**Built in function**

**Ex.no: 2**

**Aim:**

To perform some of the built in function using oracle.

**SQL Commands:**

**DATE FUNCTION**

**1. Add\_month**

This function returns a date after adding a specified date with specified number of months. **Syntax:** Add\_months(d,n); where d-date n-number of months

**Example:** Select add\_months(sysdate,2) from dual;

**2. last\_day :** It displays the last date of that month.

**Syntax:** last\_day (d); where d-date

**Example:** Select last\_day („1-jun-2009‟) from dual;

**3. Months\_between :**It gives the difference in number of months between d1 & d2.

**Syntax:** month\_between (d1,d2); where d1 & d2 -dates

**Example:** Select month\_between („1-jun-2009‟,‟1-aug-2009‟) from dual;

**4. next\_day**

It returns a day followed the specified date.

**Syntax**: next\_day (d,day);

**Example:** Select next\_day (sysdate,‟wednesday‟) from dual

**5. round**

This function returns the date, which is rounded to the unit specified by the format model.

**Syntax :** round (d,[fmt]);

where d- date, [fmt] – optional. By default date will be rounded to the nearest day

**Example:** Select round (to\_date(„1-jun-2009‟,‟dd-mm-yy‟),‟year‟) from dual;

Select round („1-jun-2009‟,‟year‟) from dual;

**NUMERICAL FUNCTIONS**

**Command Query Output**

Abs(n) Select abs(-15) from dual; 15

Ceil(n) Select ceil(55.67) from dual; 56

Exp(n) Select exp(4) from dual; 54.59

Floor(n) Select floor(100.2) from dual; 100

Power(m,n) Select power(4,2) from dual; 16

Mod(m,n) Select mod(10,3) from dual; 1

Round(m,n) Select round(100.256,2) from dual; 100.26

Trunc(m,n) Select trunc(100.256,2) from dual; 100.23

Sqrt(m,n) Select sqrt(16) from dual; 4

**CHARACTER FUNCTIONS**

**Command Query Output**

initcap(char); select initcap(“hello”) from dual; Hello

lower (char); select lower (“HELLO‟) from dual; hello

upper (char); select upper („hello‟) from dual; HELLO

ltrim (char,[set]); select ltrim (“cseit‟,”cse‟) from dual; it

rtrim (char,[set]); select rtrim (“cseit‟, “it‟) from dual; cse

replace (char,search) select replace(„jack and jue‟,„j‟,„bl‟) from dual; black and

string, replace string); blue

substr (char,m,n); select substr („information‟, 3, 4) from dual; Form

**CONVERSION FUNCTION**

**1. to\_char()**

**Syntax**: to\_char(d,[format]);

This function converts date to a value of varchar type in a form specified by date format. If

format is negelected then it converts date to varchar2 in the default date format.

**Example**: select to\_char (sysdate, ‟dd-mm-yy‟) from dual;

**2. to\_date()**

**Syntax:** to\_date(d,[format]);

This function converts character to date data format specified in the form character.

**Example:** select to\_date(„aug 15 2009‟,‟mm-dd-yy‟) from dual;

**Miscellaneous Functions**

**1. uid** – This function returns the integer value (id) corresponding to the user currently

logged in.

**Example:** select uid from dual;

**2. user** – This function returns the logins user name.

**Example:** select user from dual;

**3. nvl** – The null value function is mainly used in the case where we want to consider null values

as zero.

**Syntax;** nvl(exp1, exp2)

If exp1 is null, return exp2. If exp1 is not null, return exp1.

**Example:** select custid, shipdate, nvl(total,0) from order;

**4. vsize:** It returns the number of bytes in expression.

**Example:** select vsize(„tech‟) from dual;

**GROUP FUNCTIONS**

A group function returns a result based on group of rows.

**1. avg - Example:** select avg (total) from student;

**2. max - Example**: select max (percentagel) from student;

**2.min - Example:** select min (marksl) from student;

**4. sum - Example:** select sum(price) from product;

**COUNT FUNCTION**

In order to count the number of rows, count function is used.

**1. count(\*)** – It counts all, inclusive of duplicates and nulls.

**Example:** select count(\*) from student;

**2. count(col\_name)–** It avoids null value.

**Example**: select count(total) from order;

**3. count(distinct col\_name)** – It avoids the repeated and null values.

**Example:** select count(distinct ordid) from order;

**GROUP BY CLAUSE**

This allows us to use simultaneous column name and group functions.

**Example:** Select max(percentage), deptname from student group by deptname;

**HAVING CLAUSE**

This is used to specify conditions on rows retrieved by using group by clause.

**Example:** Select max(percentage), deptname from student group by deptname having count(\*)>=50;

**SPECIAL OPERATORS:**

**In / not in** – used to select a equi from a specific set of values

**Any -** used to compare with a specific set of values

**Between / not between** – used to find between the ranges

**Like / not like –** used to do the pattern matching

**d) Queries:**

**Q1**: **Display all the details of the records whose employee name starts with ‗A‘.**

Solution:

1. Use SELECT FROM WHERE syntax. 2. select should include all in the given format.

3. from should include employee 4. where should include condition on empname like „A%‟.

**Ans:**

SQL> select \* from emp where ename like 'A%';

EMPNO ENAME JOB DEPTNO SAL

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

2Arjun ASP 2 15000

5Akalya AP 1 10000

**Q2: Display all the details of the records whose employee name does not starts with ‗A‘.**

**Ans:**

SQL> select \* from emp where ename not like 'A%';

EMPNO ENAME JOB DEPTNO SAL

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

1 Mathi AP 1 10000

3 Gugan ASP 1 15000

4 Karthik Prof 2 30000

**Q3: Display the rows whose salary ranges from 15000 to 30000.**

**Ans:**

SQL> select \* from emp where sal between 15000 and 30000;

EMPNO ENAME JOB DEPTNO SAL

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

2 Arjun ASP 2 15000

3 Gugan ASP 1 15000

4 Karthik Prof 2 30000

**Q4: Calculate the total and average salary amount of the emp table.**

**Ans:**

SQL> select sum(sal),avg(sal) from emp;

SUM(SAL) AVG(SAL)

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

80000 16000

Q5: **Count the total records in the emp table.**

**Ans:**

SQL>select \* from emp;

EMPNO ENAME JOB DEPTNO SAL

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

1 Mathi AP 1 10000

2 Arjun ASP 2 15000

3 Gugan ASP 1 15000

4 Karthik Prof 2 30000

5 Akalya AP 1 10000

SQL> select count(\*) from emp;

COUNT(\*)

---------

5

**Q6: Determine the max and min salary and rename the column as max\_salary and min\_salary.**

Solution:

1. Use the MIN & MAX aggregate function in select clause.

2. Rename the column as min\_sal & max\_sal.

**Ans:**

SQL> select max(sal) as max\_salary, min(sal) as min\_salary from

emp; MAX\_SALARY MIN\_SALARY

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

30000 10000

**Q7: Display the month between ―1-jun-10ǁand 1-aug-10 in full.**

**Ans:**

SQL>Select month\_between („1-jun-2010‟,‟1-aug-2010‟) from dual;

**Q8: Display the last day of that month in ―05-Oct-09ǁ.**

**Ans:**

SQL> Select last\_day ('1-jun-2009') from dual;

LAST\_DAY(

---------

30-JUN-09

**Q9: Find how many job titles are available in employee table.**

Solution:

1. Use select from clause.

2. Use count function to get the result.

**Ans:**

SQL> select count(job) from

emp; COUNT(JOB)

----------

4

SQL> select count(distinct job) from

emp; COUNT(DISTINCTJOB)

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

2

**Q10: What is the difference between maximum and minimum salaries of employees in the**

**organization?**

Solution:

1. Use select from clause.

2. Use function max(),min() and find the difference between them to get the result.

**Ans:**

SQL> select max(sal), min(sal) from

emp; MAX(SAL) MIN(SAL)

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

20000 10000