DBMS SQL

Lesson 05 : SQL (Single-row) Functions

Lesson Objectives

- To understand the following topics:
  - SQL (single-row) functions
    - Number functions
    - Character functions
    - Date functions
    - Conversion functions
    - Miscellaneous Single-row functions

5.1: Types of Single Row Functions

Single Row Functions - Characteristics

- Manipulate data items
- Accept arguments and return one value
- Act on each row returned
- Return one result per row
- May modify the data type
- Can be nested
- Accept arguments which can be a column or an expression
  function_name [(arg1, arg2, …)]
- Can be used in SELECT, WHERE, and ORDER BY clauses

5.1: Types of Single Row Functions

Number Functions

- Number functions accept “numeric data” as argument, and returns “numeric values”.

CEIL (arg)	Returns the smallest integer greater than or equal to “arg”.
FLOOR (arg)	Returns the largest integer less than or equal to “arg”.
ROUND (arg,n)	Returns “arg” rounded to “n” decimal places. If “n” is omitted, then “arg” is rounded as an integer.
POWER (arg, n)	Returns the argument “arg” raised to the n^{th} power.
SQRT (arg)	Returns the square root of “arg”.
SIGN (arg)	Returns –1, 0, or +1 according to “arg” which is negative, zero, or positive respectively.
ABS (arg)	Returns the absolute value of “arg”.
MOD (arg1, arg2)	Returns the remainder obtained by dividing “arg1” by “arg2”.

Number Functions - Examples

Example 1:

```
SELECT ABS(-15) "Absolute" FROM dual;
```

```
Absolute
15
```

Example 2:

```
SELECT POWER(3,2) "Raised" FROM dual;
```

```
Raised
9
```

Number Functions - Examples

Example 3: ROUND(n,m): Returns n rounded to m places

```
Number
17.2
```

```
SELECT ROUND(17.175,1) "Number" FROM dual;
```

Example 4: TRUNC(n,m): Returns n rounded to m places

```
SELECT TRUNC(15.81,1) "Number" FROM dual;
```

```
Number
 15.8
```

5.1: Types of Single Row Functions

Character Functions

Character functions accept “character data” as argument, and returns “character” or “number” values.

LOWER (arg)	Converts alphabetic character values to lowercase.
UPPER (arg)	Converts alphabetic character values to uppercase.
INITCAP (arg)	Capitalizes first letter of each word in the argument string.
CONCAT (arg1, arg2)	Concatenates the character strings “arg1” and “arg2”.
SUBSTR (arg, pos, n)	Extracts a substring from “arg”, “n” characters long, and starting at position “pos”.
LTRIM (arg)	Removes any leading blanks in the string “arg”.
RTRIM (arg)	Removes any trailing blanks in the string “arg”.
LENGTH (arg)	Returns the number of characters in the string “arg”.
REPLACE (arg, str1, str2)	Returns the string “arg” with all occurrences of the string “str1” replaced by “str2”.
LPAD (arg, n, ch)	Pads the string “arg” on the left with the character “ch”, to a total width of “n” characters.
RPAD (arg, n, ch)	Pads the string “arg” on the right with the character “ch”, to a total width of “n” characters.

5.1: Types of Single Row Functions

Character Functions - Examples

Example 1:

```
SELECT CONCAT(‘Hello ‘,’World’)        “Concat” FROM Dual;
```

```
    Concat
Hello World
```

Example 2:

```
SELECT SUBSTR('HelloWorld',1,5) "SubString“ FROM Dual;
```

```
SubSt
Hello
```

5.1: Types of Single Row Functions

Date Functions

- Date Functions operate on Date & Time datatype values

Add_Months(date1,int1)	Returns a DATE, int1 times added, int1 can be a negative integer
Month_Between(date1,date2	Returns number of months between two dates
Last_Day(date1)	Returns the date of the last day of the month that contains the date
Next_Day(date1,char)	Returns the date of the first weekday specified as char that is later the given date
Current_Date()	Returns the current date in the session time zone. The value is in Gregorian Calendar type
Current_Timestamp	Returns the current date and time in the session time zone. The value returned is of TimeStamp with TimeZone.
Extract(datetime)	Extracts and returns the value of a specified datetime field
Round(date,[fmt])	Returns date rounded to the unit specified . The format will be according to format model fmt
Trunc(date,[fmt])	Returns date truncated to the unit specified . The format will be according to format model fmt

- Sysdate function returns the current date and time

Date Functions- Examples

- Example 1: To display today’s date:

```
SELECT sysdate FROM dual;
```

- Example 2: To add months to a date:

```
SELECT ADD_MONTHS(sysdate,10) FROM dual ;
```

- Example 3: To find difference in two dates

```
SELECT MONTHS_BETWEEN(sysdate,'01-sep-95')
FROM dual ;
```

Date Functions- Examples

- Example 3: To find out last day of a particular month.

```
SELECT last_day(sysdate) FROM dual ;
```

- Example 4: To find the date of the specified day

```
SELECT next_day(sysdate,‘Sunday’) FROM dual ;
```

- Example 5: To display date and time according to current time zone set for the database

```
SELECT sessiontimezone,current_date,current_timestamp FROM
dual ;
```

Arithmetic with Dates

- Use ‘+’ operator to Add and ‘-’ operator Subtract number of days to/from a date for a resultant date value

```
SELECT Student_code ,  (Book_actual_return_date –
        Book_expected_return_date) AS Payable_Days
        FROM  Book_Transaction
        WHERE  Book_Code = 1000;
```

5.1: Types of Single Row Functions

Conversion Functions

- Conversion functions facilitate the conversion of values from one datatype to another.

TO_CHAR (arg,fmt)	Converts a number or date “arg” to a specific character format.
TO_DATE (arg,fmt)	Converts a date value stored as string to date datatype
TO_NUMBER (arg)	Converts a number stored as a character to number datatype.
TO_TIMESTAMP(arg,fmt)	Converts character type to a value of timestamp datatype

Conversion Functions - Examples

Example 1: To display system date in format as 29 November, 1999.

```
SELECT to_char(sysdate,’DD month, YYYY) FROM dual ;
```

Example 2: To display system date in the format as 29th November, 1999.

```
SELECT to_char(sysdate,’DDth month,YYYY’) FROM dual ;
```

Conversion Functions - Examples

Example 3: To display employees whose hiredate is September 08,1981.

```
SELECT staff_code, hiredate FROM staff_master
WHERE
hiredate = TO_DATE ('September 08,1981','Month DD, YYYY');
```

Example 4: To display the value in timestamp format

```
SELECT TO_TIMESTAMP(sysdate,'DD-MM-YY') from dual;
```

5.1: Types of Single Row Functions

Miscellaneous Functions

- Some functions do not fall under any specific category and hence listed as miscellaneous functions

NVL (arg1,arg2)	Replaces and returns a null value with specified actual value
NVL2(arg1,arg2,arg3	If arg1 is not null then it returns arg2. If arg1 is null then arg3 is returned
NULLIF(arg1,arg2)	Compares both the arguments, returns null if both are equal or first argument if both are unequal
COALESCE(arg1,arg2 …argn)	Returns the first non null value in the given list
CASE	Both these functions are for conditional processing, with this IF-Then-Else logic can be applied in SQL statements
DECODE	

Miscellaneous Functions - Examples

Example 1: To display the return date of books and if not returned it should display today's date

```
SELECT book_code,
NVL(book_actual_return_date,sysdate)
FROM book_transaction;
```

Example 2: To examine expected return date of book, and if null return today's date else return the actual return date

```
SELECT book_code,
NVL2(book_expected_return_date,book_actual_return_date, sysdate)
FROM book_transaction;
```

Miscellaneous Functions - Examples

Example 3:To check if the actual return date of the book is same as the expected return date of the book

```
SELECT book_code,
NULLIF(book_expected_return_date, book_actual_return_date)
 FROM book_transaction;
```

Example 4:To track whether the expected/actual return date of the book is populated if any of the values is null it will display sysdate

```
SELECT book_code, COALESCE(book_expected_return_date,
                           book_actual_return_date, sysdate)
FROM book_transaction;
```

5.1: Types of Single Row Functions

The Case Function

- Case() function

  Conditional evaluation by doing work of an IF-THEN-ELSE statement

  Syntax

```
CASE expr when compare_expr1 then return_expr1
                    [when compare_exprn then return_exprn
ELSE else_expr]
END
```

Example

```
CASE expr when compare_expr1 then return_expr1
                    [when compare_exprn then return_exprn
ELSE else_expr]
END
```

The Decode Function

Decode () function:

Similar to CASE, Conditional evaluation by doing work of an IF-THEN-ELSE statement

Syntax

Example:

```
DECODE (<exp or coln>, <val1>,<o/p1>,<val2>,<o/p2>,
                    ……..,  <default o/p>)
```

```
SELECT  staff_code, staff_name, dept_code,
       DECODE (deptno,10,’Ten’,20,’Twenty’,’Others’)
FROM staff_master WHERE design_code = 102;
```

5.2: Tips and Tricks

Quick Guidelines

- If possible, try avoiding the SUBSTRING function in the WHERE clauses.
  - Depending on how it is constructed, using the SUBSTRING function can force a table scan instead of allowing the Optimizer to use an Index (assuming there is one).
  - Instead, use the LIKE condition, for better performance.
    - For example: Use the second query instead of using the first query.

```
WHERE SUBSTRING(column_name,1,1) = ‘b’
```

```
WHERE column_name LIKE ‘b%’
```

Summary

- In this lesson, you have learnt:
  - SQL (single-row) functions
  - Character functions
  - Number functions
  - Date functions
  - Conversion functions
  - Miscellaneous Single-row functions

Review – Questions

- Question 1: Single row functions can be broadly classified as ___.
  - Option 1: character functions
  - Option 2: numeric functions
  - Option 3: Date functions
  - Option 4: all the above

- Question 2: The function which returns the value after capitalizing the first character is ___.

Review – Questions

- Question 3: The output of the following function will be ___.
  - select to_char(17000,'$99,999.00') ‘Amount’ from dual;

- Question 4: The decode function can convert NULL values to required type.
  - True / False

- Question 5: The function which returns the last date of the month is ___.
  - Option 1: LAST_DATE
  - Option 2: LAST_DAY
  - Option 3: MONTH_LAST_DATE
  - Option 4: MONTH_LAST_DAY