

Alumno: Miguel Ángel Brea Rivera

Tema 3

1. Modelo de datos

Un **modelo de datos** es un lenguaje utilizado para la descripción de una Base de Datos. Mediante este lenguaje podemos realizar:

- Las descripciones de las **estructuras de los datos**, es decir, los tipos de datos y relaciones que existen entre ellos
- Las **restricciones de integridad**, que hace referencia a las condiciones que deben cumplir los datos, según las necesidades de nuestro modelo basado en la realidad que representen.
- Las operaciones de **manipulación de los datos**, como el insertado, el borrado y la modificación de datos.

Se debe clasificar los modelos según su nivel de abstracción, podemos distinguir los siguientes:

- Los **modelos de datos conceptuales** que describen las estructuras de datos y las restricciones de integridad, además están orientados a representar los elementos que intervienen y sus relaciones. Ejemplo, Modelo Entidad-Relación.
- Los **modelos de datos lógicos** se centran en las operaciones y se implementan en algún sistema gestor de BD. Ej.: Modelo Relacional.
- Los **modelos de datos físicos**, son estructuras de datos a bajo nivel, implementadas dentro del propio sistema gestor de BD.

Existen dos sublenguajes en los modelos de datos:

- **Lenguaje de Definición de Datos o DDL (Data Definition Language).**
- **Lenguaje de Manipulación de Datos o DML (Data Manipulation Language).**

2. Terminología del modelo relacional

El **modelo relacional** se trata de un modelo lógico que establece una estructura sobre los datos, independientemente de cómo se almacenen.

2.1. *Relación o tablas. Tuplas y dominios.*

En la relación en una tabla entre filas y columnas, destacamos los siguientes conceptos:

- **Atributo:** se trata del nombre de cada dato que se registra en la relación o tabla. Se trataría de cada una de las columnas de la tabla.
- **Tuplas:** consiste en cada uno de los elementos de la relación, que se representan por las filas y deben de cumplir que cada tupla sea única.
- **Valor:** es la intersección entre las columnas y las filas, deben estar dentro de un dominio de valores previamente establecido.

Los valores de los **dominios** deben ser atómicos, tener un nombre que los definan lógicamente, un tipo de dato y un formato.

2.2. **Grados y cardinalidad**

- **Grados:** es el número de atributos o columnas de una tabla o relación.
- **Cardinalidad:** es el número de tuplas o filas de una tabla o relación.

2.3. **Sinónimos**

Nomenclatura relacional	Nomenclatura tabla	Nomenclatura fichero
relación	tabla	Fichero
tupla	fila	Registro
atributo	columna	Campo
grado	Numero columna	Numero campo
cardinalidad	Numero fila	Numero registro

3. **Relaciones. Características de las relaciones.**

Destacamos las siguientes características entre las relaciones que se deben de conocer a la perfección:

- Los nombres de las tablas son únicos. No se repiten.
- Cada atributo toma solo un valor en cada tupla.
- No puede existir dos tuplas iguales.
- El orden de las tuplas y atributos no importan.
- Todos los datos de un atributo deben pertenecer al mismo dominio.

3.1. **Tipos de relaciones**

Existen dos tipos:

- Persistentes: únicamente pueden ser borrados por los usuarios, como las bases (conjunto de tuplas y atributos); las vistas (que almacenan una definición de consulta); instantáneas (almacenan los datos que se muestran en la consulta realizada).
- Temporales: son eliminados automáticamente por el sistema.

4. Tipos de datos

Tenemos que asignar un tipo de dato a cada atributo con las siguientes características:

- Deben de tener un nombre que estén relacionados con los datos a los que se vayan a convertir.
- Deben tener asociados un tipo de dato.

Los tipos de datos más frecuentes son: texto, numéricos, fechas, booleanos, memo (texto largo), moneda (número que representa cantidad de dinero), objeto OLE (gráficos).

5. Claves

Se trata de un atributo o un conjunto de atributos que identifiquen de modo único las tuplas de una relación. A ese atributo o conjunto de atributos se le denominan **superclaves**, podemos concluir, como dos tuplas no pueden repetirse, entonces toda la fila de una relación se puede considerar como superclave.

Se trabaja con estos tipos de claves:

- Claves candidatas.
- Claves ajenas.
- Claves primarias.
- Claves alternativas.

5.1. *Claves candidatas. Claves primarias. Claves alternativas.*

- **Claves candidatas** es el conjunto de atributos que identifiquen de manera única cada fila de una tabla, se caracteriza por su unicidad (no se repiten los datos en dos tuplas) e irreductibilidad (si elimina uno de los atributos deja de ser únicas) . Cada tabla debe tener al menos una clave candidata, aunque puede haber más de una. Si una clave candidata tiene más de un atributo se denomina **clave compuesta**.
- **Clave primaria** de una relación es aquella clave candidata que se escoge para identificar sus tuplas de modo único.
- **Claves alternativas** son aquellas claves candidatas que no han sido elegidas como primarias.

5.2. *Claves externa, ajena o secundaria.*

- Una **clave ajena, también llamada externa o secundaria**, es un atributo o conjunto de atributos de una relación cuyos valores coinciden con los valores de la clave primaria de alguna otra relación. Este tipo de clave tienen por objetivo establecer una conexión con la clave primaria que referencian. Entonces, los valores de una clave ajena deben estar presentes en la clave primaria correspondiente, o bien deben ser valores nulos.

6. Índices. Características.

- **Índice** es una estructura de datos que permite acceder a diferentes filas de una misma tabla a través de un campo o campos. Con esto se tiene la ventaja de que se permite un acceso mucho más rápido a los datos.
- Muy usado para **consultas frecuentes** a un rango de filas o una fila de una tabla.
- Los cambios en los datos de las tablas (agregar, actualizar o borrar filas) son incorporados automáticamente a los índices con transparencia total.
- Son independientes, lógica y físicamente de los datos, pueden ser creados y eliminados en cualquier momento, sin afectar a las tablas ni a otros índices.
- No es operativo o recomendable, crear un índice para cada columna, pero si tiene sentido crear un índice para un grupo de columnas.
- Al crear índices, las operaciones de modificar o agregar se ralentizan.
- Si se eliminan un índice, el acceso a los datos pueden ser más lento.

7. El valor null. Operaciones con este valor.

Defino las siguientes características con el valor null:

- Independientemente del dominio al que pertenezca un campo, éste puede tomar un valor especial denominado NULO (*NULL* en inglés) que designará la ausencia de dato.
- Cuando trabajamos con claves secundarias el valor nulo indica que la tupla o fila no está relacionada con ninguna otra tupla o fila. Este valor NULO es común a cualquier dominio.
- No es lo mismo valor nulo que espacio en blanco.
- Tampoco será lo mismo valor nulo que el valor cero.
- Este valor se va a utilizar con frecuencia en las BD y es imprescindible saber cómo actúa cuando se emplean operaciones lógicas sobre ese valor.

True && null	null
False && null	false
True null	true
False null	null
!null	null

8. Vistas

Una vista no es más que una consulta almacenada a fin de utilizarla tantas veces como se desee. Una vista no contiene datos sino la instrucción **SELECT** necesaria para crear la vista, eso asegura que los datos sean coherentes al utilizar los datos almacenados en las tablas.

Las vistas se emplean para:

- Realizar consultas complejas más fácilmente, ya que permiten dividir la consulta en varias partes
- Proporcionar tablas con datos completos
- Utilizar visiones especiales de los datos
- Ser utilizadas como tablas que resumen todos los datos
- Ser utilizadas como cursores de datos en los lenguajes procedimentales (como PL/SQL)

9. Usuarios, Roles, Privilegios

Usuario: es un conjunto de permiso que se le aplican a una conexión de Base de Dato. Tiene las siguientes funciones:

- **Ser el propietario de ciertos objetos** (tablas, vistas, etc.).
- Define una cuota de almacenamiento
- Define el tablespace por defecto para los objetos de un usuario en Oracle
- Realiza las copias de seguridad

Privilegio: es un permiso dado a un usuario para que realice ciertas operaciones que pueden ser de dos tipos, de sistemas o sobre objeto.

Rol: es un agrupación de permisos de sistema y de objetos.

10. SQL

SQL (Structured Query Language) es un lenguaje de programación diseñado para almacenar, manipular y recuperar datos almacenados en bases de datos relacionales. La primera encarnación de SQL apareció en 1974, cuando un grupo de IBM desarrolló el primer prototipo de una base de datos relacional. Relational Software (luego se convirtió en Oracle) lanzó la primera base de datos relacional comercial.

Existen estándares para SQL. Sin embargo, el SQL que puede utilizarse en cada uno de las principales RDBMS actuales viene en distintas formas. Esto se debe a dos razones:

- 1) el estándar SQL es bastante complejo, y no es práctico implementar el estándar completo
- 2) Cada proveedor de base de datos necesita una forma de diferenciar su producto de otros.

SQL enumera los comandos SQL normalmente utilizados, y se divide en las siguientes secciones:

- **Comandos SQL:** Las instrucciones SQL básicas para almacenamiento, recuperación y manipulación de datos en una base de datos relacional.
- **Manipulación de Tabla:** Cómo se utilizan las instrucciones SQL para administrar las tablas dentro de una base de datos.
- **SQL Avanzado:** Comandos SQL avanzados.
- **Sintaxis SQL:** Una página única que enumera la sintaxis para todos los comandos SQL en esta guía de referencia.

10.3. Tipos de Sentencias

PLSQL:

- DECLARE (Define un cursor para una consulta)
- OPEN (Abre un cursor para recuperar resultados de una consulta)
- FETCH (Recupera una fila de resultados de consulta)
- CLOSE (Cierra un cursor)

DML (Manipulación de datos):

- SELECT (recuperar datos)
- INSERT (inserta datos)
- DELETE (suprime filas de datos)
- UPDATE (actualiza datos)

DDL (Definición de datos)

- CREATE TABLE (añade una nueva tabla)
- DROP TABLE (suprime una tabla)
- ALTER TABLE (modifica una tabla)
- CREATE VIEW (añade una vista)
- DROP VIEW (suprime una vista)
- CREATE INDEX (construye un índice para una columna)
- DROP INDEX (suprime un índice de una columna)
- CREATE SYNOYM (define un alias para un nombre)
- DROP SYNOYM (suprime un alias)

DCL (Control de Acceso):

- GRANT (concede privilegios de acceso a usuarios)
- REVOKE (suprime privilegios de acceso a usuarios)
- COMMIT (finaliza la transacción actual)
- ROLLBACK (interrumpir la transacción actual)