**SQL Functions**

**LEFT:**

Retrieves the first specified characters from ***LEFT*** side. It starts from the LEFT and goes on to RIGHT.

Syntax: LEFT(col\_name, no.of characters)

LEFT(first\_name, 3)

**RIGHT:**

Retrieves the first specified characters from ***RiGHT*** side. It starts from the RIGHT and goes on to LEFT. It acts quite opposite to LEFT function in SQL.

Syntax: RIGHT(col\_name, no\_of\_characters)

RIGHT(last\_name, 3)

**LENGTH:**

Describes how many no.of characters are present in that particular string.

Syntax: LEN(col\_name)

LEN(fist\_name)

**SUBSTRING:**

Retrieves the ***sub part*** of the string from the whole STRING.

Syntax: SUBSTR(col\_name, Position, No of char)

SUBSTR(first\_name, 1, 3)

In SQL Index counting starts from 1.

**POSITION:**  It takes a character and a column, and provides the index where that character is for each row. Index in SQL starts from 1.

Syntax: POSITION(“char” IN col\_name)

POSITION(‘A’, in first\_name)

**STRPOS:** provides same result as POSITION function. But the syntax for achieving those results is a bit different.

Syntax: STRPOS(col\_name, ‘char’)

STRPOS(city\_name, ‘A’)

The difference we have to learn is that in POSITION and STRPOS is that they both are case sensitive in different compilers. If you want to pull an INDEX regardless of case of a letter we can use either UPPER/LOWER functions.

**CONCAT:**  As we all know that this function usually behaves like CONCATIONG two or more strings together. It’s application is same as like in many other programming languages.

Syntax: CONCAT(Col1, Col1, )

**PIPING:** We can even add STRINGS together by using PIPING || operators.

Syntax: Col1 || Col2 || ‘ ‘, || Col3

PIPING will works only in some servers.

**CAST:** This function takes an argument as any data type and converts it into desired or specified data type.

Syntax: CAST(Exp or col\_name AS Data\_type(length)

**COALESCE:**

It returns the first non-NULL value passed for each row.

Syntax: Coalesce(col1, col2, col3,….) AS alised\_col\_name

Coalesce(mobile, phone\_number, landline\_number)

**TRIM:** It removes leading or tailing spaces or unwanted or unnecessary spaces from the first or last part of the string.

It can be done through TRIM() function.

Syntax: TRIM([characters FROM ]string)

TRIM(‘ AHLAD TANGIRALA ‘)---🡪 (AHLAD TANGIRALA)

We have LTRIM(), RTRIM() etc for removal of left or right parts of the spaces to remove from the character.

**INITCAP():** It converts every part of the STRING into capital for starting letter of the string.

Syntax: INITCAP(‘STRING or CHARACTER’)

INITCAP(ahlad tangirala)--🡪 Ahlad Tangirala

**REPLACE():** It will replace one part of STRING into desired or specified string.

Syntax: REPLACE(string, old\_string, new\_string)

REPLACE(‘Ahlad Tangirala, ‘Ahlad’, ‘Akshay’)-🡪 Akshay Tangirala