



# BASES DE DATOS

## Práctica 1

### *Definición de Datos*

*UCLM-ESI*



## Objetivos

- Enseñar al alumno las sentencias que forman el lenguaje de **definición de datos** (DDL – Data Definition Language) de SQL, es decir, el subconjunto de órdenes que nos permitirán **crear** o **editar** tablas de la base de datos.



## Contenido

---

- Definición de Datos
  - Creación de Tablas
  - Restricciones de Columnas
  - Restricciones de Tablas
  - Renombrar Tablas
  - Eliminar Tablas
  - Modificar Tablas
  - Crear y Borrar Sinónimos
  - Crear Dominios
  - Ejercicios



## Bibliografía

---

- **Básica**
  - Oracle 8, Guía de aprendizaje. Michael Abbey, Michael Corey, McGraw-Hill 1997.
  - Guía de SQL, James R. Groff, Paul N. Weinberg, McGraw-Hill, 1998.
- **Complementaria**
  - La biblia de Oracle 8. Ediciones Anaya Multimedia, 1998.
  - Understanding the new SQL: A Complete Guide, Jim Melton, Alan R. Simon, Morgan Kaufmann Publisher, 1993.
  - A Guide to SQL Standard (fourth edition). Date, C.J. and Darwen, H. Addison-Wesley, 1997.
  - SQL. Manual de Referencia del Programador. Freeze, W.S. Paraninfo, 1998.



## Definición de Datos

- El lenguaje de definición de datos permite:
  - Definir y crear una nueva tabla.
  - Suprimir una tabla que ya no se necesita.
  - Cambiar la definición de una tabla existente.
  - Definir una tabla virtual (o vista) de datos.
  - Construir un índice para hacer más rápido el acceso a una tabla.
  - Controlar el almacenamiento físico de los datos por parte del SBGD.



## Definición de Datos

- La sintaxis general de la sentencia `create table` es:

```
CREATE TABLE <nombre de tabla>  
(nombre_columna1 tipo [restricción de columna],  
.....  
nombre_columnaN tipo [restricción de columna],  
[restricción_de_tabla]);
```

- La sentencia **CREATE TABLE** se utiliza para crear una tabla dentro de la cual habrá columnas que contienen datos y restricciones



# Definición de Datos

## Más en detalle...

```

CREATE TABLE <nombre tabla>
( <nombre columna> <tipo de dato>
  [NOT NULL] [UNIQUE]
  [CONSTRAINT <nombre restricción>]
  [PRIMARY KEY] [REFERENCES] [DEFAULT] [CHECK]
| [PRIMARY KEY (<lista columnas>)]
| [FOREIGN KEY (<lista columnas>) REFERENCES (nombretabla)]
| [UNIQUE (<lista columnas>)] [CONSTRAINT <nombre
  restricción>],[,...] )
| [CHECK (condición de búsqueda)]

```



# Definición de Datos

| Tipo de dato     | Descripción   |
|------------------|---|
| Char(tamaño)     | Almacena datos de tipo carácter de longitud fija, con un máximo de 2000 caracteres)   |
| varchar2(tamaño) | Almacena datos de tipo carácter de longitud variable, con un tamaño máximo de 4000  |
| varchar          | Actualmente es igual que char   |
| Long             | Almacena datos de tipo carácter de longitud variable, hasta 2 gigabytes. Solo se permite un Long por tabla. Una columna de tipo Long no puede utilizarse como parte de un índice. Una función almacenada no puede devolver un Long. Las cláusulas Where, Group By, Order By, Unique, o Connect By no pueden referenciar a una columna Long. |
| blob             | Es un objeto binario de gran tamaño, siendo el tamaño máximo 4 GB (gigabytes). Normalmente un blob se utiliza para almacenar una imagen, datos de voz, o cualquier otro bloque de datos grande no estructurado.   |
| date             | Almacena fechas desde el 1 de enero del 4712 a.C. hasta el 31 de diciembre del 4712 d.C.  |
| Integer          | Un número entero que no tiene parte fraccionaria. Normalmente un Integer será un valor de 32 bits con un rango de -2147483648 a +2147483647   |
| Smallint         | Representa un número entero que no contiene parte fraccionaria. Su precisión nunca será mayor que la de un Integer. Es un valor de 16 bits entre -32768 y +32767  |
| Number(1,d)      | Almacena datos de tipo numérico, siendo "l" la longitud y "d" el número de dígitos decimales  |
| Raw(tamaño)      | Datos binarios puros con una longitud máxima de 2000 bytes. Sirven para almacenar datos de tipo binario como sonido e imágenes digitalizadas.   |



## Definición de Datos

- **Restricciones de columnas**

**NOT NULL.** La columna no permitirá valores nulos.

**CONSTRAINT.** Permite asociar un nombre a una restricción.

**DEFAULT valor.** La columna tendrá un valor por defecto. El SGBD utiliza este valor cuando no se especifica un valor para dicha columna.

**PRIMARY KEY.** Permite indicar que esta columna es la clave primaria.

**REFERENCES.** Es la manera de indicar que este campo, es clave ajena y hace referencia a una clave candidata de otra tabla. Esta **foreign key** es sólo de una columna.

**UNIQUE.** Obliga a que los valores de una columna tomen valores únicos (no puede haber dos filas con igual valor). Se implementa creando un índice para dicha(s) columna(s).

**CHECK (condición).** Permite indicar una condición que debe de cumplir esa columna.



## Definición de Datos

- **Restricciones de tablas**

**PRIMARY KEY (columna1, columna2...).** Permite indicar las columnas que forman la clave primaria.

**FOREIGN KEY (columna1, columna2....) REFERENCES NombreTabla.** Indica las columnas que son clave ajena referenciando a una clave candidata de otra tabla.

**UNIQUE (columna1, columna2...).** El valor combinado de una o varias columnas es único.

**CHECK (condición).** Permite indicar una condición que deben cumplir las filas de la tabla. Puede afectar a varias columnas.



## Definición de Datos

- La cláusula **Foreign Key** tiene unas opciones optativas que se explican a continuación:
  - Tratamiento de nulos: Cómo debe tratar el SGBD un valor NULL en una o más columnas de la clave ajena, cuando lo compare con las filas de la tabla padre
  - Modo de borrado: Para determinar la acción que se debe realizar cuando se **elimina** una fila referenciada, se debe utilizar una **regla de supresión opcional** para la relación (CASCADE, SET NULL, SET DEFAULT, NO ACTION).
  - Modo de modificación: Una regla de actualización para la relación, que determina la acción que se debe realizar cuando se **modifica** la clave candidata de la fila referenciada (CASCADE, SET NULL, SET DEFAULT, NO ACTION).



## Definición de Datos

- Ejemplos

***areas(codigo, nombre, departamento)** (código es la clave primaria)*

***departamentos(codigo\_dpto, nombre)** (codigo\_dpto es la clave primaria)*

*La tabla áreas tiene una clave ajena*

***areas.departamento → departamentos***



## Definición de Datos

---

```
CREATE TABLE areas
```

```
(  
  codigo char(3) not null,  
  nombre char(55) not null,  
  departamento char(3) not null,  
  Primary key(codigo),  
  Foreign key(departamento) REFERENCES departamentos  
  ON DELETE SET NULL ON UPDATE CASCADE);
```

- ON DELETE Set Null Significa que si se borra algún departamento de la tabla departamentos el campo departamento de las filas de la tabla areas que le reverenciaban se pone como Null.
- ON UPDATE CASCADE Significa que si se modifica el código\_dpto de una fila de la tabla departamento, también se modificara en las filas de la tabla áreas que le referencian.



## Definición de Datos

---

```
CREATE TABLE departamentos
```

```
(  
  codigo_dpto char(3) not null,  
  nombre char(40) not null,  
  Primary key(codigo_dpto)  
);
```



## Definición de Datos

### **Renombrar una tabla**

```
RENAME TABLE <nombre tabla existente> TO <nuevo nombre tabla>
```

### **Eliminar una tabla de la base de datos**

```
DROP TABLE <nombre tabla> [CASCADE, RESTRICT]
```

Ejemplos:

```
DROP TABLE DEPARTAMENTOS CASCADE
```

(La tabla se borra, así como las posibles restricciones relativas a esta tabla)

```
DROP TABLE DEPARTAMENTOS RESTRICT
```

(La tabla se borra sólo si no se hace referencia a ella en ninguna restricción, p.e. en la definición de claves ajenas)



## Definición de Datos

### **Modificar una tabla**

La sentencia **Alter Table** se utiliza para cambiar una tabla existente. Dentro de la tabla podemos **Add** (añadir) o **Drop** (borrar) columnas y restricciones (PRIMARY KEY, FOREIGN KEY, UNIQUE, CHECK, CONSTRAINT).

```
ALTER TABLE <nombre tabla>
```

```
{ ADD <nombre columna nueva> <tipo de dato> [NOT NULL]
```

```
  MODIFY <nombre columna> [DEFAULT | DROP DEFAULT] valor
```

```
  DROP <nombre columna> [CASCADE | RESTRICT]
```

```
  ADD [PRIMARY KEY (nombre columna) |
```

```
  FOREIGN KEY (nombre columna) REFERENCES nombre_tabla |
```

```
  UNIQUE (nombre columna) | CHECK (condición)
```

```
  DROP CONSTRAINT nombre-restricción [CASCADE | RESTRICT]
```



## Definición de Datos

- Ejemplos

Agregar a la tabla areas el campo Responsable de tipo char(30)

```
alter table areas
```

```
ADD responsable char(30) not null;
```

Modificar el campo nombre de la tabla departamentos a char(50)

```
alter table departamentos
```

```
MODIFY nombre char(50);
```



## Definición de Datos

***Sinónimos***: Son nombres que puede utilizarse como sustituto o alias del nombre real de una tabla. Puede ser útil para simplificar algunas expresiones.

### ***Crear un sinónimo***

```
CREATE SYNONYM <nombre sinónimo> FOR <nombre tabla>
```

```
create synonym are for areas;
```

```
create synonym dep for departamentos;
```

### ***Borrar un sinónimo***

```
DROP SYNONYM <nombre sinónimo>
```

```
drop synonym are
```



# Definición de Datos

## ***Gestión de Dominios (SI en SQL – NO en ORACLE)***

Los dominios se usan como tipos de datos en la definición de columnas al crear tablas. Permiten definir los valores aceptados en las columnas, así como la definición de valores por defecto.

### **Creación de Dominios:**

```
CREATE DOMAIN <Nombre Dominio> AS <Tipo Datos>
[DEFAULT <valor defecto>]
[{{[CONSTRAINT <nombre restricción>] CHECK (<condición check>)}}...];
```

Ejemplos:

```
CREATE DOMAIN SeisDigitos AS CHAR(6) DEFAULT '000000';
```

```
CREATE DOMAIN SeisDigitos AS CHAR(6) DEFAULT '000000' CHECK (VALUE IS NOT NULL)
CHECK (CHAR_LENGTH(TRIM(VALUE)) = 6 AND VALUE BETWEEN '000000' AND '500000');
```



# Definición de Datos

## ***Gestión de Dominios***

### **Borrado de Dominios:**

```
DROP DOMAIN <Nombre Dominio>
```

### **Modificado de Dominios:**

```
ALTER DOMAIN <Nombre Dominio> { SET DEFAULT <Valor Defecto | DROP DEFAULT }
```

```
ALTER DOMAIN <Nombre Dominio> ADD <Restricción Dominio>
```

```
ALTER DOMAIN <Nombre Dominio> DROP CONSTRAINT <Nombre Restricción> [RESTRICT |
CASCADE ]
```



## Diccionario de Datos

---

- Conjunto de tablas que contiene las definiciones de los elementos de las bases de datos creadas por el usuario.
- Estas tablas son creadas y actualizadas directamente por el SGBD. Las tablas del diccionario de datos se pueden consultar con SELECTs
- Obtener tablas existentes en la base de datos:  
**SELECT TABLE\_NAME FROM USER\_TABLES**
- Descripción completa de una tabla existente en la base de datos:  
**DESCRIBE nombre\_tabla | DESC nombre\_tabla**