

SQL FUNCTIONS





CLASS OUTLINE

FUNCTIONS:

- String Functions
- Date Functions
- Numeric Functions
- Miscellaneous

CLAUSES:

- LIMIT Clause
- GROUP BY clause
- HAVING Clause



STRING FUNCTIONS

- The most commonly used String Functions in SQL are:

FUNCTIONS	DESCRIPTION
LOWER / LCASE	Converts a string to lower-case.
UPPER / UCASE	Converts a string to upper-case.
CHAR_LENGTH	Returns the length of the string.
SUBSTRING	Extract a Substring from a String.
INSERT	Inserts a string within a string.
CONCAT	To add two or more strings together.

STRING FUNCTIONS

- **SUBSTRING** Function: This function helps in Extracting Substring from a given String.

Syntax :	Example :
SELECT SUBSTRING (<i>string, start, length</i>)	SELECT SUBSTRING ("Welcome", 4,4);

- **INSERT** Function: This function **inserts a string within a string** at the specified position and for a certain number of characters.

Syntax :	Example :
SELECT INSERT (<i>string, position, number, string_new</i>)	SELECT INSERT ("Welcome", 4, 4, "don");



STRING FUNCTIONS

CONCAT Function: This function adds two or more strings together.

Note: If any of the expressions is a NULL value, it returns a **NULL**.

Syntax :	Example :
SELECT CONCAT (<i>string1, string2, string3,...</i>);	SELECT CONCAT ("IOTA ", "Academy ", "SQL ", "Classes!");





STRING FUNCTIONS

CONCAT Function: This function adds two or more strings together.

Que: Write a SQL statement to **concat address details** of the Customers from Customer table of CustomerDB database.

Example :

```
SELECT CONCAT(Address, ", ", City, ", ", PostalCode, ", ", Country )  
FROM Customer;
```



DATE FUNCTIONS

- The most commonly used DATE Functions in SQL are:

FUNCTIONS	DESCRIPTION
NOW	Returns the current date and time.
CURRENT_DATE	Returns the current date.
DAY	Returns the day of the month for a given date.
MONTH	Returns the month part for a given date (1 to 12).
MONTHNAME	Returns the name of the month for a given date.
YEAR	Returns the year part for a given date.
ADDDATE	Adds a date interval to a date and then returns the date.

NUMERIC FUNCTIONS

In SQL, the most commonly used numeric functions are:

- AVG,
- COUNT
- SUM
- MAX
- MIN
- ROUND
- MOD



AGGREGATE FUNCTIONS

- In SQL, aggregate functions let you perform a calculation on multiple data and return a single value.
- Commonly used Aggregate functions are **AVG, COUNT, SUM, MAX** and **MIN**.
- The aggregate functions are often used with the **GROUP BY clause** to calculate an aggregate value for each group.





AGGREGATE FUNCTIONS

- The **COUNT()** function returns the **number of the value** in a set.
- The **AVG()** function calculates the **average value** of a set of values. It **ignores NULL** in the calculation.
- The **SUM()** function returns the **sum of values** in a set.
- The **MAX()** function returns the **maximum value in a set**.
- The **MIN()** function returns the **minimum value** in a set of values.



AGGREGATE FUNCTIONS

- The **COUNT()** function returns the number of records returned by a select query.

Syntax :	Example :
SELECT COUNT (column_name1) FROM table_name ;	SELECT COUNT (CustomerName) FROM Customer;

AGGREGATE FUNCTIONS

- The **MIN()** function returns the minimum value from the selected set of values in a column.

Syntax :	Example :
SELECT MIN (column_name1) FROM table_name ;	SELECT MIN (Price) FROM Products;

AGGREGATE FUNCTIONS

- The **AVG()** function returns the average value for the selected column.

Syntax :	Example :
SELECT AVG (column_name1) FROM table_name ;	SELECT AVG (Price) FROM Products;

NUMERIC FUNCTIONS

- The **ROUND()** function rounds a number to a specified number of decimal places.

Syntax :	Example :
SELECT ROUND (AVG(column_name1), decimals) FROM table_name ;	SELECT ROUND (AVG(Price), 2) FROM Products;

ALIASES (AS)

- Aliases are used to give a table, or a column in a table, a temporary name.
- Aliases are often used to make column names more readable.
- An alias only exists for the duration of that query.
- An alias is created with the **AS** keyword.

Syntax :	Example :
SELECT column_name AS alias_name FROM table_name;	SELECT CustomerName AS Customer FROM Customer;

LIMIT CLAUSE

- The **LIMIT clause** is used to specify the number of records to return.

Syntax :	Example :
SELECT column_name(s) FROM table_name WHERE condition LIMIT number;	SELECT City, Country FROM Customer LIMIT 5 ;

GROUP BY CLAUSE

- The GROUP clause groups rows into subgroups(set of summary rows) based on values of column.
- The GROUP BY clause is often used with an aggregate function to perform calculations and return a single value for each subgroup.

Syntax :	Example :
SELECT column_name(s) FROM table_name WHERE condition GROUP BY column_name(s) ORDER BY column_name(s);	SELECT COUNT (CustomerID), City FROM Customer GROUP BY City;



THANK YOU

