

Oracle

Relaciones entre tablas vistas y procedimientos básicos

Semana #4

Instructora: Kattherine Hernandez

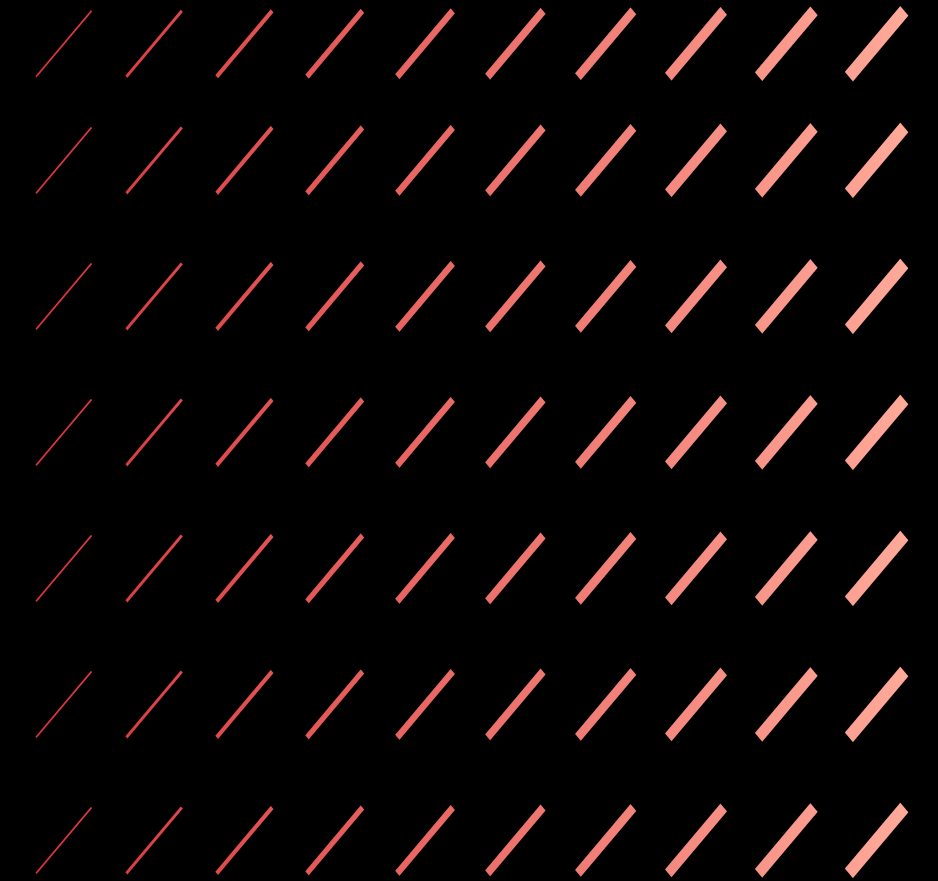
ORACLE®
DATABASE



Día 13:

Relaciones entre tablas

JOINS



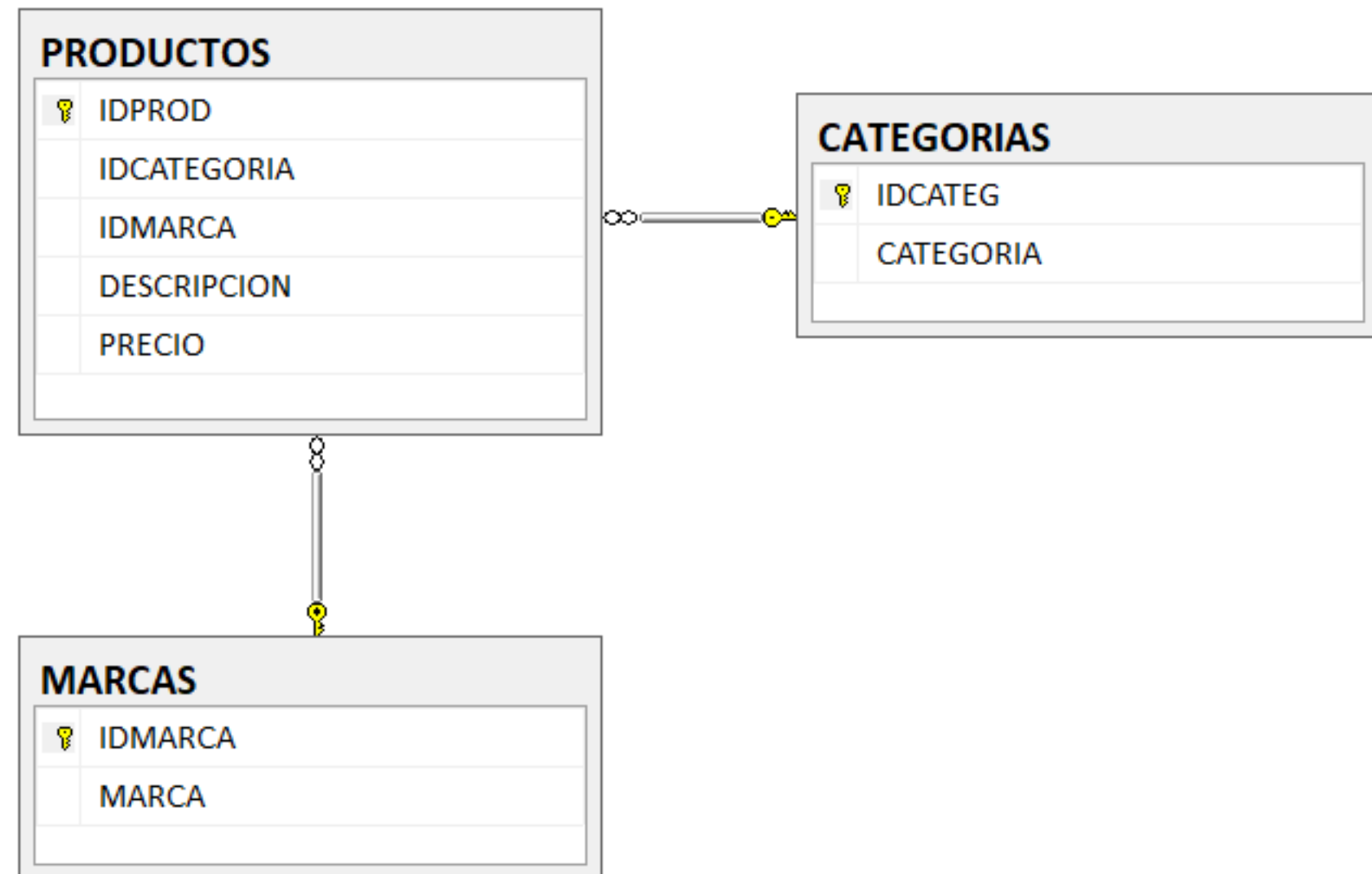
Instr: Kattherine Hernandez

Relaciones entre tablas

Las relaciones entre tablas permiten estructurar bases de datos relacionales, vinculando información de manera lógica y eficiente. Estas relaciones se establecen principalmente usando llaves primarias y llaves foráneas

Pueden ser:

- Uno a muchos (1:N)
- Uno a uno (1:1)
- Muchos a muchos (N:M)



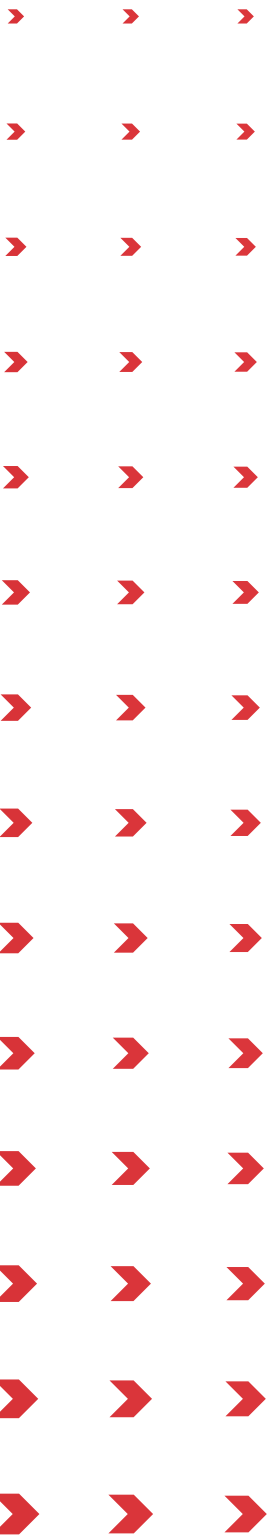
Llaves primarias y foráneas

Llaves primarias:

- Identifican de forma única cada fila en una tabla.
- Aseguran integridad de datos, evitando duplicados.
- Son esenciales para que otras tablas puedan referirse a ella.

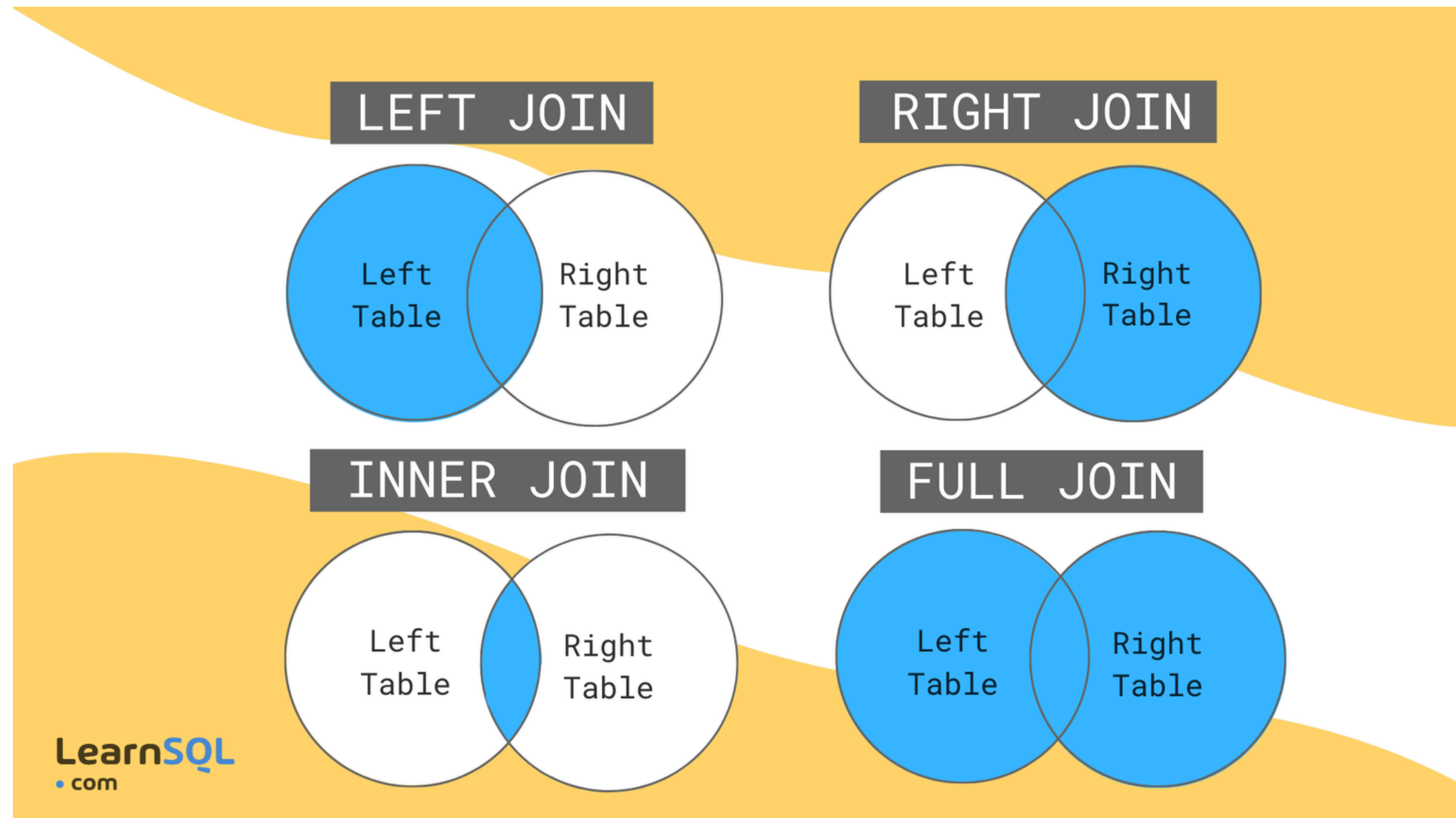
Llaves foráneas:

- Vinculan una tabla con otra, estableciendo relaciones.
- Garantizan integridad referencial, asegurando que los valores existan en la tabla relacionada.
- Permiten operaciones como ON DELETE CASCADE o ON UPDATE CASCADE.



Sentencia JOIN

Los joins se usan en consultas para combinar datos relacionados.

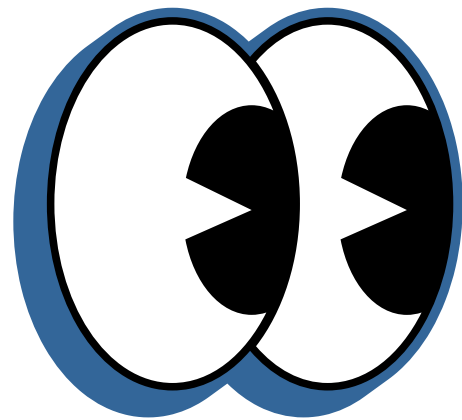


Sentencia JOIN

Los joins usan llaves primarias y foráneas para combinar tablas de manera lógica, relacionando filas basadas en estas claves.

```
SELECT T1.COLUMN1, T2.COLUMN2  
FROM TABLA1 T1  
INNER JOIN TABLA2 T2  
ON T1.LLAVE_PRIMARIA = T2.LLAVE_FORANEA;
```

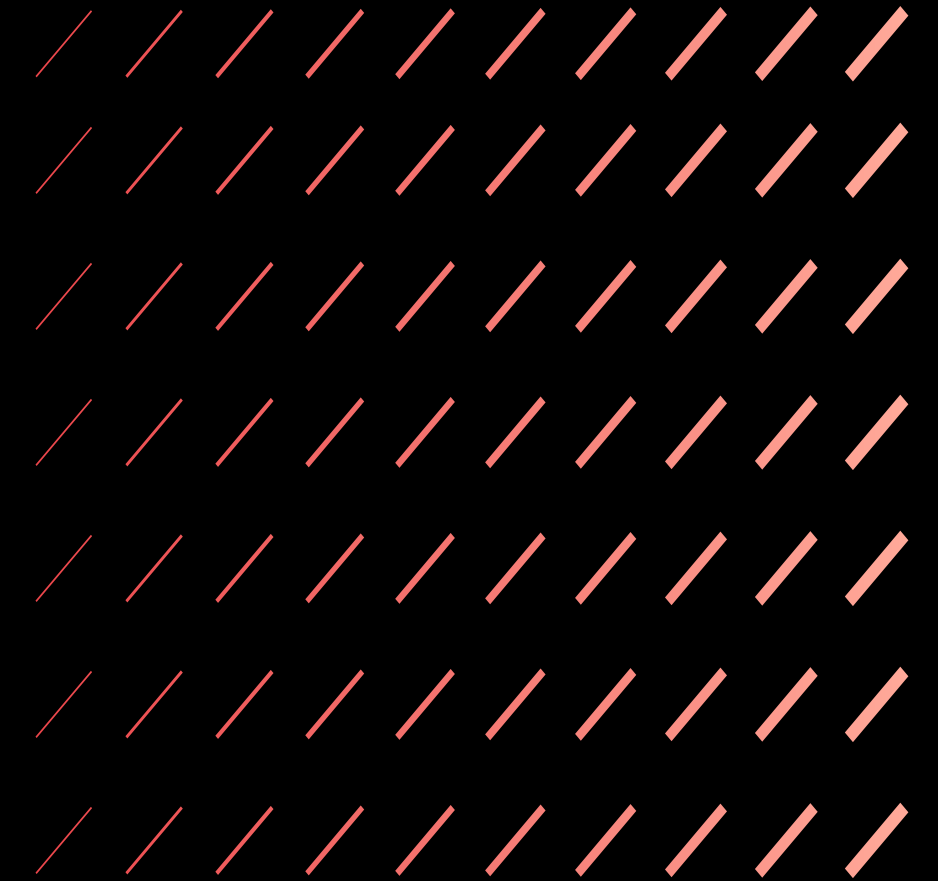
Consideraciones



- Usar llaves foráneas para garantizar que los valores relacionados existan en la tabla principal.
- Asegurarse de que los campos relacionados (clave primaria y clave foránea) compartan el mismo tipo de dato.
- Usar restricciones (NOT NULL, UNIQUE) para reforzar la integridad de los datos.
- Usar nombres claros como `cliente_id` o `producto_id` para facilitar la comprensión.
- Diseñar relaciones claras para evitar referencias circulares o datos duplicados.
- Mantener el mismo estilo y estándares para relaciones en todo el esquema.



Día 14: Vistas



Instr: Kattherine Hernandez

Vistas

Son consultas predefinidas que actúan como tablas virtuales. Son útiles para simplificar consultas complejas, mejorar la seguridad (al mostrar solo ciertos datos) y facilitar el acceso a datos frecuentes.

CREATE VIEW

CREATE OR REPLACE VIEW

DROP VIEW

```
CREATE VIEW NOMBRE_VISTA AS  
SELECT COLUMNAS  
FROM TABLAS  
WHERE CONDICIONES;
```

Ventajas

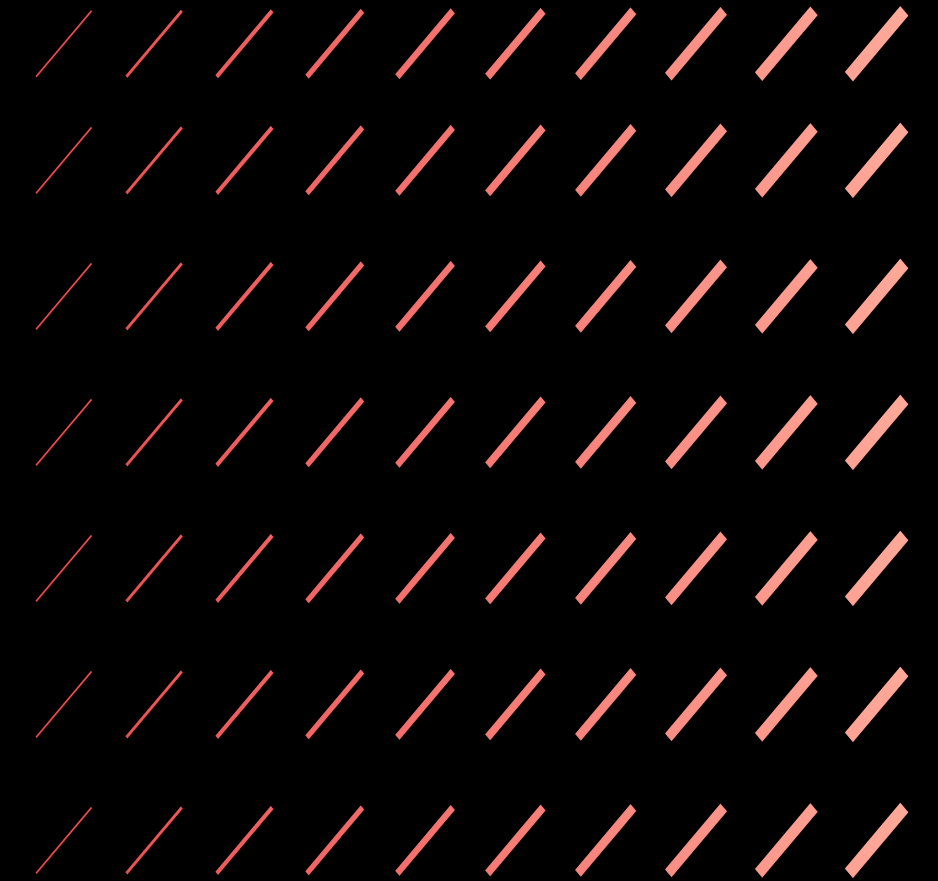


- Simplifican consultas complejas.
- Mejoran la seguridad al restringir el acceso directo a las tablas.
- Proveen abstracción, ocultando la estructura de las tablas subyacentes.
- Facilitan la reutilización de lógica SQL.



Día 15:

Procedimientos y funciones básicas



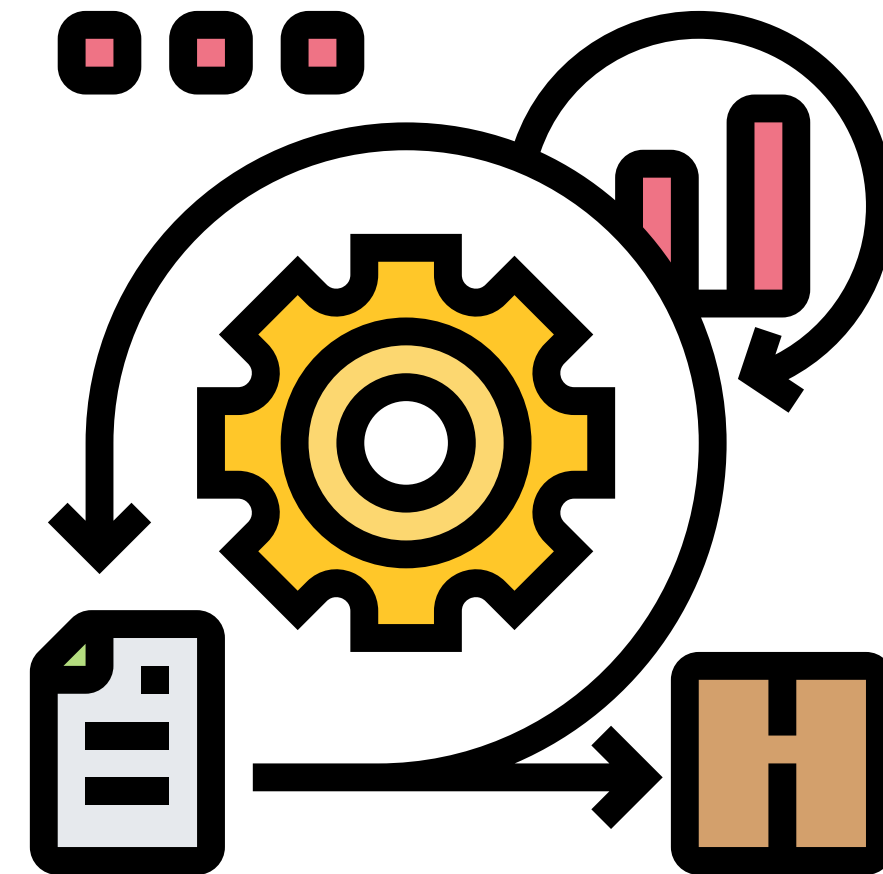
Instr: Kattherine Hernandez

Procedimientos almacenados



Son bloques de código almacenados en el servidor que se pueden ejecutar con una sola llamada.

Permiten encapsular lógica de negocio, mejorar el rendimiento y la seguridad.



Los SP

- Se ejecutan mediante comandos SQL o triggers.
- No devuelven un valor, pero pueden modificar datos o devolver valores a través de parámetros.

```
CREATE OR REPLACE PROCEDURE registrar_estudiante (  
    p_nombre    IN VARCHAR2,  
    p_apellido  IN VARCHAR2,  
    p_edad      IN NUMBER  
)  
AS  
BEGIN  
    INSERT INTO ESTUDIANTES (NOMBRE, APELLIDO, EDAD)  
    VALUES (p_nombre, p_apellido, p_edad);  
    COMMIT; -- Asegura que los cambios se guarden en la base de datos.  
END;
```

```
BEGIN  
    registrar_estudiante('Katherine', 'Hernández', 22);  
END;
```

Funciones



Una función es un bloque de código que devuelve un valor y se puede utilizar dentro de una consulta.

$$f(x)$$


```
CREATE OR REPLACE FUNCTION calcular_cuadrado (  
    p_numero IN NUMBER  
) RETURN NUMBER  
AS  
    v_cuadrado NUMBER; -- Variable para almacenar el resultado.  
BEGIN  
    v_cuadrado := p_numero * p_numero;  
    RETURN v_cuadrado; -- Devuelve el resultado.  
END;
```

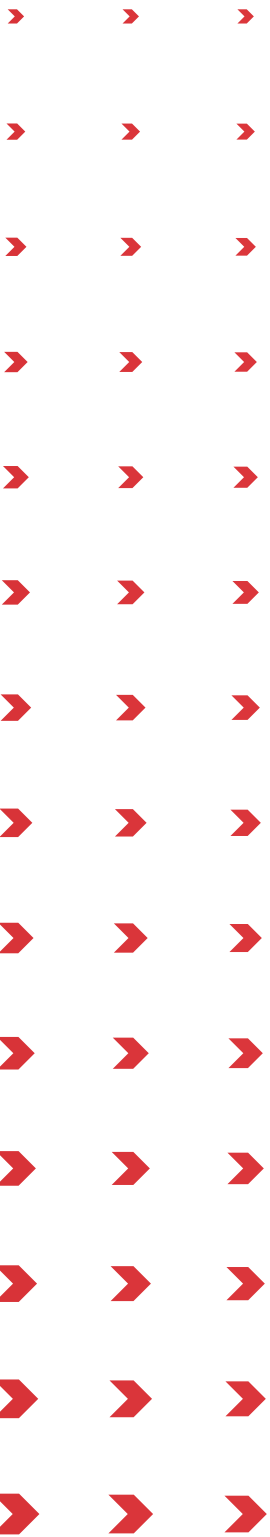
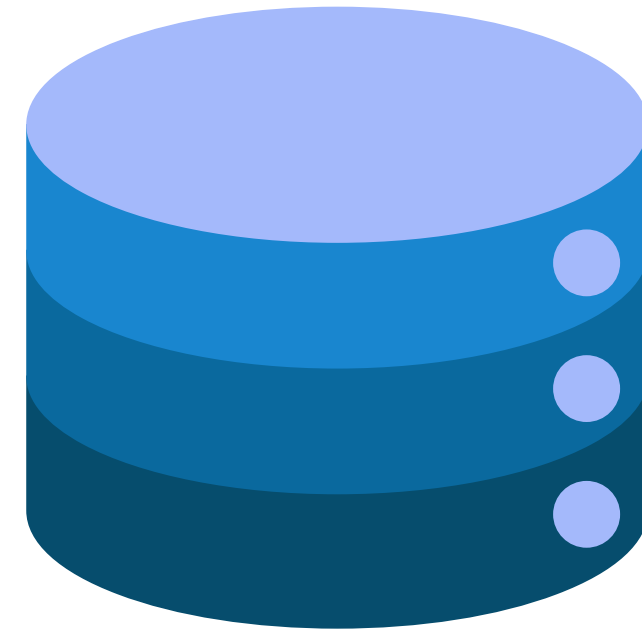
```
SELECT calcular_cuadrado(5) AS resultado FROM dual;  
-- Resultado: 25
```

Procedimientos

- No devuelven valores directamente, pero pueden modificar tablas o datos.

Funciones

- Devuelven un valor único y son utilizadas principalmente en cálculos.



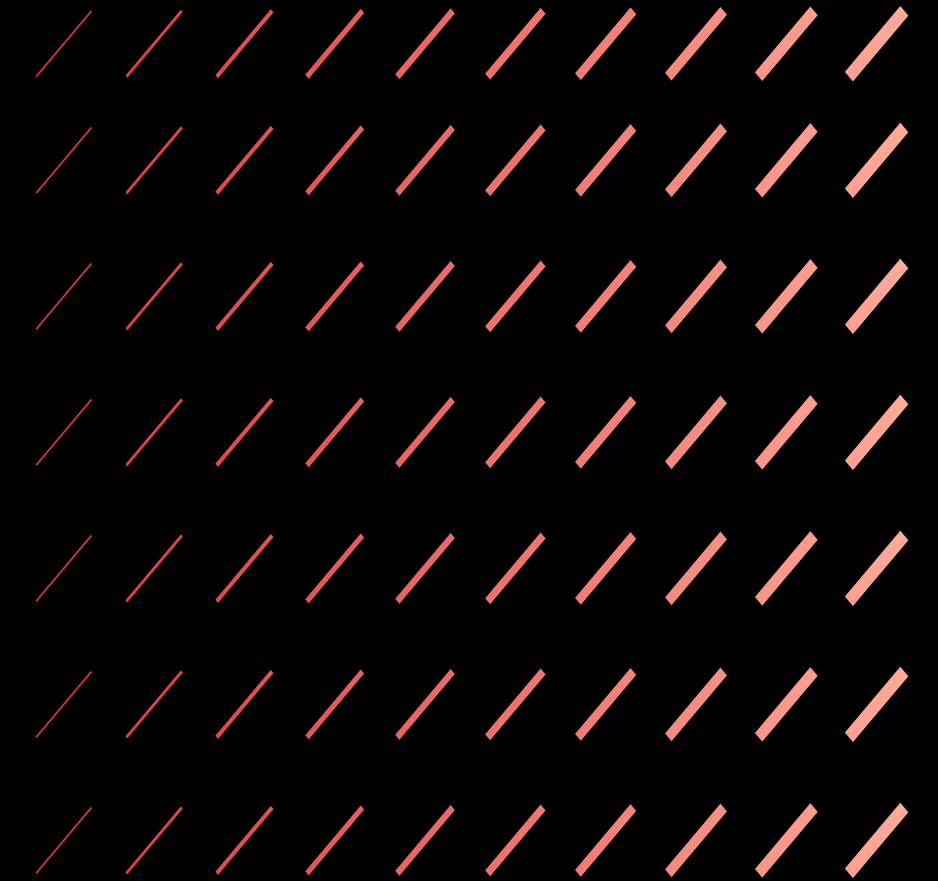
1. Crear un procedimiento que actualice el salario de un empleado en una tabla EMPLEADOS.
2. Crear una función que devuelva el salario anual de un empleado dado su salario mensual.





Día 16:

Proyecto final



Instr: Kattherine Hernandez

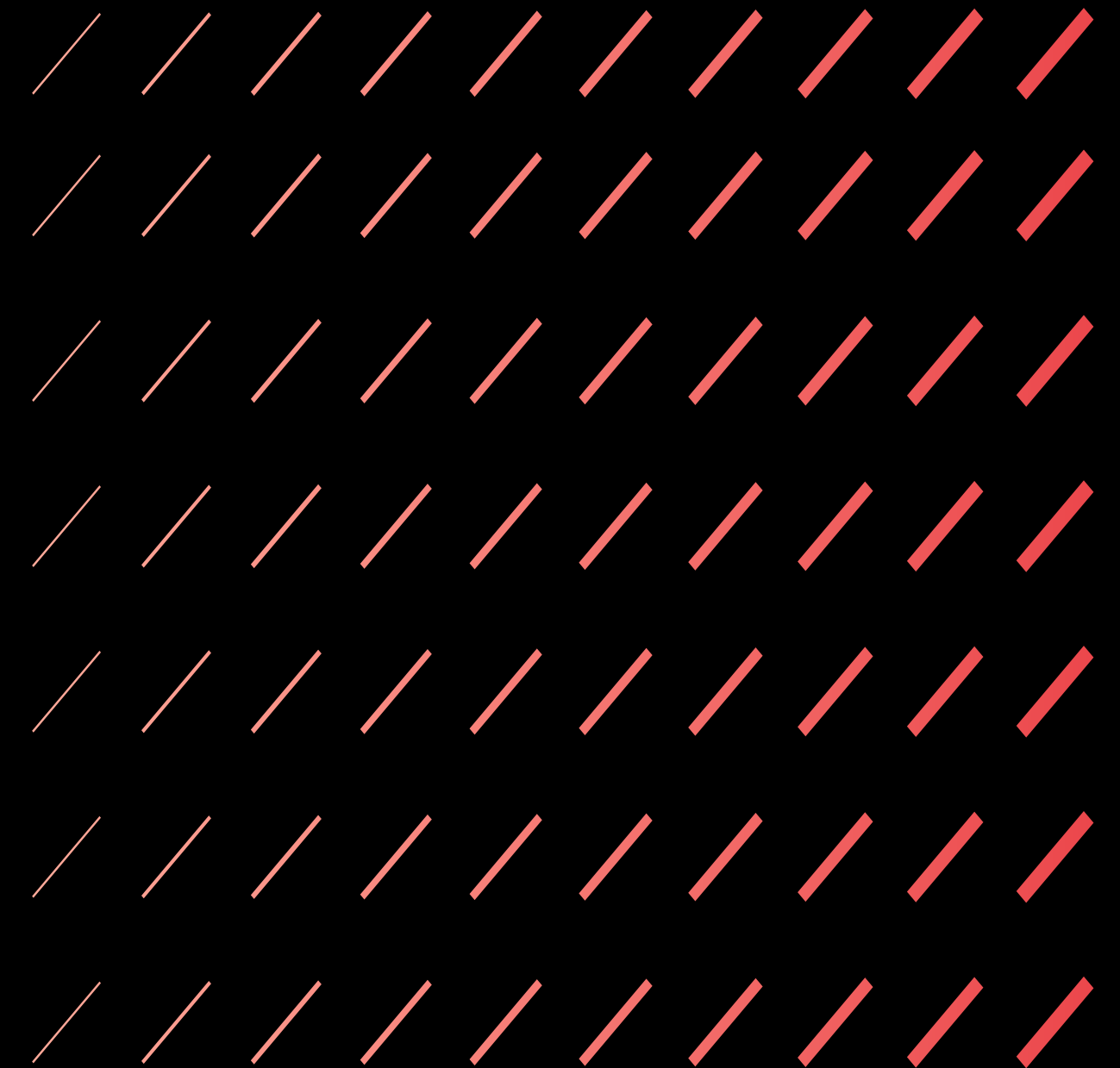


Proyecto

Desarrollar un sistema básico que permita gestionar libros, autores y préstamos de una biblioteca.

1. Cree tres tablas relacionadas.
2. Implemente un procedimiento almacenado para registrar nuevos préstamos.
3. Cree una función para calcular el total de préstamos realizados por un usuario.

**¡Felicidades por haber
llegado hasta el final!**



Instr. Katherine Hernandez