

Facultad de Ingeniería



Modelo relacional

Tema III

Semestre 2022-2

El alumno explicará y comprenderá las características y elementos que integran al Modelo Relacional aplicados al diseño lógico de bases de datos.

Modelo basado en lógica de predicados y en teoría de conjuntos, propuesto en los años 70's por Frank Codd

Propiedades:

- **No pueden existir dos relaciones que se llamen igual**
- **No pueden existir tuplas iguales**
- **No pueden existir atributos que tengan el mismo nombre**
- **No hay orden en tuplas ni en atributos**
- **Los valores de los atributos deben ser atómicos**

Atributo: Característica de una entidad

Tupla: Conjunto de atributos que caracterizan a una entidad

**(123, Fernando, Arreola, Franco)
(234, Luis, Perez,")**

Relación: Conjunto de tuplas que representa al conjunto de entidades del mundo real.

Cardinalidad: Número de tuplas de una relación

Grado: Número de atributos de una relación

Dominio: Tipo de dato correspondiente a cada atributo

Llave primaria: Es un atributo(s) que identifica de forma única una tupla de una relación

- **Naturales**
- **Candidatas**
- **Compuestas**
- **Artificiales**

Dependencia funcional: Sean X , Y subconjuntos de atributos de una relación. Diremos que Y tiene una dependencia funcional de X , o que X determina a Y , si cada valor de X tiene asociado siempre un único valor de Y .

$X \rightarrow Y$

Reglas de integridad o restricciones



num_Prestamo	nombre_Sucursal	monto
P-11	centro	700
P-14	copilco	1200
P-15	bajío	2000
P-16	coyoacan	21200
P-17	centro	800000
P-23	perisur	500000
P-93	centro	12000

X

Y

y

num_Prestamo -> nombre_Sucursal, monto CUMPLE
nombre_Sucursal -> num_Prestamo, monto NO CUMPLE

Llave foránea: Es una restricción de referencia, en la cual, un atributo de una relación X, es llave primaria en una relación Y

Restricciones de integridad:

- **Unique**
- **Check**
- **Null / Not null**
- **Validaciones a través de triggers**

Reglas de codd

- **Regla 8: Independencia física**
- **Regla 9: Independencia lógica**

**Es una estructura de datos
definida sobre un atributo(s) de
una relación que permite obtener
tuplas de forma rápida.**

Podemos dividirlos en:

- **Clustered:** Altera la forma en que las tuplas se almacenan físicamente, ya que el BDMS ordena las tuplas con base a la columna(s) que se está(n) indexando

- **Non-Clustered:** No altera la forma en que las tuplas se almacenan físicamente. Se crea un objeto donde se almacenan la(s) columna(s) que fueron seleccionadas para indexar y un apuntador a las tuplas que contienen los datos

Tipos:

- **Hash - Complejidad $O(1)$**
- **BitMap - Muy buenos en atributos con pocas variantes**
- **B-Tree - Complejidad $O(\log n)$**

Ventajas:

Un índice bien aplicado tiene mejoras bastante significativas en rendimiento.

Desventajas:

Los objetos donde se almacena la información de los índices, ocupan espacio

Consumo de recursos a la hora de actualizar, ingresar o borrar información