

M10 Bases de Datos (ITC) FIN A

ACTIVIDAD 1

|  |  |
| --- | --- |
| **Tutor:** | **Luís Sierra Betancourt** |
| **Estudiante:** | **José Ramón Ibáñez Posadas** |
| **Matricula:** | **BNL098377** |

|  |  |
| --- | --- |
| Monterrey, Nuevo León | Lunes, 30 de Septiembre de 2024 |

INTRODUCCIÓN

Las bases de datos son fundamentales en el mundo tecnológico actual, ya que permiten almacenar, organizar y gestionar grandes volúmenes de información de manera eficiente. Su origen se remonta a los primeros sistemas de archivos físicos, que fueron evolucionando con el tiempo hacia sistemas electrónicos más complejos, dando lugar a modelos jerárquicos, en red y, finalmente, al modelo relacional propuesto por Edgar Codd en la década de 1970. Hoy en día, el lenguaje SQL (Structured Query Language) es uno de los más utilizados para la interacción con bases de datos relacionales. SQL nos permite manipular datos (DML), definir estructuras (DDL), controlar permisos (DCL) y gestionar transacciones (TCL). Los dominios de SQL abarcan diversos tipos de datos como números, texto, fechas y objetos binarios, lo cual permite almacenar información de una manera adecuada y precisa según la naturaleza de los datos. Esta combinación de elementos hace que las bases de datos sean herramientas imprescindibles para cualquier empresa, aplicación o investigación que dependa de la información como recurso principal.

.

DESARROLLO

A diagram with text and a blue rectangle

Description automatically generatedMAPA MENTAL

LIGA MAPA MENTAL

<https://mm.tt/app/map/3454231343?t=Oe6N16IQZ9>

GLOSARIO

**1. Tipos de Dominios en SQL**

* **Dominio**: Un dominio en SQL se refiere al **conjunto de valores permitidos** para una columna específica en una base de datos, definido por el tipo de dato.
* **Tipos de Datos Numéricos**:
  + **INT**: Tipo de dato para **números enteros**. Ejemplo: 1, 100, -50.
  + **DECIMAL** (o **NUMERIC**): Tipo de dato para **números decimales**, con precisión y escala especificada. Ejemplo: 10.25.
  + **FLOAT/REAL**: Tipo de dato para **números de punto flotante**, que permite valores con decimales pero con una precisión menos exacta.
* **Tipos de Datos de Caracteres**:
  + **CHAR(n)**: Almacena una **cadena de longitud fija**. Ejemplo: 'ABC'.
  + **VARCHAR(n)**: Almacena una **cadena de longitud variable**, con un límite. Ejemplo: 'Hola Mundo'.
  + **TEXT**: Tipo de dato para **cadenas de texto grandes**.
* **Tipos de Datos de Fecha y Hora**:
  + **DATE**: Almacena una **fecha** (año, mes, día). Ejemplo: '2024-09-30'.
  + **TIME**: Almacena la **hora** (hora, minutos, segundos). Ejemplo: '12:34:56'.
  + **DATETIME**: Almacena tanto la fecha como la hora.
* **Tipos de Datos Lógicos**:
  + **BOOLEAN**: Almacena **valores lógicos** (verdadero o falso). Ejemplo: TRUE, FALSE.
* **Otros Tipos de Datos**:
  + **BLOB**: Tipo de dato para almacenar **objetos binarios grandes**, como imágenes o videos.
  + **ENUM**: Tipo de dato que define un **conjunto de valores posibles** para una columna. Ejemplo: 'pequeño', 'mediano', 'grande'.

**2. Elementos del Lenguaje SQL**

* **DML (Lenguaje de Manipulación de Datos)**:
  + **SELECT**: Utilizado para **consultar datos** de una tabla. Ejemplo: SELECT \* FROM empleados;.
  + **INSERT**: Utilizado para **insertar nuevos registros** en una tabla. Ejemplo: INSERT INTO empleados (nombre, edad) VALUES ('Juan', 30);.
  + **UPDATE**: Utilizado para **modificar registros** existentes. Ejemplo: UPDATE empleados SET edad = 31 WHERE nombre = 'Juan';.
  + **DELETE**: Utilizado para **eliminar registros**. Ejemplo: DELETE FROM empleados WHERE nombre = 'Juan';.
* **DDL (Lenguaje de Definición de Datos)**:
  + **CREATE**: Utilizado para **crear objetos** como bases de datos, tablas, índices. Ejemplo: CREATE TABLE empleados (id INT, nombre VARCHAR(50), edad INT);.
  + **ALTER**: Utilizado para **modificar la estructura** de un objeto existente. Ejemplo: ALTER TABLE empleados ADD salario DECIMAL(10,2);.
  + **DROP**: Utilizado para **eliminar objetos** como tablas o bases de datos. Ejemplo: DROP TABLE empleados;.
* **DCL (Lenguaje de Control de Datos)**:
  + **GRANT**: Utilizado para **conceder permisos** a los usuarios. Ejemplo: GRANT SELECT ON empleados TO usuario1;.
  + **REVOKE**: Utilizado para **revocar permisos** concedidos anteriormente. Ejemplo: REVOKE SELECT ON empleados FROM usuario1;.
* **TCL (Lenguaje de Control de Transacciones)**:
  + **COMMIT**: Utilizado para **guardar permanentemente** los cambios realizados en una transacción. Ejemplo: COMMIT;.
  + **ROLLBACK**: Utilizado para **revertir** los cambios realizados en una transacción. Ejemplo: ROLLBACK;.
  + **SAVEPOINT**: Permite crear un **punto de guardado** en una transacción, al que se puede volver con un ROLLBACK.

CONCLUSIÓN

En conclusión, las bases de datos han recorrido un largo camino desde sus primeros orígenes como sistemas de archivos físicos hasta convertirse en sofisticados sistemas relacionales que impulsan la gestión de información moderna. Su evolución ha permitido mejorar la organización, integridad y acceso a los datos, haciendo posible que sectores como el empresarial, científico, educativo y tecnológico puedan tomar decisiones basadas en información precisa y organizada. El lenguaje SQL se ha establecido como el estándar para manipular y definir datos en estos sistemas, proporcionando una estructura sólida a través de sus tipos de dominios y comandos. Desde la manipulación de registros hasta el control de transacciones y permisos, SQL facilita la administración de la información de una manera integral y eficiente. La capacidad de definir tipos de datos adecuados y la posibilidad de utilizar distintos comandos, permiten a los usuarios tener un control completo sobre sus bases de datos, maximizando el valor de la información. Así, el conocimiento de bases de datos y SQL se ha vuelto esencial para cualquier persona que desee involucrarse en el mundo de la tecnología y el manejo de la información.

BIBLIOGRAFÍA

A screenshot of a computer

Description automatically generated