



# INTRODUCCIÓN A LAS BASES DE DATOS

- Introducción al SQL.
- DDL.





**UNAC**  
CORPORACIÓN UNIVERSITARIA ADVENTISTA  
COLOMBIA

# Elementos del modelo relacional

**Relación (tabla):**

Concepto abstracto. Estructura: filas (tuplas) y columnas (atributos).



**Relación:** Conjunto de tuplas

Propiedades:

Cada “**celda**” (intersección de una fila con una columna) es atómica o univaluada.

Cada **columna** tiene un **nombre** único dentro de la relación.

Las columnas no tienen orden.

Las tuplas no tienen orden.

No hay tuplas repetidas.





**UNAC**  
CORPORACIÓN UNIVERSITARIA ADVENTISTA  
COLOMBIA

# Elementos del modelo relacional

- **Tupla:** Una relación es un conjunto de tuplas:  
Cada instancia (fila o registro) de una relación.  
**Cardinalidad** o extensión de una relación: número de tuplas.



- **Atributo:**  
Cada campo o columna de una relación.  
El conjunto de atributos forman la cabecera de la relación.  
Cada atributo está valuado o basado sobre un dominio.



**UNAC**  
CORPORACIÓN UNIVERSITARIA ADVENTISTA  
COLOMBIA

# Elementos del modelo relacional

## Claves (Keys):

Clave **candidata** de una relación:

Atributo (puede ser compuesto) cuyos valores diferencian a una tupla de las demás.

Debe cumplir **unicidad** y minimalidad (irreducibilidad).

Clave primaria de una relación:

Elegida a partir de las claves candidatas de la relación.

Las no elegidas (si las hay) se denominan claves alternativas o secundarias.

Clave **foránea** (ajena o externa):

Atributo (puede ser compuesto) en una relación R1 que referencia a la clave primaria (o alternativa) de una relación R2 (R1 y R2 no necesariamente diferentes).

Especifica de forma explícita la forma en que dos relaciones se relacionan.



# Elementos del modelo relacional

## Comparativo de términos

Relacional formal	Informal	Sistema de archivos
Relación	Tabla	Archivo
Tupla	Fila	Registro
Atributo	Columna	Campo
Instancia de una relación	Conjunto de filas de una tabla	Conjunto de registros de un archivo
Cardinalidad	Número de filas	Número de registros
Grado o aridad	Número de columnas	Número de campos
Clave primaria	Llave, identificador	Sin equivalente
Clave foránea	Llave externa o ajena, interrelación	Sin equivalente
Dominio	Conjunto de valores válidos	Conjunto de valores válidos



# SQL: DDL

## DDL:

- Lenguaje de Definición de Datos
- Permite crear objetos en la BD

## Tipos de objetos:

- **Tablas:** corresponden a las relaciones del modelo relacional
- **Índices:** estructuras que pueden ayudar a mejorar el desempeño de algunas consultas
- **Vistas:** son “tablas virtuales”\* Recordar nivel externo
- **Databases.**
- Otros...



# SQL: DDL

Una tabla se crea así:

**CREATE TABLE** nombre\_tabla

( **atributos**: cada uno con su

**tipo de datos** y

**restricciones**: de no nulidad, de clave primaria, de clave foránea, de clave alternativa, etc.

);

Un índice se crea así:

**CREATE INDEX** nombreíndice ON tabla(columna(s));

Una vista se crea así:

Ejemplo:

**CREATE VIEW** nombre\_vista **AS** consulta\_SQL;





# SQL: DDL

## Restricciones de integridad

Garantizan que los cambios hechos a la BD no ocasionen inconsistencias (según las reglas del negocio) en los datos.

1.1. Restricciones de dominio (tipo de datos): Conjunto de valores y de operaciones permitidas sobre ellos.

### Dominios esenciales de SQL (TIPO DE DATO):

**CHAR(p)**: Cadena de caracteres de longitud fija p. Máxima longitud p.

**VARCHAR(p)\***: Cadena de caracteres de longitud variable. Máxima longitud p.

**NUMBER(p,s)**: Valor numérico de precisión p y escala s (para el manejo de decimales)

**DATE** : Fechas.

**INTEGER**: Valores enteros

El programador puede definir sus propios tipos de datos ? Objeto-relacional.





**UNAC**  
CORPORACIÓN UNIVERSITARIA ADVENTISTA  
COLOMBIA

# Elementos del modelo relacional

## Restricciones de integridad

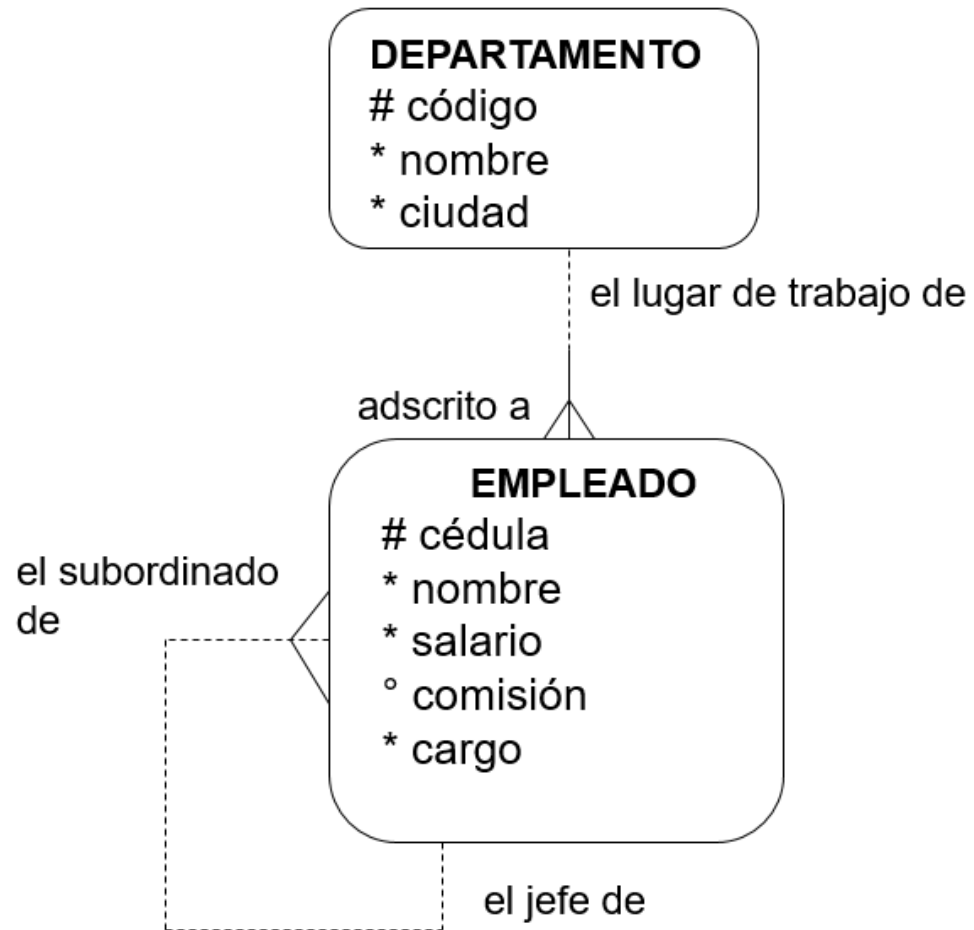
Garantizan que los cambios hechos a la BD no ocasionen inconsistencias (según las reglas del negocio) en los datos.



- **Nulos:** Un atributo puede o no admitir nulos. En SQL se especifica mediante la cláusula NOT NULL.
- **Integridad referencial:** Claves foráneas (CF). En SQL se especifica mediante las cláusulas REFERENCES y FOREIGN KEY
- **Clave Primaria (CP):** Garantiza la unicidad y obligatoriedad del o de los atributos que conforman la CP. En SQL se especifica mediante la cláusula PRIMARY KEY.



# Tomemos como ejemplo





# Define Table



```
CREATE TABLE departamento
( codigo numeric(6) PRIMARY KEY,
  nombre VARCHAR(6) NOT NULL
  UNIQUE,
  ciudad VARCHAR(12) CHECK
(ciudad IN
      ('Medellín', 'Bogotá',
      'Cali')) NOT NULL
);
```





# References

```
CREATE TABLE empleado  
( cedula numeric(10) PRIMARY KEY,  
  nombre VARCHAR(30) NOT NULL,  
  salario numeric(10,2) NOT NULL,  
  comision numeric(2) ,  
  cargo VARCHAR(20) NOT NULL,  
  jefe numeric(10) REFERENCES  
  empleado,  
  depto numeric(6) NOT NULL  
  REFERENCES      departamento  
  
);
```

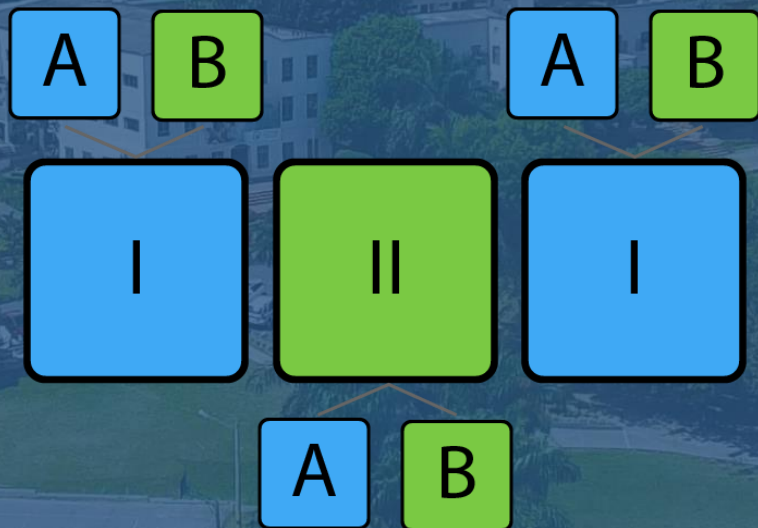


# CHECK

```
CREATE TABLE empleado
(cedula numeric (10) PRIMARY KEY,
nombre VARCHAR(30) NOT NULL,
salario numeric (10,2) NOT NULL
    CHECK(salario > 0),
comision numeric(3) NOT NULL CHECK
    (comision BETWEEN 0 AND 100),
cargo VARCHAR(20) NOT NULL,
jefe numeric (10) REFERENCES
    empleado(cedula),
depto numeric (6) NOT NULL REFERENCES
    departamento
);
```



# Especificación de una CP compuesta:

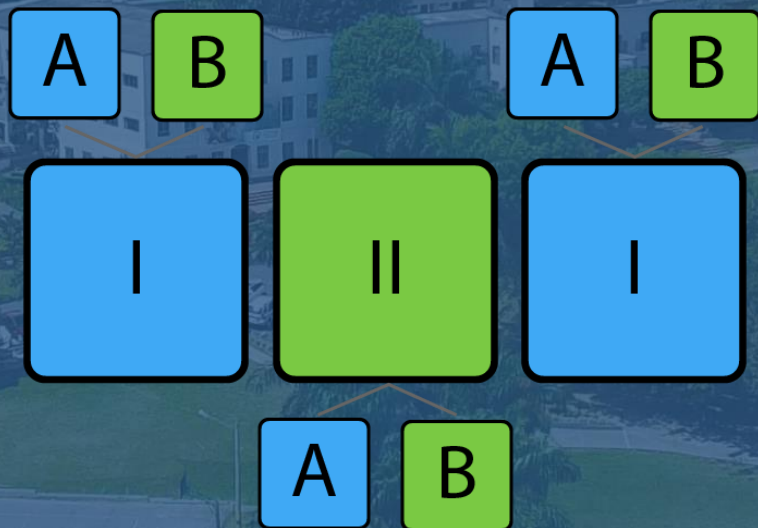


```
CREATE TABLE envio(  
    snro NUMBER(6),  
    pnro NUMBER(6),  
    cantidad NUMBER(6) NOT NULL,  
    PRIMARY KEY (snro, pnro)  
);
```





# Especificación de una CP compuesta:



```
CREATE TABLE revision(  
  codrevision numeric(5) PRIMARY KEY,  
  mi_snro numeric(6) NOT NULL,  
  mi_pnro numeric(6) NOT NULL,  
  revisor VARCHAR(20) NOT NULL,  
  CONSTRAINT fk_envio  
  FOREIGN KEY (mi_snro, mi_pnro) REFERENCES  
    envio  
);
```





**UNAC**  
CORPORACIÓN UNIVERSITARIA ADVENTISTA  
COLOMBIA

# Muchas Gracias

