

Diseño de bases de datos

Diseño de bases de datos

- Objetivos del Diseño
- Principios del Diseño de BD
- Proceso de Diseño
- Normalización
- Diseño de Tablas: Claves
- Relaciones
- Integridad referencial
- Convenciones de nomenclatura

Objetivos del Diseño

- Una base de datos correctamente diseñada permite obtener acceso a información exacta y actualizada. Puesto que un diseño correcto es esencial para lograr los objetivos fijados para la base de datos, parece lógico emplear el tiempo que sea necesario en aprender los principios de un buen diseño ya que, en ese caso, es mucho más probable que la base de datos termine adaptándose a sus necesidades y pueda modificarse fácilmente.

- Será esencial aprender a decidir qué información necesita, a dividir la información en las tablas y columnas adecuadas y a relacionar las tablas entre sí.
- El proceso de diseño de una base de datos se guía por algunos principios.
 - Evitar la información duplicada o, lo que es lo mismo, los datos redundantes, porque malgastan el espacio y aumentan la probabilidad de que se produzcan errores e incoherencias.

- Es importante que la información sea correcta y completa. Si la base de datos contiene información incorrecta, los informes que recogen información de la base de datos contendrán también información incorrecta y, por tanto, las decisiones que tome a partir de esos informes estarán mal fundamentadas.

- Un buen diseño de base de datos es, por tanto, aquél que:
 - Divide la información en tablas basadas en temas(entidades) para reducir los datos redundantes.
 - Proporciona la información necesaria para reunir la información de las tablas cuando así se precise.
 - Ayuda a garantizar la exactitud e integridad de la información.
 - Satisface las necesidades de procesamiento de los datos y de generación de informes.

Proceso de Diseño

1. Determinar la finalidad de la base de datos
2. Buscar y organizar la información necesaria:
Reúna todos los tipos de información que desee registrar en la base de datos, como los nombres de productos o los números de pedidos.
3. Dividir la información en tablas: Divida los elementos de información en entidades o temas principales, como Productos o Pedidos. Cada tema pasará a ser una tabla.

4. Convertir los elementos de información en columnas: Decida qué información desea almacenar en cada tabla. Cada elemento se convertirá en un campo y se mostrará como una columna en la tabla. Por ejemplo, una tabla Empleados podría incluir campos como Apellido y Fecha de contratación.
5. Especificar claves principales: Elija la clave principal de cada tabla. La clave principal es una columna que se utiliza para identificar inequívocamente cada fila, como Id. de producto o Código de cliente.

6. Definir relaciones entre las tablas: Examine cada tabla y decida cómo se relacionan los datos de una tabla con las demás tablas. Agregue campos a las tablas o cree nuevas tablas para clarificar las relaciones según sea necesario.
7. Ajustar el diseño: Analice el diseño para detectar errores. Cree las tablas y agregue algunos registros con datos de ejemplo. Compruebe si puede obtener los resultados previstos de las tablas. Realice los ajustes necesarios en el diseño.
8. Aplicar las reglas de normalización: Aplique reglas de normalización de los datos para comprobar si las tablas están estructuradas correctamente. Realice los ajustes necesarios en las tablas.

Algunas sugerencias para determinar las columnas de la base de datos

- No incluya datos calculados: En la mayoría de los casos, no debe almacenar el resultado de los cálculos en las tablas. En lugar de ello, puede dejar que Access realice los cálculos cuando desee ver el resultado.
- Almacene la información en sus partes lógicas más pequeñas: Puede ceder a la tentación de habilitar un único campo para los nombres completos o para los nombres de productos junto con sus descripciones. Si combina varios tipos de información en un campo, será difícil recuperar datos individuales más adelante. Intente dividir la información en partes lógicas. Por ejemplo, cree campos distintos para el nombre y el apellido, o para el nombre del producto, la categoría y la descripción.

Normalización

- Las reglas de normalización sirven para comprobar si las tablas están estructuradas correctamente.
- La normalización es más útil una vez representados todos los elementos de información y después de haber definido un diseño preliminar. La idea es asegurarse de que se han dividido los elementos de información en las tablas adecuadas. Lo que la normalización no puede hacer es garantizar que se dispone de los elementos de datos correctos para empezar a trabajar.
- Las reglas se aplican consecutivamente en cada paso para garantizar que el diseño adopta lo que se conoce como "forma normal".

Objetivos de la normalización

Eliminar la información duplicada de las tablas, evitar la redundancia de los datos.

Adaptar los cambios futuros en la estructura de las tablas, evitar problemas de actualización de los datos en las tablas.

Minimizar el impacto del cambio estructural en lo que respecta a las aplicaciones de los usuarios que acceden a los datos, proteger la integridad de los datos.

Pasos

- De 1era a 5ta Forma Normal (FN)

Primera forma normal- Crea una nueva relacion con los grupos que se repiten

- La primera forma normal establece que en cada intersección de fila y columna de la tabla existe un valor y nunca una lista de valores.
- Un atributo es atómico si sus elementos se pueden considerar como unidades indivisibles
 - Los valores no atómicos complican el almacenamiento y pueden provocar redundancia:
 - Por ejemplo, cuentas bancarias almacenadas con sus propietarios
- Respetar la atomicidad y no intentes soluciones “inteligentes”
- Supongamos que en la base de datos de la universidad a los profesores se les asigna identificadores del tipo: LAN013

Segunda forma normal- Elimina dependencias funcionales

- La segunda forma normal exige que cada columna que no sea clave dependa por completo de toda la clave principal y no sólo de parte de la clave. Esta regla se aplica cuando existe una clave principal formada por varias columnas.
- Suponga, por ejemplo, que existe una tabla con las siguientes columnas, de las cuales Id. de pedido e Id. de producto forman la clave principal:
 - Id. de pedido (clave principal)
 - Id. de producto (clave principal)
 - Nombre de producto

Este diseño infringe los requisitos de la segunda forma normal, porque Nombre de producto depende de Id. de producto, pero no de Id. de pedido, por lo que no depende de toda la clave principal. Debe quitar Nombre de producto de la tabla, ya que pertenece a una tabla diferente (a la tabla Productos).

Tercera forma normal- Elimina dependencias transitivas

- La tercera forma normal exige no sólo que cada columna que no sea clave dependa de toda la clave principal, sino también que las columnas que no sean clave sean independientes unas de otras. O dicho de otra forma: cada columna que no sea clave debe depender de la clave principal y nada más que de la clave principal.
- Ejemplo, considere una tabla con las siguientes columnas:
- IdProducto (clave principal)
- Nombre
- PrecioVenta
- Descuento

- Suponga que la columna Descuento depende del PrecioVenta sugerido. Esta tabla infringe los requisitos de la tercera forma normal porque una columna que no es clave, la columna Descuento, depende de otra columna que no es clave, la columna PrecioVenta.
- La independencia de las columnas implica que debe poder cambiar cualquier columna que no sea clave sin que ninguna otra columna resulte afectada. Si cambia un valor en el campo PrecioVenta, la columna Descuento cambiaría en consecuencia e infringiría esa regla. En este caso, la columna Descuento debe moverse a otra tabla cuya clave sea PrecioVenta.

Resumen

1. Comprueba que cada tabla tiene un número fijo de columnas y las variables son sencillas (atómicas)
2. Identifica la clave primaria
3. Comprueba que todos los atributos (menos la clave primaria) depende de TODA la clave no de PARTE de ella.
4. Si existe dependencia parcial rompe la relación en varias subrelaciones.
5. Comprueba que todos los atributos dependen de la clave y no de otros atributos (dependencias transitivas)
6. Si existe dependencias no relacionadas con la clave primaria subdivide las tablas