

SQL> SELECT ABS(52.463) FROM DUAL;

ABS(52.463)

-----------

52.463

SQL> SELECT ABS(-52.463) FROM DUAL;

ABS(-52.463)

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

52.463

SQL> -- JUST CONVERTS NEGATIVE VALUE INTO POSITIVE VAlue and display the result.

SQL> select mod(10, 2) from dual;

MOD(10,2)

----------

0

SQL> select mod(10, 3) from dual;

MOD(10,3)

----------

1

SQL> select mod(10, 7) from dual;

MOD(10,7)

----------

3

SQL> --mod gives remainder value

SQL> select power(10, 2) from dual;

POWER(10,2)

-----------

100

SQL> select power(10, 0.5) from dual;

POWER(10,0.5)

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

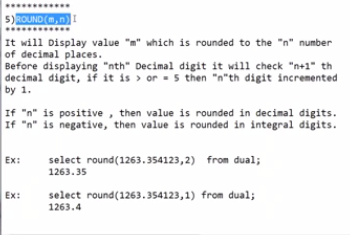
3.16227766

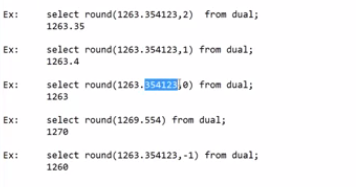
SQL> select power(10, 3) from dual;

POWER(10,3)

-----------

1000





SQL> select round(252.34636, 3) from dual;

ROUND(252.34636,3)

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

252.346

SQL> select round(252.34636, 2) from dual;

ROUND(252.34636,2)

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

252.35

SQL> select round(252.34636, 7) from dual;

ROUND(252.34636,7)

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

252.34636

SQL> select round(-252.34636, 7) from dual;

ROUND(-252.34636,7)

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

-252.34636

SQL> select round(-252.34636, 3) from dual;

ROUND(-252.34636,3)

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

-252.346

Round fun just round off the given float value to the required digits after point.

SQL> select round(252.34636, 2) from dual;

ROUND(252.34636,2)

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

252.35

From the above example, it displays the rounded number acc. To digit . if >5 then increases digit by 1.

SQL> select round(-252.3450016, 3) from dual;

ROUND(-252.3450016,3)

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

-252.345

SQL> select round(-252.3456016, 3) from dual;

ROUND(-252.3456016,3)

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

-252.346

SQL> select round(-252.3454016, 3) from dual;

ROUND(-252.3454016,3)

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

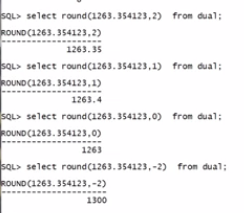
-252.345

SQL> select round(-252.3455016, 3) from dual;

ROUND(-252.3455016,3)

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

-252.346



ROUND(-252.3455016,3)

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

-252.346

SQL> select round(252.3455016, -3) from dual;

ROUND(252.3455016,-3)

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

0

SQL> select round(252.3455016, -2) from dual;

ROUND(252.3455016,-2)

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

300

SQL> select round(252.3455016, -1) from dual;

ROUND(252.3455016,-1)

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

250

SQL> select round(256.3455016, -1) from dual;

ROUND(256.3455016,-1)

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

260

SQL> select round(-256.3455016, -1) from dual;

ROUND(-256.3455016,-1)

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

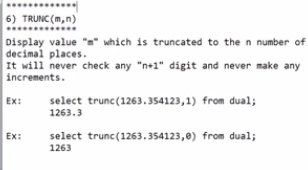
-260

SQL> select round(-256.3455016, -2) from dual;

ROUND(-256.3455016,-2)

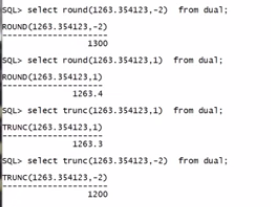
----------------------

-300



TRUNC function just do mechanical caleculation, there is no concept of rounding the given number.

But round fun gives the approximate value according to m, n values.



SQL> select round(-256.3455016, -2) from dual;

ROUND(-256.3455016,-2)

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

-300

SQL> select trunc(-256.3455016, -2) from dual;

TRUNC(-256.3455016,-2)

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

-200

SQL> select trunc(-256.3455016, -3) from dual;

TRUNC(-256.3455016,-3)

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

0

SQL> select trunc(-256.3455016, 3) from dual;

TRUNC(-256.3455016,3)

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

-256.345

SQL> select round(-256.3455016, 3) from dual;

ROUND(-256.3455016,3)

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

-256.346

SQL> select round(-256.3455016, 6) from dual;

ROUND(-256.3455016,6)

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

-256.3455

SQL> select round(256.3455016, 6) from dual;

ROUND(256.3455016,6)

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

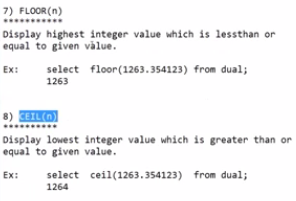
256.345502

SQL> select trunc(256.3455016, 6) from dual;

TRUNC(256.3455016,6)

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

256.345501



SQL> select floor(256.3455016) from dual;

FLOOR(256.3455016)

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

256

SQL> select floor(-256.3455016) from dual;

FLOOR(-256.3455016)

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

-257

SQL> select floor(-256.9455016) from dual;

FLOOR(-256.9455016)

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

-257

SQL> select floor(-256.0455016) from dual;

FLOOR(-256.0455016)

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

-257

SQL> select floor(256.0455016) from dual;

FLOOR(256.0455016)

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

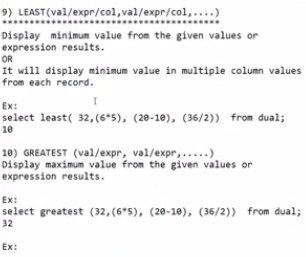
256

SQL> select floor(256.9455016) from dual;

FLOOR(256.9455016)

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

256



Least gives min value among the given numbers list or expressions list.

SQL> select min(2,34,62,5,34,5) from dual;

select min(2,34,62,5,34,5) from dual

\*

ERROR at line 1:

ORA-00909: invalid number of arguments

SQL> select least(2,34,62,5,34,5) from dual;

LEAST(2,34,62,5,34,5)

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

2

SQL> select least(2,34,62,5,-34,5) from dual;

LEAST(2,34,62,5,-34,5)

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

-34

SQL> select least(2\*19,34/2,62,5,-34,5) from dual;

LEAST(2\*19,34/2,62,5,-34,5)

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

-34

SQL> select least(-2\*19,34/2,62,5,-34,5) from dual;

LEAST(-2\*19,34/2,62,5,-34,5)

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

-38

SQL> select greatest(-2\*19,34/2,62,5,-34,5) from dual;

GREATEST(-2\*19,34/2,62,5,-34,5)

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

62

SQL> select greatest(-12\*19,34/2,62,5,-34,5) from dual;

GREATEST(-12\*19,34/2,62,5,-34,5)

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

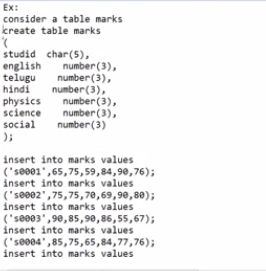
62

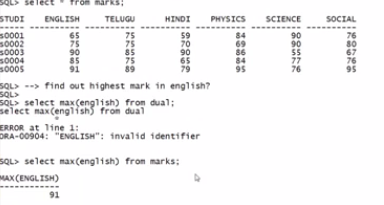
SQL> select greatest(12\*19,34/2,62,5,-34,5) from dual;

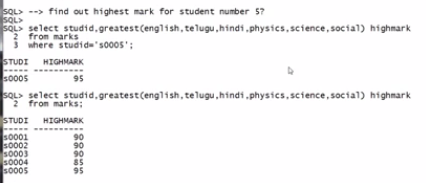
GREATEST(12\*19,34/2,62,5,-34,5)

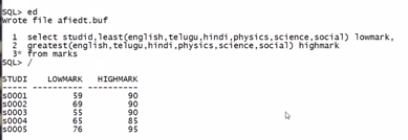
-------------------------------

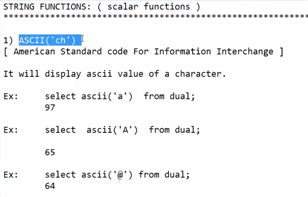
228











SQL> select ascii('a') from dual;

ASCII('A')

----------

97

SQL> select ascii('b') from dual;

ASCII('B')

----------

98

SQL> select ascii('c') from dual;

ASCII('C')

----------

99

SQL> select ascii('d') from dual;

ASCII('D')

----------

100

SQL> select ascii('e') from dual;

ASCII('E')

----------

101

SQL> select ascii('f') from dual;

ASCII('F')

----------

102

SQL> select ascii('g') from dual;

ASCII('G')

----------

103

SQL> select ascii('h') from dual;

ASCII('H')

----------

104

SQL> select ascii('i') from dual;

ASCII('I')

----------

105

SQL> select ascii('j') from dual;

ASCII('J')

----------

106

SQL> select ascii('k') from dual;

ASCII('K')

----------

107

SQL> select ascii('l') from dual;

ASCII('L')

----------

108

SQL> select ascii('m') from dual;

ASCII('M')

----------

109

SQL> select ascii('n') from dual;

ASCII('N')

----------

110

SQL> select ascii('o') from dual;

ASCII('O')

----------

111

SQL> select ascii('p') from dual;

ASCII('P')

----------

112

SQL> select ascii('q') from dual;

ASCII('Q')

----------

113

SQL> select ascii('r') from dual;

ASCII('R')

----------

114

SQL> select ascii('s') from dual;

ASCII('S')

----------

115

SQL> select ascii('t') from dual;

ASCII('T')

----------

116

SQL> select ascii('u') from dual;

ASCII('U')

----------

117

SQL> select ascii('v') from dual;

ASCII('V')

----------

118

SQL> select ascii('w') from dual;

ASCII('W')

----------

119

SQL> select ascii('x') from dual;

ASCII('X')

----------

120

SQL> select ascii('y') from dual;

ASCII('Y')

----------

121

SQL> select ascii('z') from dual;

ASCII('Z')

----------

122

SQL> select sq.currval from dual;

select sq.currval from dual

\*

ERROR at line 1:

ORA-08002: sequence SQ.CURRVAL is not yet defined in this session

SQL> select sq.curval from dual;

select sq.curval from dual

\*

ERROR at line 1:

ORA-00904: "SQ"."CURVAL": invalid identifier

SQL> select sq.nextval from dual;

NEXTVAL

----------

1

SQL> select sq.currval from dual;

CURRVAL

----------

1

SQL> select sq.currval from dual;

CURRVAL

----------

1

SQL> select ascii(sq.currval) from dual;

ASCII(SQ.CURRVAL)

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

49

SQL> select ascii(1) from dual;

ASCII(1)

----------

49

SQL> select ascii(sq.nextval) from dual;

ASCII(SQ.NEXTVAL)

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

50

SQL> select ascii(sq.nextval) from dual;

ASCII(SQ.NEXTVAL)

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

51

SQL> select ascii(sq.nextval) from dual;

ASCII(SQ.NEXTVAL)

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

52

SQL> select ascii(sq.nextval) from dual;

ASCII(SQ.NEXTVAL)

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

53

SQL> select ascii(sq.nextval) from dual;

ASCII(SQ.NEXTVAL)

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

54

SQL> select ascii(sq.nextval) from dual;

ASCII(SQ.NEXTVAL)

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

55

SQL> select ascii(sq.nextval) from dual;

ASCII(SQ.NEXTVAL)

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

56

SQL> select ascii(sq.nextval) from dual;

ASCII(SQ.NEXTVAL)

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

57

SQL> select ascii(sq.nextval) from dual;

ASCII(SQ.NEXTVAL)

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

49

SQL> select ascii(sq.nextval) from dual;

ASCII(SQ.NEXTVAL)

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

49

SQL> select ascii(sq.nextval) from dual;

ASCII(SQ.NEXTVAL)

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

49

SQL> select sq.nextval from dual;

NEXTVAL

----------

13

SQL> select sq.nextval from dual;

NEXTVAL

----------

14

SQL> select ascii(11) from dual;

ASCII(11)

----------

49

SQL> select ascii(12) from dual;

ASCII(12)

----------

49

SQL> select ascii(13) from dual;

ASCII(13)

----------

49

SQL> select ascii('A') from dual;

ASCII('A')

----------

65

SQL> select ascii('B') from dual;

ASCII('B')

----------

66

SQL> select ascii('C') from dual;

ASCII('C')

----------

67

SQL> select ascii('X') from dual;

ASCII('X')

----------

88

SQL> select ascii('Y') from dual;

ASCII('Y')

----------

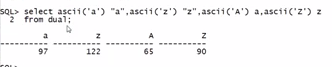
89

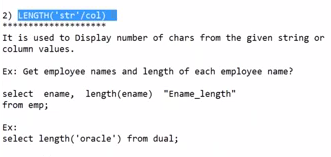
SQL> select ascii('Z') from dual;

ASCII('Z')

----------

90





SQL> SELECT LENGTH('SURESH') FROM DUAL;

LENGTH('SURESH')

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

6

SQL> SELECT LENGTH('I AM SURESH') FROM DUAL;

LENGTH('IAMSURESH')

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

11

SQL> SELECT LENGTH(98765432) FROM DUAL;

LENGTH(98765432)

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

8

SQL> SELECT LENGTH(98765432.74) FROM DUAL;

LENGTH(98765432.74)

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

11

SQL> --GET EMP NAMES AND LENGTH OF EACH NAME

SQL> SELECT ENAME, LENGTH(ENAME) FROM EMP;

ENAME LENGTH(ENAME)

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

KING 4

BLAKE 5

CLARK 5

JONES 5

SCOTT 5

FORD 4

SMITH 5

ALLEN 5

WARD 4

MARTIN 6

TURNER 6

ENAME LENGTH(ENAME)

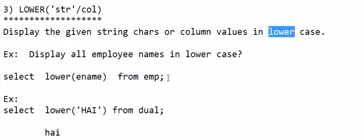
---------- -------------

ADAMS 5

JAMES 5

MILLER 6

14 rows selected.



LOWER(ENAM LENGTH(ENAME)

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

king 4

blake 5

clark 5

jones 5

scott 5

ford 4

smith 5

allen 5

ward 4

martin 6

turner 6

LOWER(ENAM LENGTH(ENAME)

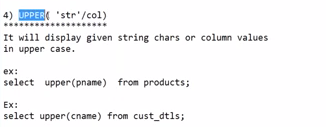
---------- -------------

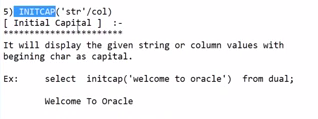
adams 5

james 5

miller 6

14 rows selected.





SQL> SELECT INITCAP(ENAME), LENGTH(ENAME) FROM EMP;

INITCAP(EN LENGTH(ENAME)

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

King 4

Blake 5

Clark 5

Jones 5

Scott 5

Ford 4

Smith 5

Allen 5

Ward 4

Martin 6

Turner 6

INITCAP(EN LENGTH(ENAME)

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

Adams 5

James 5

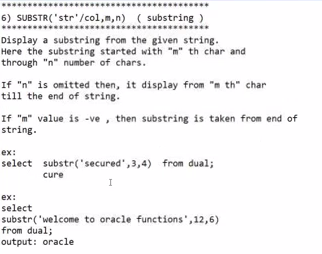
Miller 6

SQL> SELECT INITCAP('my name is suresh') from dual;

INITCAP('MYNAMEIS

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

My Name Is Suresh



M may be positive or negative.

SQL> select substr('i am suresh', 6, 6) from dual;

SUBSTR

------

suresh

SQL> select substr('i am suresh', 6, 3) from dual;

SUB

---

sur

SQL> select substr('i am suresh', 1, 4) from dual;

SUBS

----

i am

SQL> select substr('i am suresh') from dual;

select substr('i am suresh') from dual

\*

ERROR at line 1:

ORA-00938: not enough arguments for function

SQL> select substr('i am suresh',1) from dual;

SUBSTR('IAM

-----------

i am suresh

SQL> select substr('i am suresh',4) from dual;

SUBSTR('

--------

m suresh

SQL> select substr('i am suresh', 4, -6) from dual;

S

-

SQL> select substr('i am suresh', 4, -4) from dual;

S

-

SQL> select substr('i am suresh', -4, 4) from dual;

SUBS

----

resh

SQL> select substr('i am suresh', -6, 4) from dual;

SUBS

----

sure

SQL> select substr('i am suresh', -6, 6) from dual;

SUBSTR

------

Suresh

SQL> select substr('i am suresh', -11, 15) from dual;

SUBSTR('IAM

-----------

i am suresh

SQL> --find length of username for the given mail id

SQL> select substr('ssuresh446@gmail.com', 1, 9) from dual;

SUBSTR('S

---------

ssuresh44

SQL> select substr('ssuresh446@gmail.com', 1, 10) from dual;

SUBSTR('SS

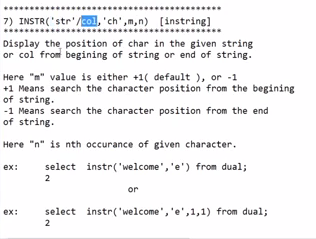
----------

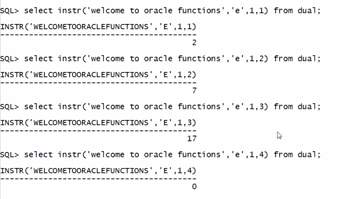
ssuresh446

SQL> -- yeah we are getting username but we dont know weather every mail has same legth or not.

SQL> -- if we know @ position then we can easily find the username , which can be find by using instr.

SQL>









SQL> --select instr('sentence/column/any code', 'char', from which position search should be started, nth occourance);

SQL> select instr('ssuresh446@gmail.com', '@', 1, 1) from dual;

INSTR('SSURESH446@GMAIL.COM','@',1,1)

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

11

SQL> select substr('SSURESH446@GMAIL.COM', (select instr('ssuresh446@gmail.com', '@', 1, 1) from dual)-1, 12)

2 ;

\*

ERROR at line 2:

ORA-00923: FROM keyword not found where expected

SQL> select substr('SSURESH446@GMAIL.COM', (select instr('ssuresh446@gmail.com', '@', 1, 1) from dual)-1, 12);

select substr('SSURESH446@GMAIL.COM', (select instr('ssuresh446@gmail.com', '@', 1, 1) from dual)-1, 12)

\*

ERROR at line 1:

ORA-00923: FROM keyword not found where expected

SQL> select substr('SSURESH446@GMAIL.COM', (select instr('ssuresh446@gmail.com', '@', 1, 1) from dual)-1, 12) from dual;

SUBSTR('SSURESH446@GMAIL.COM',(SELECTINSTR('SSUR

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

6@GMAIL.COM

SQL> select substr('SSURESH446@GMAIL.COM', 1, (select instr('ssuresh446@gmail.com', '@', 1, 1) from dual)-1) from dual;

SUBSTR('SSURESH446@GMAIL.COM',1,(SELECTINSTR('SSURESH446@GMAIL.COM','@',1,1)FROM

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

SSURESH446

SQL> select substr('sanakasuresh@GMAIL.COM', 1, (select instr('sanakasuresh@gmail.com', '@', 1, 1) from dual)-1) from dual;

SUBSTR('SANAKASURESH@GMAIL.COM',1,(SELECTINSTR('SANAKASURESH@GMAIL.COM','@',1,1)

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

Sanakasuresh