FUNCTIONS

→ Functions is a self contained block of statements, which perform some specific task.

→ The main advantage of the function is :-

① code Reusablity

* means writing the code only one time and using the same code again and again whenever we perform the same operation.

→ Functions always returns only one value.

→ Functions are divided into two types :-

① System defined / Builtin Functions
② user defined functions.

→ Function which is defined by the system along with the software is known as System defined / Builtin ~~user defined~~ functions

Scanned with CamScanner

→ Function defined by the user or programmer is known as user defined functions.

TABLES :-

→ It can be defined as collection of rows and columns. Tables are divided into two types :-

(i) DataBase Tables
(ii) Non DataBase Tables.

Database Tables :-

→ Data Base table is a table which contains the actual information.

Non Database Table :-

→ Non DataBase table is a table which does not contain any information.

Eg :- DUMMY / DUAL

→ Database tables tables are further divided into two types :-

Scanned with CamScanner

(i) System defined Tables
(ii) Data Dictionary Tables / View

System defined tables :-

→ A table which is defined by the system along with the oracle software is known as system defined tables.

→ on these tables developer cannot perform any kind of operations, such as update, delete and insert except select. These tables can also be called as read only tables. ~~Eg :- DUAL table~~

Data Dictionary table / View :-

→ one of the most important parts of Database is its data dictionary, which is read only set of tables that provides information about database.

a data dictonary tables contains :-

1. Integrity constraints information
2. the names of oracle users.
3. Privileges and roles each user has been granted.

Scanned with CamScanner

4) Auditing information, such as who has accessed or updated various Schema objects.

5) other general database Information

→ the data dictionary is structured in tables and views, just like other database data. All the data dictionary tables and views for given database are stored in that database's SYSTEM table space.

Eg:- Select * from tab;

Select * from cat;

Select * from recycle bin;

Select * from user_unused_col_tabs;

Select * from all_unused_col_tabs;

Select * from dba_unused_col_tabs;

Scanned with CamScanner

Non Data Base Table has 3 advantages :-

① It evaluates mathematical expression.

Eg :- Select 2+3+5/2 from dual;
Select (2+3+5)/2 from dual;

③ It evaluates pseudo column value.

Eg :- Select Sysdate from dual
→ Select current_date from dual;

③ It evaluates the predefined function

Select length('Sathya') from dual;

Eg :- DUAL

DUAL
→ system defined tables
→ it has one column
↳ varchar2(1)

Scanned with CamScanner

System define or predefine functions

SDF / PDF

- Single Row Functions
  1. STRING / CHAR
  2. MATH / NUMERIC
  3. DATE
  4. DATA CONVERSION
- multiple Row functions / aggregate
  1. SUM()
  2. AVG()
  3. MAX()
  4. MIN()
  5. COUNT()
     - COUNT(*)
     - COUNT(Expression)
     - COUNT (Distinct expression)
- Miscellaneous
  1. NVL()
  2. NVL2()
  3. NULLIF()
  4. COALESCE()

Scanned with CamScanner

Single Row Functions :-

→ They are those functions will work on each and every record on the table and returns more than one value.

→ These functions are broadly divided into following types :-

1) String/Char
2) Math/Numeric
3) Date
4) Data conversion

String functions :-

1) LENGTH() :-

This function ~~will work on each and every record in the table~~ counts the number of characters in the given string including spaces.

Eg :- Select length ('sathya technologies') from dual;

Ans :- (18)

select ename, length (ename) from emp;

Scanned with CamScanner

(ii) Chr :-

This function returns the high level language character from ASCII value.

Eg:- Select Chr(97) from dual;

Ans: (a)

(iii) ASCII( ) :-

This function returns ascii value from high level language character.

Eg:- Select ascii ('a') from dual;

Ans: (97)

(iv) initcap( ) :-

This function converts the first character of every word into capital letter.

Eg:- Select initcap ('sathya technology') from dual

Sathya Technology

Scanned with CamScanner

(V) upper() :-

This function converts the String into capital letter.

Eg:- Select upper ('Sathya Technology') from dual;

SATHYA TECHNOLOGY

(VI) lower() :-

This function converts the String into small letters.

Eg:- Select lower ('SATHYA TECHNOLOGY') from dual;

Sathya technology.

(VII) Reverse() :-

This functions reverses the given String:-

Eg:- select reverse('Sathya') from dual;

ayhtas.

(VIII) concat() :- This function combines two strings.

Eg:-

Select concat('satya', 'Tech') from dual;

satya techa

Scanned with CamScanner

(IX) LTrim() :-

This function removes the blank spaces or characters from the left hand side of the string.

Eg:-

Select LTRIM(' SATYA') from dual;

SATYA

Select LTRIM('SATYA', 'SA') from dual;

TYA

→ ~~[illegible]~~ Both spaces & letters cannot be deleted at a time.

(X) RTRIM() :-

This function also works like LTRIM but removes from right hand side of the string.

Eg:-

Select RTRIM('SATYA TECH ') from dual;

Select RTRIM('SATYA TECH', 'H') from dual;

(XI) TRIM() :- This function removes the blank spaces from both sides of the string. But this function cannot be used to remove characters

Select Trim(' SATYA ') from dual;

(XII) Translate() :-

→ This function replaces the given string character with another character.

Eg:- Select Translate('JACK AND JUE','J','BL') from dual;

→ It replaces only one character.

Ans: BACK AND BUE

(XIII) Replace() :-

→ This function also works like translate but it will replace with more than one character.

Eg:- Select Replace('JACK AND JUE', 'J','BL'). from dual;

Ans: Black And Blue.

(XIV) Vsize() :-

→ This function counts the number of bytes occupied by the given string.

Select Vsize('Satya Technologies') from dual;

Ans = (18)

Scanned with CamScanner

(XV) Soundex ( ) :-

→ This function returns the name whose pronouncation is same but spell differently

Eg:- Select * from Names where

table name

Sondex (Name) = Soundex ('MURTY')

column name

Ans:- murty
moorti
murthy

(XVI) Lpad( ):- This function combines the given character to the string from left side.

Eg:- Select Lpad ('Satya Tech', 15, '*') from dual;

Ans:- *****Satya Tech

(XVII) Rpad( ):- This function also works like Lpad but it is combining from right hand side of the string.

Eg:- Select Rpad ('Satya Tech', 15, '*') from dual.

Satya Tech *****;

Scanned with CamScanner

Eg:- If we want to give characters left hand side & right hand side both.

Select LPAD(RPAD('SATYA TECH', 15, '*'), 20, '*') from dual;

***** SATYA TECH *****

XVIII) SUBSTR():- This function returns the path of the string from main string.

Eg:-

select substr('SATYA TECHNOLOGIES', 7) from dual;

Technologies.

select substr('Satya Technologies', 7, 6) from dual;

Techno

XIX) Instr() :- This function ~~[illegible]~~ returns the position of the character in the given string.

Scanned with CamScanner

from where we have to start searching

Select Instr('satya Technologies', 'T', 1, 1) from dual;

which T we have to find position.

3

Select Instr('satya Technologies', 'T', 1, 2) from dual

7

Select Instr('satya Technologies', 'T', 2, 1) from dual

3

Select Instr('satya Technologies', 'T', 3, 1) from dual;

3

Select Instr('satya Technologies', 'T', 4, 1) from dual;

7

Select Instr('satya Technologies', 'T', 2, 3) from dual;

0

Scanned with CamScanner

(XX) Regexp_count() :-

This function counts the given word repeated how many number of times in the given statement/string.

Eg:- select Regexp_count('oracle is a Database, oracle is a Company', 'ora', 1) from dual;

'ora' → It is case sensitive.

1 → from where you have to start searching.

(2)

select Regexp_count('oracle is a database, oracle is a company', 'ora', 1, 'i') from dual;

'i' → If you write i then case sensitivity wouldn't matter.

(2)

→ In above example i represents irrespective of case sensitivity of the word 'ora'.

Scanned with CamScanner

(2) MATH / NUMERIC :

(i) **abs** :- This function converts the negative number into positive.

select abs (-98) from dual;

Eg :- Abs (-98) ABS (-9.8)

(98) (9.8)

(ii) **Round** :- This function rounds the given number to the nearest integer.

select round (9.6) from dual;

(10)

select round (9.5) from dual;

(10)

select round (9.4) from dual;

(9)

(iii) **floor** :- This function rounds the given number to the least integer.

Eg :- select floor (9.999) from dual;

(9)

(iv) **ceil** :- This function rounds the given number to the highest integer.

Eg :- select ceil (9.00001) from dual;

(10)

Scanned with CamScanner

(V) **mod** :- This functions returns the remainder value.

Eg :- Select mod (10,2) from dual;

(0)

(VI) **Power** :- This function returns the power value of given number

Eg :- select power (10,2) from dual;

(100)

(VII) **Sqrt** :- This function returns the square root of given number.

Eg :- select sqrt (100) from dual;

(10)

(VIII) **Sigh** :- This function returns the sigh value of the given number.

Eg :- select sigh (100) from dual;

(1)

Select sigh (0) from dual;

(0)

Select sigh (-100) from dual;

(-1)

→ It returns three value +1, 0, -1

+1 → when given number is +ve

0 → when the given number is 0

-1 → when given number is -ve.

Scanned with CamScanner

(IX) greatest :- This function returns the highest value among given numbers.

Eg :- Select greatest (5, 6, 0, -9, 9.5) from dual;

9.5

(X) least :- This function returns the least value among given numbers.

Select least (5, 6, 0, -9, 9.5) from dual

-9

(XI) log :- This function returns the logarithm value.

Select log(10, 10) from dual;

1 → base

Select log (10, 100) from dual;

2

Select log (100, 10) from dual;

0.5

(XII) ln :- This function returns the natural logarithmic value of given number.

Eg :- Select ln(10) from dual;

2.3025

Scanned with CamScanner

(XIII) Trignometric functions :-

→ This function returns the trignometric values of given number.

select sin(0) from dual;

(0)

Select cos(0) from dual;

(1)

select tan(0) from dual;

(0)

3. DATE functions :-

→ These functions are broadly divided into following types :-

(i) Add_months ()
(ii) months_between()
(iii) last_day()
(iv) next_day()
(v) extract()

Scanned with CamScanner

(i) Add_months :- This function adds the number of months between two dates.

Eg :- Select add_months (sysdate, 12) from dual;

13-march-18

(ii) months_between :-
→ This function counts the number of months between two dates.

Eg :- Select monts_between (sysdate, '13-march-15') from dual;

12

(iii) Last_day() :- This function returns the last day of the given month.

Select last_day (sysdate) from dual;

31-march-17

Scanned with CamScanner

(iv) Next_day :- This function returns the next date from given date.

Select next_day(sysdate, 'sunday') from dual;

(19-march-17)

(v) extract( ) :-

→ This function extracts date, month, year from the given date.

Select extract(day from sysdate) from dual;

(13)

select extract(month from sysdate) from dual;

(march)

select extract(year from sysdate) from dual;

(17)

Scanned with CamScanner

MULTIPLE ROW FUNCTIONS

(i) sum() :- This function finds the sum of all the values of a particular column in the table.

Eg :- Select sum(sal) from emp;

29375

(ii) avg() :- This function returns the average value of a particular column in the table.

Eg :- Select avg(sal) from emp;

2098

(iii) max() :- This function returns the maximum value of a particular column in a table.

Eg :- Select max(sal) from emp;

5000

Scanned with CamScanner

(iv) min() :- This function returns the min value of a particular column in a table.

Eg :- Select min(sal) from emp;

(800)

(v) count (*) :-

→ It counts the number of records in the given table including dublicate and null values.

Select count(*) from emp;

(14)

(vi) count(expression) :-

→ This function is also used to count the number of records in the given table including ~~both dublicate & null values~~ dublicate values excluding null values.

Eg :- Select count (mgr) from emp;

(13)

Scanned with CamScanner

(VII) count(distinct expression) :-

This function is also used to count number of records in the table excluding dublicate and nulls.

Eg :- Select count (distinct mgr) from emp;

(6)

~~Selec~~

If we have to count number of rows / records then we apply above counts commands now,

Syntax to count number of columns in the table :-

Select count(*) from user_tab_cols where table_name = 'EMP';

Scanned with CamScanner

MISCLANEOUS FUNCTIONS

→ These function will work on null values:-

(i) NVL()
(ii) NVL2()
(iii) Null if()
(iv) coalesce()

(i) NVL() :- This function converts null value into ~~[illegible]~~ ~~[illegible]~~. Non null value.

nvl(a, b)
a represents column name
b represents replaced value.

→ If column contains null then ~~replace~~ it will replace with b value.

Eg:-
Select emp.*, sal + nvl(comm, 0) totsal
from emp;

Scanned with CamScanner

(ii) **NVL2() :-**

This function is also used to convert null value into the ~~[illegible]~~ non null value.

NVL2(a, b, c)

→ a represents column name
→ b represents replaced value whose commission is not null.
→ c represents replaced value whose commission is null.

→ If column contains null replaced with c value.
→ If column does not contain null than replace with b value

Eg.-
→ select emp.*, sal+nvl2(comm, 1500, 1000) totsal from emp;

Eg:-
select emp.*, sal+nvl2(com, comm+1500, 1000) totsal from emp;

Scanned with CamScanner

Nullif (a,b) :-

→ This function is used to convert non null value into null value.

nullif (a,b)

If a and b are equal
then the result is null

If a ≠ b
then result is a value

Eg :-

```
Select ename, length(ename), job, length(job),
nullif(length(ename), length(job))
from emp;
```

(IV) coalesce :-

This function returns first occurence of non null value.

Eg :-

```
Select coalesce(null, 852, null, 0.9, null,
                8563, -9)
from dual;
```

Ans = 852

Scanned with CamScanner

Eg :-

select * from phones 1

wPhone	oPhone	hPhone
86533	86252	86531
96333	46232	
		99992

Eg :-

→ select coalesce (wPhone, oPhone, hPhone) from phones 1;

Ans :-

86533
96333
9992

GROUP-BY Clause :-

→ A group by statement specifies that an sql select statement returns a list that is grouped by one or more columns usually in order to apply some sort of aggregate function to certain column.

→ Group by clause is used to group rows based on distinct values that exist for specified columns.

Scanned with CamScanner

DISTINCT clause :-

→ If we want to select unique value from a column or group of column.

Syntax for selecting unique value from a column :-

Select distinct col. from <Table Name>

Syntax for selecting unique value from group of columns :-

Select Distinct (col 1, col 2 .. col n) from <Table Name>

Eg :- Select desig from Emp ;

Select ename from Emp ;

Select distinct desig from emp ;

Select distinct desig from emp ;

→ no, duplicate values will be there

from this if there are two duplicate values it will show that. But

customer

cust id	cust name	city	Gender
1	abc	chennai	M
2	def	Banglore	M
3	ghi	Banglore	M
4	jkl	Hyderabad	F

Select distinct city from customer;

Chennai
Banglore
Hyderabad

Eg :- from multiple columns

~~Select Distinct (cust id)~~

Department

d id	d name	loc
d1	Production	Hyd
d1	"	"
d1	"	"
d2	Sales	Bangl
d2	"	"

Select Distinct (d id, dname, loc) from Department

GROUP - BY clause

→ Group by clause is used in collaboration with Select statatement to arrange identical data into groups.

→ It is used with Select query

→ can be used with aggregate functions.

→ It is followed by where clause and precedes by order by clause.

Table tutorial

id	title	subject	duration	upload date
8	while	C	8:13	4-04-13
9	For	C	5:30	"
10	Break	C	7:62	"
11	continue	C	6:56	"
12	union	C	3:21	"
13	Typedy	C	7:86	"
14	Select	SQL	4:56	4-05-13
15	Insert	SQL	5:32	"
16	operator	SQL	6:37	"
17	logical	SQL	5:11	"
18	orderby	SQL	4:19	"

Scanned with CamScanner

Syntax

Select col 1, col 2, col 3 . col n aggregate function from < table - name >
group by < col 1 >, < col 2 >, < col 3 >
. < col n

Eg :-

Select subject, count(*) from tutorial group by subject ;

subject	count(*)
C	8
SQL	5

Select Subject, sum (duration) from tutorial group by subject ;

Subject	duration
C	48 : 17
SQL	25 : 55

Scanned with CamScanner