Trabajo 3 de Base De Datos 1

Gustavo Ángel Ortega Pérez

Hennys Sebastian Tous De Avila

Universidad de Cartagena

**Texto, Carta

Descripción generada automáticamente**

Programa: Ingeniería de Software

Asignatura: Estructura de Datos

Tutor: Jhon Arrieta Arrieta

Cartagena de Indias – agosto 3 del año 2022

**Tabla de Contenido**

1. Introducción.
2. Objetivo.
3. Justificación.
4. Resultados
5. Síntesis y argumentos individuales.
6. Bibliografía

**Introducción**

En el presente documento vamos a explorar el uso de un Gestor de Bases de Datos, haciendo uso de sentencias DML para la creación, eliminación y modificación de registros teniendo en cuenta las tablas establecidas en la actividad anterior.

**Objetivos**

* Conocer los distintos Gestores para manejar bases de datos e interactuar con su interfaz.
* Hacer uso de las Sentencias DML para operar con datos dentro de cada una de las tablas.
* Observar el comportamiento de los datos en tablas intermedias o débiles al momento de correr sentencias DML.

**Justificación**

Principalmente hemos decidido tomar riendas de este documento para ahondar nuestros conocimientos dentro del uso de Gestores para bases de Datos, teniendo en cuenta sentencias DML.

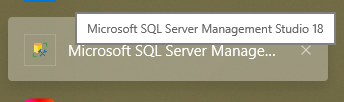
Esto con la finalidad de aprender a cómo manejar registros mediante sentencias DML de tipo Insert, Update y Delete,

**Resultados**

A continuación con el desarrollo de esta 3ra actividad ustedes como grupo deben demostrar que han estudiado y practicado lo suficiente el tema de **Sentencias DML de tipo Insert, Update y Delete**

1. Usar el Cliente Manejador de BD Gráfico para establecer conexión con el respectivo Motor de BD.

* Ejecutamos Microsoft SQL Server Management



* Una vez entremos, nos va a aparecer una ventana en donde debemos colocar el login para ingresar, en este caso usaremos la autenticación del sistema y presionaremos connect.

Interfaz de usuario gráfica

Descripción generada automáticamente

1. Utilizar el archivo **backup\_bd\_grupo\_xyz.sql** para realizar el proceso de restauración de la bd.

* Una vez dentro de Microsoft SQL Server Management vamos a dar click derecho en la carpeta

1. Verificar que se han creado todas las tablas de la **bd\_grupo\_xyz**

Tabla

Descripción generada automáticamente

1. Confirmar si las tablas están conectadas mediante las respectivas llaves foráneas, por ejemplo: volviendo a generar el Modelo Relacional o revisando cada tabla para ver si contiene o no sus llaves foráneas.

Interfaz de usuario gráfica, Tabla

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Diagrama

Descripción generada automáticamente

1. Volver a crear el usuario para **bd\_grupo\_xyz** con las mismas credenciales utilizadas en el punto #5 de la 2da Actividad.

Para el Login

Diagrama, Tabla

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Para el usuario

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media

Interfaz de usuario gráfica, Aplicación, Correo electrónico

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

1. Desconectarse del Motor de BD y volver a conectarse utilizando las credenciales del usuario **bd\_grupo\_xyz.**

Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

1. Mostrar las bases de datos del usuario **bd\_grupo\_xyz**

Texto

Descripción generada automáticamente

De ahora en adelante, solo cuando se le pida (INSTRUCCIÓN SQL o GUI) ustedes deben dar respuesta al desarrollo de las operaciones SQL de dos maneras:

1. **INSTRUCCIÓN SQL**: Se debe utilizar la función de editor de SQL suministrada por el Cliente Gráfico MBD, e ingresar y ejecutar la respectiva Instrucción SQL.
2. **GUI**: Usar la función netamente gráfica (generalmente un formulario con campos y botones para ingresar, cambiar o eliminar los datos de las columnas y registros de cada tabla)

**Observaciones**:

* Para caso del modo **INSTRUCCIÓN SQL**, en cada ejercicio deben escribir la instrucción SQL utilizada, tomar recortes de pantalla de la zona donde fue ingresada, ejecutada y el resultado de la instrucción.
* Para el caso de **GUI**, en cada ejercicio se debe tomar pantallazo de la zona donde se ingresan los datos y se da la respectiva orden para ejecutar la instrucción.

**NOTA**: solo tomar recortes de pantalla de la zona interesa mostrar, evitar tomar pantallazos de toda la pantalla porque evita visualizar correctamente el resultado de cada ejercicio.

Tomar alguna de las tablas de la base de datos bd\_grupo\_xyz y realizar las siguientes operaciones:

1. Insertar al menos 2 registros usando el formato de **Inserción total** de SQL.

DESCRIPCIÓN DETALLADA DE LOS PASOS PARA RESOLVER ESTE PUNTO

**INSERT INTO Alumno VALUES ('001', 'Alejandro', 4);**

**INSERT INTO Alumno VALUES ('002', 'Rodolfo', 7);**

**Texto

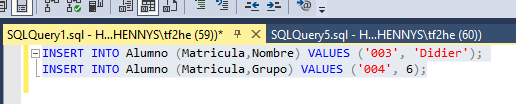
Descripción generada automáticamente con confianza media**

Para hacer inserción total simplemente insertamos los datos en el orden en el que están las columnas de dicha tabla.

1. Insertar al menos 2 registros usando el formato de **Inserción parcial** de SQL

**INSERT INTO Alumno (Matricula,Nombre) VALUES ('003', 'Didier');**

**INSERT INTO Alumno (Matricula,Grupo) VALUES ('004', 6);**

****

Para hacer inserción parcial, debemos especificar en que columnas queremos insertar los datos. Las columnas excluidas deben aceptar null o tener un valor por defecto para que estas no generen errores.

1. Insertar al menos 5 registros usando el formato de **Inserción masiva** de SQL

**INSERT INTO Alumno**

**VALUES ('005','Angel', 1),**

**('006','Roberto',2),**

**('007','Fido', 2),**

**('008','Carlos',3),**

**('009','Henry', 7),**

**('010','Armando',8);**

**Texto

Descripción generada automáticamente**

Para hacer inserción masiva usamos la misma inserción total pero esta vez separamos las filas de datos por comas, para así insertar mas de un registro en un solo INSERT

1. Mostrar todos los registros Insertado en la tabla:

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto

Descripción generada automáticamente**

Tabla

Descripción generada automáticamente con confianza media

Usamos la opción grafica de SQL Server para mostrar los primeros 1000 registros.

Podemos ver que las inserciones anteriores se encuentran en la tabla.

1. Cambiar o actualizar todos los datos de alguna de las columnas de la tabla

UPDATE Alumno Set Grupo = 1;



Usamos UPDATE en la tabla Alumno para cambiar toda la columna de Grupo a 1.

1. Mostrar todos los registros Insertado en la tabla:

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto

Descripción generada automáticamente**

Tabla

Descripción generada automáticamente con confianza media

Usamos la opción grafica de SQL Server para mostrar los primeros 1000 registros.

Podemos ver que los datos de la columna Grupo han sido cambiados a 1

1. Cambiar o actualizar los datos sólo un registro de la TablaX cuya clave primaria sea igual a Y o Z valor.

UPDATE Alumno Set Nombre = 'Citripio', Grupo = 2 where Nombre IS NULL;

**Interfaz de usuario gráfica, Texto

Descripción generada automáticamente con confianza media**

Usamos UPDATE para cambiar un registro donde el nombre sea NULL por Citripio y cambiamos su grupo a 2.

1. Mostrar todos los registros Insertado en la tabla:

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto

Descripción generada automáticamente**

Imagen que contiene Tabla

Descripción generada automáticamente

Usamos la opción grafica de SQL Server para mostrar los primeros 1000 registros.

Podemos ver que el nombre y el grupo del registro 4 han sido cambiados.

1. Cambiar o actualizar sólo los datos de aquellos registros de la TablaX que cumplan con Y **o** Z condición

**UPDATE Alumno**

**Set**

**Grupo = 5**

**where Nombre = 'Angel' or Matricula ='009';**

**Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente**

Usamos UPDATE y Where con el condicional or para decirle que cambie el grupo a 5 donde el registro tenga por nombre Angel o si la matricula es 009.

1. Mostrar todos los registros Insertado en la tabla:

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto

Descripción generada automáticamente**

Imagen que contiene Tabla

Descripción generada automáticamente

Usamos la opción grafica de SQL Server para mostrar los primeros 1000 registros.

Podemos ver que el grupo del registro 5 y 9 ha cambiado.

1. Cambiar o actualizar los datos sólo de aquellos registros de la TablaX que cumplan con Y **y**  Z condición

UPDATE Alumno

Set

Grupo = 8

where Nombre = 'Didier' and Matricula ='003';

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Usamos UPDATE y Where con el condicional and para decirle que cambie el grupo a 8 donde el registro tenga por nombre Didier y su matrícula sea 003.

1. Mostrar todos los registros Insertado en la tabla:

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto

Descripción generada automáticamente**

**Imagen que contiene Tabla

Descripción generada automáticamente**

Usamos la opción grafica de SQL Server para mostrar los primeros 1000 registros.

Podemos ver que el grupo del registro 3 ha cambiado.

1. Cambiar o actualizar a mayúsculas los datos de algunas columnas de la TablaX cuyos registros cumplan con alguna condición específica.

UPDATE Alumno

Set

Nombre = UPPER(Nombre)

where Matricula ='008' or Matricula ='005';

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza baja

Usamos UPDATE y Where con el condicional or para decirle que cambie a mayúsculas los nombres donde la matricula sea 008 o si la matricula es 005.

1. Mostrar todos los registros Insertado en la tabla:

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto

Descripción generada automáticamente**

Tabla

Descripción generada automáticamente con confianza media

Usamos la opción grafica de SQL Server para mostrar los primeros 1000 registros.

Podemos ver que los nombres de los registros 5 y 8 han cambiado a mayusculas.

1. Eliminar solo un registro de la TablaX de datos cuya llave primaria sea igual a Y valor

**DELETE FROM Alumno where Matricula = 8;**

****

Usamos DELETE FROM para especificar que queremos borrar los registros de la tabla Alumno donde la matricula sea 8 (o 008).

1. Mostrar todos los registros Insertado en la tabla:

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto

Descripción generada automáticamente**

Tabla

Descripción generada automáticamente con confianza baja

Usamos la opción grafica de SQL Server para mostrar los primeros 1000 registros.

Podemos ver que se ha borrado un registro.

1. Eliminar sólo aquellos registros de la TablaX que cumplan con Y **o**  Z condición

DELETE FROM Alumno where Grupo = 2 or Grupo = 8;



Usamos DELETE FROM para especificar que queremos borrar los registros de la tabla Alumno donde el grupo sea igual a 2 o 8.

1. Mostrar todos los registros Insertado en la tabla:

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto

Descripción generada automáticamente**

Tabla

Descripción generada automáticamente

Usamos la opción grafica de SQL Server para mostrar los primeros 1000 registros.

Podemos ver que se borraron 2 registros.

1. Eliminar todos los registros de la TablaX

**DELETE FROM Alumno;**



Usamos solamente DELETE FROM y el nombre de la tabla para borrar todos los registros de dicha tabla.

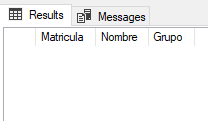
1. Mostrar todos los registros Insertado en la tabla:

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto

Descripción generada automáticamente**



Usamos la opción grafica de SQL Server para mostrar los primeros 1000 registros.

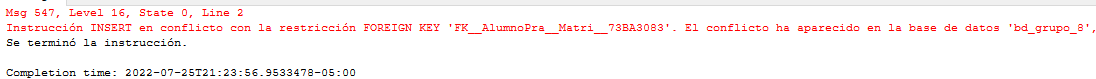
Podemos ver que se han borrado todos los registros.

Las siguientes operaciones tienen como objetivo comprobar el concepto de integridad referencial, el cual fue aplicado a la bd **bd\_grupo\_xyz** en la 2da Actividad, cuando se crearon las llaves foráneas con integridad DELETE RESTRICT UPADATE CASCADE.

Seleccionar al menos dos tablas conectadas entre sí mediante una relación de llave foránea. [TablaA] ->----<R>----|- [TablaB]. Para efectos de simplificación la explicación del ejercicio, llamaremos [TablaDébil] a la tabla que tiene la llave foránea, ósea, la [TablaA] y la otra la llamaremos [TablaFuerte], es decir, a la [TablaB].

1. Insertar al menos 1 registro en [TablaDebil], debe dar error.

**INSERT INTO AlumnoPractica VALUES (4.8, '2022/07/26', 005, 14);**

****

Hacemos inserción total dentro de la tabla intermedia AlumnoPractica, pero esta lanza error debido a que los valores asignados a las llaves foráneas de las tablas alumno y práctica, no existen en dichas tablas.

1. Mostrar todos los registros Insertado en la [TablaDebil]:

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Aplicación, Word

Descripción generada automáticamente**

Usamos la opción grafica de SQL Server para mostrar los primeros 1000 registros.

Podemos ver que no hay registros.

1. Insertar al menos 3 registros en [TablaFuerte], no debe dar error

INSERT INTO Alumno

VALUES ('001', 'Angel', 2),

('002', 'Miguel', 6),

('003', 'Cole', 1),

('004', 'Cristian', 3),

('005', 'Daniel', 6);

INSERT INTO Practica

VALUES ('Alta', 'Contado', 1),

('Media', 'Altura', 2),

('Baja', 'Arte', 3),

('Baja', 'Soldadura', 4),

('Alta', '3D', 5);

Texto, Pizarra

Descripción generada automáticamente

Insertamos registros masivamente en ambas tablas fuertes (Debido a que es una relación de muchos a muchos);

1. Mostrar todos los registros Insertado en la [TablaFuerte]:

Tabla Alumno

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

**Tabla

Descripción generada automáticamente**

Tabla Practica

Interfaz de usuario gráfica, Aplicación, Tabla

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Tabla

Descripción generada automáticamente

Usamos la opción grafica de SQL Server para mostrar los primeros 1000 registros.

Podemos ver que si se insertaron los registros en ambas tablas.

1. Insertar al menos 8 registros en [TablaDebil] pero de la siguiente manera:

* Los **2 primeros** registros de [TablaDebil] deben estar conectados al **I** registro Insertado en [TablaFuerte]
* Los **siguientes 3** registros de tabla [TablaDebil] deben estar conectados con el **II** registro de [TablaFuerte].
* Los **últimos 3** registros insertados en [TablaDebil] deben estar conectados con el **III r**egistro de tabla [TablaFuerte]

No debe dar error:

INSERT INTO AlumnoPractica

VALUES (4.8,'2022/07/01','001',1),

(5.0,'2022/08/11','001',4),

(3.9,'2022/01/20','002',3),

(1.0,'2022/03/23','002',2),

(2.5,'2022/12/12','002',5),

(4.5,'2022/04/10','003',3),

(3.2,'2022/06/03','003',2),

(4.1,'2022/05/30','003',1);

Texto

Descripción generada automáticamente

Insertamos masivamente en la tabla AlumnoPractica teniendo en cuenta que debemos usar matriculas y números de practicas ya existentes. Además de esto insertamos una fecha y la nota de dicha práctica.

1. Mostrar todos los registros Insertado en la [TablaDebil]:

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

**Tabla

Descripción generada automáticamente**

Usamos la opción grafica de SQL Server para mostrar los primeros 1000 registros.

Podemos ver que si se insertaron los registros en la tabla intermedia.

1. Modificar o **cambiar el valor de la llave primera** del **primer** registro insertado en [TablaFuerte].

UPDATE Alumno SET Matricula = '020' where Matricula = '001';

UPDATE Practica Set P = 8 Where P = 1;



Usamos UPDATE en ambas tablas fuertes para cambiar los datos de un solo registro teniendo en cuenta una condicion. En la tabla alumno donde la matricula sea 001 la cambiamos por 020 y en la tabla practica donde el numero de practica sea 1 lo cambiamos por 8.

1. Mostrar todos los registros Insertado en la [TablaFuerte]:

Tabla Alumno

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

Tabla

Descripción generada automáticamente

Tabla Practica

Interfaz de usuario gráfica, Aplicación, Tabla

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Tabla

Descripción generada automáticamente

Usamos la opción grafica de SQL Server para mostrar los primeros 1000 registros de ambas tablas.

Podemos ver que los UPDATE realizados a cada tabla si tuvieron efecto.

1. Mostrar todos los registros Insertado en la [TablaDebil]:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Tabla

Descripción generada automáticamente

Usamos la opción grafica de SQL Server para mostrar los primeros 1000 registros.

Podemos ver que si se actualizaron los registros en la tabla intermedia.

1. Modificar o **cambiar el valor de la llave foránea** del IV **registro** insertado en [TablaDebil].

Update AlumnoPractica

Set Matricula = '002', P = 4

where Matricula = '003' and P = 2;

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media

Usamos UPDATE en la tabla intermedia para cambiar la Matricula por 002 y el numero de practica P por 4 en el registro número cuatro 4 donde cuya matricula sea igual a 003 y su numero de practica sea 2.

1. Mostrar todos los registros Insertado en la [TablaDebil]:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

**Tabla

Descripción generada automáticamente**

Usamos la opción grafica de SQL Server para mostrar los primeros 1000 registros.

Podemos ver que si se actualizo el registro numero cuatro.

1. Mostrar todos los registros Insertado en la [TablaFuerte]:

Tabla Alumno

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

Tabla

Descripción generada automáticamente

Tabla Practica

Interfaz de usuario gráfica, Aplicación, Tabla

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Tabla

Descripción generada automáticamente

Usamos la opción grafica de SQL Server para mostrar los primeros 1000 registros de ambas tablas.

Podemos ver que los registros de las tablas fuertes siguen intactos debido a que los cambios solo fueron hechos en la tabla intermedia.

1. Eliminar un registro de la [TablaFuerte]:

**DELETE FROM Alumno where Matricula = '003';**



Usamos DELETE FROM para especificar que queremos borrar los registros de la tabla alumno donde matricula sea igual a 003.

1. Mostrar todos los registros Insertado en la [TablaFuerte]:

Tabla Alumno

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Tabla

Descripción generada automáticamente

Tabla Practica

Interfaz de usuario gráfica, Aplicación, Tabla

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Tabla

Descripción generada automáticamente con confianza media

Usamos la opción grafica de SQL Server para mostrar los primeros 1000 registros de ambas tablas.

Podemos ver que se elimino un registro de la tabla Alumnos y la tabla Practica sigue intacta.

1. Mostrar todos los registros Insertado en la [TablaDebil]:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Tabla, Excel

Descripción generada automáticamente

Usamos la opción grafica de SQL Server para mostrar los primeros 1000 registros.

Podemos ver que se eliminaron los registros vinculados al registro eliminado en la tabla Alumno.

1. Eliminar un registro de la [TablaDebil]:

DELETE FROM AlumnoPractica WHERE Nota = 1;



Usamos DELETE FROM para especificar que queremos borrar los registros de la tabla AlumnoPractica donde la nota sea igual a 1.

1. Mostrar todos los registros Insertado en la [TablaDebil]:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Tabla

Descripción generada automáticamente

Usamos la opción grafica de SQL Server para mostrar los primeros 1000 registros.

Podemos ver que se eliminaron los registros donde la nota sea 1.

1. Mostrar todos los registros Insertado en la [TablaFuerte]:

Tabla Alumno

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Tabla

Descripción generada automáticamente

Tabla Practica

Interfaz de usuario gráfica, Aplicación, Tabla

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Tabla

Descripción generada automáticamente con confianza media

Usamos la opción grafica de SQL Server para mostrar los primeros 1000 registros de ambas tablas.

Podemos ver que los registros de las tablas fuertes siguen intactos debido a que los cambios solo fueron hechos en la tabla intermedia.

1. Eliminar todos los registros de la [TablaDebil] vinculados al 2do registro de la [TablaFuerte]

DELETE FROM AlumnoPractica WHERE Matricula = '002';



Usamos DELETE FROM para especificar que queremos borrar los registros de la tabla AlumnoPractica donde la matricula sea 002.

1. Mostrar todos los registros Insertado en la [TablaDebil]:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Tabla

Descripción generada automáticamente

Usamos la opción grafica de SQL Server para mostrar los primeros 1000 registros.

Podemos ver que se eliminaron los registros donde la matricula sea 002.

1. Mostrar todos los registros Insertado en la [TablaFuerte]:

Tabla Alumno

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Tabla

Descripción generada automáticamente

Tabla Practica

Interfaz de usuario gráfica, Aplicación, Tabla

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Tabla

Descripción generada automáticamente con confianza media

1. Eliminar el 2do Registro de la [TablaDebil]

**DELETE FROM AlumnoPractica WHERE P = 4;**

****

Usamos DELETE FROM para especificar que queremos borrar los registros de la tabla AlumnoPractica donde el número de practica sea 4.

1. Mostrar todos los registros Insertado en la [TablaFuerte]:

Tabla Alumno

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Tabla

Descripción generada automáticamente

Tabla Practica

Interfaz de usuario gráfica, Aplicación, Tabla

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Tabla

Descripción generada automáticamente con confianza media

Usamos la opción grafica de SQL Server para mostrar los primeros 1000 registros de ambas tablas.

Podemos ver que los registros de las tablas fuertes siguen intactos debido a que los cambios solo fueron hechos en la tabla intermedia.

**Síntesis**

Gestor de Base de Datos

Un Sistema Gestor de Base de Datos (SGBD es un sistema que permite la creación, gestión y administración de bases de datos, así como la selección y gestión de las estructuras necesarias para almacenar y recuperar información de la forma más eficiente posible.

Actualmente podemos organizarlos en dos tipo, a su vez daremos ejemplos:

* Gestores Relacionales: Estos gestores se basan principalmente en los Modelos relacionales, de tal manera que estos establecen relaciones entre sus datos. Entre ellos podemos encontrar:
  + MySQL
  + SQL Server
  + PostgreSQL
  + Oracle
* Gestores no relacionales: Estos gestores no requieren estructurar datos fijamente como en tablas. Son usados principalmente para gestionar grandes cantidades de datos. Entre ellos podemos encontrar:
  + MongoDB
  + Cassandra
  + Redis

Sentencias DML

Las sentencias DML (Data Manipulation Language) permiten a los usuarios la introducción, modificación, consulta o eliminación de registros que se encuentran en una base de datos. Las principales sentencias DML son:

* SELECT: Esta sentencia se utiliza para realizar consultas sobre los registros.
* INSERT: Esta sentencia se utiliza para insertar registros en una base de datos.
* UPDATE: Esta sentencia se utiliza para modificar los valores de uno o varios registros.
* DELETE: Esta sentencia se utiliza para eliminar los registros de una tabla.

**Conclusión**

En el presente documento, hemos podido observar el comportamiento de los datos frente a las sentencias DML ejecutadas para la realización de los ítems. No obstante, también hemos podido percatarnos de los métodos o funciones que nos ofrece cada Gestor de bases de datos para facilitarnos ciertas tareas.

**Referencias**

W3 Schools. (s. f.). *SQL tutorial*. <https://www.w3schools.com:443/sql/>

Marín, R. (2022, 21 abril). *Los gestores de bases de datos más usados en la actualidad.* Gestores de Bases De Datos. https://www.inesem.es/revistadigital/informatica-y-tics/los-gestores-de-bases-de-datos-mas-usados/