Integer Data type

DDL for integer-type data is

CREATE TABLE customer (

Customer\_id int,

credit integer

);

INSERT INTO customer (customer\_ID, credit) VALUES(10,12345.5);

Float Data type

DDL for float-type data is

CREATE TABLE customer (

Customer\_id int,

credit float

);

INSERT INTO customer (customer\_ID, credit) VALUES(10,12345.57654);

Date and Time Data type

Date

DDL for date-type data is

CREATE TABLE customer (

Customer\_id int,

Transaction\_date date, -- date type column declaration

Credit float

);

INSERT INTO customer (customer\_ID, transaction\_date, credit) VALUES(10,11-10-2022,12345.57654);

INSERT INTO customer (customer\_ID, transaction\_date, credit) VALUES(11,2022-10-12,12345.57654);

Time

DDL for time-type data is  
CREATE TABLE customer (

Customer\_id int,

Transaction\_date date,

Transaction\_time time,

Credit float

);

INSERT INTO customer (customer\_ID, transaction\_date, transaction\_time, credit) VALUES(10,11-10-2022,12345.57654);

String Data Type

Char

DDL for character type data is

CREATE TABLE customer (

Customer\_id int,

Customer\_name char(60) -- Here 60 defines the maximum length of the value which can be stored in in this column

);

INSERT INTO customer (customer\_ID, customer\_name) VALUES(10,’JK’);

Varchar

DDL for this data type is   
create table test(

col1 varchar(120) -- This means it can hold 120 single-byte characters, 60 double-byte characters, 40 three-byte characters, or 30 4 byte characters.

);

SQL String Functions

RIGHT()

Example:

SELECT RIGHT(‘This is String Right function’,14)

Let’s look at one more example for customer table:

Select right(Name,4) as output from customer;

LEFT()

Example:

SELECT left(‘This is String Left function’,4)

Let’s look at one more example for customer table:

Select left(Name,4) as output from customer;

Length()

Example:

SELECT length‘This is String length function’)

Let’s look at one more example for customer table:

Select length(Name) as output from customer;

TRIM()

Example:

SELECT TRIM(' This is String TRIM function ');

RTRIM()

Example:

select len(‘ This is String RTRIM function ’)

LTRIM()

Example:

select len(‘ This is String RTRIM function ’)

RPAD()

Example:

Select RPAD(‘6’,4,’0’) as new\_column;

Another Example:

SELECT

first\_name,

RPAD(first\_name, 10, '\*') AS padded\_value

FROM customers;

LPAD()

Example:

Select LPAD(‘8’,4,’0’) as new\_column;

Let’s see another example

SELECT

last\_name,

RPAD(LPAD(last\_name, 10, '#'), 15, '\*') AS padded\_value

FROM customers;

Example:

select replace(‘String Function ’,’n’,’$’) -- This will replace all the occurrences of ‘n’ by a ‘$’.

Let's look at another example:

SELECT Name,

       value,

       REPLACE (REPLACE (REPLACE(value, 'A', '5'), 'C', 9), 'D', 4) as new\_value

FROM datatable;

REVERSE()

Example:

Select reverse (‘String’)

Example:

Select substring(‘This is String function’,3,7)

CAST()

Example 1:

Select CAST(‘2022-12-12’ as VARCHAR) as date\_varchar -- Saves the date value as a varchar data type

Here, we are using the CAST function to convert the SCORE column from type FLOAT to INTEGER

SELECT First\_Name,CAST (Score AS Integer) Int\_Score FROM Satisfaction;

CONCATENATE()

Example:

Select concatenate(‘This’,’is’,’an’,’example’) as variable -- This will create a new value and store in the column variable

CONCATENATE\_WS()

For Example:

SELECT CONCAT\_WS(‘-‘,‘This’,’is’,’an’,’example’) -- This will add ‘-‘ separator after every string

UPPER Function

Example 1:

Select Upper('this is data wrangling for string') As column1\_new from test\_table;

Example 2:

Select Upper(‘This IS data wrangling for string’) As column1;

Example 3:

Select upper(Address) from customers

Example 5:

Using upper with update statement:

Update Customers Set Name=upper(Name)

We can also use upper to update for a specific condition using WHERE clause:

Update Customers Set Address= UPPER(Address) Where Name=’FRED’

Example 6:

Select \* From Customers Where upper(Address)=’Tennessee’

LOWER Function

Example 1:

Select LOWER(‘This IS data wrangling for string’) As column1\_new from test\_table;

Example 2:

Select **lower(Address)** from customers

Example 3:

Update Customers Set Name=lower(Name)

Example 4:

Select \* From Customers Where lower(Address)=’new york’

INITCAP Function

For Example:

Select INITCAP(‘this is data wrangling for string’)

Practical Exercises

Practical Exercise 1

SELECT LENGTH(last\_name) as "Length",

CONCAT(first\_name, ' ', last\_name) as "Full Name",

SUBSTRING(first\_name, 1, 1) as "Initial",

REPLACE(email, '@example.com', '') as "Username",

UPPER(last\_name) as "Last Name (Uppercase)"

FROM users

WHERE LOWER(first\_name) = 'john';

Practical Exercise 2

SELECT LENGTH(name), UPPER(name), SUBSTRING(name, 1, 3)

FROM users

WHERE TRIM(email) = 'packt.wrangling@example.com';

Practical Exercise 3

SELECT employee\_id,

CONCAT(first\_name, ' ', last\_name) as full\_name,

LENGTH(first\_name) as first\_name\_length,

UPPER(last\_name) as last\_name\_uppercase

FROM employees;

Practical Exercise 4

SELECT

UPPER(first\_name) as first\_name,

LOWER(last\_name) as last\_name,

CONCAT(first\_name, ' ', last\_name) as full\_name,

LENGTH(email) as email\_length,

SUBSTRING(email, 1, INSTR(email, '@')-1) as username

FROM customers;