

Experiment 4

Queries using Conversion functions (to_char, to_number and to_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr), date functions (Sysdate, next_day, add_months, last_day, months_between, least, greatest, trunc, round, to_char, to_date)

```
SQL> select *from emp;
```

ENO	ENAME	SALARY	LOC
101	ali	15000	vja
102	haji	20000	hyd
103	mohammad	42000	vja
104	ravi	23000	gnt
105	irfath	50000	hyd

a) Conversion Functions:

1. to_char: to_char is used to convert the attribute values to char.

```
SQL> select to_char(salary,'$99999.99') from emp;
```

TO_CHAR(SALARY)

```
-----  
$15000.00  
$20000.00  
$42000.00  
$23000.00  
$50000.00
```

```
SQL> SELECT TO_CHAR (123.4567, '99999.9') FROM DUAL;
```

TO_CHAR(

```
-----  
123.5
```

```
SQL> SELECT TO_CHAR(1234.56789,'9,999.00') FROM DUAL;
```

```
TO_CHAR(1
```

```
-----
```

```
1,234.57
```

```
SQL> SELECT TO_CHAR(SYSDATE, 'YYYY/MM/DD') FROM DUAL;
```

```
TO_CHAR(SY
```

```
-----
```

```
2025/01/24
```

```
SQL> SELECT TO_CHAR (23, '000099') FROM DUAL;
```

```
TO_CHAR
```

```
-----
```

```
000023
```

```
SQL> SELECT TO_CHAR (234.5678, '000.000') FROM DUAL;
```

```
TO_CHAR(
```

```
-----
```

```
234.568
```

```
SQL> SELECT TO_CHAR (234.5678, '00.00') FROM DUAL;
```

```
TO_CHA
```

```
-----
```

```
#####
```

```
SQL>SELECT TO_CHAR(sysdate, 'Month DD, YYYY') from dual;
```

```
TO_CHAR(SYSDATE,'M
```

```
-----
```

```
January 24, 2025
```

2. **to_number**: to_number is used to convert the attribute value to number.

```
SQL> SELECT TO_NUMBER('1210.73', '9999.99') FROM DUAL;
```

```
TO_NUMBER('1210.73','9999.99')
```

```
-----
```

```
1210.73
```

3. **to_date:** to_date is used for convert and display the attribute values as date.

```
SQL> select to_date('01-01-2025', 'MM-DD-YYYY') from dual;
```

```
TO_DATE('
```

```
-----
```

```
01-JAN-25
```

b) String functions:

1. Concatenation: CONCAT is used to Combines result from several different fields.

Syntax:

```
CONCAT(string1, string2)
```

```
SQL> select concat (eno, loc) from emp;
```

```
CONCAT(ENO,LOC)
```

```
-----
```

```
101vja
```

```
102hyd
```

```
103vja
```

```
104gnt
```

```
105hyd
```

2. lpad: Used to pad the string to left to length n characters. string_pad parameter is optional. If not specified, string will be padded spaces to the left-side of string.

Syntax:

```
lpad ('string', n , 'string_pad')
```

```
SQL> select lpad(ename,10,'*') from emp;
```

```
LPAD(ENAME
```

```
-----
```

```
*****ali
```

```
*****haji
```

```
**mohammad
```

*****ravi

****irfath

SQL> Select lpad('tech', 7) from dual;

LPAD('T

tech

3. rpad: Used to pad the string to right to length n characters. string_pad parameter is optional.
If not specified, string will be padded spaces to the right-side of string.

Syntax:

 rpad ('string', n , 'string_pad')

SQL> select rpad(ename,10,'*') from emp;

RPAD(ENAME

ali*****

haji*****

mohammad**

ravi*****

irfath****

SQL> Select rpad('tech', 7) from dual;

RPAD('T

tech

4. ltrim: It is used to remove all white spaces or specified characters from the left end side of a string

Syntax: ltrim(string)

SQL> select ltrim('*****hi*****', '*') from dual;

LTRIM('***

hi*****

```
SQL> SELECT LTRIM(' Sample ') from dual;
```

```
LTRIM('
```

```
-----
```

```
Sample
```

5.rltrim: It is used to remove all white spaces specified characters from the right end side of a string

Syntax: rtrim(string)

```
SQL> select rtrim('*****hi*****','*') from dual;
```

```
RTRIM('*
```

```
-----
```

```
*****hi
```

```
SQL> SELECT rTRIM('Sample ') from dual;
```

```
RTRIM('
```

```
-----
```

```
Sample
```

6. lower: Returns a string in uppercase converted to lowercase

Syntax: LOWER(string):

```
SQL> SELECT LOWER('sANd') from dual;
```

```
LOWE
```

```
----
```

```
Sand
```

```
SQL> select lower(ename) from emp;
```

```
LOWER(ENAM
```

```
-----
```

```
ali
```

```
haji
```

```
mohammad
```

```
ravi
```

```
irfath
```

7. upper: Returns a string in lowercase converted to uppercase

Syntax: UPPER(string):

```
SQL> SELECT upper('sANd') from dual;
```

```
UPPE
```

```
----
```

```
SAND
```

```
SQL> select upper(ename) from emp;
```

```
UPPER(ENAM
```

```
-----
```

```
ALI
```

```
Haji
```

```
MOHAMMAD
```

```
RAVI
```

```
IRFATH
```

8. initcap: Returns a string with each word's first character in uppercase and the rest in lowercase.

Syntax: INITCAP(character-expression)

```
SQL> select initcap (ename) from emp;
```

```
INITCAP(EN
```

```
-----
```

```
Ali
```

```
Haji
```

```
Mohammad
```

```
Ravi
```

```
Irfath
```

9. length: Returns the length of the string.

Syntax: LENGTH(string)

```
SQL> select ename,length(ename) from emp;
```

```
ENAME    LENGTH(ENAME)
```

```
-----
```

```
ali
```

```
3
```

haji	4
mohammad	8
ravi	4
irfath	6

```
SQL> select length(ename) from emp where salary=20000;
LENGTH(ENAME)
```

```
-----
4
```

10. substr: Displays the string Starting with the **pos** character in string and select the next characters.

Syntax: SUBSTR(str,pos,len);

```
SQL> select ename, substr(ename,4) from emp;
ENAME SUBSTR(
```

```
-----
ali
haji      i
mohammad  ammad
ravi      i
irfath    ath
```

```
SQL> SELECT SUBSTR('San Diego',2,4) from dual;
SUBS
```

```
----
an D
```

11. instr: instr() function return the location of starting position of the sub string in the existing value.

```
SQL> select instr('welcome to CRRCOE','to') from dual;
INSTR('WELCOMETOCRRCOE','TO')
```

```
-----
9
```

Returns the position of the first occurrence of substring substr in string str

```
SQL> SELECT INSTR('foobarbar', 'bar') from dual;
```

```
INSTR('FOOBARBAR','BAR')
```

```
-----
```

4

c) Date functions:

1. Sysdate(): sysdate() function returns the current system date.

```
SQL> select sysdate from dual;
```

```
SYSDATE
```

```
-----
```

24-JAN-25

2. next_day(): It returns the date of next coming day .

```
SQL> select next_day(sysdate,'sunday') from dual;
```

```
NEXT_DAY(
```

```
-----
```

26-JAN-25

3. add_months(): It returns the next date after adding number of months in the arguments.

Syntax: add_months(date1, n)

```
SQL> select add_months(sysdate,5) from dual;
```

```
ADD_MONTH
```

```
-----
```

24-JUN-25

4. last_day(): Returns the last day of the date specified.

Syntax: LAST_DAY(date1)

```
SQL> select last_day(sysdate) from dual;
```

```
LAST_DAY(
```

```
-----
```

31-JAN-25

5. months_between(): It returns the numbers of months between given two dates.

Syntax: MONTHS_BETWEEN(date1,date2)

```
SQL> select months_between('02-feb-2025','02-feb-2024') from dual;
```

```
MONTHS_BETWEEN('02-FEB-2025','02-FEB-2024')
```

```
-----
```

```
12
```

```
SQL> select months_between(sysdate,'02-feb-2024') from dual;
```

```
MONTHS_BETWEEN(SYSDATE,'02-FEB-2024')
```

```
-----
```

```
11.7320232
```

6. least(): it returns least value from the given argument or attributes.

```
SQL> select least(300,450,100,440) from dual;
```

```
LEAST(300,450,100,440)
```

```
-----
```

```
100
```

7. greatest(): it returns maximum values from the given arguments or attributes in the relation

```
SQL> select greatest(300,450,100,440) from dual;
```

```
GREATEST(300,450,100,440)
```

```
-----
```

```
450
```

8. trunc(): The TRUNC() function returns a DATE value truncated to a specified unit.

```
SQL> select trunc(sysdate,'mm') from dual;
```

```
TRUNC(SYS
```

01-JAN-25

9. round(): Round function round a number to a specified length or precision.

SQL> select round(12.49,0) from dual;

ROUND(12.49,0)

12

SQL> select round(12.51,0) from dual;

ROUND(12.51,0)

13

10. to_char(): it convert the given date type attribute values to text and return the date in the specific format.

SQL> select to_char(sysdate,'yyyy-mm-dd') from dual;

TO_CHAR(SY

2025-01-24

11. to_date(): It will convert either a character string or an expression into a date value

SQL> SELECT TO_DATE('20100105', 'YYYYMMDD') FROM DUAL;

TO_DATE('

05-JAN-10