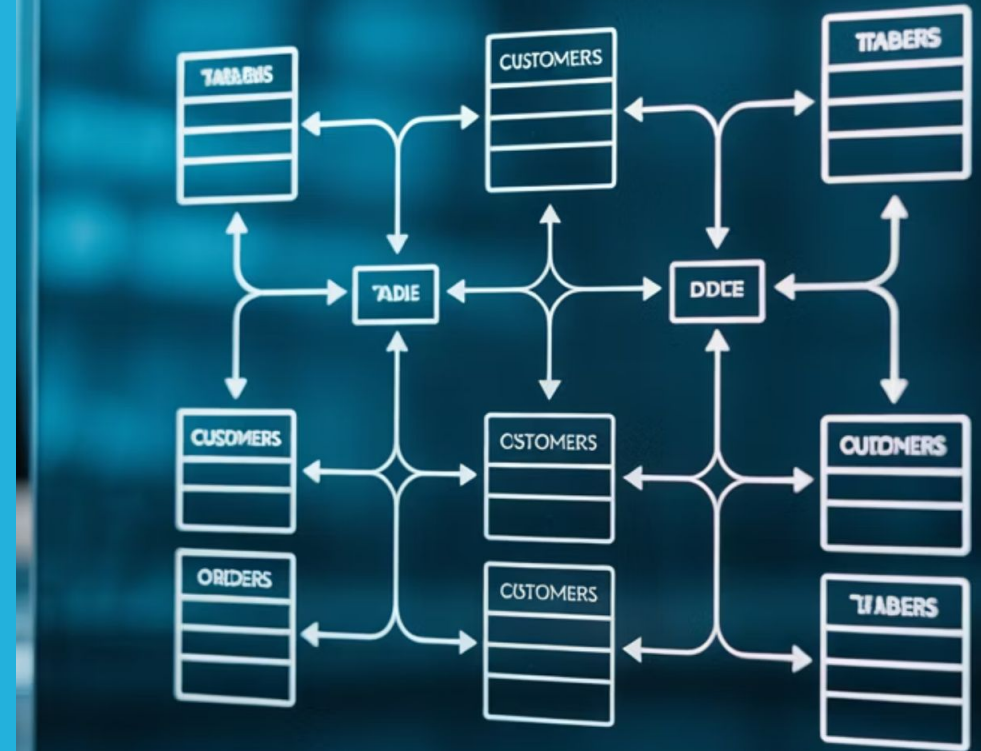


AE5 – Modelado de Datos con Diagramas Entidad-Relación (ERD)

Módulo 5: Fundamentos de Bases de Datos Relacionales

Bootcamp Full Stack Python

Aprendizaje esperado: Elaborar un modelo de datos de acuerdo a estándares de modelamiento para resolver un problema de baja complejidad.



Objetivos de Aprendizaje



Comprender el modelo Entidad-Relación (E-R)



Identificar entidades, atributos y relaciones



Diferenciar entre modelo conceptual y modelo relacional



Transformar un ERD en tablas relacionales



Practicar con MySQL y VS Code



¿Por qué diseñar bases de datos?

Eficiencia en el acceso a los datos

Optimiza la velocidad y rendimiento de las consultas

Garantizar integridad y consistencia

Asegura que los datos sean precisos y confiables

Seguridad y escalabilidad

Permite crecer y adaptarse a nuevas necesidades

Evitar redundancia y errores

Elimina duplicados y minimiza inconsistencias

Ejemplo sencillo:

Imagina una tienda online. ¿Qué pasa si el cliente "Juan Pérez" aparece registrado 10 veces con correos distintos? → Mala experiencia, pérdida de control.

El Proceso de Abstracción



Análisis de requerimientos
¿Qué datos necesito?



Modelo conceptual (E-R)
Dibujar entidades y relaciones



Modelo lógico (relacional)
Convertir a tablas



Modelo físico
Implementar en MySQL

El Modelo Conceptual (E-R)

Componentes principales:

Entidad

Objeto real que almacena datos (Ej: Cliente, Producto)

Atributos

Características de una entidad (Ej: nombre, precio)

Relación

Asociación entre entidades (Ej: Cliente realiza Pedido)



Identificación de Entidades

Ejemplo: Sistema de Biblioteca

Libro

Almacena información sobre los libros disponibles

Usuario

Contiene datos de las personas que usan la biblioteca

Préstamo

Registra cuándo un usuario toma prestado un libro

Regla: una entidad debe ser única y representar algo importante.

Mini actividad:

En parejas, ¿qué entidades tendría un sistema de cine?

Atributos e Identificadores

Atributos: propiedades de la entidad.

Clave primaria (PK): identificador único.

Ejemplo: Usuario

id_usuario (PK)

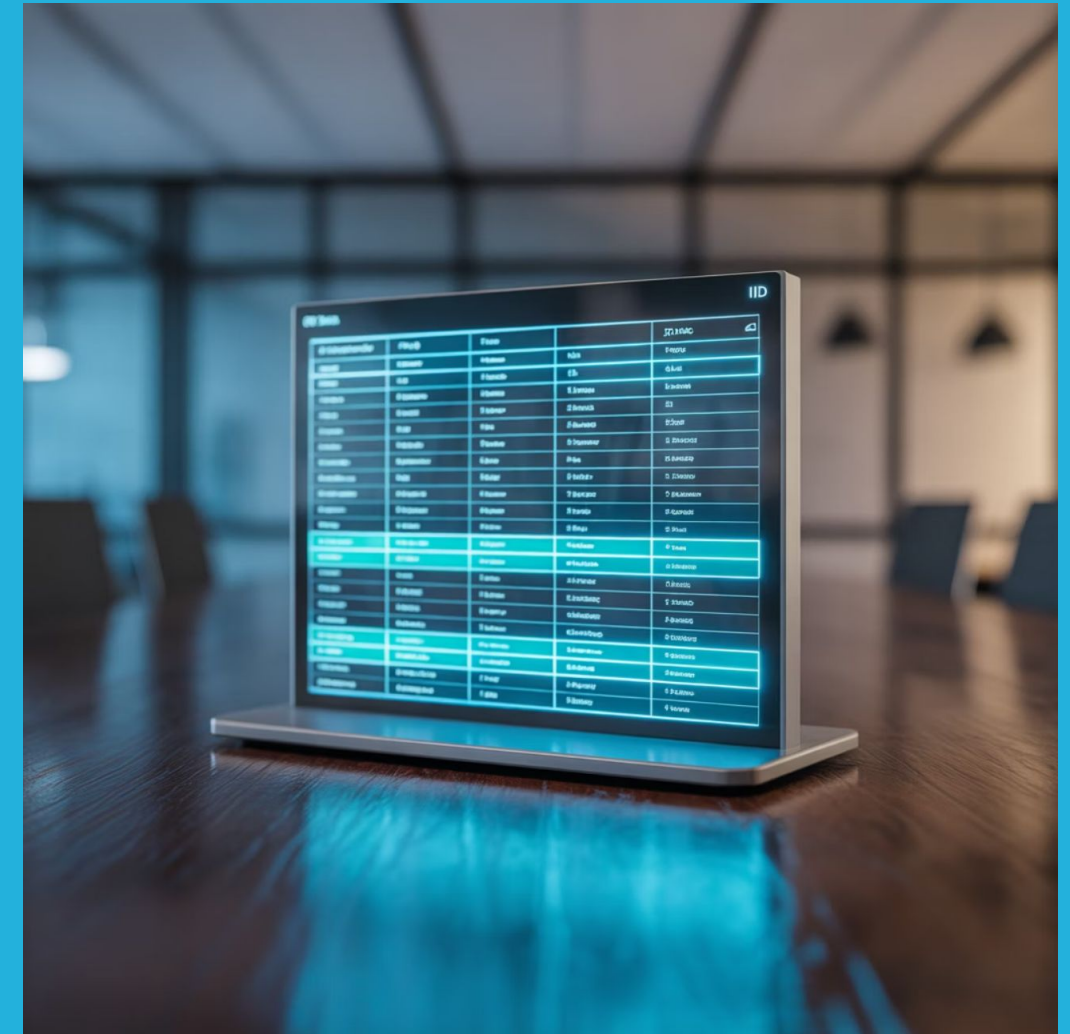
Identificador único del
usuario

nombre

Nombre completo del
usuario

correo (único)

Dirección de email del
usuario

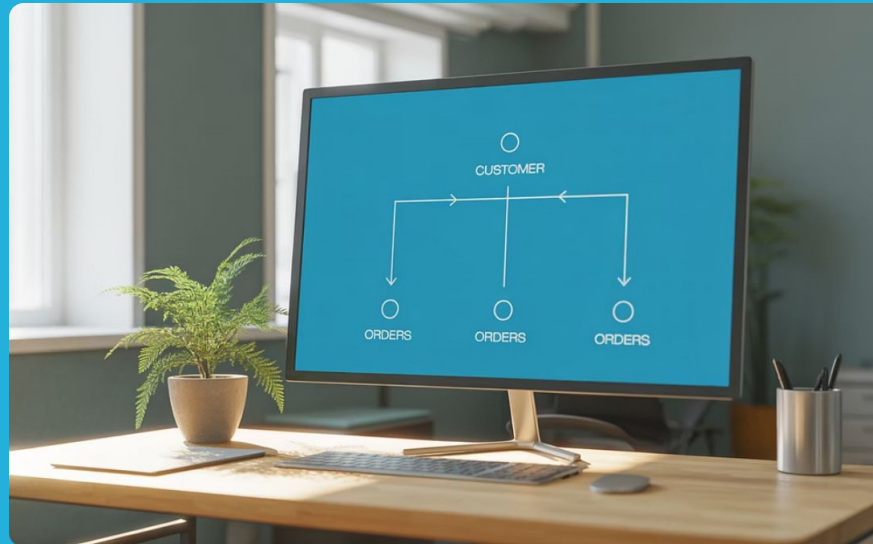


Relaciones entre Entidades



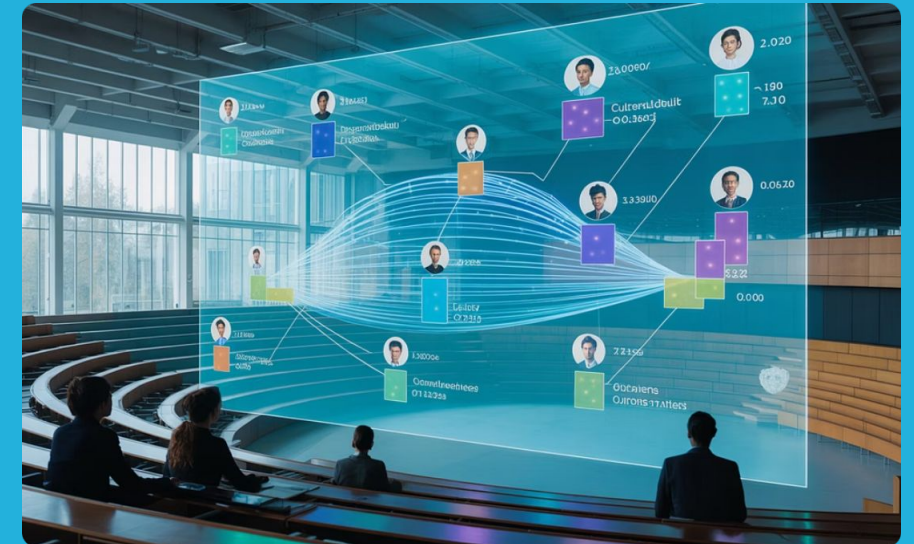
1:1

Un pasaporte pertenece a un ciudadano.



1:N

Un cliente hace muchos pedidos.



N:M

Un estudiante toma varios cursos y un curso tiene varios estudiantes.

Entidades Fuertes y Débiles

Tipos de entidades:

Fuerte

Independiente (Cliente)

Existe por sí misma

Débil

Depende de otra entidad (Pedido depende de Cliente)

No puede existir sin la entidad fuerte

Ejemplo:

Un pedido no existe sin un cliente.



Del Modelo Conceptual al Relacional



Entidades

Se convierten en Tablas



Atributos

Se convierten en Columnas



Relación 1:N

Se implementa con clave foránea (FK)



Relación N:M

Requiere tabla intermedia



Ejemplo de Transformación

Modelo E-R:

Cliente (id_cliente, nombre, email)

Pedido (id_pedido, fecha, id_cliente)

Modelo Relacional (SQL):

```
CREATE TABLE Cliente (  
    id_cliente INT PRIMARY KEY,  
    nombre VARCHAR(100),  
    email VARCHAR(100) UNIQUE  
);  
  
CREATE TABLE Pedido (  
    id_pedido INT PRIMARY KEY,  
    fecha DATE,  
    id_cliente INT,  
    FOREIGN KEY (id_cliente) REFERENCES Cliente(id_cliente)  
);
```


Normalización

Beneficios:

- Elimina redundancia
- Garantiza consistencia
- Mejora la integridad de datos

Ejemplo:

No repetir datos del cliente en cada pedido.



Diccionario de Datos

Ejemplo tabla Cliente:

Campo	Tipo	Restricción	Descripción
id_cliente	INT	PK	Identificador único
nombre	VARCHAR(100)	NOT NULL	Nombre completo
email	VARCHAR(100)	UNIQUE	Correo electrónico

El diccionario de datos documenta cada campo de la base de datos, facilitando su comprensión y mantenimiento.

Aplicación en la Vida Real

Entidades:

- Usuario
- Post
- Comentario
- Like

Relaciones:

- Usuario crea Post (1:N)
- Usuario da Like a Post (N:M)

Ejercicio Práctico en VS Code

Actividad: Crea tablas para este ERD:

Usuario

id, nombre, email

Publicación

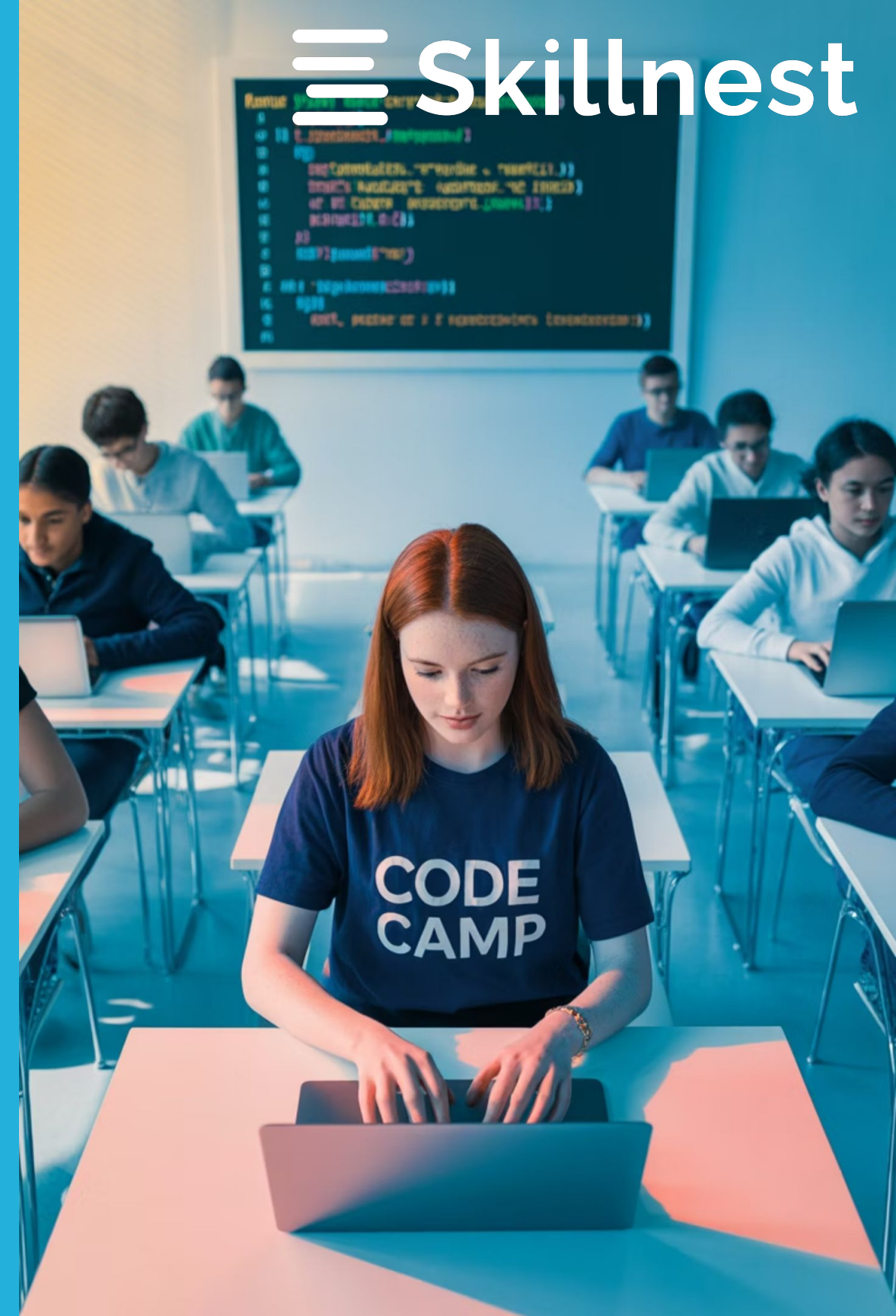
id, texto, id_usuario

Comentario

id, contenido, id_usuario, id_publicacion

Código inicial:

```
CREATE TABLE Usuario (  
  id INT PRIMARY KEY,  
  nombre VARCHAR(100),  
  email VARCHAR(100) UNIQUE  
);
```



Mini Desafío en Grupos


Diseña un modelo E-R para un sistema de "Pedidos de Comida Online"

Entidades mínimas:

- Cliente
- Pedido
- Producto

Relación:

- Cliente realiza Pedido
- Pedido contiene Producto

 Trabajen en grupos para diseñar un diagrama E-R completo que resuelva este problema.



Discusión y Retroalimentación

¿Qué entidades definieron?
Comparen sus modelos con los de otros grupos

¿Cómo resolvieron la relación N:M entre Pedido y Producto?
Expliquen su estrategia para implementar esta relación

¿Qué dudas surgieron?
Compartan los desafíos que encontraron durante el ejercicio

Herramienta MySQL Workbench

Ventajas:

- Permite crear ERDs de forma visual
- Genera automáticamente el SQL
- Ideal para documentar y comunicar el diseño



Cierre Reflexivo

**El buen diseño evita
problemas en el futuro
Invertir tiempo en planificar
ahorra horas de correcciones**

**ERDs ayudan a pensar antes
de programar
Visualizar la estructura antes
de escribir código**

**Lo importante no es
memorizar, sino entender las
relaciones
Comprender los conceptos
fundamentales**

👉 **¿Cómo cambia la calidad de una aplicación cuando la base de datos está bien diseñada?**

Gracias y Próximos Pasos



✓ **Practicar creando ERDs para distintos sistemas**
Aplica lo aprendido a diferentes contextos



✓ **Explorar Workbench y generar código SQL**
Familiarízate con las herramientas profesionales



✓ **Próxima sesión: consultas SQL (SELECT, JOIN)**
Prepárate para aprender a extraer información de las bases de datos

**Innovate.
Collaborate.
Succeed.**