ST. FRANCIS INSTITUTE OF TECHNOLOGY

MT. POINSUR, BORIVALI (W), MUMBAI

LAB MANUAL

EXPERIMENT NO. 4

Aim: Perform Simple queries based on Numeric, Character and Date SQL functions.

Theory:

Explain different Numeric, Character and Date SQL functions.

Lab Manual:

a) Numeric Functions: It accept numeric input and return numeric values as output.

i. Abs (n): This function returns the absolute value of the given number. **SQL> select ABS (-15) from dual;**

```
Absolute
----------
   15
```

ii. Cos: COS returns the cosine of an angle expressed in radians.

```
SQL> select cos (0) from dual;
COS (0)
----------
   1
```

iii. Sin: `SIN` returns the sine of an angle expressed in radians.

```
SQL> select sin (30 * 3.14/180) "Sine" from dual;

Sine
----------
   .5
```

iv. Tan: `TAN` returns the tangent of an angle expressed in radians.

```
SQL> select tan (135 * 3.14/180) "Tan" from dual;
Tan
----------
   -1
```

v. Round: `ROUND` returns *a value* rounded to *integer* specified to be placed to the right of the decimal point.

```
SQL> select round (15.193, 1) "Round" from dual;

Round
----------
   15.2
```

vi. Truncate: The `TRUNC` function returns *a number value (x)* truncated by a number (y) to its decimal places.

SQL> select Trunc(15.79, 1) "Truncate" from dual;

Truncate

15.7

vii. Power: `POWER` returns *n2* raised to the *n1* power.
SQL> select Power (3, 2) "Power" from dual;
Power

9

b) Character Functions: It accept character input and return character values as output.
a) LOWER: Converts mixed case or uppercase character string to lowercase. **SQL> select lower (city) from emp;**

LOWER (CITY)
----
mumbai
pune
nagpur

b) UPPER: Converts mixed case or lowercase character string to uppercase.
SQL> select upper (city) from emp;

UPPER (CITY)

MUMBAI
PUNE
NAGPUR

c) INITCAP: Converts first letter of each word to uppercase and remaining letters to lowercase.
SQL> select initcap (f_name) from emp;

INITCAP (F_NAME)

Anil
Sunil
Smita

d) CONCATE: Use to concatenate value of one or more columns and display it or concatenate string with the columns and display it.
SQL> select concat(ID, f_name) from emp;
CONCAT (ID, F_NAME)

E01 Anil

E02 Sunil
E03 Smita

e) LTRIM: Use to trim/cut characters contained in the set from the left side.
SQL> select Ltrim('Nagpur', 'Nag') as Leftm from emp where f_name= 'Anil';
LEF
------
pur

f) RTRIM: Use to trim/cut characters contained in the set from the rightside. **SQL>**
select Rtrim ('Mumbai', 'bai') from emp where f_name= 'Sunil';

RTR

Mum

g) SUBSTRING: It returns a part of string, beginning at a given character position and ends where the substring_length specified ends. If the staring character of substring not specified then it takes first character as default.
SQL> Select substr ('Mumbai', 1, 3) "Substring" from dual;
Substring

Mum

h) LENGTH: It takes the character as input and returns the length of the string as an output. Output is measured in bits by default, for measuring in bytes we need to write Lengthb.
SQL> select Length ('Nagpur') as Len from emp where f_name= 'Smita';
LEN

6

i) REPLACE: It takes the string as input in which character needs to be replaced. Also takes old and new characters that are to be replaced with each other. So it searches the old character occurrence in the string and replaces it with the new character.
SQL> Select replace ('Jack and Jue','J', 'BL') "Changed_String" from dual;
Changed_String

Black and Blue

c) Date Time Functions: Oracle stores dates in default date format as DD-MON-YY. It basically comprises of: century, year, month, day, hours, minutes, and seconds.

a) TO_CHAR: Converts a date or number to a string
SQL> Select to_char (DOJ, 'Month, DD, Year') as "Date_in_string" from Month;
Date_in_string

JAN, 01, 2013

JAN, 01, 2013

b) ADD MONTH (D, N): It takes as input a date and adds one month to that date and displays the result.

SQL> Select add_months (DOL, 1) as "Added month" from Month;

Added month

01-MAR-13
01-MAY-13

c) LAST DAY (D): It takes as input a date and for that date returns the last date of its month.

SQL> Select Last_day (DOL, 1) as "Last day of month" from Month;

Last day of month

28-FEB-13
30-APRIL-13

d) MONTHS_BETWEEN (D1, D2): It takes as input 2 dates and returns number of months between these two dates.

SQL> Select months_between (DOL, DOJ) as "Bet_month" from Month;

Bet_month

1
3

e) MONTH (DATE): It takes as input a date and returns the month number of that date as a result.

SQL> Select month (DOL) as "Month no" from Month;

Month no

2
4

f) MONTHNAME (DATE) It takes as input a date and returns the month name of that date as a result.

SQL> Select monthname (DOL) as "Month name" from Month;

Month name

February
April

g) DAYNAME (DATE): It takes as input a date and returns the weekday name of that date as a result.

SQL> Select dayname (DOL) as "Week day" from Month;

Week day

Friday
Monday

h) DAYOFMONTH (DATE): It takes as input a date and returns which day it is of that month, the result is always between 1 t0 31.

SQL> Select dayofmonth (DOL) as "Day no" from Month;
Day no

1
1

i) DAYOFWEEK (DATE): It takes as input a date and returns which day it is of the week, the result is always between 1 t0 7.

SQL> Select dayofweek (DOL) as "Week Day no" from Month;
Week Day no

6
2

j) DAYOFYEAR (DATE): It takes as input a date and returns which day it is of that year, the result is always between 1 t0 365.

SQL> Select dayofyear ('27-02-03') as "Year Day no" from dual;
Year Day no

34

k) NOW (): It is a date function used to display from dual the system current date and time together.

SQL> Select Now ( );
Now ()

'2014-06-17 23:50:26'

l) CURDATE (): It is a date function used to display from dual the system current date.

SQL> Select Curdate ( );
Curdate ()

'2014-06-17'

m) CURTIME (): It is a date function used to display from dual the system current time.

SQL> Select Curtime ( );

Curtime ()
----

n) NEXTDAY(): It is a date function used to display the next day.

SELECT NEXT_DAY('02-FEB-2001','TUESDAY') "NEXT DAY"

LAB MANUAL

FROM DUAL;

NEXT DAY

06-FEB-2001

'23:50:26'

Implementation:

5. Apply SQL function for given exercise

Exercise:

1. Create table Flight(F_id, DoT, DoI, Time, Pass_name, Source, Destination, Fare) where DOT is Date of travel and DOI is the date of issue

```
Create table flight(F_id number(5) primary key, DoT varchar(11), DoI varchar(11), TTime varchar(6), Pass_name varchar(40), Source varchar(40),Destination varchar(40), Fare number(5))
```

Table created.

2. Insert 5 values

```
select * from flight
```

F_ID	DOT	DOI	TTIME	PASS_NAME	SOURCE	DESTINATION	FARE
1	01-Jan-2021	1-Jan-2020	10:00	Andrei	Mumbai	Goa	4000
2	01-Feb-2021	1-Jan-2020	01:00	Sam	Mumbai	Delhi	4000
3	01-Mar-2021	1-Jan-2020	02:00	Ram	Mumbai	Kerala	4000
4	01-Apr-2021	1-Jan-2020	03:00	Shyam	Mumbai	Kashmir	4000
5	01-May-2021	1-Jan-2020	04:00	Mansi	Mumbai	Daman	4000

3. Display today's date and time in the prompt.

```
select current_timestamp from dual
```

CURRENT_TIMESTAMP
07-APR-21 06.04.23.610767 AM US/PACIFIC

LAB MANUAL

4. Display the absolute value of -184.

```
select abs(-184) from dual
```

ABS(-184)
184

5. Select a value from the dual and for that value find its cube.

```
select Power (4, 3) from dual
```

POWER(4,3)
64

6. Display the date (doI) 2 months after date of Issue of Ticket.

```
Select add_months (DOI, 2) as "Added month" from Flight
```

Added month
01-MAR-20
01-MAR-20
01-MAR-20
01-MAR-20
01-MAR-20

7. Display the last day of month of date of Travel.

```
select last_day(DoT) from Flight
```

LAST_DAY(DOT)
31-JAN-21
28-FEB-21
31-MAR-21
30-APR-21
31-MAY-21

8. Display the month between date of travel and date of Issue.

```
select months_between (DoT, DoI) "Bet_month" from flight
```

Bet_month
12
13
14
15
16

9. Display the next occurrence of Monday from the day of Travel.

```
select next_day(DoT,'Monday') as "next_monday" from flight
```

next_monday
04-JAN-21
08-FEB-21
08-MAR-21
05-APR-21
03-MAY-21

10. Display the First letter of Pass_name into capitals.

```
select initcap(Pass_name) from flight
```

INITCAP(PASS_NAME)
Andrei
Sam
Ram
Shyam
Mansi

11. Display the Pass_name into upper case.

LAB MANUAL

```
select upper(Pass_name) from flight
```

UPPER(PASS_NAME)
ANDREI
SAM
RAM
SHYAM
MANSI

12. Display the Destination & source name into Lower case.

```
select lower(destination), lower(source) from flight
```

LOWER(DESTINATION)	LOWER(SOURCE)
goa	mumbai
delhi	mumbai
kerala	mumbai
kashmir	mumbai
daman	mumbai

13. Display the first 3 characters of the Destination place name.

```
select substr((destination), 1,3) "latsDest." from Flight
```

latsDest.
Goa
Del
Ker
Kas
Dam

14. Display the last 3 characters of the Destination place name.

```
select substr(reverse(destination), 1,3) "latsDest." from Flight
```

latsDest.
aoG
ihl
ala
rim
nam

15. Display the pass name that begins with ‘m’ and replace with ‘B’.

```
select replace (pass_name , 'M' , 'D') "Changed_string" from flight
```

Changed_string
Andrei
Sam
Ram
Shyam
Dansi

16. Display only 3 characters from the 3rd character with names of Source.

```
select substr(source, 3,3) "Sou" from Flight
```

Sou
mba
mba
mba
mba
mba

LAB MANUAL

17. Display the rounded value of fare up to 2 characters.

```
SELECT ROUND(Fare, 2) AS RoundValue from flight
```

ROUNDVALUE
4000
4000
4000
4000
4000

18. Display 20th September 2008 in the date format.

```
select TO_DATE('20, September, 2008') from dual
```

TO_DATE('20,SEPTEMBER,2008')
20-SEP-08

19. Display the day truncated up to the year for the DOT in the Flight table.

```
select trunc(To_date (dot,'DD-MM-YYYY'),'YEAR') from flight
```

TRUNC(TO_DATE(DOT,'DD-MM-YYYY'),'YEAR')
01-JAN-21
01-JAN-21
01-JAN-21
01-JAN-21
01-JAN-21

LAB MANUAL

20. Take DOT as input and display the output like “my travel Date is ”.

```
select concat('my travel date is ' , DOT) from flight
```

CONCAT('MYTRAVELDATEIS',DOT)
my travel date is 01-Jan-2021
my travel date is 01-Feb-2021
my travel date is 01-Mar-2021
my travel date is 01-Apr-2021
my travel date is 01-May-2021

Conclusion:
With the help of this experiment, we learned character, numeric and date & time functions. These built-in functions help us in retrieving data easily as per the requirement.