**Date Functions (ALL Oracle Proprietary functions)**

To get today’s date use the **sysdate** function.

select **sysdate** from dual;

To get only current time use **to\_char** function and specify the format.

select **to\_char**(sysdate, 'hh:mi:ss') from dual;

To get today’s date and time together use **to\_char** function and specify the format.

select to\_char(sysdate, 'DD-MON-YYYY hh:mi:ss') from dual;

To show either AM or PM after the time use the format as PM

select to\_char(sysdate,'hh:mi:ss **PM**') from dual;

OR

select to\_char(sysdate,'hh**24**:mi:ss') from dual;

To get today’s date, time **and timezone** use the **systimestamp** function.

select **systimestamp** from dual;

Calculations with date:

select sysdate + 3 from dual;

----------------------------------------------------

select **'07-dec-04' + 3** from dual; -- Gives error

Use the **to\_date** function to convert the string into date.

select **to\_date('**07-dec-04') + 3 from dual;

select to\_date('07-dec-04') - to\_date('03-dec-04') from dual;

select to\_date('07-dec-04') - to\_date('13-dec-04') from dual; -- Negative value

---------------------------------------------------

**Months\_between(d1,d2)**

If d1 > d2 then +ve else -ve

select months\_between(sysdate, hiredate) from emp

select round(months\_between(sysdate,hiredate),0) As "Months Between" from emp

Assignment - Find the years between using months\_between

select months\_between('13-dec-04','24-jul-04') from dual

select round(months\_between('13-dec-04','24-jul-04'),0) from dual

**Add\_months(date,n)** [n can be +ve or -ve]

select add\_months('4-dec-04',3) from dual; Gives 04-Mar-05

select add\_months('4-dec-04',-3) from dual; Gives 04-Sep-04

-----------------------------------------------------------------------------------------------

**Next\_Day(date,'char')** -- Gives the date of the next week for the specified day in the char parameter.

select next\_day('04-Dec-04','Wed') from dual; Gives the date coming on next Wednesday after 4-Dec-04. The answer is *08-Dec-04*

OR

Instead of character parameter a numeric value can be also given.

It has Sunday as 1 Monday as 2 ...Saturday as 7

select next\_day('04-Dec-04', 4) from dual. The answer is *08-Dec-04*

Note: The number cannot be negative.

----------------------------------------------------------------------

**Last\_Day(date)** - Returns the last date of the month.

select last\_day(sysdate) from dual;

select last\_day('23-Feb-2024') from dual;

--------------------------------------------------------------------------

**Round (Date, fmt)**

If month is the fmt, then rounds the date to the months extremities. If date is between 1 and 15 then it rounds the date to the first of the same month. If it is 16 to end then rounds to first date of the next date.

E.g.

select round(to\_date('04-dec-04'),'Month') from dual; ----> 01-DEC-04

select round(to\_date('15-dec-04'),'Month') from dual; ----> 01-DEC-04

select round(to\_date('16-dec-04'),'Month') from dual; ----> 01-JAN-05

select round(to\_date('02-Feb-04'), 'Year') from dual; --> 01-JAN-04

select round(to\_date('30-Jun-04'), 'Year') from dual; --> 01-JAN-04

select round(to\_date('1-Jul-04'), 'Year') from dual; --> 01-JAN-05

**select round(to\_date('17-Jun-24'),'Day') from dual; -- Sunday 16-Jun-2024**

**select round(to\_date('20-Jun-24'),'Day') from dual; -- Sunday 23-Jun-2024**

**Trunc (Date, fmt)**

select trunc(to\_date('23-dec-04'),'Month') from dual; Gives 01-Dec-04

select trunc(to\_date('07-Dec-04'),'Month') from dual; Gives 01-Dec-04

select trunc(to\_date('23-Dec-04'),'Year') from dual; Gives 01-Jan-04

select trunc(to\_date('02-Feb-04'),'Year') from dual; Gives 01-Jan-04

**select trunc(to\_date('17-Jun-24'),'Day') from dual; -- Sunday 16-Jun-2024**

**select trunc(to\_date('20-Jun-24'),'Day') from dual; -- Still Sunday 16-Jun-2024**

Use the **To\_Char** function in where clause while filtering date on the parts of date like dd or mm or yy.

To see all the records of employees joined after 1982

select \* from emp

where to\_char(hiredate,'yy') >= 82;

Or where to\_char(hiredate,'yyyy’) >= 1982

To see all the records of employees who have joined from October of any month.

select \* from emp

where to\_char(hiredate, 'mm') >= 10;

(Note 🡪 Only number 1 to 12 can be used for comparison)

select ename, hiredate, to\_char(hiredate, 'Month') from emp

To see all the records of employees who have joined from 21st date.

select \* from emp

where to\_char(hiredate, 'dd') >= 21;

To see all the records of employees who have joined from April-81

select \* from emp

where to\_char(hiredate, 'mm') > 04 and to\_char(hiredate, 'yy') >= 81

To see all the records of employees who have joined after 28-Sep-81

select \* from emp

where hiredate > '28-Sep-81'

**DATE FORMAT MODEL**

# YEARS

select to\_char(hiredate,'YYYY') || '-' || to\_char(hiredate,'YEAR') FROM EMP;

# MONTHS

select to\_char(hiredate,'MONTH') || '-' || to\_char(hiredate,'MM')

|| '-' || TO\_CHAR(hiredate,'MON') FROM EMP;

# DAYS

select to\_char(hiredate,'DAY') || '-' || to\_char(hiredate,'DD')

|| '-' || TO\_CHAR(hiredate,'DY') FROM EMP;

* For the **D** format code, 1-7 are valid values.
* For the **DAY** format code, Sunday-Saturday are valid values.
* For the **DD** format code, 1-31 are valid values.
* For the **DDD** format code, 1-366 are valid values.

For the **DY** format code, Sun-Sat are valid values.

ADVANCED FORMATS –

# CENTURY

SELECT TO\_CHAR(SYSDATE, '**SCC**') FROM DUAL;

**SYEAR** The spelled out year.

select to\_char(HIREDATE,**'SYEAR**') "YEAR"

FROM EMP

**Q**  To get the quarter of the year (1,2,3 and 4)

select ename, hiredate, to\_char(HIREDATE,'Q') "Quarter"

FROM EMP;

# Suffixes –

select to\_char(sysdate, **'ddth'**) from dual;

select to\_char(sysdate, **'ddsp'**) from dual;

select to\_char(sysdate, **'ddspth'**) from dual;

**Practical Usage of to\_date function:**

**Date values** are in **Varchar data type** and that too in yyyy-mm-dd.

The below example shows still how to compare the date values!!

create table admissions

(rollno int,

**adm\_date varchar(20)**

);

Insert into admissions (rollno, adm\_date) Values(1, '2023-01-01');

Insert into admissions (rollno, adm\_date) Values(2, '2023-01-02');

Insert into admissions (rollno, adm\_date) Values(3, '2023-01-03');

Insert into admissions (rollno, adm\_date) Values(4, '2023-01-04');

commit;

We want to see all the admissions done after 2-Jan-2024.

Select rollno, adm\_date

from admissions

where **to\_char(to\_date(adm\_date,'yyyy-mm-dd'),'dd-mm-yy')** > '02-01-24';

**Filtering Only Date Values From Date And Time Values:**

create table visitors\_2023

(v\_id int,

v\_name varchar(40),

v\_date date);

Insert Into visitors\_2023 (v\_id, v\_name, v\_date)

Values(1,'James', **to\_date('25-DEC-2023 10:45:12', 'DD-MON-YYYY HH24:MI:SS')**);

Insert Into visitors\_2023 (v\_id, v\_name, v\_date)

Values(2,'Smith', to\_date('26-DEC-2023 18:27:23', 'DD-MON-YYYY HH24:MI:SS'));

Insert Into visitors\_2023 (v\_id, v\_name, v\_date)

Values(3,'Roger', to\_date('26-DEC-2023 20:08:38', 'DD-MON-YYYY HH24:MI:SS'));

commit;

-- Display all visitors who came on 26-DEC-2023, irrespective of their entry time:

Select v\_name

from visitors\_2023

where v\_date = '26-DEC-2023'; -- No rows shown!!!!!

-- Use the trunc function to ignore the time!!

Select v\_name

from visitors\_2023

where **trunc**(v\_date) = '26-DEC-2023';

-- To see visitor names who came before 7:00 pm on 26-Dec-2023

Select v\_name

from visitors\_2023

where to\_char(v\_date, 'HH24:MI:SS') <= '19:00:00'

and to\_char(v\_date, 'DD-MON-YYYY') = '26-DEC-2023';

**Filtering Today's Details:**

create table trainings

(trid int,

technology varchar(50),

training\_date date);

Insert into trainings Values(1,'Azure', '2-May-2024');

Insert into trainings Values(2,'SQL', '8-May-2024');

Insert into trainings Values(3,'Power BI', Sysdate);

Commit;

Select \* from trainings;

-- Show today's training details:

Select \* from trainings

where training\_date = Sysdate; -- No output................

Select \* from trainings

where **to\_char(training\_date, 'DD-MON-YYYY')**

**=**

**to\_char(Sysdate, 'DD-MON-YYYY');**