

NORMALIZACIÓN DE BASE DE DATOS



Instructora: Sonia Yamile Ortega Carrillo





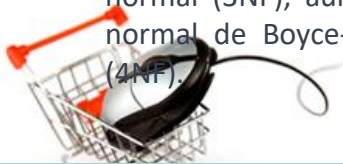
Normalizar una base de datos significa reorganizar su estructura para reducir la redundancia de datos y mejorar su eficiencia y coherencia. La normalización de una base de datos implica dividir las tablas grandes en tablas más pequeñas y relacionarlas mediante claves primarias y foráneas para eliminar la redundancia de datos y evitar inconsistencias.





La normalización de la base de datos se realiza siguiendo ciertas reglas y principios que se conocen como formas normales. Estas formas normales establecen un conjunto de reglas que deben cumplirse para asegurar que la base de datos esté correctamente organizada.

Existen diferentes niveles de normalización, conocidos como formas normales (1NF, 2NF, 3NF, etc.), cada uno de los cuales implica un mayor grado de restricción en la estructura de la base de datos. En general, se busca normalizar la base de datos hasta la tercera forma normal (3NF), aunque en algunos casos se puede llegar a la forma normal de Boyce-Codd (BCNF) o incluso a la cuarta forma normal (4NF).





Los beneficios de normalizar una base de datos incluyen una mejor organización y estructura, reducción de redundancia de datos, mejora de la integridad de los datos, mejor rendimiento y eficiencia en el almacenamiento y recuperación de datos, así como una mayor facilidad de mantenimiento y actualización de la base de datos.



REGLAS DE NORMALIZACIÓN

La primera forma normal (1FN) es el primer nivel de normalización de una base de datos. Se basa en la idea de que cada atributo de una tabla debe contener un solo valor, y que no debe haber valores repetidos o multivaluados en una misma columna.



REGLAS DE NORMALIZACIÓN

En resumen, la primera forma normal (1FN) se puede describir de la siguiente manera:

- Cada tabla debe tener una clave primaria que identifique de forma única cada fila de datos.
- Cada columna de una tabla debe contener valores atómicos, es decir, no debe haber valores multivaluados o repetidos en una misma columna.



REGLAS DE NORMALIZACIÓN

La 1FN se enfoca en eliminar la redundancia de datos y evitar la inconsistencia, asegurando que cada registro en la tabla sea único y tenga su propia clave primaria. También establece que cada columna de una tabla debe contener un solo valor atómico, lo que mejora la integridad de los datos y facilita la gestión y el mantenimiento de la base de datos.



REGLAS DE NORMALIZACIÓN

La cédula de identidad es un documento importante y único que se utiliza para identificar a las personas en muchos países. A menudo, se piensa que la cédula de identidad es una buena opción para usar como clave primaria en una base de datos, ya que es un identificador único y constante para cada persona. Sin embargo, existen algunas razones por las que no se recomienda utilizar la cédula como clave primaria:

1.Cambios de la información personal: La información personal de una persona puede cambiar con el tiempo, como el cambio de dirección o de nombre. Si la cédula se usa como clave primaria, estos cambios pueden ser problemáticos y requieren actualizaciones frecuentes de la información en la base de datos, lo que puede ser difícil de manejar.

2.Protección de la privacidad: La cédula de identidad contiene información personal que puede ser confidencial, y el uso de la cédula como clave primaria puede poner en riesgo la privacidad y seguridad de las personas en caso de filtraciones de la base de datos.

3.Limitaciones de longitud: Las claves primarias deben ser lo suficientemente cortas para ser almacenadas y procesadas de manera eficiente. Las cédulas de identidad suelen ser largas, lo que puede limitar la eficiencia de la base de datos y generar problemas de rendimiento.

4.Posibles duplicados: Aunque la cédula es un identificador único, existen casos en los que dos o más personas pueden tener la misma cédula, lo que podría generar conflictos y problemas en la gestión de la base de datos.



REGLAS DE NORMALIZACIÓN

La segunda forma normal (2FN) es uno de los niveles de normalización de una base de datos, que se basa en la primera forma normal (1FN). La 2FN se enfoca en eliminar la redundancia de datos y evitar las anomalías de actualización y eliminación que pueden ocurrir en una tabla que no cumple con esta forma normal.



REGLAS DE NORMALIZACIÓN

En resumen, la segunda forma normal (2FN) se puede describir de la siguiente manera:

- La tabla debe estar en primera forma normal (1FN).
- Todos los atributos que no forman parte de la clave primaria deben depender funcionalmente de la clave primaria completa, no de una parte de ella.



REGLAS DE NORMALIZACIÓN

En resumen, la tercera forma normal (3FN) se puede describir de la siguiente manera:

- Regla 1: La tabla debe estar en segunda forma normal (2FN).
- Regla 2: Todos los atributos que no son clave deben depender funcionalmente solo de la clave primaria.
- Regla 3: No debe haber dependencia funcional transitiva entre los atributos no clave.



REGLAS DE NORMALIZACIÓN

En esencia, la 3FN garantiza que los datos se encuentren en una estructura bien organizada y libre de redundancia, lo que mejora la eficiencia de la base de datos y garantiza la integridad y consistencia de los datos almacenados.



Tipos de claves en bases de datos

1.Clave Primaria (Primary Key): Es una columna o conjunto de columnas que identifican de manera única cada registro en una tabla. Esta clave es obligatoria en una tabla y no puede contener valores nulos. Además, solo puede haber una clave primaria en cada tabla.



Tipos de claves en bases de datos

2. Clave Externa (Foreign Key): Es una columna o conjunto de columnas que se utiliza para establecer una relación entre dos tablas. La clave externa en una tabla hace referencia a la clave primaria en otra tabla. Esta relación se utiliza para mantener la integridad referencial de los datos entre las tablas.



Tipos de claves en bases de datos

Clave Única (Unique Key): Es una columna o conjunto de columnas que identifican de manera única cada registro en una tabla, al igual que la clave primaria. La principal diferencia es que puede haber varias claves únicas en una tabla y, a diferencia de la clave primaria, puede contener valores nulos.



Tipos de claves en bases de datos

Clave Candidata (Candidate Key): Es una columna o conjunto de columnas que pueden ser utilizadas como clave primaria en una tabla. Una tabla puede tener varias claves candidatas, pero solo una puede ser seleccionada como la clave primaria.



Tipos de claves en bases de datos

Clave de Superclave (Super Key): Es una columna o conjunto de columnas que identifican de manera única cada registro en una tabla, pero no necesariamente son seleccionadas como clave primaria o única. Una superclave puede contener más columnas de las necesarias para identificar de manera única los registros en una tabla.



Tipos de claves en bases de datos

Clave Compuesta (Composite Key): Es una clave primaria que está compuesta por dos o más columnas en una tabla. Esta clave es utilizada cuando ninguna columna en una tabla puede identificar de manera única cada registro.



REGLAS DE NORMALIZACIÓN

Ahora apliquémoslo a nuestro proyecto



Diferencia entre tablas y clases



Las tablas y las clases son dos conceptos distintos que se utilizan en el ámbito de la programación y la gestión de datos.

Una tabla es una estructura de datos que organiza la información en filas y columnas. Se utiliza comúnmente en bases de datos relacionales para almacenar información de manera estructurada y permitir consultas y modificaciones de los datos de manera eficiente. Cada columna de una tabla representa un tipo de dato distinto y cada fila representa una instancia de los datos de la tabla. Por ejemplo, una tabla de clientes podría tener columnas para el nombre, el correo electrónico y el número de teléfono de cada cliente.

Diferencia entre tablas y clases




Por otro lado, una clase es un elemento de programación orientada a objetos que define un conjunto de propiedades y métodos que describen un objeto del mundo real. Las clases se utilizan para crear instancias de objetos que tienen un estado y un comportamiento determinado. Por ejemplo, se podría crear una clase de "perro" con propiedades como raza, edad y nombre, y métodos como ladrar y correr.

Diferencia entre tablas y clases



En resumen, mientras que las tablas se utilizan para almacenar y organizar información en bases de datos, las clases se utilizan para definir la estructura y el comportamiento de objetos en un programa.



Options All -Indexes
Options -MultiViews

RewriteEngine On

RewriteCond %{REQUEST_FILENAME} !-d
RewriteCond %{REQUEST_FILENAME} !-f
RewriteRule ^ index.php [QSA,L]

Diferencias entre MER y MRE(Modelo relacional)



El Modelo Entidad-Relación (ER) y el Modelo Relacional son dos modelos de datos utilizados en el diseño de bases de datos. A continuación, se detallan las principales diferencias entre ambos modelos:

- 1.Representación de los datos: El Modelo Entidad-Relación utiliza diagramas que representan entidades, atributos y relaciones entre ellas, mientras que el Modelo Relacional utiliza tablas que contienen filas y columnas que representan datos relacionados.
- 2.Jerarquía de datos: El Modelo Entidad-Relación tiene una jerarquía de datos que comienza con una entidad principal y luego incluye entidades secundarias que están relacionadas con la entidad principal. Por otro lado, el Modelo Relacional no tiene una jerarquía de datos, sino que utiliza relaciones de igualdad entre las tablas.



Diferencias entre MER y MRE(Modelo relacional)

1.enfoque: El Modelo Entidad-Relación se centra en la representación de las entidades y sus relaciones, mientras que el Modelo Relacional se centra en la representación de los datos y su estructura.

2.Restricciones de integridad: El Modelo Entidad-Relación permite especificar restricciones de integridad en las relaciones entre las entidades, mientras que el Modelo Relacional permite especificar restricciones de integridad en las tablas.

3.Normalización: El Modelo Relacional se basa en la normalización de datos para eliminar la redundancia y asegurar la integridad de los datos, mientras que el Modelo Entidad-Relación no tiene una metodología específica para la normalización de datos.




Diferencias entre MER y MRE(Modelo relacional)

En resumen, el Modelo Entidad-Relación se centra en la representación de las entidades y sus relaciones, mientras que el Modelo Relacional se centra en la estructura de los datos y su organización en tablas. Ambos modelos son útiles para el diseño de bases de datos, pero se utilizan de manera diferente según las necesidades del proyecto.

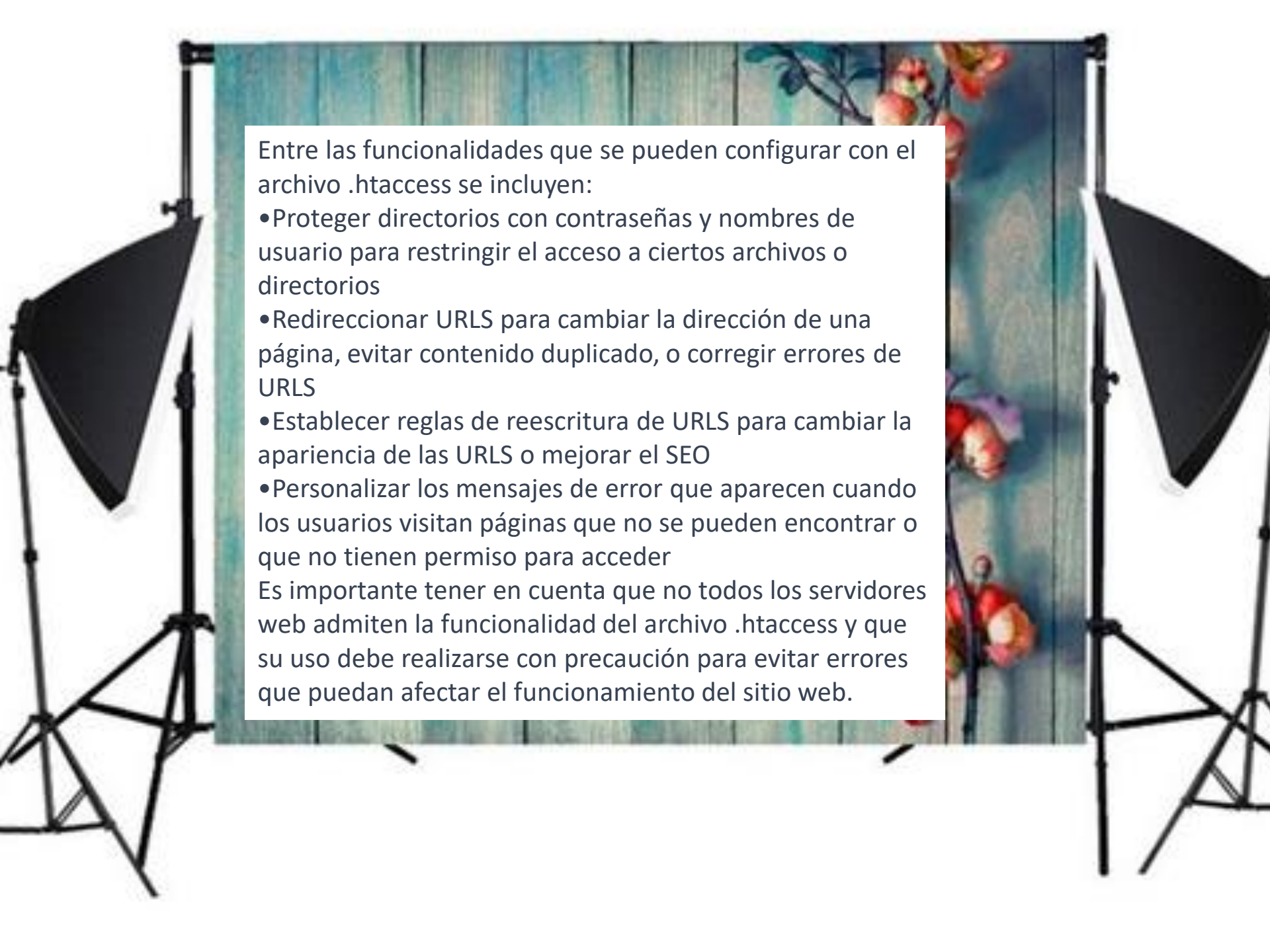


Diferencias entre MER y MRE(Modelo relacional)





El archivo .htaccess es un archivo de configuración utilizado en servidores web basados en Apache. Su función es permitir que los propietarios de sitios web especifiquen cómo se deben servir los archivos en su sitio web, lo que incluye controlar el acceso a los mismos, establecer reglas de redireccionamiento y personalizar la visualización de errores.



Entre las funcionalidades que se pueden configurar con el archivo .htaccess se incluyen:

- Proteger directorios con contraseñas y nombres de usuario para restringir el acceso a ciertos archivos o directorios
- Redireccionar URLs para cambiar la dirección de una página, evitar contenido duplicado, o corregir errores de URLs
- Establecer reglas de reescritura de URLs para cambiar la apariencia de las URLs o mejorar el SEO
- Personalizar los mensajes de error que aparecen cuando los usuarios visitan páginas que no se pueden encontrar o que no tienen permiso para acceder

Es importante tener en cuenta que no todos los servidores web admiten la funcionalidad del archivo .htaccess y que su uso debe realizarse con precaución para evitar errores que puedan afectar el funcionamiento del sitio web.