERENCE</u>	Compares two SOUNDEX values, and returns an integer value
<u>FORMAT</u>	Formats a value with the specified format
<u>LEFT</u>	Extracts a number of characters from a string (starting from left)
<u>LEN</u>	Returns the length of a string
<u>LOWER</u>	Converts a string to lower-case
<u>LTRIM</u>	Removes leading spaces from a string
<u>NCHAR</u>	Returns the Unicode character based on the number code
<u>PATINDEX</u>	Returns the position of a pattern in a string
<u>QUOTENAME</u>	Returns a Unicode string with delimiters added to make the string a valid SQL Server delimited identifier
<u>REPLACE</u>	Replaces all occurrences of a substring within a string, with a new substring
<u>REPLICATE</u>	Repeats a string a specified number of times
<u>REVERSE</u>	Reverses a string and returns the result
<u>RIGHT</u>	Extracts a number of characters from a string (starting from right)
<u>RTRIM</u>	Removes trailing spaces from a string

<u>SOUNDEX</u>	Returns a four-character code to evaluate the similarity of two strings
<u>SPACE</u>	Returns a string of the specified number of space characters
<u>STR</u>	Returns a number as string
<u>STUFF</u>	Deletes a part of a string and then inserts another part into the string, starting at a specified position
<u>SUBSTRING</u>	Extracts some characters from a string
<u>TRANSLATE</u>	Returns the string from the first argument after the characters specified in the second argument are translated into the characters specified in the third argument.
<u>TRIM</u>	Removes leading and trailing spaces (or other specified characters) from a string
<u>UNICODE</u>	Returns the Unicode value for the first character of the input expression
<u>UPPER</u>	Converts a string to upper-case

SQL Server Math/Numeric Functions

Function	Description
<u>ABS</u>	Returns the absolute value of a number
<u>ACOS</u>	Returns the arc cosine of a number
<u>ASIN</u>	Returns the arc sine of a number
<u>ATAN</u>	Returns the arc tangent of a number
<u>ATN2</u>	Returns the arc tangent of two numbers
<u>AVG</u>	Returns the average value of an expression
<u>CEILING</u>	Returns the smallest integer value that is \geq a number
<u>COUNT</u>	Returns the number of records returned by a select query
<u>COS</u>	Returns the cosine of a number
<u>COT</u>	Returns the cotangent of a number
<u>DEGREES</u>	Converts a value in radians to degrees

<u>EXP</u>	Returns e raised to the power of a specified number
<u>FLOOR</u>	Returns the largest integer value that is <= to a number
<u>LOG</u>	Returns the natural logarithm of a number, or the logarithm of a number to a specified base
<u>LOG10</u>	Returns the natural logarithm of a number to base 10
<u>MAX</u>	Returns the maximum value in a set of values
<u>MIN</u>	Returns the minimum value in a set of values
<u>PI</u>	Returns the value of PI
<u>POWER</u>	Returns the value of a number raised to the power of another number
<u>RADIANS</u>	Converts a degree value into radians
<u>RAND</u>	Returns a random number
<u>ROUND</u>	Rounds a number to a specified number of decimal places
<u>SIGN</u>	Returns the sign of a number
<u>SIN</u>	Returns the sine of a number
<u>SQRT</u>	Returns the square root of a number
<u>SQUARE</u>	Returns the square of a number
<u>SUM</u>	Calculates the sum of a set of values
<u>TAN</u>	Returns the tangent of a number

SQL Server Date Functions

Function	Description
<u>CURRENT_TIMESTAMP</u>	Returns the current date and time
<u>DATEADD</u>	Adds a time/date interval to a date and then returns the date
<u>DATEDIFF</u>	Returns the difference between two dates
<u>DATEFROMPARTS</u>	Returns a date from the specified parts (year, month, and day values)
<u>DATENAME</u>	Returns a specified part of a date (as string)

<u>DATEPART</u>	Returns a specified part of a date (as integer)
<u>DAY</u>	Returns the day of the month for a specified date
<u>GETDATE</u>	Returns the current database system date and time
<u>GETUTCDATE</u>	Returns the current database system UTC date and time
<u>ISDATE</u>	Checks an expression and returns 1 if it is a valid date, otherwise 0
<u>MONTH</u>	Returns the month part for a specified date (a number from 1 to 12)
<u>SYSDATETIME</u>	Returns the date and time of the SQL Server
<u>YEAR</u>	Returns the year part for a specified date

SQL Server Advanced Functions

Function	Description
<u>CAST</u>	Converts a value (of any type) into a specified datatype
<u>COALESCE</u>	Returns the first non-null value in a list
<u>CONVERT</u>	Converts a value (of any type) into a specified datatype
<u>CURRENT_USER</u>	Returns the name of the current user in the SQL Server database
<u>IIF</u>	Returns a value if a condition is TRUE, or another value if a condition is FALSE
<u>ISNULL</u>	Return a specified value if the expression is NULL, otherwise return the expression
<u>ISNUMERIC</u>	Tests whether an expression is numeric
<u>NULLIF</u>	Returns NULL if two expressions are equal
<u>SESSION_USER</u>	Returns the name of the current user in the SQL Server database

<u>SESSIONPROPERTY</u>	Returns the session settings for a specified option
<u>SYSTEM_USER</u>	Returns the login name for the current user
<u>USER_NAME</u>	Returns the database user name based on the specified id