

Bases de datos

Modelos de datos y normalización

Modelo de datos

Una representación abstracta y estructurada de la información que se almacena

Componentes

- Modelo conceptual
- Modelo lógico
- Modelo físico

Modelo Conceptual

Objetivo

- Identificar y definir los objetos de la base de datos y las relaciones entre ellos
- Representación visual de alto nivel; sin detalles técnicos
- Modelo Entidad-relación

Modelo Entidad-relación

- Entidades: Objetos que deben ser almacenados
- Atributos: Características que describen las entidades
- Relaciones: Cómo las entidades se relacionan entre sí

Diagrama Entidad-relación



Entidad



Relación



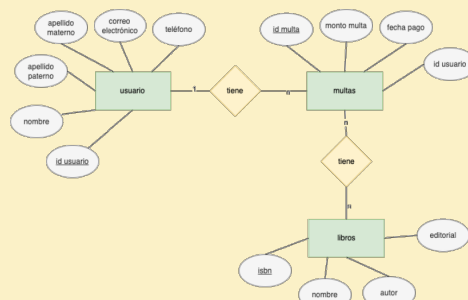
Atributo



Atributo llave



Conector



Depenencias de existencia

Una entidad puede solo existir como "hijo" de otra

Cardinalidad	<p># de registros de una entidad A que se relaciona con un registro de una entidad B</p> <ul style="list-style-type: none"> • 1-1: Un registro de A se relaciona únicamente con un registro de B • 1-n: Un registro de A se relaciona con 0 o varios registros de B • n-n: Varios registros de A se relacionan con varios registros de B <p>Se hace con una tabla adicional, solo con las relaciones</p>
---------------------	---

Primary Key / Clave primaria	<p>Número identificador del renglón en la tabla; único</p> <p>Se pueden usar para referenciar el renglón en otras tablas como clave foránea.</p>
-------------------------------------	--

Normalización

Significado y beneficios	<ul style="list-style-type: none"> • Elimina la redundancia: usa menos memoria • Mejora la integridad de los datos: Si un dato se guarda dos veces, puede que no se actualice en un lugar y haya errores al buscar cosas en la tabla • Eliminación de anomalías: Sin la normalización puede ser necesario insertar valores incompletos (no inicializados, nulos, etc) • Eficiencia en la consulta y el mantenimiento: Entre más grande y redundante la tabla, más tardadas pueden ser las consultas y facilita hacer cambios a la base.
---------------------------------	---

1NF (1st normal form)	<ul style="list-style-type: none"> • Solo valores atómicos • No hay repeticiones de grupos de datos o columnas que se puedan agrupar en una tabla separada • Cada fila es única y se puede identificar con una clave primaria
------------------------------	--

(Varias columnas para clases → una columna para clase y varios renglones con los mismos registros de alumno y una clase distinta por cada uno)

2NF	<ul style="list-style-type: none"> • cumple con 1nf • todos los campos no clave dependen completamente de la clave primaria
------------	---

One (student_id, name) and one (student_id, class) table; no more redundancy.

3NF

- cumple con 2NF
- No hay dependencias transitivas; Un atributo no clave no depende de otro atributo no clave

one (student_id, name) and one (student_id, class) table; no more redundancy.

Dependencia transitiva

Dadas las dependencias $A \rightarrow B$ y $B \rightarrow C$, entonces existe la dependencia transitiva de C con respecto a A si la dependencia $A \rightarrow B$ no existe directamente. En términos prácticos, A sería una clave primaria, B un atributo no clave y C otro atributo no clave el cual depende de B.

(student_id, class_id, class_name) \rightarrow (student_id, class_id) & (class_id, class_name)

Modelo Lógico

Objetivo

Representar la información de manera lógica con tablas, relaciones y restricciones

Modelo Relacional

Modelo para gestionar bases de datos propuesto por Edgar F. Codd. en 1970

Estructura

Los datos dentro de las tablas se almacenan registros (filas), los cuales representan un objeto real, cada uno con varios atributos (columnas, campos).

Cada fila se debe identificar con una **llave primaria** única.

Las relaciones se establecen con llaves foráneas, columnas con las llaves primarias de otra tabla.

Las relaciones pueden tener cardinalidad

Propiedades de las Tablas

Cada tabla tiene un nombre y éste es distinto del nombre de todas las demás.

Los valores de los campos son atómicos: en cada registro, cada campo toma un solo valor.

No hay dos campos en la misma tabla que se llamen igual.

El orden de los campos no importa.

Cada registro es distinto a los demás.

El orden de los registros no importa.

NULL

La ausencia de un valor

Integridad de datos

- Evita la duplicidad
- Impide almacenar datos incorrectos
- Impide atracción de datos
- Impide la eliminación de información

Reglas de integridad de datos

1. Restricción de dominios (tipos de dato)
2. Regla de integridad de entidades: Ninguna llave primaria puede ser nula
3. Regla de integridad referencial: Cant make a referenced key null?

Reglas de nulos

1. Ningún campo de llave externa puede ser nulo
2. Regla de integridad de entidades: Ninguna llave primaria puede ser nula
3. Regla de integridad referencial: Cant make a referenced key null?

Reglas de llaves externas: Borrado

1. Restringir: No se permite borrar el registro de la tabla dominante (#1) si existe un registro en una tabla subordinada (#n)
2. Propagar: El eliminar la dominante, el DBMS borra la subordinada
3. Anular: El eliminar la dominante, convierte las referencias en la tabla subordinada a NULL

Reglas de llaves externas: Modificación

1. Restringir: Lo mismo cambiando eliniar por modificar
2. Propagar:
3. Anular:

MISSING SYMBOLOGY

Reglas de negocio

Restricciones específicas, dependientes del uso en la vida real, que se imponen a los datos

Álgebra relacional

Definición

Un conjunto de operaciones que se aplican a relaciones (tablas) para producir nuevas tablas como resultados.

Operaciones de conjuntos

- Unión
- Intersección
- Diferencia

Operaciones que remueven parte de una tabla

- Proyección: Table \rightarrow new table with only some columns
- Selección: Table \rightarrow new table with only some rows

Producto carteseano

$$\{(r, s) \mid \forall r \in R, s \in S\}$$

R		S	
A	B	C	D
a ₁	b ₁	c ₁	d ₁
a ₂	b ₂	c ₂	d ₂

R×S			
A	B	C	D
a ₁	b ₁	c ₁	d ₁
a ₁	b ₁	c ₂	d ₂
a ₂	b ₂	c ₁	d ₁
a ₂	b ₂	c ₂	d ₂

Natrual join

Producto cartesiano pero solo con los renglones que cumplan con cierta comparación "=" entre dos columnas

Theta join

Lo mismo pero con mayor, menor, etc.