

Facultad de Ingeniería



Modelo relacional

Tema III

Semestre 2021-2



UNAM
UNIVERSIDAD NACIONAL
AUTÓNOMA DE MÉXICO

Ejercicio



Una base de datos para una pequeña empresa debe contener información acerca de clientes, artículos y pedidos. Hasta el momento se registran los siguientes datos en documentos varios:

Para cada cliente: Número de **cliente** (único), direcciones de envío (varias por cliente), saldo, límite de crédito (depende del cliente, pero en ningún caso debe superar los 3.000.000 pesos), descuento.

Para cada **artículo**: Número de artículo (único), fábricas que lo distribuyen, existencias de ese artículo en cada fábrica, descripción del artículo.

Para cada **pedido**: Cada pedido tiene una cabecera y el cuerpo del pedido. La cabecera está **formada por el número de cliente**, dirección de envío y fecha del pedido. El cuerpo del pedido son varias líneas, en cada línea **se especifican el número del artículo** pedido y la cantidad. Además, se ha determinado que se debe almacenar la información de las **fábricas**. Sin embargo, dado el uso de distribuidores, se usará: Número de la fábrica (único) y teléfono de contacto. Y se desean ver cuántos **artículos (en total) provee la fábrica**. También, por información estratégica, se podría incluir información de **fábricas alternativas** respecto de las que ya fabrican artículos para esta empresa.

Nota: El # de artículos provistos es la suma de las existencias de cada artículo.

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

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**

Ejercicio



Se desea crear un sistema de base de datos que permita obtener información sobre las películas que se exhiben actualmente en las distintas salas cinematográficas ubicadas dentro de la Ciudad de México. Para cada película, se deben almacenar los siguientes datos: título, clasificación (A, B, C o D), género (acción, musical, terror, etc.); nombre y país de origen de director principal, razón social y fecha de fundación de la compañía productora; nombre y domicilio de las salas en donde se exhibe; nombre, nacionalidad, sexo, fecha de nacimiento y honorarios de los actores que participen, así como el nombre del actor principal.

**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