

# Capítulo IV

## Diseño Lógico-Modelo Relacional

**Identificadores y Restricciones de Integridad**

# \* IDENTIFICADORES SQL

- Se utilizan para identificar objetos como por ejemplo, nombres de tablas, nombre de vistas y de columnas.
- Restricciones del SQL estándar:
  - No pueden tener mas de 128 caracteres de longitud
  - Debe comenzar con una letra
  - No pueden contener espacios en blanco
  - Se permiten letras del alfabeto mayúscula y minúscula y carácter de subrayado.

TIPO DE DATO	DECLARACIONES			
Booleano	Boolean			
Carácter	Char	Varchar		
Bit	Bit	Bit varying		
Numérico exacto	Numeric	Decimal	Integer	Smallint
Numérico aproximado	Float	Real	Double precision	
Fecha y hora	Date	Time	Timestamp	

TIPO DE DATO	DECLARACIONES
Booleano	<p>Boolean: True y False donde True &gt; False A menos que una restricción NOT NULL lo especifique, acepta un UNKOWN (descoconocido)</p>
Carácter	<p>CHARACTER (VARYING) (longitud) Puede abreviarse como</p> <p>Char: en el caso de tipo de dato carácter de longitud fija. Ejemplo: nombre_pila CHAR (10) Si almacenamos Ana, los siete restantes a la derecha se rellenan con blancos, almacenandose ANA<b>ppppppp</b>.</p>
	<p>VARCHAR: en el caso de longitud variable, sólo se almacenan los caracteres almacenados. Ejemplo: nombre_pila CHAR (10) Si almacenamos Ana, se almacenan 3 caracteres.</p>

TIPO DE DATO	DECLARACIONES
Bit:	<p>Se utiliza para definir cadenas de bits (1 y 0). El formato es igual que el de cadenas</p> <p>Bit</p> <p>Bit varying</p>
Numérico exacto	<p>Permite definir números con una representación exacta, donde cada número esta compuesto por dígitos una coma opcional y un signo también opcional.</p> <p>Debe definirse una precisión (número de dígitos significativos incluyendo los números decimales pero excluyendo la coma y una escala que indica la cantidad de posiciones decimales</p> <div data-bbox="672 833 1327 943"> <p>Numeric (presicion ( , escala))</p> <p>Decimal (presicion ( , escala))</p> </div> <div data-bbox="1365 833 1890 968"> <p>Almacenan los números en notación decimal, la escala predeterminada es cero</p> </div> <div data-bbox="672 1005 1299 1058"> <p>ejemplo salario DECIMAL(7,2)</p> </div> <div data-bbox="1365 1005 1890 1096"> <p>Permite almacenar valores hasta 99.999,99</p> </div> <p>Integer Se utiliza para enteros, positivos de gran tamaño</p> <p>Smallint Enteros positivos pequeños. Máximo valor 32.767</p>

TIPO DE DATO	DECLARACIONES
Numérico aproximado	<p>Se utiliza para definir números que no tienen una representación exacta. Son similares a los de notación científica. Ejemplo: <math>3E6</math> equivale a <math>3 \times 10^6</math>,  <math>-5E-4</math> a <math>-5 \times 10^{-4}</math></p> <p>Float (precision) <span>La precisión controla la mantiza</span></p> <p>Real Double precision <span>La precision del REAL Y DOUBLE PRECISION depende de la implementación</span></p>
Fecha y hora	<p>El estandar de iso subdivide los tipos de datos de fecha y hora en:</p> <p>Date : almacena Year, month, day</p> <p>Time : almacena hour, minute, second</p> <p>Timestamp: almacena la fecha y la hora</p>

# \* Restricciones o Controles de integridad

1. Datos requeridos
2. Restricciones de dominio
3. Integridad de entidades
4. Integridad referencial

Pueden definirse en las instrucciones CREATE y ALTER TABLE

# 1. RESTRICCIONES A LOS DATOS

## NOT NULL

Permite que impongamos restricciones a los campos que obligatoriamente deben contener datos; como por ejemplo

- sucursal
- nombre de una persona

Son definidos al momento de la creación de la tabla

NOMBRE_SUCURSAL	VARCHAR (12) NOT NULL
Nombre	VARCHAR (10) NOT NULL



## 2. RESTRICCIONES DE DOMINIO

### CHECK

Toda columna tiene un dominio, es decir un conjunto de valores legales; por ejemplo el campo SEXO puede ser 'F' o 'M'.

Estas restricciones pueden ser definidas al momento de la creación de la tabla con la opción CHECK o haciendo un ALTER a una tabla creada.

sexo CHAR NOT NULL

CHECK (sexo in ('M','F'))

color\_bandera VARCHAR NOT NULL

CHECK (color\_bandera in ('rojo', 'azul'))

### DEFAULT

Como su propio nombre indica, esta restricción introduce un valor por defecto en una columna cuando no se indica ningún valor para insertar. Con esta restricción aseguramos la integridad de dominio, ya que aseguramos valores válidos para nuevos registros que se inserten.

Precio money

set DEFAULT (0)

### 3. RESTRICCIONES DE ENTIDADES

Las llaves primarias que identifican a cada entidad, deben ser definidas con un valor unívoco y no nulo en cada fila.

Por ejemplo, cada fila de la tabla empleado debe tener en el campo `codigo_empleado` un único valor y nunca este valor podrá ser nulo.

**PRIMARY KEY (`codigo_empleado`)**

En el caso de una llave compuesta

**PRIMARY KEY (`no_sucursal`, `no-cta`)**

La cláusula PRIMARY Key sólo puede especificarse una sola vez por tabla.

# RESTRICCIONES DE ENTIDADES

<b>UNIQUE:</b>	<p>Esta restricción obliga a que todos los valores de una determinada columna no estén repetidos en otros registros.</p> <p>Si tenemos varias restricciones UNIQUE en una misma tabla, todas deben ser cumplidas a la vez para cada registro.</p>
----------------	---

## 4. INTEGRIDAD REFERENCIAL

Las llaves foráneas enlazan cada fila de la tabla hijo (es decir la tabla que la contiene) con la fila de la tabla padre donde esta la misma como llave primaria.

La integridad referencial quiere decir que si la llave foránea (externa) contiene un valor, dicho valor debe hacer referencia a una fila existente y válida dentro de la tabla padre.

El estándar ISO de SQL permite la definición de llaves primarias con la clausula FOREIGN KEY en las instrucciones CREATE y ALTER TABLE.

FOREIGN KEY (Codigo\_DEPTO) references Tabla\_EMPLEADO

SQL rechaza cualquier intento de efectuar una operación INSERT o UPDATE a una tabla hijo cuando dicho valor a insertar no existen en la tabla padre.

## 4. INTEGRIDAD REFERENCIAL

En el caso del borrado o actualización de un padre que tenga varios hijos, el SQL soporta su manejo de cuatro posibles formas:

1. **CASCADE**: Borra o actualiza la fila de la tabla padre y borra o actualiza automáticamente todas las filas correspondientes en la tabla hijo (lo hace en cascada)
2. **SET NULL**: Borra o actualiza la fila del padre y se asigna el valor NULL a las filas correspondientes en la tabla hijo
3. **SET DEFAULT**: Borra o actualiza la fila en la tabla padre y se asigna a cada campo de la tabla hijo (llave foránea) el valor especificado como default.
4. **NO ACTION**: Se rechaza la operación de borrado o actualización de la tabla padre. Esta es la opción predeterminada si no se especifica una regla ON DELETE