

CURSOS
INTERSEMESTRALES



PROTECO

DDL

Data Definition Language
Y Tipos de Dato

SQL1

Semestre 2020-1

Tipos de dato en Oracle

Al igual que en otros lenguajes SQL y mas exáctamente Oracle 18c tiene sus propios tipos de dato.

Es importante destacar que SQL es técnicamente un Estándar de datos por el cual muchos lenguajes se rigen.



Cadenas de Caracteres

SQL 2003 (especificación)	ORACLE 11g	DB2 9.5	SQL Server 2008	PostgreSQL 8.x	MySQL 5.x
CHAR [ACTER] [(n)]	CHAR[ACTER] [(n)]	CHAR[ACTER] [(n)]	CHAR[ACTER] [(n)]	CHAR[ACTER] [(n)]	CHAR [(M)]
CHAR[ACTER] VARIING(n) ó VARCHAR (n)	CHAR[ACTER] VARIING(n) ó VARCHAR(n) ó VARCHAR2 (n)	CHAR[ACTER] VARIING(n) ó VARCHAR(n) ó LONGVARCHAR	CHAR[ACTER] VARIING(n) ó VARCHAR [(n)] TEXT	CHAR[ACTER] VARIING(n) ó VARCHAR(n)	VARCHAR(M) ó TINYTEXT ó TEXT ó MEDIUMTEXT ó LONGTEXT
CLOB ó CHARACTER LARGE OBJECT	CLOB ó LONG [VARCHAR]	CLOB [(n) [K M,G]]	VARCHAR (MAX)	TEXT	BINARY [(M)] VARBINARY (M)

Longitudes de Caracteres

Tipo dato	de	Oracle
CHAR		1-2000
VARCHAR2		1-4000
NCHAR (unicode)		1-2000
NVARCHAR (unicode)		1-4000
CLOB		8 TB
NCLOB		8 TB

Tipo de dato	DB2
CHAR	1-254
VARCHAR	1-32672
LONG VARCHAR	1-32700
CLOB	2 GB
GRAPHIC	1-127
VARGRAPHIC	1-16, 336
LONG VARGRAPHIC	16, 350



Longitudes de Caracteres

Tipo dato	de	SQL Server
CHAR		1-8000
VARCHAR		1-8000
TEXT		2 GB
NCHAR		1-4000
NVARCHAR		
NTEXT		1 GB
VARCHAR (MAX)		2 GB
NVARCHAR		

Tipo dato	de	MySQL
CHAR		1-255
VARCHAR		1-65535
TINYTEXT		0-255
TEXT		0-65535
MEDIUM TEXT		0-16777215
LONGTEXT		0-4294967295

Tipo de dato	PostgreSQL
CHAR	1 GB
VARCHAR	1-GB
TEXT	1-GB



Tipos de dato Numérico exacto

Los tipos de datos numéricos exactos pueden ser números enteros finitos o números con parte decimal finita.

Al definir un atributo de una tabla con un tipo de dato numérico exacto se puede definir 2 elementos:

- Precisión: Número total de dígitos: Parte entera + parte decimal
- Escala: Número de dígitos que formarán a la parte decimal.



Tipo de dato Numérico exacto

SQL 2003 (especificación)	ORACLE 11g	DB2 9.5	SQL Server 2008	PostgreSQL 8.x	MySQL 5.x
INT[EGER]	NUMBER (n)	INT[EGER] ó BIGINT	INT[EGER] ó BIGINT	INTEGER ó BIGSERIAL ó SERIAL ó BIGINT	INT[(M)] ó MEDIUMINT[(M)] ó BIGINT[(M)]
SMALLINT	SMALLINT NUMBER (n)	SMALLINT	SMALLINT ó TINYINT	SMALLINT	SMALLINT[(M)] ó TINYINT[(M)]
NUMERIC [(P[, S])] ó DEC[IMAL] [(P[, S])]	NUMERIC [(P[, S])] ó DEC[IMAL] [(P[, S])] ó NUMBER[(P[, S])]	NUMERIC [(P[, S])] ó DEC[IMAL] [(P[, S])]	NUMERIC [(P[, S])] ó DEC[IMAL] [(P[, S])] MONEY SMALLMONEY	NUMERIC [(P[, S])] ó DEC[IMAL] [(P[, S])]	

Tipos de Dato numérico Aproximados

Se emplean cuando no se conoce con exactitud los valores de la precisión y/o escala que puede tener los valores de una columna.

EN ORACLE

- INTEGER, SMALLINT se convierten a NUMBER(38)
- NUMERIC, DECIMAL se convierten a NUMBER(38)
- El rango se define de 1×10^{-130} a 9×10^{125} (38 nueves).

Se recomienda que todos los tipos de datos de tipo numérico sean de tipo "NUMBER", ya que es el Estándar de Oracle



SQL 2003 (especificación)	ORACLE 11g	DB2 9.5	SQL Server 2008	PostgreSQL 8.x	MySQL 5.x
FLOAT[(P)]	FLOAT[(P)] ó NUMBER	FLOAT[(P)]	FLOAT[(P)]	-----	FLOAT[(P[,S])]
REAL	REAL NUMBER	REAL	REAL	REAL	REAL[(P[,S])]
DOUBLE PRECISION	DOUBLE PRECISION NUMBER	DOUBLE [PRECISION]	DOUBLE PRECISION	DOUBLE PRECISION	DOUBLE [PRECISION]

Cadenas Binarias

Se emplean para almacenar secuencia de bytes en la base de datos, es decir, se almacenan archivos binarios: fotos, música, videos, huellas, documentos, etc.

SQL 2003 (especificación)	ORACLE 11g	DB2 9.5	SQL Server 2008	PostgreSQL 8.x	MySQL 5.x
BLOB ó BINARY OBJECT	LARGE BLOB ó LONGRAW ó RAW (n)	BLOB [(n) [K M G]]	VARBINARY (MAX) ó VARBINARY [(n)] ó IMAGE	BYTEA	BINARY [(n)] ó VARBINARY (M) ó TINYBLOB ó BLOB ó MEDIUM BLOB ó LONG BLOB



Tipo de dato	de Oracle
BLOB	8 TB
RAW	4000 BYTES

Tipo de dato	DB2
BLOB	2 GB

Tipo de dato	SQL Server
BINARY	8000
VARBINARY	8000
IMAGE	2147483647
VARBINARY (MAX)	2 GB



Primeros Pasos

1.- Abrir una Terminal

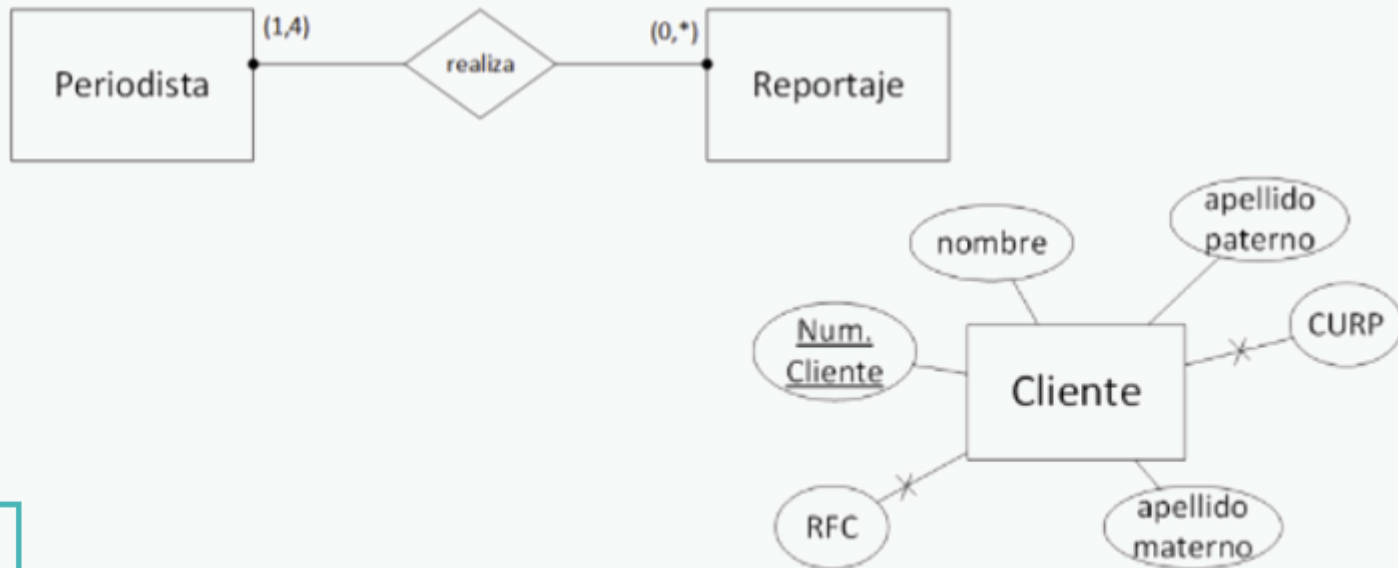
```
su -l oracle  
sqlplus / as sysdba  
startup  
exit  
sqlplus / as sysdba
```



De Dibujitos al Código

Los diagramas (En Cualquier notación), nos ayudan bastante para la creación del código SQL, algunos entornos nos permiten pasar directamente del modelo al código con algunos ajustes extras

Modelo E/R



De Dibujitos al código

Usando la sentencia "CREATE TABLE"

Estudiante

- ◆ Estudiante_Id
- ◆ Edad
- ◆ Correo_electrónico
- ◆ Carrera
- ◆ Promedio
- ◆ Estatura







```
CREATE TABLE estudiante(  
    estudiante_id NUMBER(10),  
    edad NUMBER(10),  
    correo_electronico VARCHAR2(40),  
    carrera VARCHAR2(40),  
    promedio NUMBER(3,2),  
    estatura NUMBER(2,3)  
);
```



De Dibujitos al código

Usando la sentencia "CREATE TABLE"

Estudiante







 Estudiante_Id
 Edad
 Correo_electrónico
 Carrera
 Promedio
 Estatura

```
CREATE TABLE estudiante(  
    estudiante_id NUMBER(10) PRIMARY KEY,  
    edad NUMBER(10),  
    correo_electronico VARCHAR2(40),  
    carrera VARCHAR2(40),  
    promedio NUMBER(3,2),  
    estatura NUMBER(2,3)  
);
```



Ejemplo

AUTOMOVIL









 automovil_id
 color
 num_ruedas
 num_puertas
 marca
 año

```
CREATE TABLE automovil(  
    automovil_id NUMBER(5) PRIMARY KEY,  
    color VARCHAR2(40),  
    num_ruedas NUMBER(10),  
    num_puertas NUMBER(10),  
    marca VARCHAR2(20),  
    año NUMBER(4)  
);
```



EJERCICIO 1

USUARIO

 USUARIO_ID	VARCHAR(40)	NOT NULL
 NOMBRE	VARCHAR(40)	NOT NULL
 APELLIDO_PATERNO	VARCHAR(40)	NOT NULL
 APELLIDO_MATERNO	VARCHAR(40)	NOT NULL
 NUM_MATRICULA	NUMERIC(18,0)	NOT NULL
 NUM_SEMESTRE	NUMERIC(18,0)	NULL
 USERNAME	VARCHAR(40)	NOT NULL
 PASSWORD	VARCHAR(40)	NOT NULL
 CON_PRESTAMO	CHAR(2)	NOT NULL
 CON_PRESTAMO_VENCIDO	CHAR(2)	NOT NULL








BIBLIOTECA

 BIBLIOTECA_ID	NUMERIC(10,0)	NOT NULL
 NOMBRE	VARCHAR(20)	NOT NULL
 FOLIO	CHAR(5)	NOT NULL
 LATITUD	VARCHAR(40)	NOT NULL
 LONGITUD	VARCHAR(40)	NOT NULL
 DIRECCION_WEB	VARCHAR(40)	NOT NULL
 DIRECCION	VARCHAR(40)	NOT NULL








EJERCICIO 2

COMPUTADORA

 COMPUTADORA_ID	NUMERIC(40,0)	NOT NULL
 MARCA	VARCHAR(40)	NOT NULL
 AÑO	VARCHAR(40)	NOT NULL
 MODELO	VARCHAR(40)	NOT NULL
 PROCESADOR	VARCHAR(40)	NOT NULL
 CANTIDAD_RAM	VARCHAR(40)	NOT NULL
 MARCA_TARJETA_GRAFICA	VARCHAR(40)	NULL






TELEVISION

 TELEVISION_ID	NUMERIC(40,0)	NOT NULL
 MODELO	VARCHAR(40)	NOT NULL
 AÑO	VARCHAR(40)	NOT NULL
 RESOLUCION	VARCHAR(40)	NOT NULL
 MARCA	VARCHAR(40)	NOT NULL



Ejercicio

Teniendo el siguiente modelo:

CANCION		
 CANCION_ID	NUMERIC(10,0)	NOT NULL
 AÑO	DATE	NULL
 FOTO	BINARY(40)	NULL
 LETRA	VARCHAR(40)	NULL
 NOMBRE	VARCHAR(40)	NOT NULL

- Crear la tabla
- Modificar la columna nombre por "Nombre_cancion"
- Agregar la columna "Álbum de tipo varchar2"
- Agregar la columna "Artista" de tipo Varchar2
- Eliminar la columna Letra

