

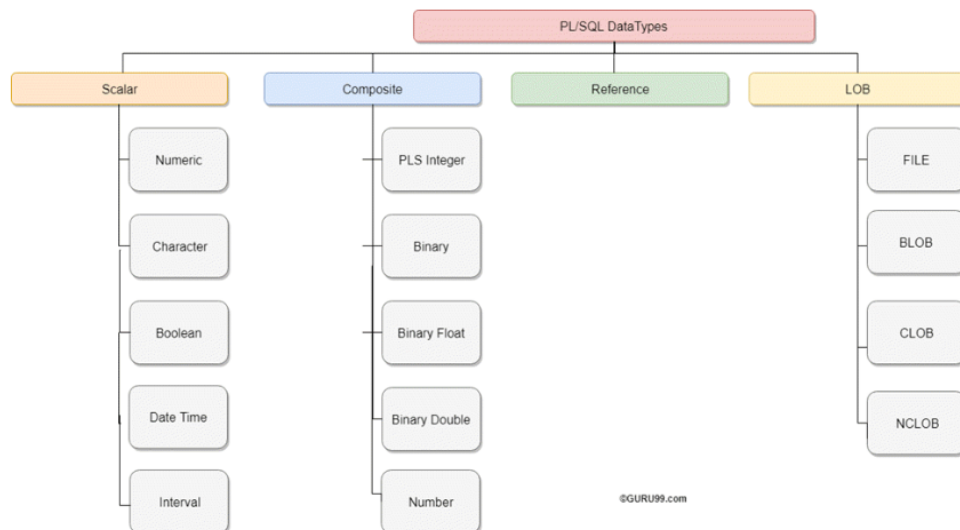
TIPOS DE DATOS

¿Qué son los tipos de datos PL/SQL?

Tipos de datos en PL/SQL se utilizan para definir cómo los datos serán almacenados, manejados y tratados por Oracle durante el almacenamiento y procesamiento de datos. Los tipos de datos están asociados con el formato de almacenamiento específico y las restricciones de rango. En Oracle, a cada valor o constante se le asigna un tipo de datos.

La principal diferencia entre PL/SQL y [SQL](#) tipos de datos es, el tipo de datos SQL se limita a la columna de la tabla, mientras que los tipos de datos PL/SQL se utilizan en la [bloques PL/SQL](#)

A continuación se muestra el diagrama de diferentes Oracle Tipos de datos PL/SQL:



Diferentes tipos de datos en PL/SQL

Tipo de datos de carácter PL/SQL

Este tipo de datos básicamente almacena caracteres alfanuméricos en formato de cadena.

Los valores literales siempre deben estar entre comillas simples al asignarlos al tipo de datos CHARACTER.

Este tipo de datos de carácter se clasifica además de la siguiente manera:

- CHAR Tipo de datos (tamaño de cadena fijo)
- VARCHAR2 Tipo de datos (tamaño de cadena variable)
- VARCHAR Tipo de datos
- NCHAR (tamaño de cadena fijo nativo)

Tipo de datos	Descripción	Sintaxis
CHAR	<p>Este tipo de datos almacena el valor de la cadena y el tamaño de la cadena se fija en el momento de declarar el variable.</p> <ul style="list-style-type: none"> • Oracle La variable se rellenaría en blanco si la variable no ocupara todo el tamaño que se ha declarado para ella, por lo tanto Oracle asignará la memoria para el tamaño declarado incluso si la variable no la ocupó por completo. • La restricción de tamaño para este tipo de datos es de 1 a 2000 bytes. • El tipo de datos CHAR es más apropiado para usar siempre que se maneje el tamaño de datos fijo. 	<p>grade CHAR;</p> <p>manager CHAR (10):= 'guru99';</p> <p>Explicación de sintaxis:</p> <ul style="list-style-type: none"> • La primera declaración declaró la variable "grado" del tipo de datos CHAR con el tamaño máximo de 1 byte (valor predeterminado). • La segunda declaración declaró la variable 'administrador' del tipo de datos CHAR con el tamaño máximo de 10 y asignó el valor 'guru99' que es de 6 bytes. Oracle asignará la memoria de 10 bytes en lugar de 6 bytes en este caso.
VARCHAR2	<p>Este tipo de datos almacena la cadena, pero la longitud de la cadena no es fija.</p> <ul style="list-style-type: none"> • La restricción de tamaño para este tipo de datos es de 1 a 4000 bytes para el tamaño de la columna de la tabla y de 1 a 32767 bytes para las variables. • El tamaño se define para cada variable en el momento de la declaración de la variable. • Pero Oracle asignará memoria sólo después de que se defina la variable, es decir, Oracle considerará solo la longitud real de la cadena que está almacenada en una variable para la asignación de memoria en lugar del tamaño que se ha dado para una variable en la parte de declaración. • Siempre es bueno utilizar VARCHAR2 en lugar del tipo de datos CHAR para optimizar el uso de la memoria. 	<p>manager VARCHAR2(10) := 'guru99';</p> <p>Explicación de sintaxis:</p> <ul style="list-style-type: none"> • La declaración anterior declaró la variable 'administrador' del tipo de datos VARCHAR2 con el tamaño máximo de 10 y le asignó el valor 'guru99' que es de 6 bytes. Oracle asignará memoria de sólo 6 bytes en este caso.
VARCHAR	<p>Esto es sinónimo del tipo de datos VARCHAR2.</p>	<p>manager VARCHAR(10) := 'guru99';</p> <p>Explicación de sintaxis:</p>

Tipo de datos	Descripción	Sintaxis
	<ul style="list-style-type: none"> Siempre es una buena práctica utilizar VARCHAR2 en lugar de VARCHAR para evitar cambios de comportamiento. 	<ul style="list-style-type: none"> La declaración anterior declaró la variable 'administrador' del tipo de datos VARCHAR con el tamaño máximo de 10 y asignó el valor 'guru99' que es de 6 bytes. Oracle asignará memoria de sólo 6 bytes en este caso. (Similar a VARCHAR2)
	<ul style="list-style-type: none"> NVARCHAR2 (tamaño de cadena variable nativa) LARGO y LARGO CRUDO 	
NCHAR	<p>Este tipo de datos es el mismo que el tipo de datos CHAR, pero el juego de caracteres será el juego de caracteres nacional.</p> <ul style="list-style-type: none"> Este juego de caracteres se puede definir para la sesión utilizando NLS_PARAMETERS. El juego de caracteres puede ser UTF16 o UTF8. La restricción de tamaño es de 1 a 2000 bytes. 	<p>native NCHAR(10);</p> <p>Explicación de sintaxis:</p> <ul style="list-style-type: none"> La declaración anterior declara la variable 'nativa' del tipo de datos NCHAR con un tamaño máximo de 10. La longitud de esta variable depende del (número de longitudes) por byte tal como se define en el juego de caracteres.
NVARCHAR2	<p>Este tipo de datos es el mismo que el tipo de datos VARCHAR2, pero el juego de caracteres será el juego de caracteres nacional.</p> <ul style="list-style-type: none"> Este juego de caracteres se puede definir para la sesión utilizando NLS_PARAMETERS. El juego de caracteres puede ser UTF16 o UTF8. La restricción de tamaño es de 1 a 4000 bytes. 	<p>Native var NVARCHAR2(10):='guru99';</p> <p>Explicación de sintaxis:</p> <ul style="list-style-type: none"> La declaración anterior declara la variable 'Native_var' del tipo de datos NVARCHAR2 con un tamaño máximo de 10.
LARGO y LARGO	<p>Este tipo de datos se utiliza para almacenar texto grande o datos sin procesar hasta un tamaño máximo de 2 GB.</p>	<p>Large_text LONG;</p> <p>Large_raw LONG RAW;</p> <p>Explicación de sintaxis:</p>

- Se utilizan principalmente en el diccionario de datos.
- El tipo de datos LONG se utiliza para almacenar datos del juego de caracteres, mientras que LONG RAW se utiliza para almacenar datos en formato binario.
- El tipo de datos LONG RAW acepta objetos multimedia, imágenes, etc., mientras que LONG solo funciona con datos que se pueden almacenar utilizando un juego de caracteres.

- La declaración anterior declara la variable 'Large_text' del tipo de datos LONG y 'Large_raw' del tipo de datos LONG RAW.

Nota: No se recomienda el uso del tipo de datos LARGO Oracle. En su lugar, se debe preferir

NÚMERO PL/SQL Tipo de datos

Este tipo de datos almacena números de punto fijo o flotante de hasta 38 dígitos de precisión. Este tipo de datos se utiliza para trabajar con campos que contendrán solo datos numéricos. La variable se puede declarar con precisión y detalles de dígitos decimales o sin esta información. Los valores no necesitan estar entre comillas al asignarse para este tipo de datos.

A NUMBER(8,2);

B NUMBER(8);

C NUMBER;

Explicación de sintaxis:

- En lo anterior, la primera declaración declara que la variable 'A' es de tipo de datos numéricos con precisión total 8 y dígitos decimales 2.
- La segunda declaración declara que la variable 'B' es de tipo de datos numéricos con precisión total 8 y sin dígitos decimales.
- La tercera declaración es la más genérica, declara que la variable 'C' es de tipo numérico sin restricción en precisión o decimales. Puede tener hasta un máximo de 38 dígitos.

Tipo de datos booleano PL/SQL

Este tipo de datos almacena los valores lógicos. Oracle El tipo de datos booleano representa VERDADERO o FALSO y se utiliza principalmente en declaraciones condicionales. No es necesario que los valores estén entre comillas al asignarlos para este tipo de datos.

Var1 BOOLEAN;

Explicación de sintaxis:

- En lo anterior, la variable 'Var1' se declara como tipo de datos BOOLEANO. La salida del código será verdadera o falsa según la condición establecida.

Tipo de datos de fecha PL/SQL

Este tipo de datos almacena los valores en formato de fecha, como fecha, mes y año. Siempre que una variable se define con el tipo de datos FECHA junto con la fecha, puede contener información de hora y, de forma predeterminada, la información de hora se establece en 12:00:00 si no se especifica. Los valores deben estar entre comillas al asignarse para este tipo de datos.

El Oracle El formato de hora para entrada y salida es 'DD-MON-AA' y nuevamente se establece en NLS_PARAMETERS (NLS_DATE_FORMAT) en el nivel de sesión.

```
newyear DATE:='01-JAN-2015';
```

```
current_date DATE:=SYSDATE;
```

Explicación de sintaxis:

- En lo anterior, la variable "año nuevo" se declara como tipo de datos FECHA y se le asigna el valor del 1 de enero.st, fecha 2015.
- La segunda declaración declara la variable current_date como tipo de datos DATE y le asigna el valor con la fecha actual del sistema.
- Ambas variables contienen la información de tiempo.

Tipo de datos LOB PL/SQL

Este tipo de datos se utiliza principalmente para almacenar y manipular grandes bloques de datos no estructurados como imágenes, archivos multimedia, etc. Oracle prefiere LOB en lugar del tipo de datos LONG ya que es más flexible que el tipo de datos LONG. Las siguientes son las principales ventajas del tipo de datos LOB sobre el tipo de datos LARGO.

- El número de columnas en una tabla con tipo de datos LONG está limitado a 1, mientras que una tabla no tiene restricción en el número de columnas con tipo de datos LOB.
- La herramienta de interfaz de datos acepta el tipo de datos LOB de la tabla durante la replicación de datos, pero omite la columna LARGA de la tabla. Estas columnas LARGAS deben replicarse manualmente.
- El tamaño de la columna LARGA es de 2 GB, mientras que LOB puede almacenar hasta 128 TB.

- Oracle está mejorando constantemente el tipo de datos LOB en cada una de sus versiones de acuerdo con los requisitos modernos, mientras que el tipo de datos LONG es constante y no recibe muchas actualizaciones.

Por lo tanto, siempre es mejor utilizar el tipo de datos LOB en lugar del tipo de datos LONG. A continuación, se muestran los diferentes tipos de datos LOB. Pueden almacenar hasta un tamaño de 128 terabytes.

1. BLOB
2. CLOB y NCLOB
3. ARCHIVOB

Tipo de datos	Descripción	Sintaxis
BLOB	Este tipo de datos almacena los datos LOB en formato de archivo binario hasta un tamaño máximo de 128 TB. No almacena datos basados en los detalles del conjunto de caracteres, por lo que puede almacenar datos no estructurados, como objetos multimedia, imágenes, etc.	Binary_data BLOB; Explicación de sintaxis: <ul style="list-style-type: none"> • En lo anterior, la variable 'Binary_data' se declara como BLOB.
CLOB y NCLOB	El tipo de datos CLOB almacena los datos LOB en el juego de caracteres, mientras que NCLOB almacena los datos en el juego de caracteres nativo. Dado que estos tipos de datos utilizan almacenamiento basado en juegos de caracteres, no pueden almacenar datos como multimedia, imágenes, etc. que no se pueden poner en una cadena de caracteres. El tamaño máximo de estos tipos de datos es 128 TB.	Charac_data CLOB; Explicación de sintaxis: <ul style="list-style-type: none"> • En lo anterior, la variable 'Charac_data' se declara como tipo de datos CLOB.
ARCHIVOB	<ul style="list-style-type: none"> • BFILE son los tipos de datos que almacenaron los datos en formato binario no estructurado fuera de la base de datos como un archivo del sistema operativo. • El tamaño de BFILE es para un sistema operativo limitado, son archivos de solo lectura y no se pueden modificar. 	