# DECODE FUNCTION

In Oracle/PLSQL, the **decode** function has the functionality of an IF-THEN-ELSE statement.

The syntax for the **decode** function is:

**decode( expression ,**

**search_1 , result_1 ,**

**search_2 , result_2,**

**default**

**)**

*expression* is the value to compare.

*search* is the value that is compared against *expression*.

*result* is the value returned, if *expression* is equal to *search*.

*default* is optional. If no matches are found, the decode will return *default*. If *default* is omitted, then the decode statement will return null (if no matches are found).

```
create table e
(id int,
 name char(20),
 dept int,
 sal float
 )

describe e


Insert into e values(111,'dinesh',10,10000)

insert into e values(112,'shovan',20,15000)

Insert into e values(113,'vidhan',30,13000)
```

```
insert into e values(114,'vinush',10,12000)

insert into e values(115,'sudhir',40,18000)

select * from e

--simple decode function
select name, DECODE (dept,
                                10,'cse',
                                20,'ece',
                                30,'eee',
                                'unknown') from e

select dept,count(dept) from e group by dept

-- decode fn to increment the salary based on the department
select name,sal,dept,DECODE( dept,
                                    10,(sal + (sal*10)/100),
                                          20,(sal + (sal*20)/100)
                                            ) from e

--summing up the salary after increment
select sum(DECODE( dept,
                                    10,(sal + (sal*10)/100),
                                          20,(sal + (sal*20)/100)
                                            )) from e

/* Formatted on 2008/01/18 13:17 (Formatter Plus v4.8.0) */
SELECT   dept, COUNT (dept),
         SUM (DECODE (dept,
                      10, (sal + (sal * 10) / 100),
                      20, (sal + (sal * 20) / 100),
                      30, (sal + (sal * 5) / 100),
                      40, (sal + (sal * 50) / 100)
                     )
             )
    FROM e
GROUP BY dept

select DECODE(dept,
                    10,'cse',
                    20,'ece',
                    30,'eee',
                    40,'admin'
                   ),count(dept),SUM(DECODE( dept,
                                                 10,(sal +
(sal*10)/100),
                                                       20,(sal +
(sal*20)/100),
                                                       30,(sal +
(sal*5)/100),
                                                       40,(sal +
(sal*50)/100)
                                                       )
                                           ) from e GROUP BY dept
```

By Dinesh

```
select * from e

--Inside decode u cannot use other operators for comparisio9n exception
"=" so this query is invalid
select name,sal,DECODE (sal,
                                <=12000,'A0',
                                    >15000,'A1',
                                 ) from e

--DECODE does not range checking abilities. Some other features are
supported by DECODE , it can be
-- combined with other functions like SIGN(), TRUNC(), GREATESt(),
LEAST()

select * from e

select max(sal) from e

-- first chech cond. sal<10000  & then checks exp1 then exp2 etc...
/* Formatted on 2008/01/18 13:17 (Formatter Plus v4.8.0) */
SELECT NAME, sal,
       DECODE (GREATEST (sal, 10000),
               10000, sal + 5000,
               15000, sal + 1000,
               13000, sal + 500,
               12000, sal + 1500,
               sal + 0
              )
  FROM e

/* Formatted on 2008/01/18 13:17 (Formatter Plus v4.8.0) */
SELECT NAME, sal,
       DECODE (LEAST (sal, 10000),
               10000, sal + 5000,
               15000, sal + 1000,
               13000, sal + 500,
               12000, sal + 1500,
               sal + 0
              )
  FROM e

 /* Formatted on 2008/01/18 13:17 (Formatter Plus v4.8.0) */
SELECT NAME,
       DECODE (GREATEST (sal, 100),
               1000, DECODE (NAME,
                             'dinesh', (sal + 2000),
                             'shovan', (sal + 1000)
                            ),
               sal
              )
  FROM e
```

By Dinesh