

Curso Bases de Datos

Radium Rocket

Clase 2

Modelado de Bases de Datos

Agenda

- Conceptos básicos de modelado de bases de datos.
- Entidades, atributos y relaciones en un sistema.
- Diagramas ER y transformación en modelos relacionales.
- Formas normales (1NF, 2NF, 3NF) y aplicación práctica.

Modelado de Bases de Datos

Introducción al Modelado de Bases de Datos

¿Qué es el modelado de datos?

- Proceso de estructurar datos de manera eficiente.
- Herramienta esencial para evitar redundancias y errores.

Importancia del modelado:

- Optimización del diseño.
- Mejora de la consulta y la escalabilidad.
- Base para sistemas de información bien diseñados.

Diagramas Entidad-Relación

Diagramas Entidad-Relación (ER)

Entidades

Representan objetos, conceptos o elementos del mundo real sobre los cuales se desea almacenar información.

Tipos de Entidades:

- Fuertes (o regulares): Aquellas que tienen existencia independiente.
Ejemplo: Empleados, Departamentos.
- Débiles: Aquellas que dependen de otra entidad para existir.
Ejemplo: Detalles de Pedidos dependen de Pedidos.

En un sistema de gestión de empleados, las entidades principales podrían ser:

- Empleados: Representa a los trabajadores de la organización.
- Departamentos: Representa las divisiones dentro de la organización.
- Proyectos: Representa las tareas o actividades asignadas.

Diagramas Entidad-Relación (ER)

Atributos

Representan las características o propiedades de las entidades.

Tipos de Atributos:

- Simples: No se pueden dividir en partes más pequeñas. Ejemplo: nombre, edad.
- Compuestos: Pueden dividirse en partes más pequeñas. Ejemplo: dirección (compuesto por calle, ciudad, código postal).
- Multivaluados: Pueden tener múltiples valores. Ejemplo: teléfonos de un empleado.
- Derivados: Pueden calcularse a partir de otros atributos. Ejemplo: edad derivada de fecha_nacimiento.

Para la entidad Empleados:

`empleado_id (simple, clave primaria).`

`nombre (simple).`

`fecha_nacimiento (simple).`

`dirección (compuesto).`

Diagramas Entidad-Relación (ER)

Relaciones

Asociaciones entre dos o más entidades.

Componentes:

- Cardinalidad: Indica el número de ocurrencias en la relación. Puede ser:
 - Uno a uno (1:1).
 - Uno a muchos (1:N).
 - Muchos a muchos (M:N).
- Participación: Indica si la relación es obligatoria o opcional.
 - Total: Todas las instancias de una entidad están relacionadas.
 - Parcial: Algunas instancias de una entidad están relacionadas.

Ejemplo:

En un sistema de empleados:

- Empleado trabaja en Proyecto: Relación M:N.
- Empleado pertenece a Departamento: Relación 1:N.

Ejemplo

Ejemplo

Empleado trabaja en Proyecto (M:N):

Un empleado puede trabajar en varios proyectos.

Un proyecto puede tener varios empleados asignados.

Tabla intermedia: asignaciones para resolver la relación M:N.

Atributos: empleado_id, proyecto_id, horas_asignadas.

Empleado pertenece a Departamento (1:N):

Un empleado pertenece a un solo departamento.

Un departamento puede tener varios empleados.

Ejemplo

Entidades Principales:

Empleados, Departamentos, Proyectos.

Relaciones:

1:N entre Empleado y Departamento.

M:N entre Empleado y Proyecto, resuelta con una tabla intermedia Asignaciones.

Diagramas ER: Identifican las asociaciones y cardinalidades para estructurar correctamente el sistema.

Normalización de bases de datos

Normalización in a nutshell

¿Qué es la normalización?

Proceso para estructurar tablas y eliminar redundancias.

Mejora la integridad de los datos y evita inconsistencias.

Resumen de las Formas Normales:

1NF: Eliminar atributos multivaluados.

2NF: Eliminar dependencias parciales.

3NF: Eliminar dependencias transitivas.

1NF

Tabla inicial: pedidos

pedido_id	cliente	productos	cantidades
1	Juan	TV, Laptop	1, 1
2	Ana	Smartphone	2

1NF

1NF (Primera Forma Normal): Eliminar Atributos Multivaluados

Regla: Cada celda debe contener un único valor.

Problema: La columna productos y cantidades contienen múltiples valores separados por comas.

Solución:

Dividir las filas para que cada celda tenga un solo valor.

1NF

pedido_id	cliente	producto	cantidad
1	Juan	TV	1
1	Juan	Laptop	1
2	Ana	Smartphone	2

2NF

2NF (Segunda Forma Normal): Eliminar Dependencias Parciales

Regla: Todos los atributos no clave deben depender completamente de la clave primaria.

Problema: cliente depende de pedido_id, pero no de la combinación de pedido_id y producto.

Solución:

Crear una tabla separada para almacenar la información de los clientes.

2NF

pedido_id			producto			cantidad		
1			TV			1		
1			Laptop			1		
2			Smartphone			2		

pedido_id		cliente	
1		Juan	
2		Ana	

3NF

3NF (Tercera Forma Normal): Eliminar Dependencias Transitivas

Regla: Ningún atributo no clave debe depender de otro atributo no clave.

Problema: Si la tabla pedidos incluye información adicional como la dirección del cliente, esta depende de cliente y no de pedido_id.

Solución:

Crear una tabla separada para almacenar información de los clientes.

3NF

cliente_id	nombre	direccion
101	Juan	Calle 1
102	Ana	Calle 2

Por qué normalización

Reducción de Redundancia: Los datos no se repiten innecesariamente.

Evitar Inconsistencias: Las modificaciones se realizan en un solo lugar.

Mejoras en Escalabilidad: El diseño es modular y fácil de extender.

Ideas sobre 2NF y 3NF

Segunda Forma Normal (2NF):

Entra en juego solo si hay claves primarias compuestas.

Elimina las dependencias parciales, es decir, asegura que cada atributo no clave dependa de toda la clave primaria compuesta.

Tercera Forma Normal (3NF):

Asegura que no haya dependencias transitivas, es decir, ningún atributo no clave debe depender de otro atributo no clave.

Se aplica independientemente de si la clave primaria es compuesta o simple.

Ejemplo 2NF

Tabla inicial: inscripciones

estudiante_id	curso_id	curso_nombre	estudiante_nombre
1	101	Matemáticas	Ana López
2	102	Física	Juan Pérez
1	103	Química	Ana López

Ejemplo 2NF

estudiante_id	estudiante_nombre
1	Ana López
2	Juan Pérez

curso_id	curso_nombre
101	Matemáticas
102	Física
103	Química

estudiante_id	curso_id
1	101
2	102
1	103

Ejemplo 3NF

Tabla inicial: ventas

venta_id	cliente_id	cliente_dir	total
1	101	Calle A	500
2	102	Calle B	300
3	101	Calle A	200

Ejemplo 3NF

cliente		
cliente_id	cliente_direccion	
101	Calle A	
102	Calle B	
venta		
venta_id	cliente_id	total
1	101	500
2	102	300
3	101	200