Ex.No.8 .04.2025	STRING FUNCTONS

AIM

To perform sql operations by using string functions.

CREATING A TABLES

SQL> CREATE TABLE SYSTEM(ID NUMBER(5), NAME VARCHAR2(20), PLACE VARCHAR2(20));

Table created.

SQL> CREATE TABLE DEPART(D_ID NUMBER(5), D_NAME VARCHAR2(20), EMAIL VARCHAR(30));

Table created.

INSERTING VALUES INTO TABLE

SQL> INSERT INTO SYSTEM VALUES(1, 'Praveen', 'Erode');
1 row created.

SQL> INSERT INTO SYSTEM VALUES(2, 'Shangav', 'Salem');
1 row created.

SQL> INSERT INTO SYSTEM VALUES(3, 'Karthik', 'Chennai');
1 row created.

SQL> INSERT INTO SYSTEM VALUES(4, 'Jegan', 'Coimbatore');
1 row created.

SQL> INSERT INTO SYSTEM VALUES(5, 'Sanjai', 'Madurai');
1 row created.

SQL> INSERT INTO SYSTEM VALUES(6, 'Mounish', 'Tiruppur');
1 row created.

SQL> INSERT INTO SYSTEM VALUES(7, 'Kamalesh', 'Trichy');
1 row created.

SQL> INSERT INTO SYSTEM VALUES(8, 'Babu', 'Namakkal');

1 row created.

SQL> INSERT INTO DEPART VALUES(201, 'AI', 'Praveen.23ai@kongu.edu'); 1 row created.

SQL> INSERT INTO DEPART VALUES(202, 'DS', 'shangav.23ds@kongu.edu');
1 row created.

SQL> INSERT INTO DEPART VALUES(203, 'CS', 'karthik.23cs@kongu.edu');
1 row created.

SQL> INSERT INTO DEPART VALUES(204, 'SE', 'jegan.23se@kongu.edu');
1 row created.

SQL> INSERT INTO DEPART VALUES(205, 'ML', 'sanjai.23ml@kongu.edu');
1 row created.

SQL> INSERT INTO DEPART VALUES(206, 'IOT', 'mounish.23iot@kongu.edu');
1 row created.

SQL> INSERT INTO DEPART VALUES(207, 'DA', 'kamalesh.23da@kongu.edu');
1 row created.

SQL> INSERT INTO DEPART VALUES(208, 'IT', 'babu.23it@kongu.edu');
1 row created.

ASCII (CHR):

SELECT ASCII('a') AS ascii_value FROM dual;

ASCII_VALUE

97

VALUE(CHR):

SELECT CHR(75) AS character FROM dual;

C

-

K

CONCAT:

SELECT NAME || ' lives in ' || PLACE AS student_details FROM SYSTEM;

STUDENT_DETAILS

Praveen lives in Erode
Shangav lives in Salem
Karthik lives in Chennai
Jegan lives in Coimbatore
Sanjai lives in Madurai
Mounish lives in Tiruppur
Kamalesh lives in Trichy
Babu lives in Namakkal

8 rows selected.

UPPERCASE & LOWERCASE:

SELECT UPPER(NAME) AS upper_name, LOWER(NAME) AS lower_name FROM SYSTEM;

UPPER_NAME	LOWER_NAME
Praveen	Praveen
SHANGAV	shangav
KARTHIK	karthik
JEGAN	jegan
SANJAI	sanjai
MOUNISH	mounish
KAMALESH	kamalesh
BABU	babu

8 rows selected.

LENGTH(STR):

SELECT NAME, LENGTH(NAME) AS name_length FROM SYSTEM;

NAME	NAME_LENGTH
Praveen	5
Shangav	7
Karthik	7
Jegan	5
Sanjai	6
Mounish	7
Kamalesh	8
Babu	4

8 rows selected.

REPLACE(STR):

```
SELECT REPLACE(NAME, 'Karthik', 'Karthikeyan') AS replaced_name FROM SYSTEM;
```

```
REPLACED_NAME
-------------------------------------------
Praveen
Shangav
Karthikeyan
Jegan
Sanjai
Mounish
Kamalesh
Babu

8 rows selected.
```

SUBSTR():

```
SELECT SUBSTR('Praveen rocks', 7, 5) AS substring FROM dual;

SUBST
------
Rocks
```

RPAD() & LPAD():

```
SELECT RPAD(NAME, 10, '*') AS right_pad, LPAD(NAME, 10, '*') AS left_pad FROM SYSTEM;

RIGHT_PAD        LEFT_PAD
----------       ----------
Praveen*****      *****Praveen
Shangav***        ***Shangav
Karthik***        ***Karthik
Jegan*****        *****Jegan
Sanjai****        ****Sanjai
Mounish***        ***Mounish
Kamalesh**        **Kamalesh
Babu******        ******Babu

8 rows selected.
```

LTRIM & RTRIM:

```
SELECT LTRIM('   KAVIN') AS ltrimmed, RTRIM('KAVIN   ') AS rtrimmed FROM dual;

LTRIM  RTRIM
------         ------
KAVIN  KAVIN
```

REVERSE:

```
SELECT REVERSE(NAME) AS reversed_name FROM SYSTEM;

REVERSED_NAME
----------------------
yyappanI
vagnahS
kihtraK
nageJ
iajnaS
hsinuoM
hselamaK
ubaB

8 rows selected.
```

EXTRACTING THE DOMAIN OF EMAIL:

```
SELECT SUBSTR(EMAIL, INSTR(EMAIL, '@') + 1) AS email_domain FROM DEPART WHERE
EMAIL IS NOT NULL;

EMAIL_DOMAIN
---------------------------------
kongu.edu
kongu.edu
kongu.edu
kongu.edu
kongu.edu
kongu.edu
kongu.edu
kongu.edu

8 rows selected.
```

CHANGING THE DOMAIN NAME:

```
SELECT REPLACE(EMAIL, SUBSTR(EMAIL, INSTR(EMAIL, '@') + 1), 'gmail.com') AS
updated_mail FROM DEPART;


UPDATED_MAIL
--------------------------------------------------
kavin.23ai@gmail.com
shangav.23ds@gmail.com
karthik.23cs@gmail.com
jegan.23se@gmail.com
sanjai.23ml@gmail.com
mounish.23iot@gmail.com
kamalesh.23da@gmail.com
babu.23it@gmail.com

8 rows selected.
```

SEARCHING OPERATIONS:

LIKE KEYWORD:

```
SELECT D_NAME FROM DEPART WHERE D_NAME LIKE 'D%';


D_NAME
----------
DS
DA
SQL> SELECT D_NAME FROM DEPART WHERE D_NAME LIKE '_A%';


D_NAME
----------
DA
SQL> SELECT D_NAME FROM DEPART WHERE D_NAME NOT LIKE 'S%';
```

D_NAME

AI

DS

CS

ML

IOT

DA

IT

7 rows selected.

NULL:

SELECT * FROM DEPART WHERE EMAIL IS NULL;

no rows selected

NOT NULL:

SELECT * FROM DEPART WHERE EMAIL IS NOT NULL;

D_ID	D_NAME	EMAIL
201	AI	kavin.23ai@kongu.edu
202	DS	shangav.23ds@kongu.edu
203	CS	karthik.23cs@kongu.edu
204	SE	jegan.23se@kongu.edu
205	ML	sanjai.23ml@kongu.edu
206	IOT	mounish.23iot@kongu.edu
207	DA	kamalesh.23da@kongu.edu
208	IT	babu.23it@kongu.edu

8 rows selected.

CONTENTS	MARKS ALLOTED	MARKS OBTAINED
Aim,Algorithm,SQL,PL/SQL	30	
Execution and Result	20	
Viva	10	
Total	60	

RESULT

Thus, multiple string operations were carried out. These operations facilitated the effective manipulation, formatting, and searching of string data according to various conditions.