



CURSO

SOFTWARE TESTING

UNIDAD 9 – Bases de Datos - SQL

PRESENTACIÓN

En esta unidad repasaremos nociones básicas de SQL

TEMARIO

1. – Base de datos
2. – SQL
3. – Comandos.
4. – Ejemplos.

Base de Datos

Es una colección de datos organizada en un formato estructurado que es definido como *metadatos* que describe esa estructura. Puede pensar en los metadatos como información sobre los datos almacenados, que define cómo se almacenan éstos en una base de datos.

Existen bases de datos

- Relacionales
- No Relacionales

Base de datos Relacional

- Una *entidad* es una persona, lugar, cosa, evento o concepto sobre el cual los datos son recolectados, como un artista, un libro o una transacción de ventas. Cada entidad comprende uno o más atributos (columnas).
- Los datos se almacenan en una relación en tuplas (filas). Una *tupla* es un conjunto de datos cuyos valores hacen una instancia de cada atributo definido por esa relación. Cada tupla representa un registro.

Tabla (Entidad)

Diagram illustrating a table structure and its data rows.

The table has four columns: **Alu_Legajo**, **Alu_Nombre**, **Alu_Apellido**, and **Alu_FecIngreso**.

The data rows are:

Alu_Legajo	Alu_Nombre	Alu_Apellido	Alu_FecIngreso
100100	Claudio	Hernández	1/1/2021
100101	Pedro	Rodriguez	1/1/2020
100102	Raul	Ortiz	1/12/2020
100103	Ana	Soliz	1/1/1999
100104	Natalia	Gomez	20/5/2020

Annotations:

- A blue arrow points from the label **Columna (Atributo)** to the **Alu_Apellido** column.
- A red arrow points from the label **Fila (Registro Tupla)** to the row containing **100101 Pedro Rodriguez 1/1/2020**.

Base de datos No Relacional (ejemplo)

MongoDB es una base de datos orientada a documentos. Esto quiere decir que en lugar de guardar los datos en registros, guarda los datos en documentos. Estos documentos son almacenados en una representación similar a JSON.

Una de las diferencias más importantes con respecto a las bases de datos relacionales, es que **no es necesario seguir un esquema**. Los documentos de una misma colección (concepto similar a una tabla de una base de datos relacional), pueden tener esquemas diferentes.

Imaginemos que tenemos una colección a la que llamamos Personas. Un documento podría almacenarse de la siguiente manera:

```
{ Nombre: "Pedro",  
  Apellidos: "Martínez Campo",  
  Edad: 22,  
  Aficiones: ["fútbol", "tenis", "ciclismo"],  
  Amigos: [ { Nombre:"María", Edad:22 }, { Nombre:"Luis", Edad:28 } ] }
```


Base de datos No Relacional (ejemplo)

En la misma colección podríamos guardar un documento como este:

```
{ Nombre: "Luis", Estudios: "Administración y Dirección de Empresas",  
Amigos:12 }
```

Este documento no sigue el mismo esquema que el primero. Tiene menos campos, algún campo nuevo que no existe en el documento anterior e incluso un campo de distinto tipo.

Esto que es algo impensable en una base de datos relacional, es algo totalmente válido en **MongoDB**.

Modelo Relacional

- Un identificador único es un atributo o conjunto de atributos que únicamente identifican cada fila de datos en una relación

Alu_Legajo	Alu_Nombre	Alu_Apellido	Alu_FecIngreso
100100	Claudio	Hernández	1/1/2021
100101	Pedro	Rodriguez	1/1/2020
100102	Raul	Ortiz	1/12/2020
100103	Ana	Soliz	1/1/1999
100104	Natalia	Gomez	20/5/2020

CodigoLibro	NombreLibro	GeneroLibro
45444	Bella y Bestia	Infantil
45333	Cuentos de Terror	Terror
55662	10 indiecitos	Misterio
104055	Detective Forense	Policial
40566	Lugar del Crimen	Policial

Modelo Relacional

Fecha	Tipo Cbte	PtoVenta	NumeroCbte	Estado	Vendedor	Cliente
01/03/2021	FAC A	00001	500	Pago	01	332
22/05/2021	FAC A	00001	550	CC	02	441
05/06/2021	FAC B	00001	8	Pago	02	983
01/02/2021	FAC A	00001	490	Pago	01	42

Renglon	Tipo Cbte	PtoVenta	NumeroCbte	Articulo	Cant	Precio Unitario
1	FAC A	00001	500	Pantalon	1	3000
2	FAC A	00001	500	Camisa	1	1500
3	FAC A	00001	500	Cinto	2	800
4	FAC A	00001	500	Pantalon	1	3000

Modelo Relacional

- Cada atributo de una tupla contiene sólo un valor.
- Cada tupla en una relación contiene el mismo número de atributos.
- Cada tupla es diferente, lo que significa que la combinación de los valores de todos los atributos de una tupla dada no puede ser como ninguna otra tupla en la misma relación.

Legajo	Nombre	Apellido	Fec.Nac	Carrera	Materias
234234	Claudio	Hernández	01/06/1968	Sistemas	Matemática 1, Programación 1
121222	Luis	Lopez	01/07/1999	Sistemas	Programación 1, Computación 1
455333	Silvana	Carrizo	04/08/2000	Sistemas	Matemática 1
455333	Silvana	Carrizo	04/08/2000	Sistemas	Matemática 1

Modelo Relacional

- Si un atributo no es parte de la clave principal (identificador único) entonces depende de la clave principal.

Profesor	IDMateria	IDSede	NombreMateria	NombreSede
Claudio Hernández	01	UTN_1	Testing	Medrano
Pablo Castillo	02	UTN_1	Python	Medrano
Claudio Hernández	01	V	Testing	Virtual

Modelo Relacional

Curso	
PK	<u>Profesor</u> <u>ID Materia</u> <u>Id Sede</u>

Materias	
PK	<u>Id Materia</u>
	Nombre Materia

Sedes	
PK	<u>Id_Sede</u>
	Nombre Sede

Profesor	IDMateria	IDSede	NombreMateria	NombreSede
Claudio Hernández	01	UTN_1	Testing	Medrano
Pablo Castillo	02	UTN_1	Python	Medrano
Claudio Hernández	01	V	Testing	Virtual

SQL (Lenguaje de Consulta Estructurada)

- **Lenguaje de definición de datos (DDL, Data Definition Language)**
 - Ej: CREATE TABLE, DROP TABLE, ALTER TABLE
- **Lenguaje de control de datos (DCL, Data Control Language)**
 - Ej. GRANT, REVOKE
- **Lenguaje de manipulación de datos (DML, Data Manipulation Language)**
 - Ej. SELECT, INSERT, DELETE, UPDATE

Bases de datos - Productos en mercado

- SQL Server
- Oracle
- MariaDB
- MySQL
- PostgreSQL

SQL - Comandos

- Insertar datos en tablas (INSERT)
- Eliminar datos de tablas (DELETE)
- Modificar datos de tablas (UPDATE)

SQL - Insert

- INSERT INTO nombre_tabla (columna1, columna2, columna3,...)
VALUES (valor1, valor2, valor3, ...)
- INSERT INTO nombre_tabla VALUES (valor1,valor2,valor3)

EJEMPLO:

```
INSERT INTO Alumnos(Alu_Legajo, Alu_Nombre, Alu_Apellido, Alu_FecIngreso) VALUES  
(100150,'Rafael','Guido','10/10/2020')
```

```
INSERT INTO Alumnos VALUES (100150,'Rafael','Guido','10/10/2020')
```

SQL - Update

- UPDATE nombre_tabla
SET columna1 = valor1, columna2 = valor2
WHERE columna3 = valor3

EJEMPLO:

```
UPDATE Alumnos SET Alu_Apellido = 'Rodríguez' WHERE Alu_Legajo = '100101'
```

SQL - Delete

- DELETE FROM nombre_tabla WHERE nombre_columna = valor

EJEMPLO:

```
DELETE Alumnos WHERE Alu_Legajo='100150'
```

SQL - Select

Clausulas:

- Where
- Order by
- Group by
- Funciones (Count, max, min, sum)
- Having
- Inner Join
- Left Join, Right Join
- Cast
- In

Select * from Clientes

PREGUNTAS



Ejercicio en clase

Practica: Crear tablas de Sistema de Alumnos

Usamos el diseñador de tablas de Management Studio para crear tablas y relaciones



Fin de a Unidad 9.