Oracle PL/SQL – Data Types

PL/SQL Data Types

- PL/SQL variables, constants and parameters must have a valid data type, which specifies a storage format, constraints, and valid range of values.

- PL/SQL data types can be broken down into the following categories:

  - Scalar
  - Reference
  - Composite
  - LOB (Large Object)

Scalar Data Type

- A scalar type is a data type that holds a single value.
- Scalar types can be broken down into subcategories, that include
  - Character/String
  - Number
  - Boolean
  - Date/Time

Character/String Type

- PL/SQL character or string types include everything from sinngl echaracter values to large strings up to 32K in size

- These types can store letters, numbers, and binary data.

- Frequently used are char and varchar2

Example:

```
DECLARE
v_string VARCHAR2(10); -- variable v_string can store up to 10 bytes
```

Different Character Types

DATA TYPE	DESCRIPTION
CHAR	Fixed-length character string with maximum size of 32,767 bytes
VARCHAR	VARCHAR is an ANSI-standard SQL type, synonymous with VARCHAR2. Oracle recommends VARCHAR2.
VARCHAR2	Variable-length character string with maximum size of 32,767 bytes
LONG	Variable-length character string with maximum size of 32,760 bytes
LONG RAW	Variable-length binary or byte string with maximum size of 32,760 bytes
NCHAR	Fixed-length national character string with maximum size of 32,767 bytes
NVARCHAR2	Variable-length national character string with maximum size of 32,767 bytes
RAW	Variable-length binary or byte string with maximum size of 32,767 bytes, not interpreted by PL/SQL
ROWID	Physical row identifier, the address of a row in an ordinary table
UROWID	Universal row identifier (physical, logical, or foreign row identifier)

Number Type

- The NUMBER datatype supports **integer, real numbers** and f**loating-point** values.

- Precision and scale should be defined for floating point values

Precision:

- The number of total digits allowed for the value.
- The maximum precision for the NUMBER type is 38.

Scale:

- The number of digits allowed to the right of the decimal place (if scale is positive), or number of digits to round to the left of the decimal place (if scale is negative).
- Scale can range from –84 to 127.

PL/SQL Numeric Types

DATA TYPE	DESCRIPTION
PLS_INTEGER / BINARY_INTEGER	Signed integer in range -2,147,483,648 through 2,147,483,647, represented in 32 bits
BINARY_FLOAT	Single-precision IEEE 754-format floating-point number
BINARY_DOUBLE	Double-precision IEEE 754-format floating-point number
NUMBER(prec, scale)	Fixed-point or floating-point number with absolute value in range 1E-130 to (but not including) 1.0E126. A NUMBER variable can also represent 0.
INTEGER	ANSI and IBM specific integer type with maximum precision of 38 decimal digits
FLOAT	ANSI and IBM specific floating-point type with maximum precision of 126 binary digits (approximately 38 decimal digits)

Boolean Type

- Boolean accepts values of TRUE, FALSE and NULL.

- Quotes should not be used while assigning values to Boolean variable. If quotes are used, they cause an error condition

- Since SQL has no data type equivalent to BOOLEAN, they cannot be used in
  - SQL statements
  - Built-in SQL functions (such as TO_CHAR)
  - PL/SQL functions invoked from SQL statements

Example:

```
DECLARE
    v_boolean BOOLEAN;
BEGIN
    v_boolean := TRUE;
END;
```

Date/Time Type

- Date/Time types includes the below
  - DATE
  - TIMESTAMP
  - INTERVAL

- The DATE PL/SQL type stores the century, year, month, day, hour, minute, and second.

- Dates can be converted between Character types and the DATE type using the below built-in functions
  - TO_DATE
  - TO_CHAR

- The default date format is set by the Oracle initialization parameter NLS_DATE_FORMAT. For example, the default might be 'DD-MON-YY'

TIMESTAMP Type

PL/SQL TIMESTAMP supports the below types

- ✓ **TIMESTAMP** (same as DATE but also provides sub second times up to nine digits)
- ✓ **TIMESTAMP WITH TIMEZONE** (same as TIMESTAMP but it returns timestamp relative to GMT or UTC)
- ✓ **TIMESTAMP WITH LOCAL TIMEZONE** (same as timestamp but returns time corresponding to the location of the client accessing the data server)

```
SET SERVEROUTPUT ON;
DECLARE
        v_datetime1 TIMESTAMP := SYSTIMESTAMP;
        v_datetime2 TIMESTAMP WITH TIME ZONE := SYSTIMESTAMP;
        v_datetime3 TIMESTAMP WITH LOCAL TIME ZONE := SYSTIMESTAMP;
        v_date DATE := SYSDATE;
BEGIN
        dbms_output.put_line(v_datetime1);
        dbms_output.put_line(v_datetime2);
        dbms_output.put_line(v_datetime3);
        dbms_output.put_line(v_date);
END;
```

- Large object (LOB) data type refers large data items such as text, graphic images, video clips, and sound waveforms.

- Predefined PL/SQL LOB data types are

Data Type	Description	Size
BFILE	Used to store large binary objects in operating system files outside the database.	System-dependent. Cannot exceed 4 gigabytes (GB).
BLOB	Used to store large binary objects in the database	8 to 128 terabytes (TB)
CLOB	Used to store large blocks of character data in the database.	8 to 128 terabytes (TB)
NCLOB	Used to store large blocks of NCHAR data in the database.	8 to 128 terabytes (TB)

Composite

- Composite types differ from Scalar types in that they have internal components.
- They can contain multiple scalar variables that are referred to as attributes.
- Composite types include records, nested tables, index-by tables, and varrays.

Reference

- Oracle PL/SQL supports two types in Reference category
  - REF CURSOR
  - REF
- Reference types provide memory structures, but unlike Scalar and Composite types, they can point to different storage locations throughout the program.