

**HENRY**

Data Science



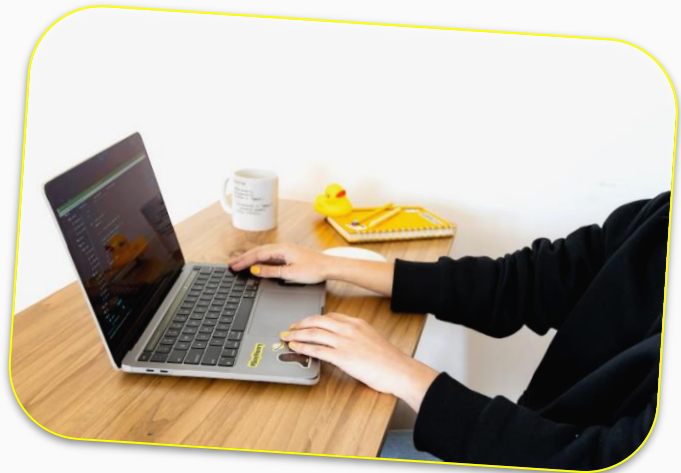
# **Clase 5**

## **sistemas de gestión de bases de datos**





# Agenda



- Modelos relacionales
- Introducción a SQL
- Introducción a bases de datos



# **OBJETIVOS DE LA CLASE**

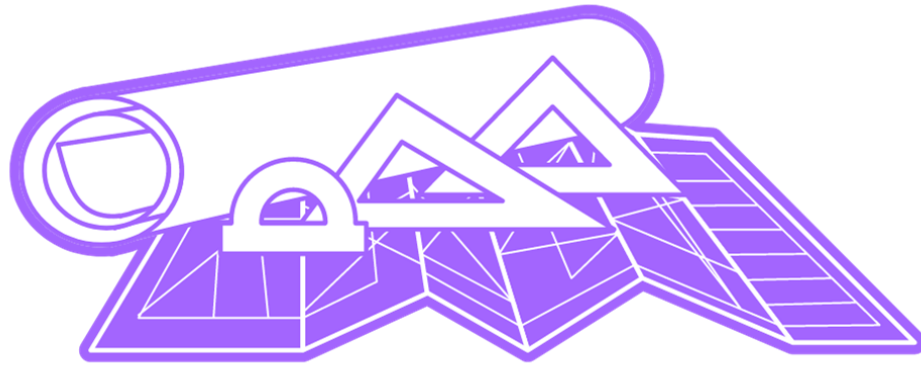
*Al finalizar esta lecture estarás en la capacidad de...*

- **Reconocer** el concepto de Sistemas de Gestión de Bases de Datos y de Modelo de Entidad Relación.
- **Conocer** el Lenguaje Estructurado de Consulta (SQL).
- **Comprender** la diferencia entre bases de datos transaccionales y analíticas.

# **Modelos relacionales**



El **modelo de datos** es el equivalente al plano de un edificio y representa de forma conceptual aquello que quiere diseñar.





---

## En pocas palabras...

El **modelado de datos** es el proceso mediante el cual se definen los requisitos de negocios y se diseñan las mejores estructuras de datos para soportarlo.



---

# **Modelo entidad-relación**



## Para modelar la realidad y traducirla en estructuras...



Se usa el modelo de **ENTIDAD-RELACIÓN**.



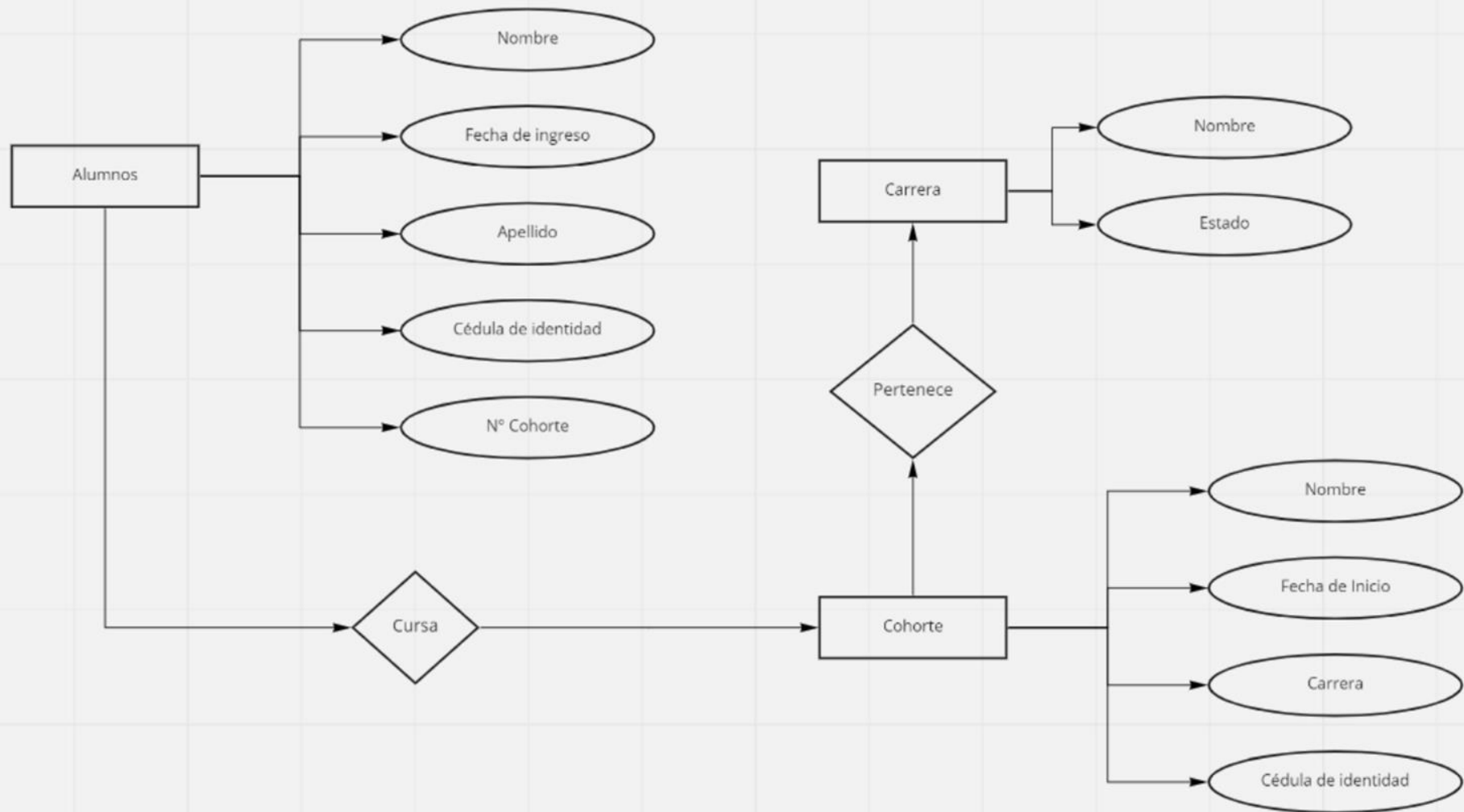
Para construirlo se basa en el relevamiento del modelo de negocios de la organización a través de **entidades**, **atributos** y **relaciones**.





---

# Relaciones

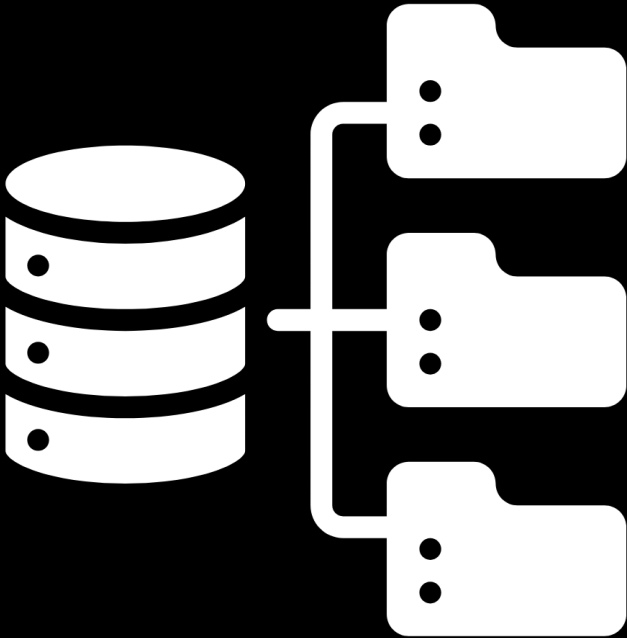




---

# **Bases de datos**

# ¿Qué son?



Colección de datos almacenados de forma coherente y permanente.

Proviene de entidades, objetos o hechos de la realidad.



**Se  
representa  
mediante  
dos  
elementos**

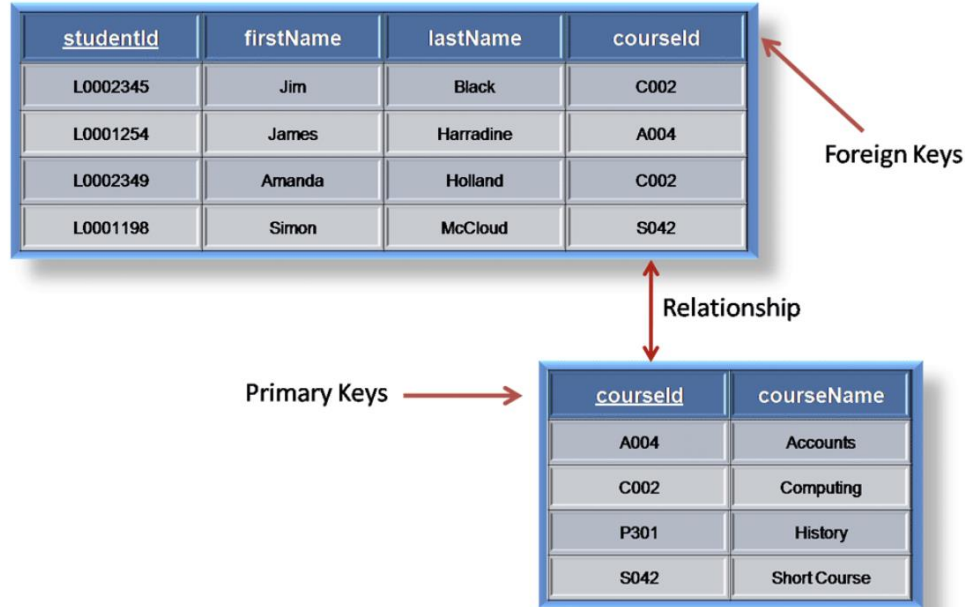
**Primary key:** única e inequívoca a un elemento (registro) de la entidad.

**Foreing key:** es una **primary key** en otra tabla. Hace referencia a los valores relacionados de otra entidad.



---

# **Tipos de relaciones**



Las relaciones pueden ser 1-1 (uno a uno), 1-M (uno a muchos), N-M (mucho a muchos), lo que se denomina cardinalidad.



---

# Introducción a SQL





# ¿Qué es SQL?



SQL por sus siglas en inglés significa **Lenguaje de Consulta Estructurada** (Structured Query Language).



Lenguaje diseñado para interactuar con las bases de datos relacionales. SQL se subdivide a su vez entre distintos tipos de **sublenguajes** como DDL, DML, DCL y TCL.

# Sublenguajes



DDL: Data Definition Language

DML: Data Manipulation Language

DCL: Data Control Language

TCL: Transaction Control language



# DDL: Data Definition Language

---



Sentencias que permiten definir la estructura de una base de datos, esta estructura está compuesta por “objetos” (no confundir con POO en Python).



Los tipos de objetos que se pueden generar son: bases de datos, tablas, vistas, procedimientos.



Las acciones que se pueden ejecutar son: crear, modificar, eliminar.



## Bases de datos

```
CREATE DATABASE henry; # Crear.  
ALTER DATABASE henry; # Modificar.  
DROP DATABASE henry; # Borrar.
```



# Tablas

```
CREATE TABLE alumno (  
    cedulaIdentidad INT NOT NULL AUTO_INCREMENT,  
    nombre VARCHAR(20),  
    apellido VARCHAR(20),  
    fechaInicio DATE,  
    PRIMARY KEY (cedulaIdentidad)  
);
```

```
ALTER TABLE alumno  
ADD direccion VARCHAR(20);
```

```
ALTER TABLE alumno  
CHANGE direccion direccion VARCHAR(50);
```

```
DROP TABLE alumno;
```



---

# Vistas

```
CREATE VIEW datosAlumnos AS  
SELECT *  
FROM alumno;
```

```
ALTER VIEW datosAlumnos AS  
SELECT *  
FROM alumno;
```

```
SELECT *  
FROM datosAlumnos;
```

```
DROP VIEW datosAlumnos;
```

---

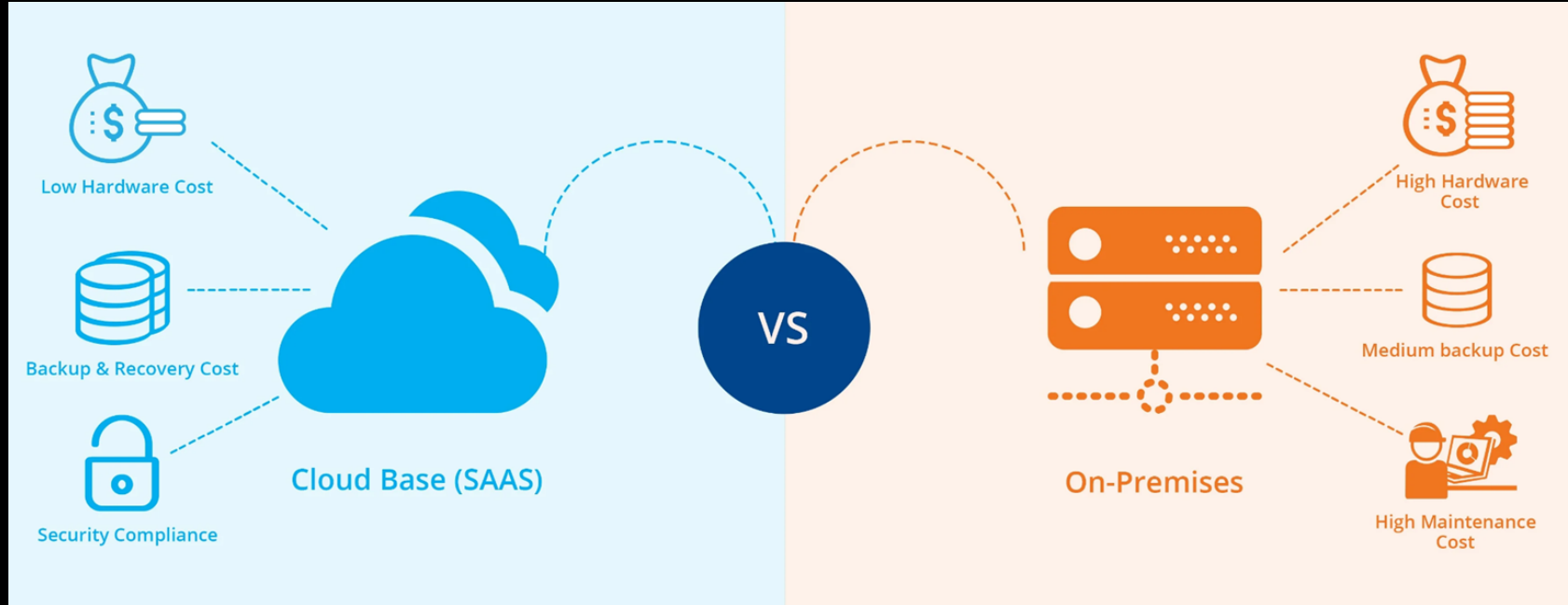


# Procedimientos

```
DELIMITER $$  
CREATE PROCEDURE contarAlumnos (IN ape VARCHAR(20))  
    BEGIN  
        SELECT COUNT(*) FROM alumno WHERE apellido = ape;  
    END $$  
DELIMITER ;  
  
CALL contarAlumnos("Alvarez");  
  
DROP PROCEDURE contarAlumnos;
```



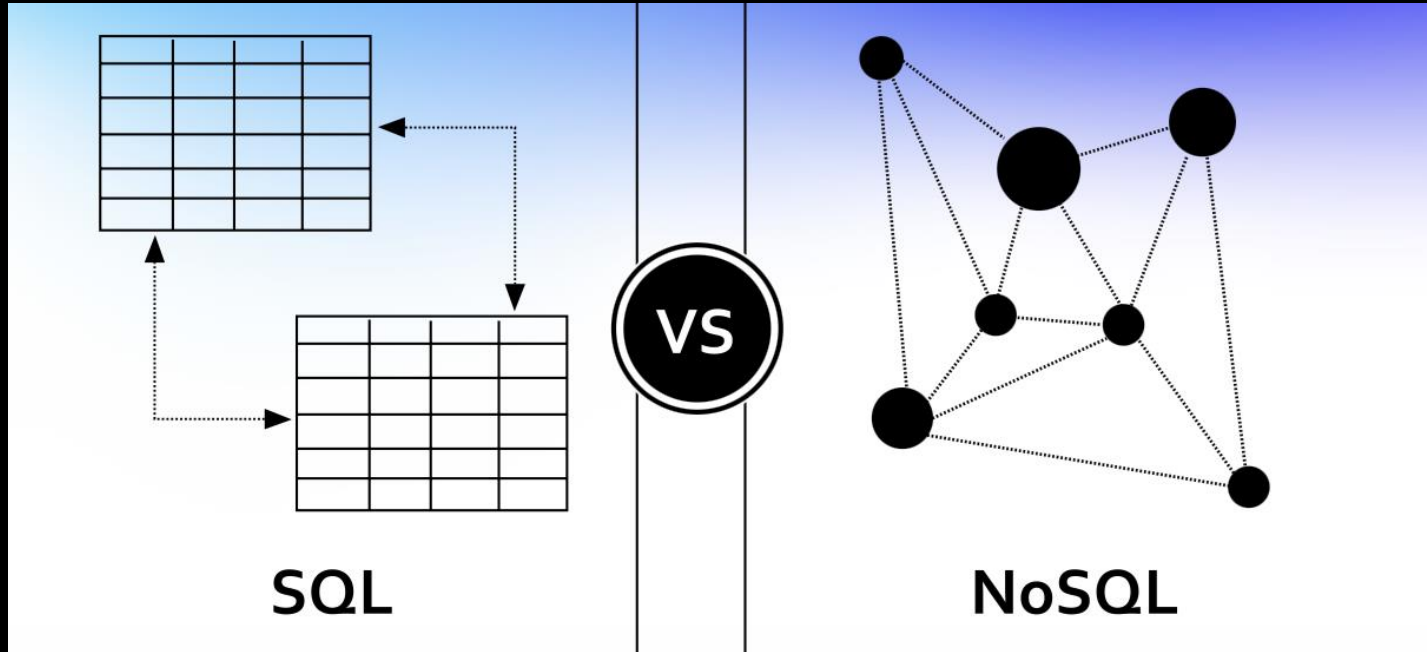
# On-Premise VS cloud





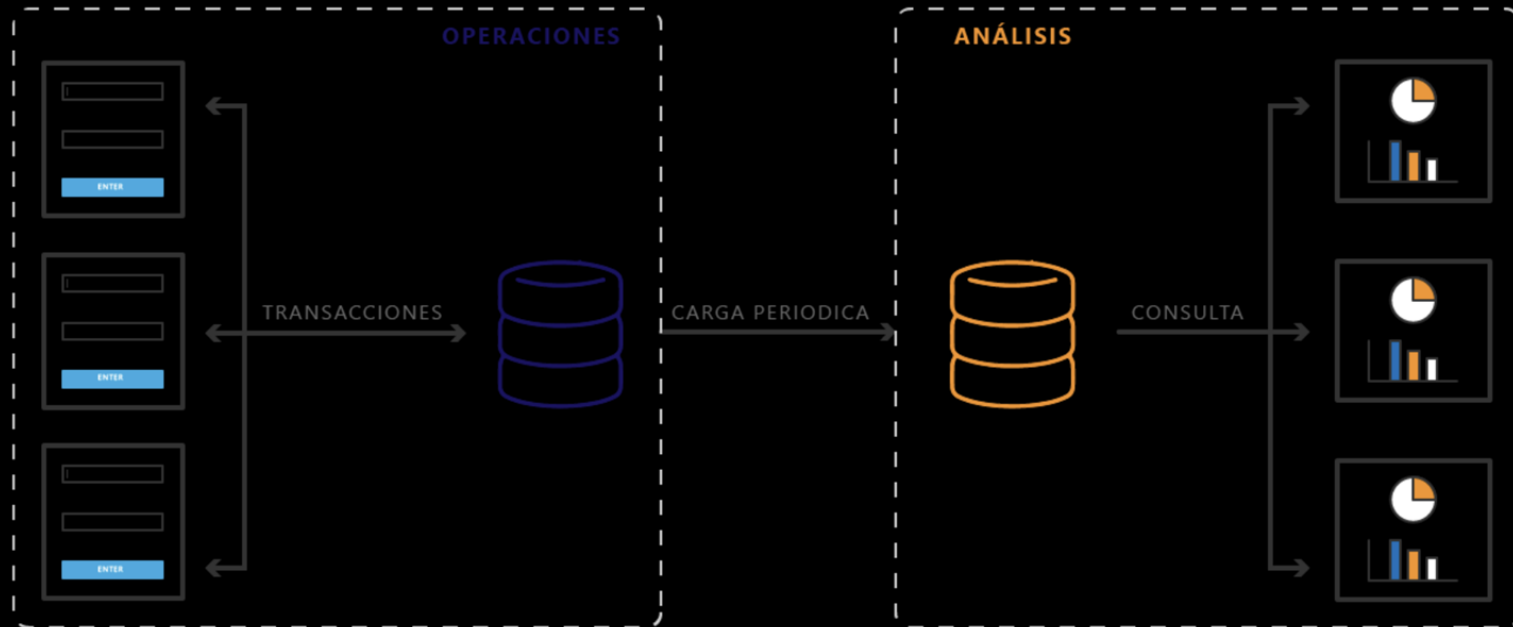


# Relacionales vs No relacionales





# Analíticas vs Operacionales



# HENRY

