# SQL CLASS – 33

# TOPICS– **Date** **Conversion Function**

**Conversion Function –**

* The Conversation functions convert a value from one data type to another.
* The Data type conversion in Oracle is of two types.

1. Implicit Data type Conversion
2. Explicit Data type Conversion

**Implicit Data Type Conversion –**

* Implicit Date type conversion work according to the convention specified by oracle.
* The Assignment succeeds if the Oracle server can convert the date type of value.
* CHAR to NUMBER conversion succeed only if the character string represents a valid NUMBER.
* CHAR to DATES conversion succeed only if the character string represents the default format of DD-MON-YY.

**In Assignment Operator –box**

|  |  |
| --- | --- |
| Varchar2 / Char | Number |
| Varchar2 / Char | **Date** |
| Number | **Varchar2** |
| Date | **Varchar2** |

**Explicit Data Type Conversion –**

* SQL provided three functions to convert a value from one data type

to another.

* The explicit conversation function is

1. TO\_CHAR (To character conversion)
2. TO\_DATE (To Date conversion)
3. TO\_NUMBER (To Number Conversion)

**TO\_CHAR CONVERSION –**

* This function can be used in two ways.

1. TO\_CHAR (Number Conversion)
2. TO\_CHAR (Date Conversion)

**TO\_CHAR (NUMBER CONVERSION) –**

**Syntax**-TO\_CHAR (NUMBER, format)

• Converts Number of Number data type to a value of VARCHAR2 data type.

• 'format' is the optional number format that can be used.

**Decimal Indicator – D**

* It returns the specified position of the decimal character.
* The default decimal delimiter is period (**.**)
* Only one decimal indicator can be specified in a number format model.

**Syntax-**

(Number,99D99)

**Example –**

SELECT 1234, TO\_CHAR (1234,'9999D99') FROM DUAL;

O/P: 1234.00

SELECT 1234, TO\_CHAR (1234,'999D99') FROM DUAL;

O/P: #######

SELECT 12345.10, TO\_CHAR (12345.10,'99999D9') FROM DUAL;

O/P: 12345.1

SELECT 12345.10, TO\_CHAR (12345.10,'999D9') FROM DUAL;

O/P: ######

**Group Separator – G**

* Returns the specified position of the Group separator
* Multiple Group separators can be specified.

**Syntax –**

(Number, 9G999)

**Examples –**

CREATE TABLE Employee (Empid NUMBER,

Ename VARCHAR2 (25),

Elocation VARCHAR2 (25),

Salary Number);

INSERT INTO Employee VALUES (1, 'Sam', 'Pune', 55000);

INSERT INTO Employee VALUES (2, 'Merry', 'UK', 66000);

INSERT INTO Employee VALUES (3, 'Johns', 'UK', 77000);

INSERT INTO Employee VALUES (4, 'Tom', 'Mumbai', 88000);

INSERT INTO Employee VALUES (5, 'Jerry', 'Pune', 990000);

SELECT TO\_CHAR (1234567,'99G99G9999') FROM DUAL;

SELECT Salary, TO\_CHAR (Salary,'9G999') FROM Employee;

**Local Currency Indicator: L**

* Returns the specified position of the local currency symbol.

**Syntax –**

(Number, 'L999 OR 999L')

**Example –**

SELECT 1234, TO\_CHAR (1234,'L9999') FROM DUAL;

SELECT Salary, TO\_CHAR (Salary,'L99999') Currency

FROM Employee

WHERE Empid = 4;

SELECT Salary, TO\_CHAR (Salary,'L99G99D9','NLS\_CURRENCY = IndRupees') Salary

FROM Employee

WHERE Empid = 5;

**Trailing Minus Indicator -** MI

* Return negative value with a trailing minus sign'-'.
* Returns positive value with a trailing Blank.
* 'MI' Format should be declared as trailing argument only.

**Syntax –**

(Number,9999MI)

**Examples –**

SELECT -10000, TO\_CHAR (-10000,'L99G99D9MI') FROM DUAL;

SELECT Salary, TO\_CHAR (Salary, 'L99G99D9MI') FROM Employee;

**Number Indicator – PR**

* Returns negative number in '<>'.
* It can appear only as trailing declaration.

**Syntax –**

(Number,9999PR)

**Examples –**

SELECT TO\_CHAR (-1000,'L99G999D99PR') FROM DUAL;

SELECT Salary, TO\_CHAR (Salary,'L99999PR') FROM Employee;

**Roman Number Indicator-**

* **RN** - Returns Upper Roman Number.
* **rn** - Returns Lower Roman Number.
* The value can be an integer between 1 and 3999.

**Syntax –**

(Number, 'RN')

(Number, 'rn')

**Example –**

SELECT 12, TO\_CHAR (12,'RN'), TO\_CHAR (12,'RN') FROM DUAL;

**O/p-** XII and xii

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**THANK YOU**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_