**Character function**

**Case manipulation function:**

1. This function can act as only in letters.

**1.Upper:**

1. It is used **to convert** all the letters to **upper case.**
2. If there are characters in the string that are not letters, they are **unaffected by this function.**

**Syntax:**

1. Select upper(‘mahe’) from dual;
2. Select upper(‘mahe’),upper(‘usha’) from dual;
3. Select \* from emp where ename=upper(‘allen’);

**2.Lower:**

1. Lower function is used to convert all the letters to lower case.

**Syntax:**

1. Select lower(‘mahe’) from dual;
2. Select lower(‘mahe’),lower(‘usha’) from dual;
3. Select \* from emp where ename=lower(‘allen’);

**3.Initcap:**

1. It is used to convert first letter of all the word in a string to uppercase and rest of the words to lower case.

**Syntax:**

1. Select initcap(‘mahe’) from dual;
2. Select initcap(‘mahe’),initcap(‘usha’) from dual;
3. Select \* from emp where ename=initcap(‘allen’);

**Character manipulation function:**

1. This function can act as only in letters.
2. But it has more pre-defined method than case manipulation function.

**1.Concat:**

* CONCAT function **allows you to concatenate two strings together.**
* Concat functions **accept only two arguments.**

**SYNTAX:**

CONCAT( string1, string2 )

Ex:

SQL> select ename,job, concat(ename,job) from emp;

**Concatenate Single Quotes**

SELECT CONCAT('Let''s', ' learn Oracle') FROM dual;

**Syntax:**

SQL> select concat('hi',' bye')

from dual;

* select concat('hi','hello','bye') from dual

SQL> /

select concat('hi','hello','bye')

\*

ERROR at line 1:

ORA-00909: invalid number of arguments

**2.Length:**

1. It is used to **find the length.**

**Syntax:**

Select ename,length(ename) from emp;

**3.Reverse:**

1. It is **used to reverse the even string.**

**Syntax:**

Select ename,reverse(ename) from emp;

**4.Replace:**

1. It is **used to replace the string to be replaced.**

**Syntax:**

Replace(String,String-to-be-replaced,replacement-string)

SQL> select replace('jspider','j','q') from dual;

REPLACE

-------

Qspider

SQL> select replace('jspiders','s','m') from dual;

REPLACE(

--------

Jmpiderm

SQL> select replace('jspiders','sp','n') from dual;

REPLACE

-------

Jniders

SQL> select replace('jspiders','sp','mahe') as result from dual;

RESULT

----------

Jmaheiders

**5.Substring:**

1. It is **used to extract substring in the given string.**
2. Parameters or Arguments
3. String the source string.
4. Start position the starting position for extraction.
5. The **first position in the string is always 1.**
6. Length Optional. It is the number of characters to extract. If this parameter is omitted, the SUBSTR function will return the entire string where position and length are both integers.

**Syntax:**

Substr(string,start\_position,length);

Note:

If starting position is don’t know then we did not perform substring operation. But we can extract that using instr function.

**6.Trim:**

1. It is used to remove some spaces or character in a given string.

**Types:**

1. LTRIM
2. RTRIM
3. TRIM

**LTRIM:**

1. It is **used to remove some spaces or a character** in a given string from **left hand side.**

**RTRIM:**

1. It is **used to remove some spaces or a character** in a given string from **right hand side.**

**TRIM:**

1. It is **used to remove some spaces or a character** in a given string from **both sides.**

**SYNTAX:**

TRIM(LEADING/TRAILING/BOTH TRIM\_CHARACTER FROM STRING);

Leading-ltrim

Trailing-rtrim

Both-trim both side

SQL> select trim(both 'ee' from 'eewelcome') from dual;

select trim(both 'ee' from 'eewelcome') from dual

\*

ERROR at line 1:

ORA-30001: trim set should have only one character

**Note:**

In trim we cannot have more than one character as a trim character.

**7.INSTR:**

1. Instring is **used to find the position of the substring** in a given string.
2. If substring is not present in the given string then **it returns zero [0].**

**Syntax:**

INSTR(STRING,SUB\_STRING,START\_POSITION,LENGTH(nth occurance))