ÁREA 3. DESARROLLO DE SOFTWARE DE APLICACIÓN

SUBÁREA 3.3 BASES DE DATOS

TEMAS IMPORTANTE:

Conceptos de Bases de Datos:

Las bases de datos son una herramienta fundamental en el desarrollo de aplicaciones y sistemas informáticos. A continuación, se explican algunos de los conceptos clave de las bases de datos:

* Base de datos: Una base de datos es un conjunto organizado de datos que se almacena y se gestiona en un sistema informático. Las bases de datos permiten el almacenamiento, la gestión y la consulta de grandes cantidades de datos de una forma eficiente y organizada.
* Tablas: Las tablas son la forma en que se organizan los datos dentro de una base de datos relacional. Cada tabla representa una entidad del mundo real, como clientes, productos o pedidos, y contiene una serie de columnas que representan las propiedades de esa entidad.
* Filas: Las filas de una tabla representan instancias de la entidad que está representada por esa tabla. Cada fila contiene los valores de las propiedades de esa entidad en particular.
* Claves primarias: Una clave primaria es un atributo o conjunto de atributos que identifica de forma única cada fila de una tabla. Las claves primarias se utilizan para asegurar que no haya filas duplicadas en la tabla y para relacionar las filas de una tabla con las filas de otras tablas.
* Relaciones: Las relaciones son una forma de conectar las filas de diferentes tablas en una base de datos. Las relaciones se establecen utilizando claves primarias y claves foráneas, y permiten representar las relaciones entre las entidades del mundo real.
* Consultas: Las consultas son el medio por el cual se accede a los datos de una base de datos. Las consultas permiten buscar, filtrar y ordenar los datos de una o varias tablas, y pueden ser tan simples o complejas como sea necesario.
* Transacciones: Las transacciones son un conjunto de operaciones que se realizan en una base de datos y que se consideran como una unidad lógica. Las transacciones permiten asegurar la integridad de los datos en la base de datos, y se utilizan para garantizar que las operaciones se realicen de forma segura y consistente.

Modelo Entidad-Relación:

El Modelo Entidad-Relación (ER) es un modelo conceptual utilizado para representar la estructura de una base de datos en términos de entidades, atributos y relaciones. El modelo ER se utiliza en la fase de diseño de la base de datos para representar las necesidades de información del negocio o del sistema que se está desarrollando.

El modelo ER consta de tres elementos básicos:

* Entidades: Las entidades son objetos o conceptos del mundo real que se representan en la base de datos. Las entidades se representan como rectángulos en el modelo ER, y se identifican mediante un nombre descriptivo.
* Atributos: Los atributos son características o propiedades de las entidades que se desean almacenar en la base de datos. Los atributos se representan como óvalos en el modelo ER, y se identifican mediante un nombre descriptivo.
* Relaciones: Las relaciones son asociaciones entre dos o más entidades. Las relaciones se representan como rombos en el modelo ER, y se identifican mediante un nombre descriptivo. Las relaciones pueden ser de varios tipos, como uno a uno, uno a muchos y muchos a muchos.

En el modelo ER, las entidades se describen mediante atributos, y las relaciones se describen mediante las entidades que participan en la relación y su cardinalidad (es decir, el número de instancias de una entidad que se asocian con el número de instancias de otra entidad).

El modelo ER se utiliza como base para diseñar la estructura física de la base de datos, mediante la creación de tablas que representan las entidades y los atributos, y las relaciones se implementan mediante claves foráneas.

El modelo ER es una herramienta útil para la representación de la estructura de una base de datos y es ampliamente utilizado en el diseño de bases de datos relacionales. Sin embargo, el modelo ER no proporciona información sobre el comportamiento de la base de datos, lo que se resuelve mediante la adición de restricciones de integridad y reglas de negocio.

Modelo Relacional:

El modelo relacional es un modelo de datos utilizado en la gestión de bases de datos relacionales. El modelo relacional se basa en la teoría de conjuntos y álgebra relacional y se utiliza para representar la estructura de la base de datos y las relaciones entre las tablas.

En el modelo relacional, la información se organiza en tablas, donde cada tabla representa una entidad o una relación. Cada tabla consta de filas y columnas, donde las filas representan instancias de la entidad o la relación y las columnas representan los atributos o propiedades de la entidad o la relación.

En el modelo relacional, se utilizan claves para establecer las relaciones entre las tablas. La clave principal de una tabla es un conjunto de uno o varios atributos que identifican de manera única cada fila en la tabla. Las claves foráneas se utilizan para establecer relaciones entre las tablas, donde la clave foránea de una tabla es un conjunto de atributos que se refieren a la clave principal de otra tabla.

El modelo relacional utiliza el lenguaje SQL (Structured Query Language) para realizar consultas y manipular los datos en la base de datos. Las operaciones SQL incluyen la selección, la inserción, la actualización y la eliminación de datos en las tablas.

Las ventajas del modelo relacional incluyen la flexibilidad para agregar, modificar y eliminar datos, así como la capacidad de realizar consultas y análisis complejos de datos. Además, el modelo relacional proporciona una separación clara entre la estructura de la base de datos y la lógica de la aplicación, lo que facilita el mantenimiento y la escalabilidad de la base de datos.

Disparadores (Triggers):

Los disparadores o triggers son objetos de la base de datos que se ejecutan automáticamente en respuesta a ciertos eventos o acciones, como la inserción, actualización o eliminación de datos en una tabla. Los disparadores se definen para una tabla específica y se activan cuando se produce un evento determinado en esa tabla.

Los disparadores se utilizan para realizar acciones adicionales en la base de datos, como la validación de datos, la actualización de otras tablas o la realización de cálculos complejos. Por ejemplo, un disparador podría actualizar una tabla de totales después de que se haya realizado una inserción o actualización en una tabla de detalles.

Los disparadores se definen mediante un conjunto de instrucciones SQL que se ejecutan automáticamente cuando se produce el evento asociado. El disparador se puede configurar para que se ejecute antes o después del evento y se puede definir para ejecutar una única instrucción SQL o un conjunto de instrucciones.

Los disparadores se utilizan comúnmente en aplicaciones empresariales y de bases de datos para garantizar la integridad de los datos, realizar cálculos complejos o automatizar tareas de mantenimiento. Sin embargo, es importante tener cuidado al utilizar disparadores, ya que pueden afectar el rendimiento de la base de datos y complicar el mantenimiento y la depuración.

Consultas en SQL:

SQL (Structured Query Language) es un lenguaje de programación utilizado para interactuar con las bases de datos relacionales. Las consultas en SQL se utilizan para recuperar información de una o varias tablas en una base de datos. A continuación, se describen algunos de los conceptos y comandos básicos para realizar consultas en SQL:

SELECT: Es el comando principal utilizado para realizar consultas en SQL. Se utiliza para seleccionar una o varias columnas de una tabla o varias tablas, y puede incluir cláusulas como WHERE, GROUP BY, HAVING y ORDER BY para filtrar, agrupar y ordenar los resultados.

WHERE: Se utiliza para filtrar los datos en una consulta en SQL. Permite especificar una o varias condiciones que deben cumplirse para que los registros se incluyan en los resultados de la consulta.

GROUP BY: Se utiliza para agrupar los registros en una consulta en SQL. Permite agrupar los resultados según una o varias columnas y realizar operaciones de agregación, como SUM, COUNT, AVG, MIN y MAX, en las columnas agrupadas.

HAVING: Se utiliza para filtrar los resultados de una consulta después de que se hayan aplicado las cláusulas GROUP BY y las funciones de agregación. Permite especificar condiciones que deben cumplirse para que los grupos se incluyan en los resultados de la consulta.

ORDER BY: Se utiliza para ordenar los resultados de una consulta en SQL. Permite especificar una o varias columnas por las que se deben ordenar los resultados, en orden ascendente o descendente.

JOIN: Se utiliza para combinar varias tablas en una consulta en SQL. Permite especificar cómo se deben relacionar las tablas y cómo se deben combinar los datos de las tablas relacionadas.

Manejadores de Bases de Datos:

Un manejador de bases de datos (DBMS, Database Management System) es un software que permite la creación, gestión y utilización de bases de datos. Los DBMS proporcionan una interfaz entre el usuario y la base de datos, permitiendo al usuario realizar operaciones como la creación, consulta, actualización y eliminación de datos. A continuación, se describen algunos de los manejadores de bases de datos más comunes:

MySQL: Es un sistema de gestión de bases de datos relacional de código abierto muy popular y ampliamente utilizado en aplicaciones web. MySQL se utiliza en una amplia variedad de aplicaciones, desde sitios web hasta sistemas de gestión de inventarios y registros médicos.

PostgreSQL: Es un sistema de gestión de bases de datos relacional de código abierto conocido por su capacidad para manejar grandes cantidades de datos y su compatibilidad con SQL. PostgreSQL es ampliamente utilizado en aplicaciones empresariales y de bases de datos.

Microsoft SQL Server: Es un sistema de gestión de bases de datos relacional desarrollado por Microsoft. Se utiliza principalmente en entornos empresariales y es compatible con una amplia variedad de lenguajes de programación.

Oracle Database: Es un sistema de gestión de bases de datos relacional desarrollado por Oracle. Oracle es ampliamente utilizado en entornos empresariales y se considera uno de los sistemas de bases de datos más potentes y completos del mercado.

MongoDB: Es un sistema de gestión de bases de datos NoSQL que se utiliza principalmente en aplicaciones web y móviles. MongoDB es conocido por su escalabilidad y flexibilidad, y es compatible con una variedad de lenguajes de programación.

Bases de Datos Distribuidas:

Las bases de datos distribuidas son sistemas de bases de datos que se distribuyen geográficamente en varias ubicaciones físicas, y están diseñados para trabajar juntos como si fueran una sola base de datos. A diferencia de una base de datos centralizada, donde toda la información se almacena en un solo lugar, una base de datos distribuida divide la información entre varias ubicaciones y las interconecta mediante redes de comunicación.

A continuación, se describen algunos conceptos clave de las bases de datos distribuidas:

* Fragmentación: La fragmentación se refiere a la división de una tabla en fragmentos, donde cada fragmento se almacena en un nodo de la red. Hay tres tipos de fragmentación: horizontal, vertical y mixta.
* Replicación: La replicación se refiere a la creación de copias de datos y su distribución en varios nodos de la red. La replicación se utiliza para mejorar la disponibilidad de los datos y reducir el tiempo de respuesta de las consultas.
* Transacciones distribuidas: Una transacción distribuida se compone de varias operaciones que se realizan en diferentes nodos de la red. Para que una transacción distribuida sea exitosa, todas las operaciones deben completarse con éxito.
* Control de concurrencia: El control de concurrencia se refiere a la coordinación de las transacciones concurrentes en una base de datos distribuida. El objetivo es evitar conflictos de actualización y garantizar la consistencia de los datos.
* Sincronización: La sincronización se refiere a la coordinación de la copia de datos entre los nodos de la red. La sincronización es necesaria para garantizar la consistencia de los datos y evitar conflictos de actualización.
* Seguridad: La seguridad es un aspecto crítico en las bases de datos distribuidas. Debido a que los datos se almacenan en varios nodos de la red, la seguridad debe ser implementada en todos los nodos para evitar accesos no autorizados y proteger la integridad de los datos.

Las bases de datos distribuidas se utilizan en entornos empresariales y de alta disponibilidad, donde es necesario tener acceso a los datos desde diferentes ubicaciones geográficas y garantizar su disponibilidad y seguridad.