

Agenda

- DUAL table
- Using HELP
- SQL Functions
 - Numeric Functions
 - String Functions
 - Date and Time Functions
 - Control Flow Functions
 - NULL Functions
 - Misc/Info Functions
 - Group Functions
- ~~GROUP BY clause~~
- ~~HAVING clause~~

Dual table

- It is a virtual table in memory to support SELECT query
- Oracle has a dual table
- ANSI SQL is influenced by oracle and hence it is now added in ANSI Standard
- IN ANSI SQL this table is optional

```
SELECT 2+2;  
SELECT 2+2 FROM DUAL;  
  
SELECT USER();  
SELECT USER() FROM DUAL;  
  
SELECT DATABASE();  
SELECT DATABASE() FROM DUAL;
```

Using HELP

```
HELP SELECT;  
HELP INSERT;  
HELP FUNCTIONS;  
HELP Numeric Functions;  
HELP String Functions;
```

1. Numeric Functions

```
HELP Numeric Functions  
  
HELP POW;
```

```
SELECT POW(2,3);
SELECT POW(4,-2); -- 1/(4^2)

HELP SQRT;
SELECT SQRT(4);
SELECT SQRT(16);

HELP ROUND;

SELECT ROUND(123.45);
SELECT ROUND(123.456,1);
SELECT ROUND(123.456,2);
SELECT ROUND(123.456,-1);
SELECT ROUND(123.456,-2);
SELECT ROUND(165.123,-2);

SELECT ROUND(price,2) FROM books;

-- CEIL -> nearest highest integer
-- FLOOR -> nearest lowest integer
HELP CEILING;
HELP FLOOR;
SELECT CEILING(3.14), FLOOR(3.14);
SELECT CEILING(-3.14), FLOOR(-3.14);
```

2. String Functions

```
HELP string functions;

SELECT LOWER("SUNBEAM");

SELECT LOWER(ename) FROM emp;

SELECT upper("sunbeam");

SELECT upper(subject) FROM books;

SELECT LEFT("sunbeam",2), RIGHT("sunbeam",2);

SELECT LEFT(ename,2), RIGHT(ename,2) FROM emp;

-- display all emps whose initial letter of names falls in the range of B to J
SELECT * FROM emp WHERE LEFT(ename,1) BETWEEN 'B' AND 'J';

HELP SUBSTRING;
SELECT SUBSTRING("sunbeam",3,2);
--pos from left
SELECT SUBSTRING("sunbeam",-3,2);
--pos from right

SELECT CONCAT("Sunbeam"," ", "Infotech");
```

```
SELECT CONCAT(ename,"-",job) FROM emp;

-- display the output as
-- ename is working in dept deptno as job
-- SMITH is working in dept 20 as CLERK
SELECT CONCAT(ename," is working in dept ",deptno," as ",job) FROM emp;

SELECT LENGTH("sunbeam");
SELECT LENGTH(" sunbeam ");
SELECT TRIM(" sunbeam ");
SELECT LENGTH(TRIM(" sunbeam "));

SELECT LPAD("9388",10,'X');
SELECT RPAD("8983",10,'X');

-- display output as
-- ***sunbeam***
SELECT LPAD("sunbeam",10,'*');
SELECT RPAD(LPAD("sunbeam",10,'*'),13,'*');
```

3.Date and Time Functions

```
HELP Date and Time Functions;

SELECT NOW();
SELECT SYSDATE();

-- not to take if understanding is low
SELECT NOW(),SLEEP(5),SYSDATE();

SELECT DATE("2000-01-01 11:10:20");
SELECT TIME("2000-01-01 11:10:20");

SELECT DATE(NOW());
SELECT TIME(NOW());

SELECT DATE_ADD(NOW(),INTERVAL 4 DAY);
SELECT DATE_ADD(NOW(),INTERVAL 1 MONTH);

SELECT DATEDIFF(NOW(),"2023-01-01");
--return no of days

SELECT TIMESTAMPDIFF(MONTH,"2023-01-01",NOW());

-- display the exact experience of emps in terms of year and months
SELECT empno,ename,sal,TIMESTAMPDIFF(YEAR,hire,NOW()) AS years FROM emp;
SELECT empno,ename,sal,TIMESTAMPDIFF(YEAR,hire,NOW())%12 AS months FROM emp;

SELECT empno,ename,sal,TIMESTAMPDIFF(YEAR,hire,NOW()) AS
years,TIMESTAMPDIFF(YEAR,hire,NOW())%12 AS months FROM emp;
```

```
SELECT
DAY(NOW()),MONTH(NOW()),YEAR(NOW()),HOUR(NOW()),MINUTE(NOW()),SECOND(NOW());

-- display all emps hired in 1982.
SELECT * FROM emp WHERE YEAR(hire)=1982;
```

4. Flow Control Functions

```
-- display ename,deptno,deptname(10=ACCOUNTING, 20=RESEARCH, 30=SALES)
SELECT ename,deptno, CASE
WHEN deptno=10 THEN "ACCOUNTING"
WHEN deptno=20 THEN "RESEARCH"
WHEN deptno=30 THEN "SALES"
ELSE "UNKNOWN"
END FROM emp;

--OR

SELECT ename, CASE deptno
WHEN 10 THEN "ACCOUNTING"
WHEN 20 THEN "RESEARCH"
WHEN 30 THEN "SALES"
ELSE "UNKNOWN"
END FROM emp;

-- display ename,sal and category of emp.
-- category -> sal>2500 = RICH , sal<=2500 = POOR
SELECT ename,sal,IF(sal>2500,"RICH","POOR") AS category FROM emp;
```

5. NULL Functions

- NULL is a special value in RDBMS
- It is irrespective of datatype
- NULL is not 0,0.0,'0',"NULL"
- It is used to represent absent/empty value.

```
-- display comm of emp and if comm is null display salary.
SELECT comm FROM emp;
SELECT IF(comm IS NULL,sal,comm) AS comm FROM emp;
SELECT IFNULL(comm,sal) AS comm FROM emp;
-- if arg1=NULL result = arg2 else result=arg1

SELECT ename,sal,NULLIF(sal,3000) FROM emp;
-- makes null if arg1==arg2

SELECT NULLIF(NULL,"Hello");
-- comparison with NULL always return null
```

6. Misc/Info Functions

```
SELECT VERSION();
```

```
SELECT DATABASE();
```

```
SELECT USER();
```