
Base de Datos I

Curso No. TWB22B

Introducción

CY350 Conceptos RDBMS y SQL

Instructor

Por favor preséntese usted:

- Nombre
- Experiencia en RDBMS
- Metas que espera lograr

Descripción del Curso

- ◆ Este curso está diseñado para introducir a los estudiantes a los conceptos de bases de datos relacionales y SQL
- ◆ El alcance del curso es proporcionar un entendimiento básico de los conceptos relacionados a las bases de datos relacionales tales como la normalización de bases de datos, SQL, diccionario de datos, UDT, una introducción al DB2 8.x, MySQL y PostgreSQL.

Audiencia

Estudiantes, profesionales, y desarrolladores
quienes deseen conocer acerca de los conceptos
básicos de las bases de datos relacionales y SQL

Prerequisitos

- ◆ Conceptos básicos de programación
- ◆ Entender el almacenamiento de datos usando archivos
- ◆ Codificar programas simples en C
- ◆ Estas habilidades pueden ser obtenidas completando el curso: CY320 : Introducción a la Programación con C

Objetivos del curso

- ◆ Comprender los conceptos de RDBMS
- ◆ Explicar el proceso de diseño de base de datos y el modelo conceptual
- ◆ Definir normalización y entender la necesidad de normalizar
- ◆ Entender SQL y tablas de datos
- ◆ Trabajar con SQL avanzado y explicar sus conceptos tales como claves primaria y foránea
- ◆ Entender los fundamentos de DB2 y sus capacidades
- ◆ Explicar cómo crear y administrar objetos de la base de datos
- ◆ Explicar los tipos de datos y definiciones de columnas
- ◆ Explicar el acceso a base de datos y conceptos de seguridad

Objetivos del curso

- ◆ Entender los fundamentos de MySQL y sus capacidades
- ◆ Entender los fundamentos de PostgreSQL y sus capacidades

Agenda

Cada unidad en este curso es de dos horas de duración

Resumen

- Se han visto los objetivos globales del curso y la agenda
- Empecemos

Fundamentos de RDBMS

Unidad 1:

Introducción a RDBMS

Objetivos del Aprendizaje

- Definir una base de datos
- Describir el DBMS
- Enumerar los diversos modelos de datos
- Trabajar con RDBMS
- Describir los fundamentos de la Administración de Transacciones
- Explicar la necesidad de las propiedades ACID de una transacción

Inicios del Manejo de Datos

- Gran cantidad de datos organizados en dispositivos de almacenamiento secundarios (discos y cintas) como *archivos de datos*
- Los archivos de datos llamados también archivos planos, son una colección de registros, cada uno teniendo un conjunto de campos
- Las relaciones en los archivos de datos deben ser manejadas por los programadores.
- Dependiendo de la aplicación, los archivos de datos se pueden organizar como archivos secuenciales o archivos de acceso aleatorio.
- Los archivos de datos son completamente dependientes del dispositivo físico que se utilice.

Desventajas de Archivos Planos

- **No Concurrencia:** Los datos no pueden ser accedidos y modificados por más de una persona a la vez
- **No Integridad:** Si más de una aplicación utiliza la misma información, los datos pueden ser corrompidos
- **No Relación entre Datos:** Es difícil establecer relaciones entre los datos
- **No Reutilización:** Un archivo plano diseñado para un sistema no se puede reutilizar para otro sistema
- **Baja Seguridad:** Es difícil establecer niveles altos de seguridad sobre los datos que se almacena en los archivos planos

¿Qué es una Base de Datos?



**¿Cómo puedo
almacenar juntos
diferentes tipos de
datos?**

¿Qué es una Base de Datos? ... 2

Colección organizada de datos interrelacionados

ü Organizado

- Manera que los datos necesitan ser almacenados

ü Colección

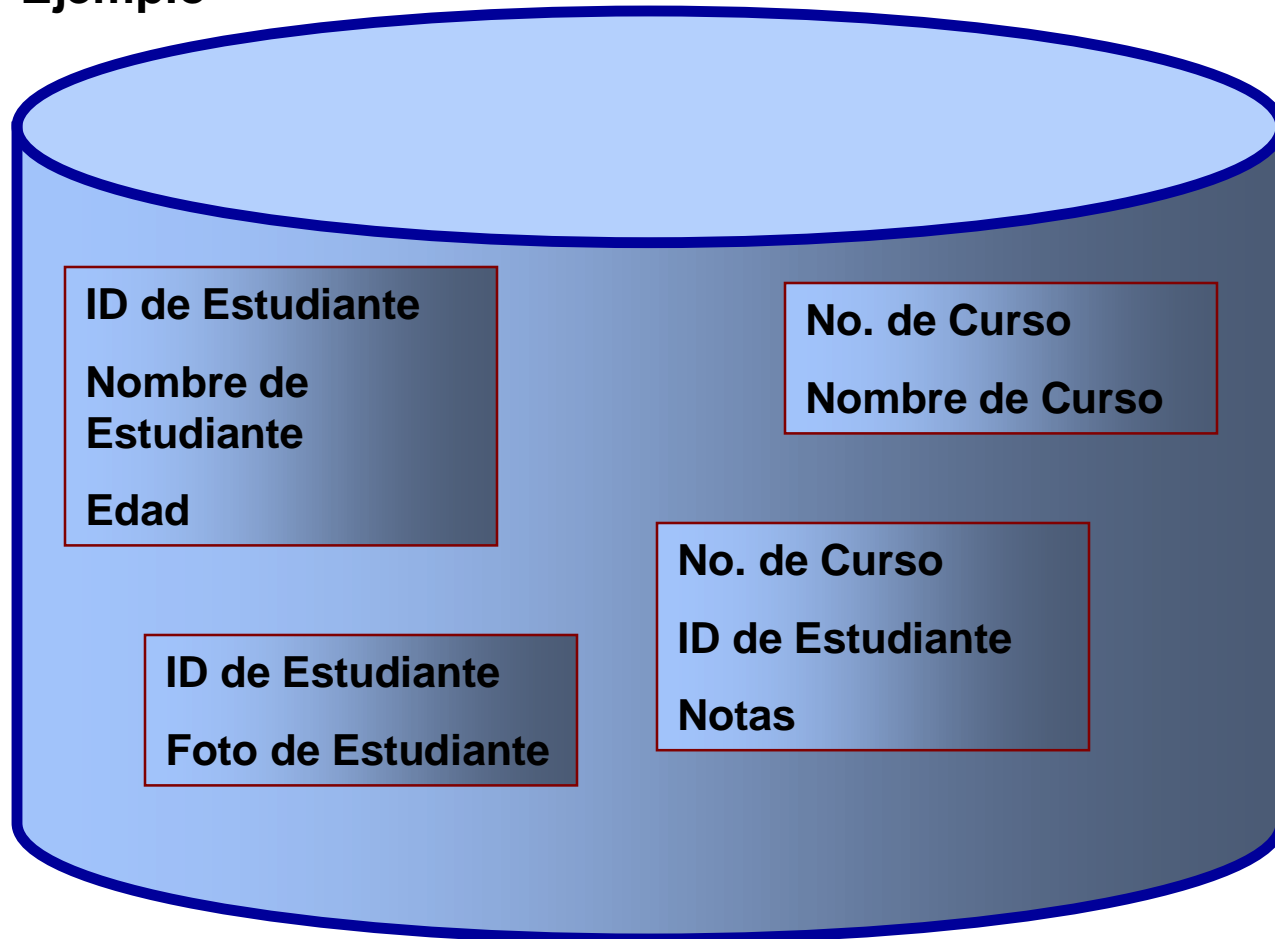
- Grupo de datos
- Colecciones de datos que pertenecen a diversos tipos de datos

ü Inter-relacionado

- Datos que proporcionan información razonable y coherente

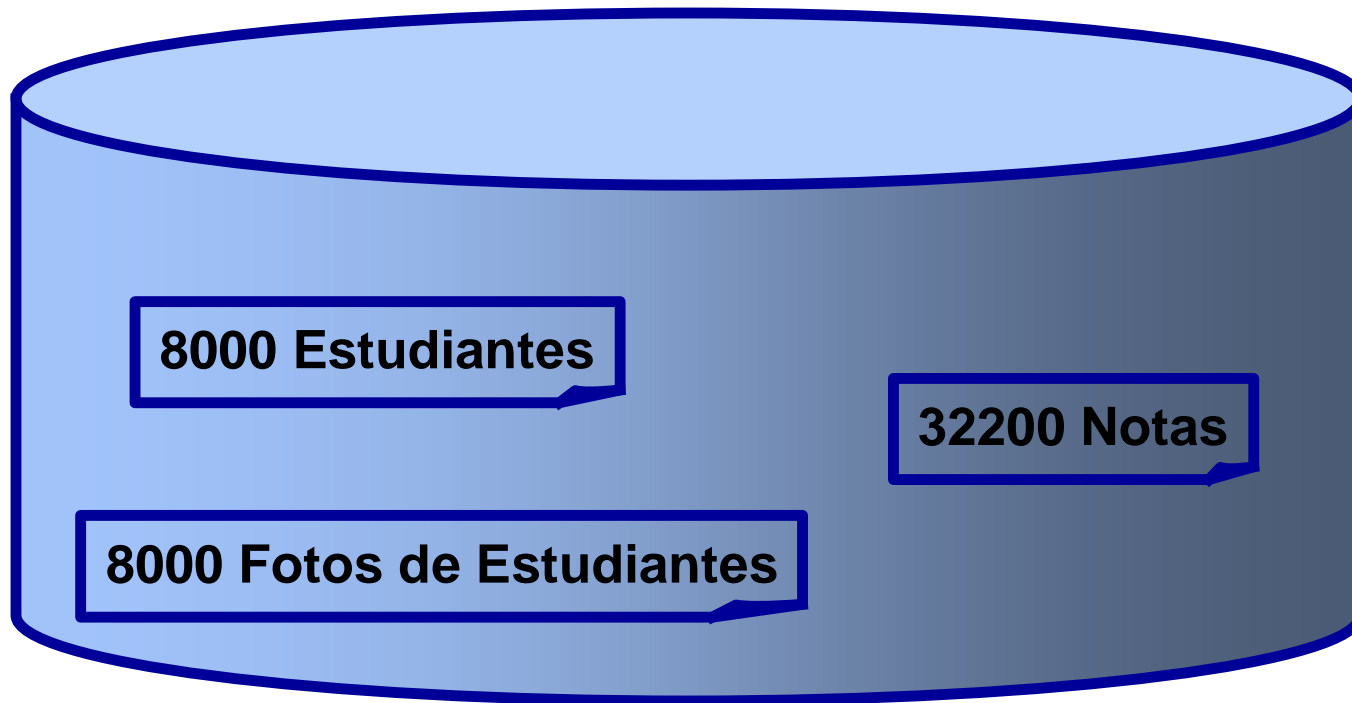
¿Qué es una Base de Datos? ... 3

Ejemplo



¿Qué es una Base de Datos? ... 4

Una base de datos se usa para almacenar y recuperar grandes cantidades de datos



- Físicamente, los datos en una computadora se almacenan en forma de archivos
- Es una colección computarizada de datos, almacenados como archivos de datos

Sistema Administrador de Base de Datos

- Se requiere un sistema apropiado para administrar una base de datos
- Las tareas incluyen:
 - Recuperar datos
 - Agregar datos
 - Modificar datos
 - Remover datos
- Un DBMS se puede definir como un sistema que trabaja sobre las bases de datos, y ayuda a los usuarios a administrar la base de datos de acuerdo a sus necesidades

Propiedades de un DBMS

- Proporciona al programador funcionalidades:
 - ü Para crear bases de datos
 - ü Para agregar, modificar y eliminar elementos hacia/desde una base de datos
 - ü Para ordenar registros en una base de datos
- Proporciona comandos especiales para llevar a cabo tareas sin programación de bajo nivel
- Proporciona independencia de datos

Independencia de Datos

- **Independencia de datos física**

Los programas de aplicación que usan la base de datos no deben ser afectados por cambios en la organización física subyacente de la base de datos

- **Independencia de datos lógica**

Capacidad de cambiar el esquema lógico de la base de datos sin causar que las aplicaciones se rescriban

Modelos de Datos

Un modelo de datos es la estructura subyacente de una base de datos.

En esta unidad, se estudian los modelos lógicos basados en registros.

Los modelos lógicos basados en registros son llamados así porque la base de datos es estructurada en torno a registros.



El Modelo Relacional

- Una o más tablas representan datos y las relaciones entre diversos datos
- Una tabla es una colección de registros en una base de datos
- Una tabla tiene múltiples **columnas**
- Cada columna tiene un nombre único y contiene conjuntos de datos
- Cada conjunto se llama **fila**
- El valor de los datos se obtiene por la intersección de una fila y una columna

El Modelo Relacional... 2

RegistrosdeEstudiante

ID- Estudiante	Nombre- Estudiante	Hospedaje	Código- Disciplina
93PS274	James	Ram Bhawan	A1
97PS087	Alice	Meera Bhawan	A2

CodigosdeDisciplina

Codigo- Disciplina	Disciplina
A1	Ingeniería Química
A2	Ingeniería Civil

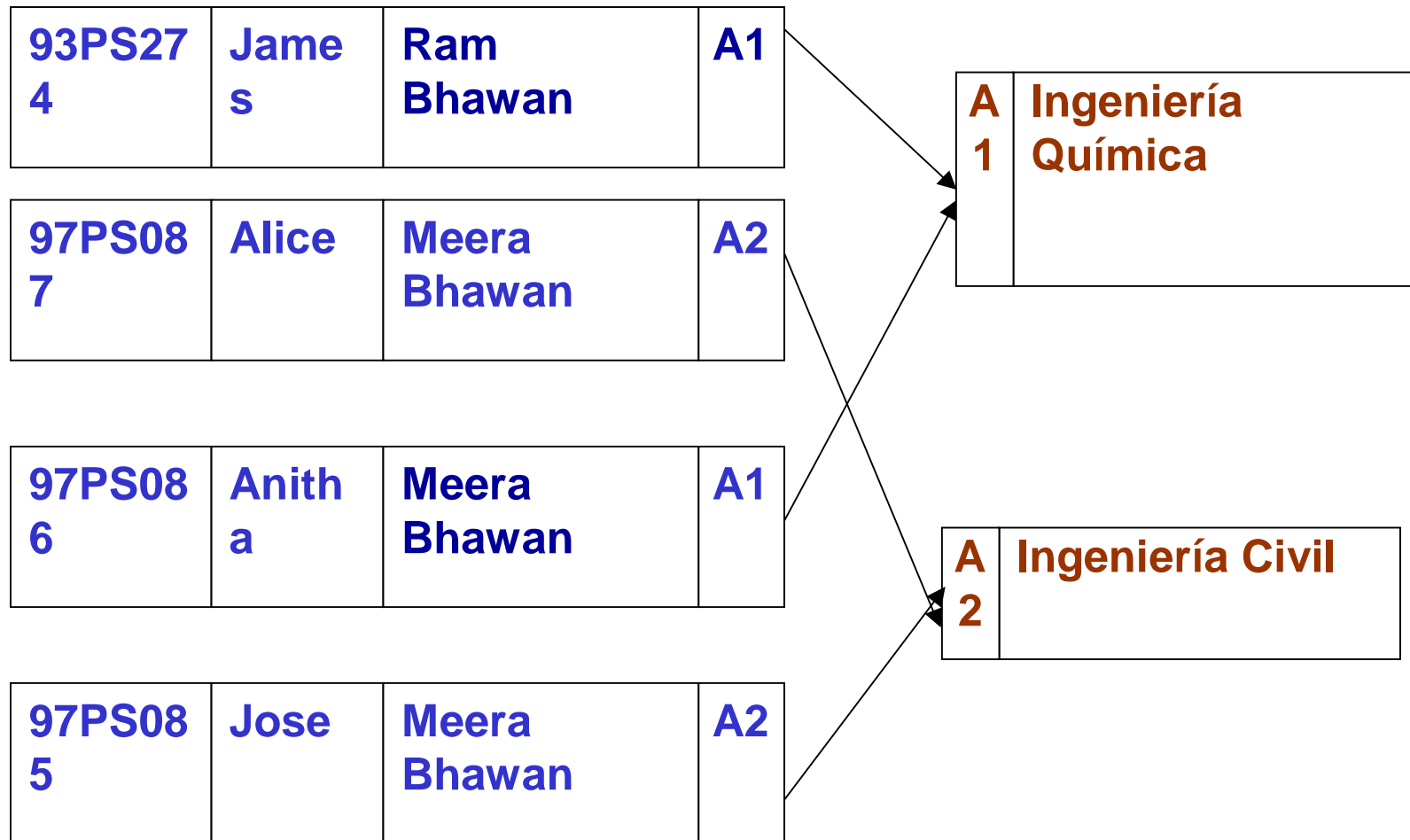
Características del Modelo Relacional

- Colección de tablas que contienen los datos forman la base de RDBMS
- Es soportado por un modelo matemático formal
 - ü Algebra Relacional
 - ü Cálculo Relacional
- Soporta eficientes maneras de recuperación y administración de datos
- Operaciones relacionales:
 - ü Seleccionar (Select)
 - ü Proyectar (View)
 - ü Unir (join)
- Operaciones de conjuntos:
 - ü Unión
 - ü Intersección
 - ü Producto cartesiano
- Capacidad de hacer cumplir las restricciones de integridad de datos

El Modelo de Red

- Los datos se representan mediante una colección de registros
- Las relaciones entre los datos se representan por enlaces
- Los enlaces pueden ser vistos como punteros
- Los registros en la base de datos se organizan en una colección de grafos arbitrarios

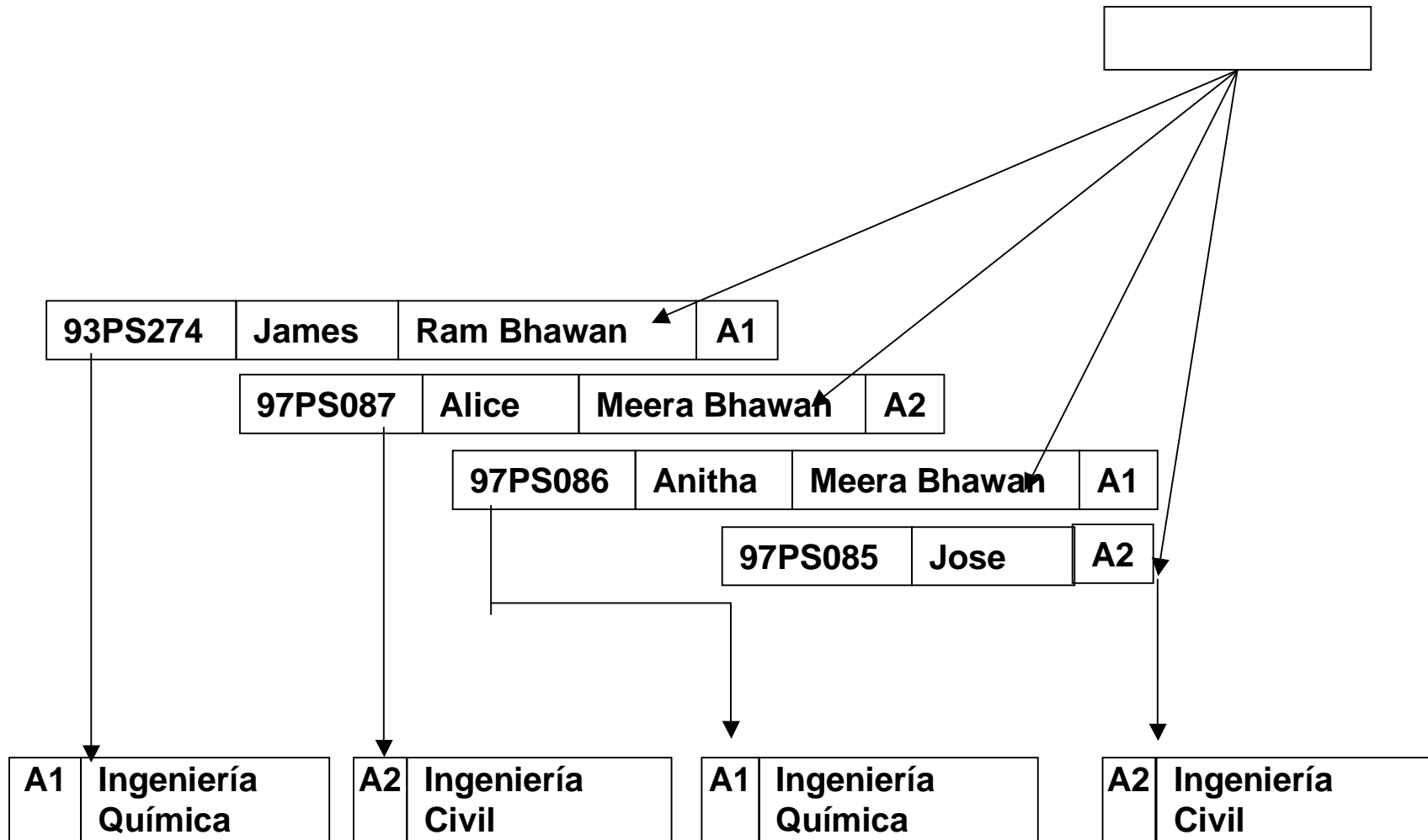
El Modelo de Red... 2



El Modelo Jerárquico

- Similar al Modelo de Red
- Los datos se representan mediante una colección de registros
- Las relaciones entre los datos se representan por enlaces
- Los registros están organizados como una colección de árboles en vez de grafos arbitrarios

El Modelo Jerárquico... 2



El Modelo Relacional vs Otros

- Los otros modelos están estrechamente ligados con la implantación subyacente de la base de datos
- El Modelo Relacional es superior porque soporta la independencia de datos
- El Modelo Relacional establece relaciones entre los registros según los valores que contienen
- Los otros usan punteros y enlaces complejos

¿Qué es un RDBMS?

- RDBMS es un DBMS basado en el modelo relacional
- Organiza los datos y las relaciones entre los datos en tablas que se componen de columnas y filas
- Permite la definición de las estructuras de datos, de las operaciones de almacenamiento y recuperación, y de las restricciones de integridad
- Ciertos campos pueden ser designados como **CLAVES** para acelerar la búsqueda

Proveedores de RDBMS

Proveedor	Producto	Última Versión
IBM	DB2 UDB	8.x
Oracle Corporation	Oracle	10
Microsoft	Microsoft SQL Server	2005
MySQL	MySQL	5.x
PostgreSQL	PostgreSQL	8.x

Administración de Transacciones

- Una transacción es una colección de operaciones que realiza una única unidad lógica de trabajo
- Una transacción se considera incompleta si todas las operaciones no ocurren
- La administración de transacciones implica algunas propiedades importantes llamadas propiedades ACID (Atomicity, Consistency, Isolation, and Durability).
- Una serie de operaciones ocurre entre las sentencias `BEGIN TRANSACTION` y `END TRANSACTION`

Atomicidad

- Una transacción se ejecuta exactamente una vez
- Una transacción es atómica
- Se hace todo el trabajo o no se hace nada

Consistencia

- Una transacción es una unidad de integridad porque:
 - ü Preserva la consistencia de datos
 - ü Transforma un estado consistente de datos en otro estado consistente
- La consistencia requiere que los datos comprometidos con una transacción se preserven semánticamente

Aislamiento

- Una transacción es también una unidad de aislamiento
- La Administración de Transacciones permite que las transacciones concurrentes se comporten como si cada una fuera la única transacción ejecutándose en el sistema
- El aislamiento requiere que parezca que sólo una transacción está manipulando la base de datos
- Una transacción nunca debe poder ver las etapas intermedias de otra transacción

Durabilidad

- Una transacción es también una unidad de recuperación
- Si una transacción tiene éxito, el sistema garantiza que persistirán sus actualizaciones
- Este requisito de persistencia se llama *durabilidad*

Ventajas de DBMS

- La redundancia de datos puede ser minimizada
- La integridad, la seguridad y la consistencia de los datos pueden ser aseguradas
- Mejor rendimiento en el almacenamiento y recuperación de datos
- Se aseguran operaciones fáciles de manipulación de datos
- Los datos pueden ser distribuidos
- Los datos pueden ser adaptados a los cambios del modelo del negocio

Resumen

- Se definió qué es una base de datos
- Se describió qué es el DBMS
- Se presentaron los diversos modelos de datos
- Se explicó que significa trabajar con un RDBMS
- Se presentaron los fundamentos de la Administración de Transacciones
- Se explicó la necesidad de las propiedades ACID de una transacción

Unidad 2:

Diagrama Entidad Relación

Objetivos del aprendizaje

- Explicar los conceptos de diseño de base de datos
- Enumerar los pasos para el diseño de base de datos
- Discutir acerca de las entidades y sus relaciones
- Describir los diagramas Entidad-Relación (ER) y sus notaciones
- Indicar las pautas para construir un modelo ER
- Transformar Diseño Conceptual a Diseño Lógico

Diseño de Base de Datos

- ¿Cómo se decide el número de tablas?
- ¿Qué datos se almacenan en una tabla particular?
- Se debe formular el diseño del contenido de las tablas que pueden almacenar los datos
- El proceso de formulación se denomina “proceso de diseño de base de datos”

Pasos para el Diseño de una BD



El Modelo Conceptual

Modelación
de Datos

Diagramas
ER

Entidades
y
Relaciones

Modelación de Datos

- El propósito principal de la identificación de entidades es la producción del **diagrama Entidad-Relación (ER)** que:
 - ü Muestra la naturaleza de las relaciones significativas que existen entre los tipos de entidades
 - ü Es una representación en la aplicación del 'mundo real'

Entidades y Relaciones

- Las *entidades* son objetos del mundo real que son relevantes al sistema de información
- Las propiedades de una entidad son sus *Atributos* que determinan los datos que deben ser almacenados
- Considere como ejemplo un tipo de entidad 'carro' con los atributos:
 - ü Nro-Reg (número de registro del carro)
 - ü Modelo
 - ü Capacidad del motor
 - ü Color
- Los datos reales se pueden describir como los *valores de los atributos de las entidades*

Entidades y Relaciones...2

- Un atributo único de un tipo de entidad se llama *clave* o el identificador de la entidad.
- Dos o más atributos juntos designados como la clave forman una *clave compuesta*
- El identificador o clave debe identificar unívocamente un registro
- Las entidades son asociadas por *relaciones*
- En la sentencia, “un cliente ha pedido ciertas mercancías”, “ha pedido” es la relación

¿Por qué Diagramas ER?

- La modelación ER es una buena ayuda para diseñar bases de datos
- Un modelo ER es una representación gráfica del sistema
- Es un modelo de datos conceptual de alto nivel
- Un modelo ER también soporta la percepción de los datos por un usuario
- Es independiente de la plataforma particular del DBMS y del hardware

Componentes del Diagrama ER



Entidades

**Nombre
de
Entidad**

- Representa una colección de objetos
- Los miembros (instancias) tienen ciertas características como un objeto
- Pueden ser fácilmente identificados
- Juega un papel necesario

Entidades...2

Dos tipos de entidades:



**Entidad
Fuerte**



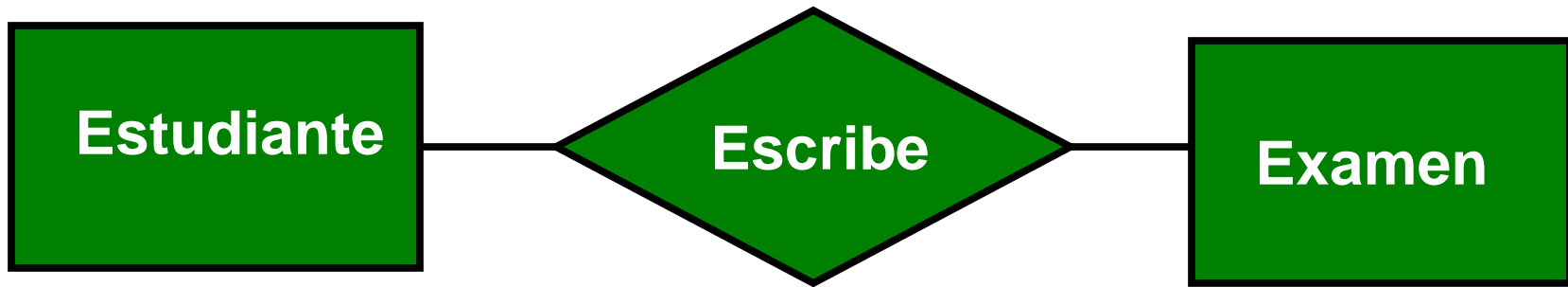
**Entidad
Débil**

Relaciones

- Las entidades están conectadas unas a otras a través de relaciones
- Las relaciones representan un conjunto de conexiones entre objetos



Relación Entre Entidades



Cardinalidad

- Especifica si una entidad está relacionada con una o más instancias de otra entidad en una misma relación

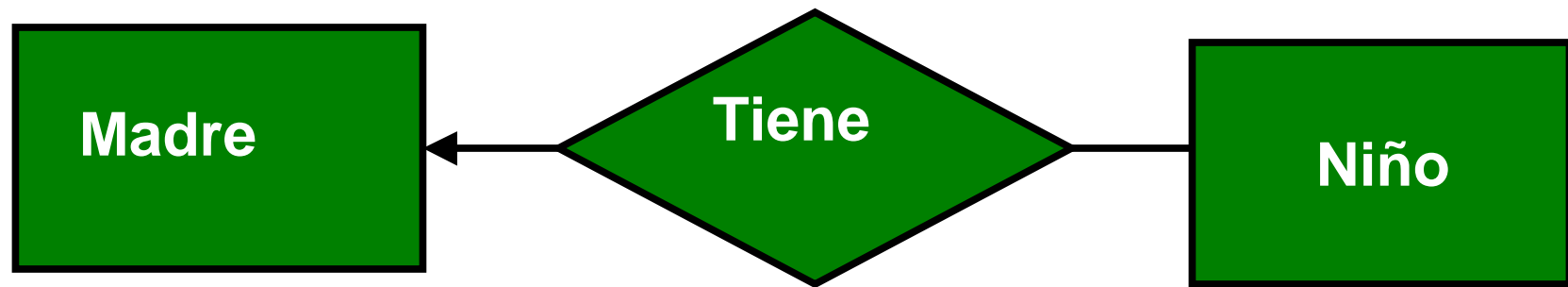
Relación Uno-a-Uno

- También se denota por 1:1
- Un hombre se puede casar sólo con una mujer y una mujer se puede casar sólo con un hombre



Relación Uno-a-Muchos

- También se denota por 1:M o M:1
- Una madre puede dar a luz a muchos niños
- Un niño puede tener solamente una madre biológica



Relación Muchos-a-Muchos

- También se denota por M:N
- Más de una persona puede ser autor de un libro
- Un autor puede escribir muchos libros



Opcionalidad

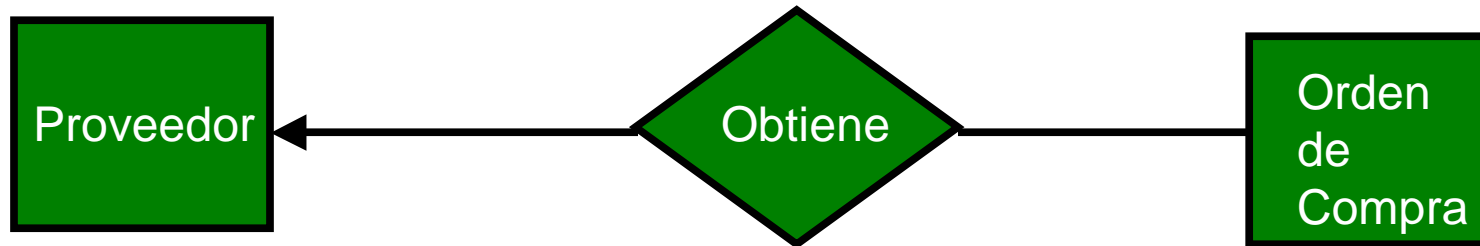
- Una relación puede ser opcional o mandatoria
- La opcionalidad puede ser diferente a cada lado de la relación
- Un estudiante no puede existir sin un curso
- Un curso puede existir antes de que los estudiantes se hayan matriculado



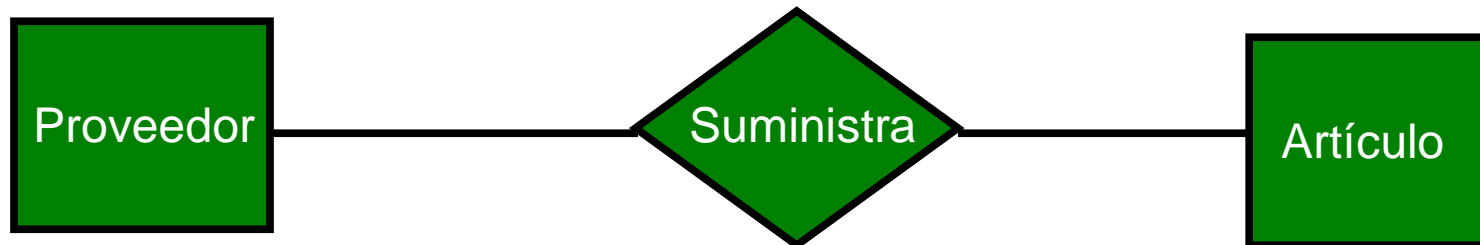
Desarrollar Diagramas ER

- Considere un Sistema de Orden De Compra
- Identifique las entidades del sistema
- Estas son:
 - Proveedor
 - Orden de Compra
 - Artículo

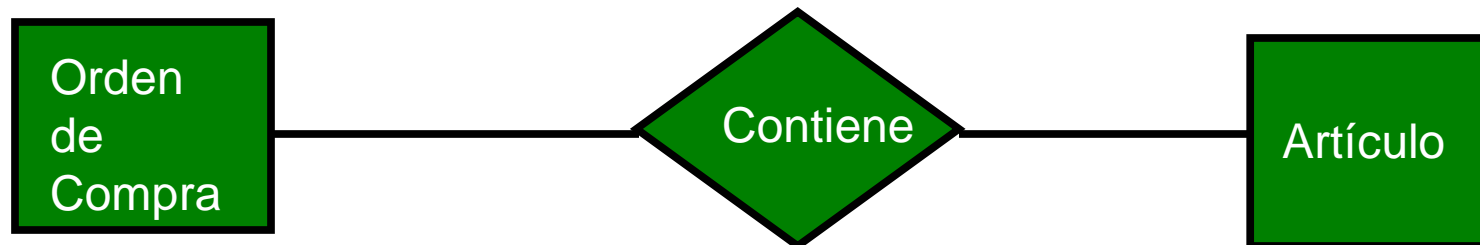
Desarrollar Diagramas ER...2



Relación (1:M) Proveedor:Orden de Compra



Relación (M:N) Proveedor:Artículo



Relación (M:N) Orden de Compra: Artículo

Desarrollar Diagramas ER...3

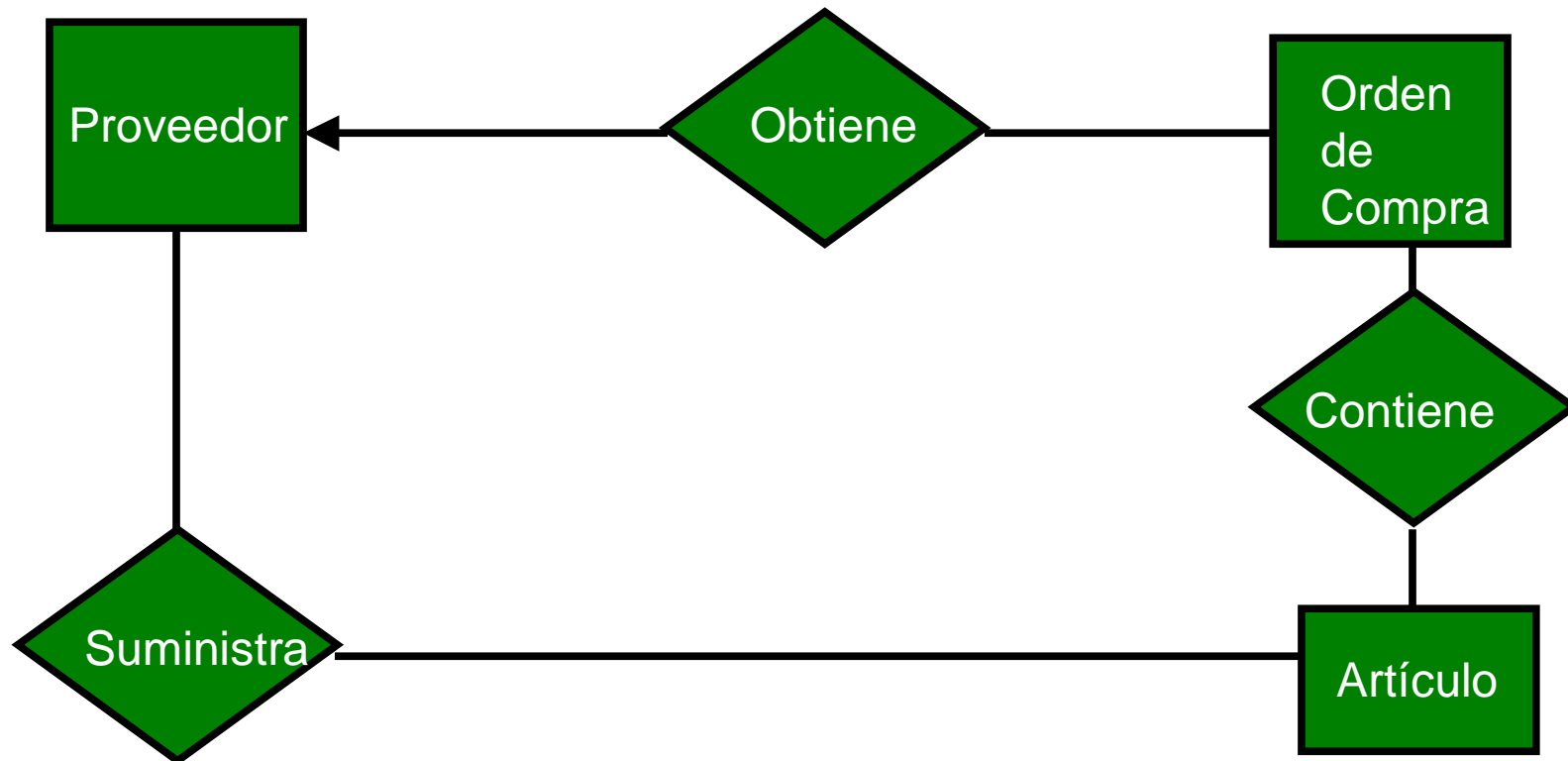



Diagrama ER de un Sistema de Orden de Compra

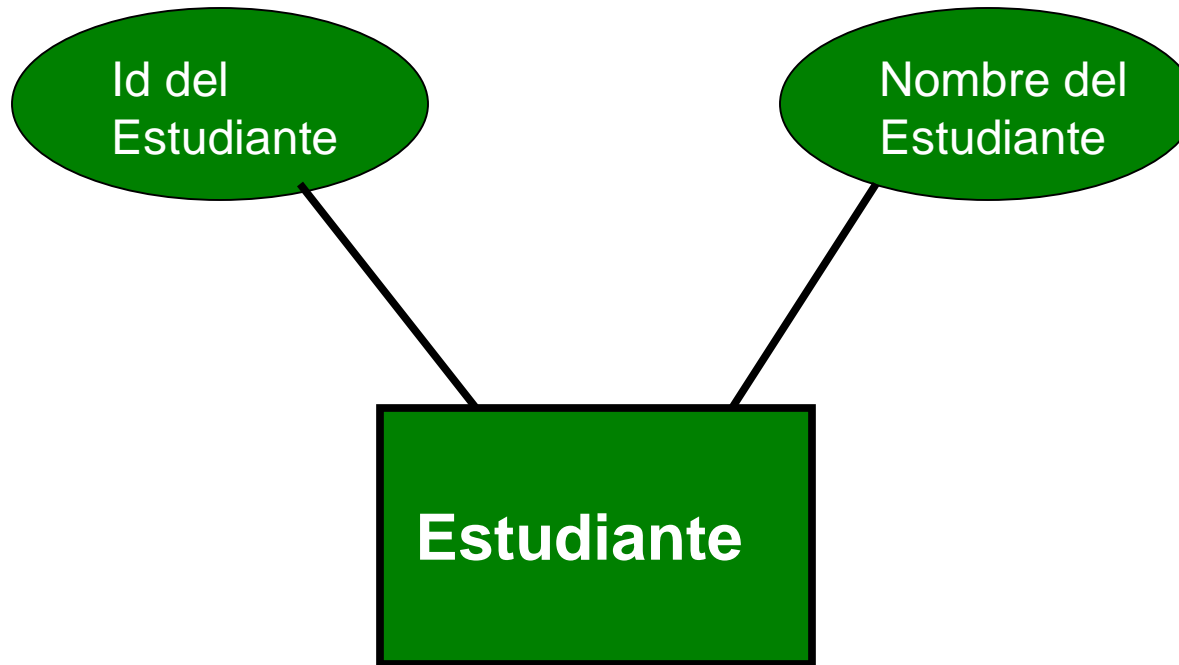
Atributos



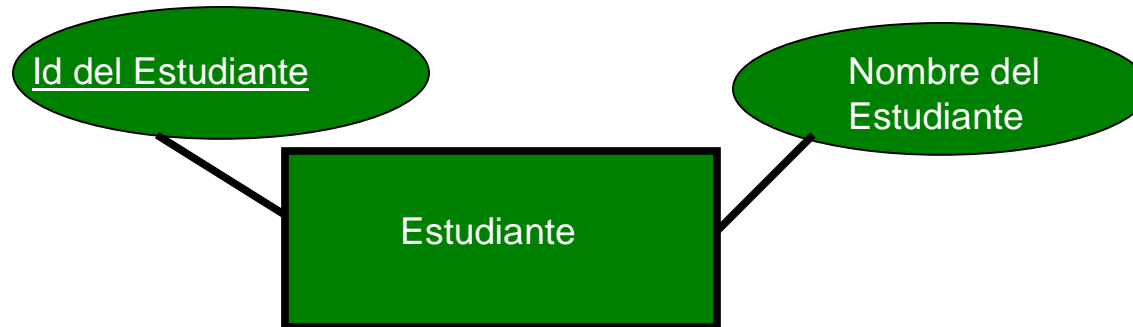
Nombre del
Atributo

- Todos los datos referentes a una entidad están contenidos en sus atributos
- Un atributo es alguna propiedad de la entidad que es interesante en un contexto específico
- El valor del atributo se almacena en la base de datos
- Cada instancia del conjunto de entidades tendrá el mismo conjunto de atributos, pero puede contener valores diferentes

Mostrar los Atributos en los Diagramas ER

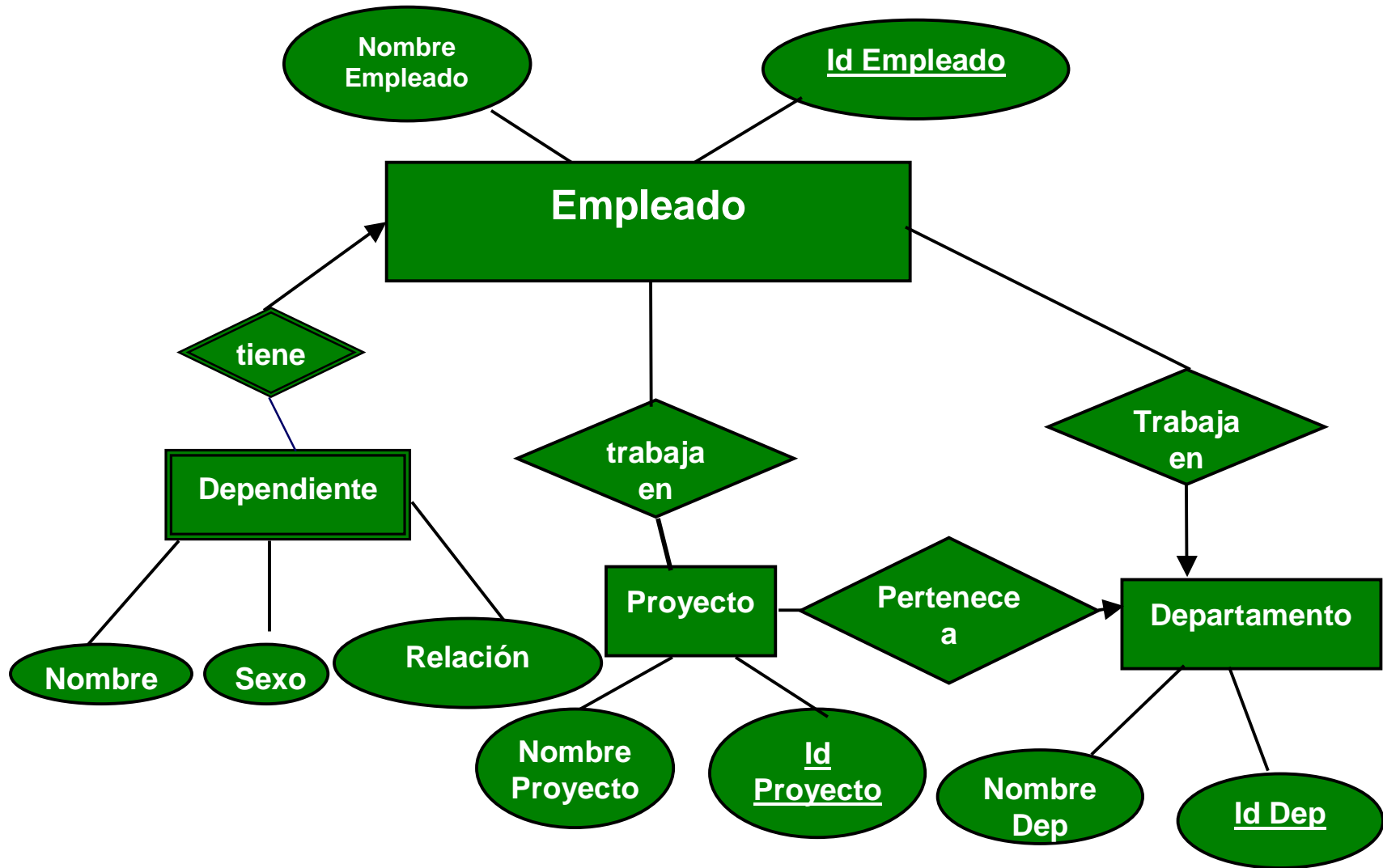


Mostrar las Claves en el Diagrama ER



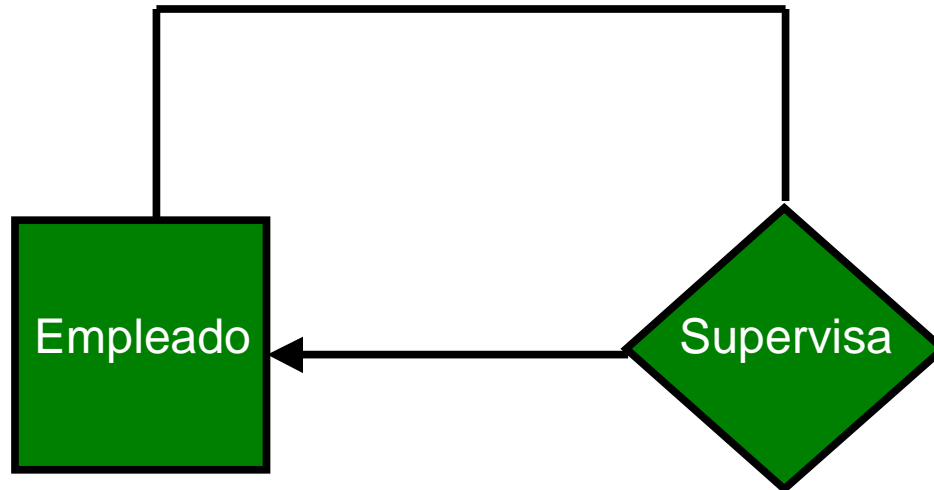
- Una clave identifica de manera única las ocurrencias de un tipo de entidad
- Una clave candidata es un candidato potencial a convertirse en una clave primaria.
- La clave candidata seleccionada como el identificador se conoce como clave primaria

Diagrama E/R de HLX Products Ltd



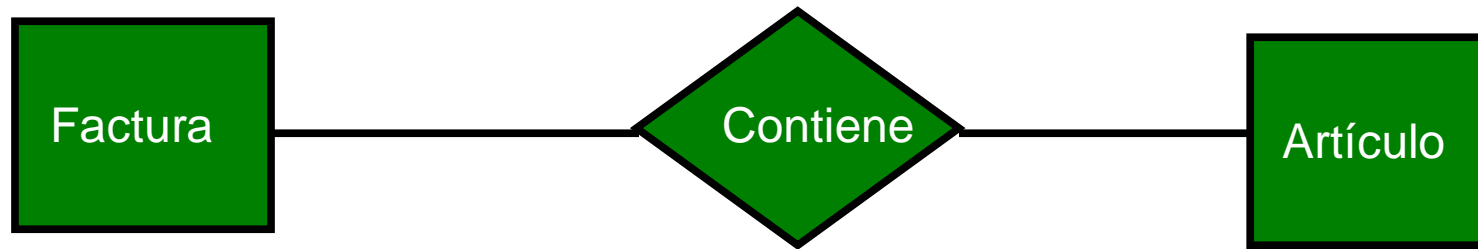
Grados de una Relación

Relación Unaria



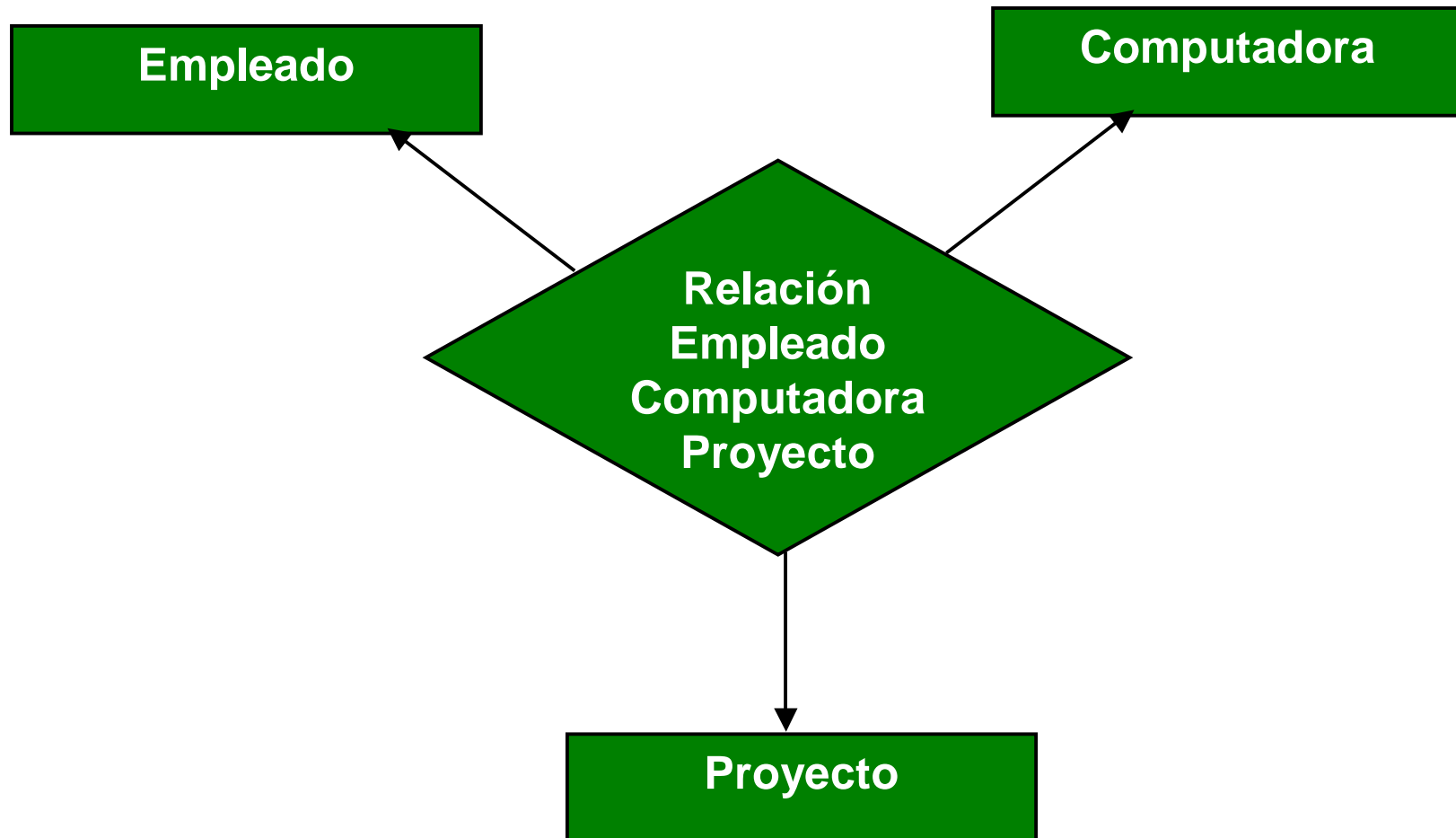
El número de entidades que participan en la relación es uno

Relación Binaria



El número de entidades que participan en la relación es dos

Relación Ternaria

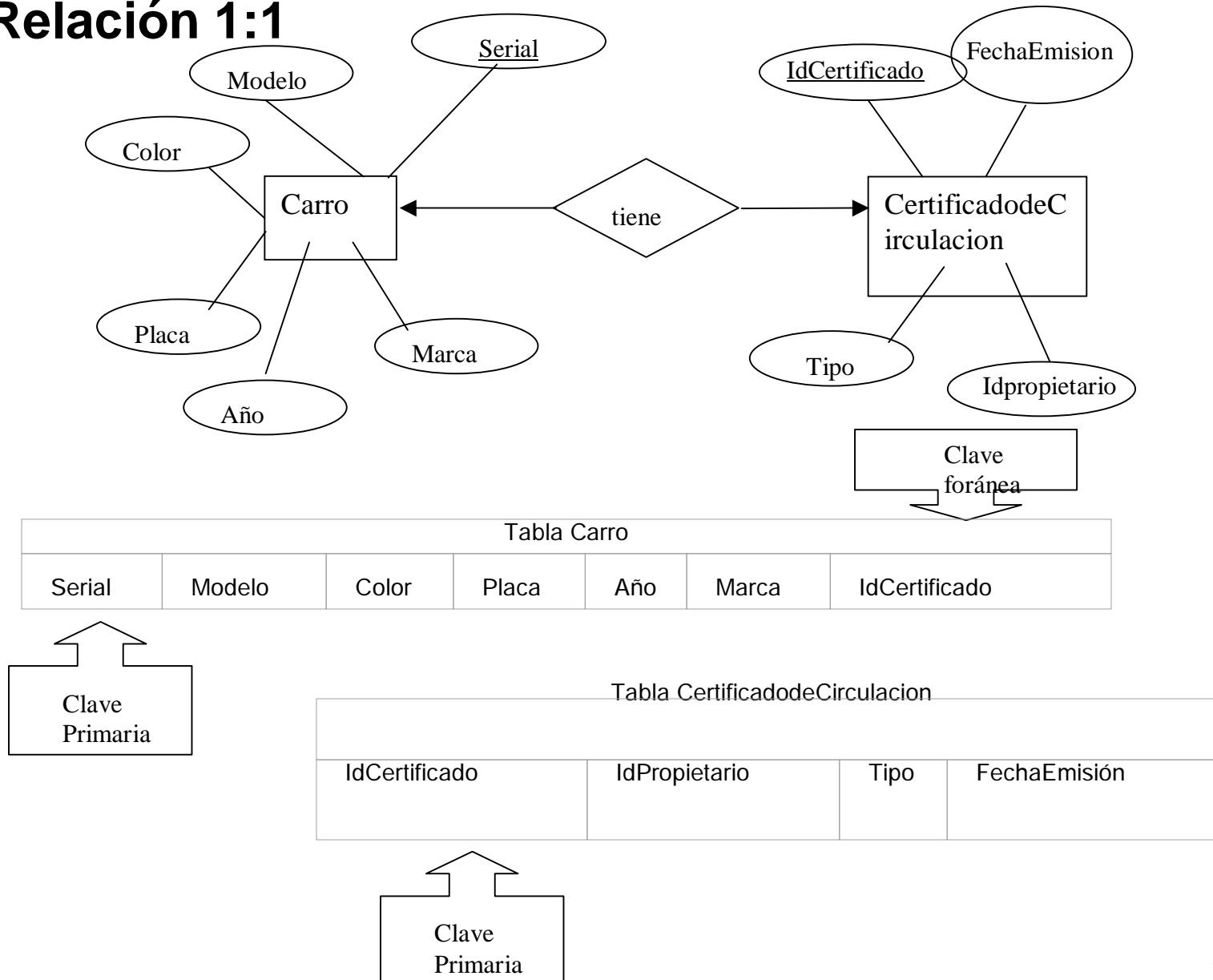


Pasos para Construir un Modelo ER

- Identifique las entidades
- Elimine las entidades duplicadas
- Enumere los atributos de cada entidad
- Marque las claves primarias
- Defina las relaciones
- Examine cada tipo de entidad para determinar cómo se relaciona con otras entidades
- Describa la cardinalidad de las relaciones
- Elimine las relaciones redundantes

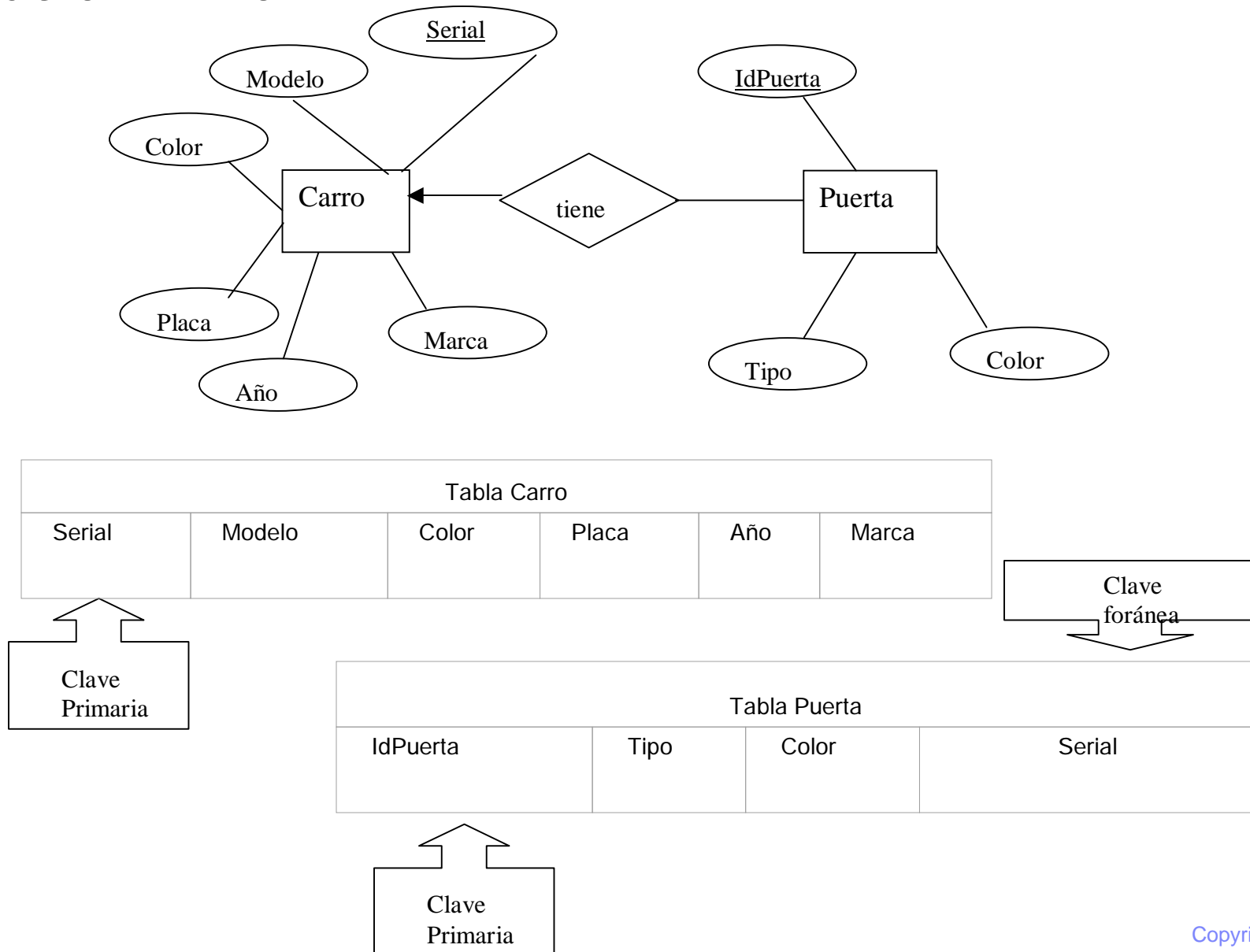
Transformación Diseño Conceptual a Diseño Lógico

Relación 1:1



Transformación Diseño Conceptual a Diseño Lógico

Relación 1:M o M:1



Transformación Diseño Conceptual a Diseño Lógico

Relación M:N

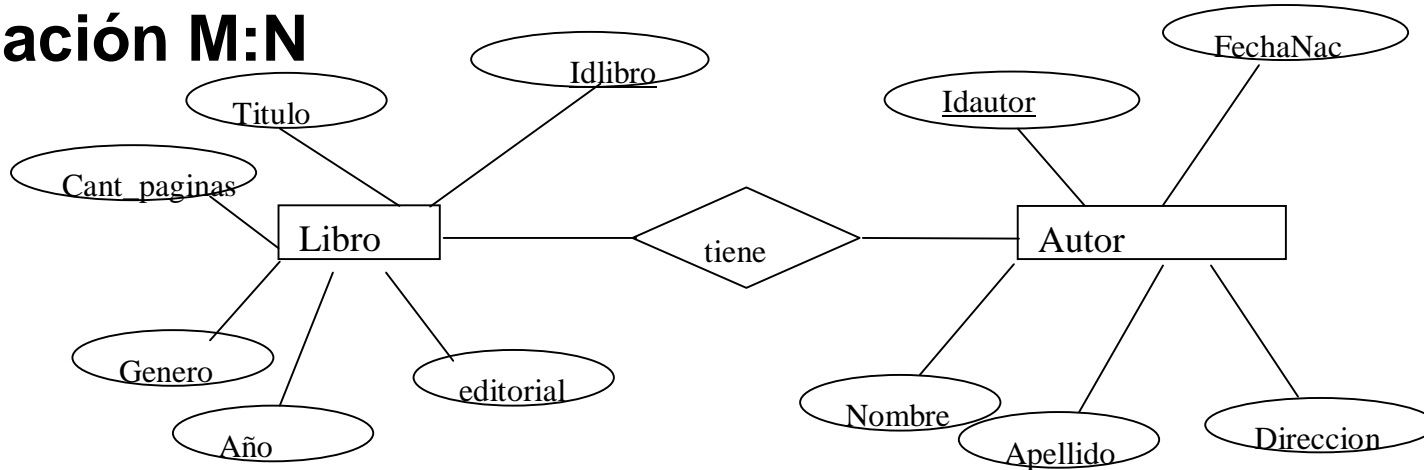


Tabla Cliente				
Idautor	Nombre	Apellido	Direccion	FechaNac

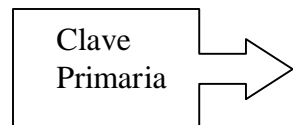
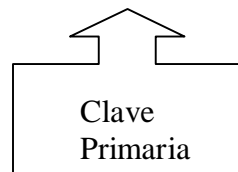


Tabla Libro					
Idlibro	Título	cant_paginas	Género	Año	Editorial

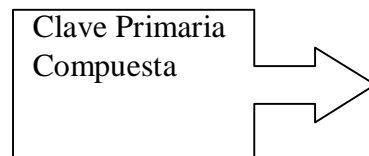
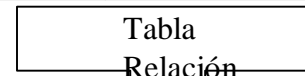


Tabla Autor-Libro	
Idautor	Idlibro

Resumen

- Se explicaron los conceptos necesarios para realizar el diseño de base de datos
- Se describieron los pasos para el diseño de base de datos
- Se estudiaron los conceptos de entidades y sus relaciones
- Se describieron los diagramas ER y sus notaciones
- Se presentaron las pautas para construir un modelo ER
- Se mostraron la manera de como transformar Diseño Conceptual en Diseño Lógico

Unidad 3:

Normalización de la Base de

Datos

Objetivos del Aprendizaje

- Explicar la normalización
- Describir la importancia de la normalización en las bases de datos
- Apreciar la necesidad de la normalización en las bases de datos
- Discutir diversas clases de formas normales

Normalización

- *Normalización* es el proceso de organizar los datos para minimizar la duplicación
- Generalmente implica dividir una base de datos en dos o más tablas y definir las relaciones entre ellas
- El objetivo es aislar los datos en una tabla

Proceso de Normalización

Tabla Estudiante

ID Estudiante	Nombre Estudiante	Ubicación Domicilio	ID Curso	Curso
S101	Joseph Daniel	New York	CS201	Sistema de Administración de Base de Datos
S102	Jerry Jones	San Jose	CS202	Análisis y Diseño Orientado a Objetos
S101	Joseph Daniel	New York	CS202	Análisis y Diseño Orientado a Objetos
S103	Carl Baker	Washington	CS203	Ingeniería de Software

Proceso de Normalización

- Hay un requerimiento del negocio que solicita cambiar el nombre del curso “Análisis y Diseño Orientado a Objetos” a “Análisis Orientado a Objetos”
- ¿Cómo puede ser incorporado este cambio?
- Se puede evitar la duplicación si se divide la tabla Estudiante en dos tablas diferentes y después se enlazan con una relación de tabla

Proceso de Normalización... 2

Tabla Estudiante

ID Estudiante	Nombre Estudiante	Ubicación Casa
S101	Joseph Daniel	New York
S102	Jerry Jones	San Jose
S103	Carl Baker	Washington

Proceso de Normalización... 3

Tabla Curso

ID Curso	Curso
CS201	Sistema de Administración de Base de Datos Relacional
CS202	Análisis y Diseño Orientado a Objetos
CS203	Ingeniería de Software

Proceso de Normalización... 4

Tabla Estudiante-Curso

ID Estudiante	ID Curso
S101	CS201
S101	CS202
S102	CS202
S103	CS203

Descomposición sin Pérdida (Loss-less Join)

- Los datos deben ser analizados cuidadosamente antes de que se puedan dividir las tablas
- Cuando se hace una unión (join) de las tablas divididas, se debe obtener la tabla original sin normalizar
- Esta unión se llama Descomposición sin Pérdida “Loss-Less Join”
- No debe haber pérdida de datos después de la normalización

Necesidad de Normalizar



Flexibilidad

**Integridad
de Datos**

Eficiencia

Primera Forma Normal (1NF)

- Debe satisfacer las siguientes condiciones:
 - Los campos de variables repetidos o grupos variables repetidos deben ser eliminados
 - Todas las filas deben contener el mismo número de campos

Primera Forma Normal...2

Información de la Orden de Compra

Id Orden	Fecha Orden	Cliente	Email Cliente	Grado	Puntos Credito	Nombre Item	Cant Reque	Precio Unitario
1	11/30/1998	Joe Smith	Joe@ewr.com	A	1	Martillo	5	\$25
1	11/30/1998	Joe Smith	Joe@ewr.com	A	1	Sierra	10	\$30
1	11/30/1998	Joe Smith	Joe@ewr.com	A	1	Clavos	15	\$40
2	12/3/1998	Sam	Sam@ter.com	B	4	Sierra	20	\$30
2	12/3/1998	Sam	Sam@ter.com	B	4	Clavos	25	\$40

Problemas por no Estar en 1NF

- Suponga que se necesita cambiar la dirección Email del Cliente Joe Smith
- Se debe cambiar en 3 lugares
- Esto sería bastante extenso cuando el cliente coloca x órdenes con y items cada uno
- Se tendría que cambiar en $x*y$ lugares

Aplicar 1NF

Tabla Orden en 1NF

IdOrden	Fecha orden	Cliente	EmailCliente	Grado	Puntos Credito
1	11/30/1998	Joe Smith	joe@iewr.com	A	1
2	12/3/1998	Sam	sam@ter.com	B	4

Aplicar 1NF... 2

Tabla ArtículoOrden

IdOrden	Nombreltem	Cant Reque	Precio Unitario
1	Martillo	5	\$25
1	Sierra	10	\$30
1	Clavos	15	\$40
2	Sierra	20	\$30
2	Clavos	25	\$40

Segunda Forma Normal (2NF)

- Un diseño relacional se dice que está en segunda forma normal (2NF) si, y sólo si, está en 1NF y, además, cada columna que no está en la clave primaria es dependiente totalmente la clave primaria
- La 2NF se aplica a las tablas que tienen claves primarias compuestas por dos o más columnas.
- Si una Tabla está en 1NF y su clave primaria es simple (tiene un sola columna), entonces también está en 2NF.

Segunda Forma Normal (2NF)...2

- La Tabla **Orden** está en 2FN ya que su clave primaria es simple
- La Tabla **ArticuloOrden** tiene como clave primaria a las columnas IdOrden y NombreItem, en este caso se aplica la 2NF

Segunda Forma Normal (2NF)...2

La tabla **ArticuloOrden** se divide en 2 Tablas:

- DetalleOrden
- Item

Tabla DetalleOrden

IdOrden	IdItem	Cant Reque	Precio UnitarioVenta
1	1	5	\$25
1	2	10	\$30
1	3	15	\$40
2	2	20	\$30
2	3	25	\$40

Segunda Forma Normal (2NF)...3

En la tabla **Item** se crea el campo IdItem para identificar univocamente los registros

Tabla Item

IdItem	Nombre Item	Precio Unitario
1	Martillo	\$25
2	Sierra	\$30
3	Clavos	\$40

Tercera Forma Normal (3NF)

- Para que un diseño relacional esté en 3NF, tiene que estar en 2NF
- Cada columna no-clave tiene que ser mutuamente independiente
- No debe tener ninguna dependencia transitiva

Dependencia Transitiva

- Si se tiene tres columnas A, B y C en una tabla, donde A es la columna clave
- La dependencia puede ser especificada como:
 $A \rightarrow B, C$
- Si B depende de A, y C depende de B, entonces C también depende de A
- Esta clase de dependencia se llama *Dependencia Transitiva*

¿Por qué no es 3NF?

Las columnas no-clave PuntosCredito y Grado no son mutuamente independiente

Las columnas no-clave cliente y emailCliente no son mutuamente independiente

Tabla Orden

IdOrden	Fecha orden	Cliente	EmailCliente	Grado	Puntos Credito
1	11/30/1998	Joe Smith	joe@iewr.com	A	1
2	12/3/1998	Sam	sam@ter.com	B	4

Tabla Cliente en 3NF

IdCliente	Cliente	EmailCliente	Puntos Credito
1	Joe Smith	joe@iewr.com	1
2	Sam	sam@ter.com	4

Tabla PuntosCredito en 3NF

Puntos Credito	Grado
1	A
2	A
3	A
4	B
5	B
6	B
7	C
8	C
9	C

Tabla Orden en 3NF

IdOrden	FechaOrden	IdCliente
1	11/30/1998	1
2	12/3/1998	2

Cuarta Forma Normal (4NF)

- Tiene que estar en 3NF
- La tabla no debe contener dos o más columnas independientes de múltiples valores

Persona	PaísesVisitados	PremiosRecibidos
---------	-----------------	------------------

Una Tabla Ambigua

Persona	Países Visitados	Premios Recibidos
Scott	Germany	
Scott	France	
Scott		Grammy
Scott		Oscar
Scott		Premio Noble

4NF Datos de la Tabla PaísesVisitados

Persona	PaísesVisitados
Scott	Germany
Scott	France

4NF Datos de la Tabla PremiosRecibidos

Persona	PremiosRecibidos
Scott	Grammy
Scott	Oscar
Scott	Premio Noble

Factores que Afectan la Normalización

- Datos univaluados versus multievaluados
- Dependencia completa de la clave
- Hechos independientes contra hechos dependientes
- La presencia de restricciones mutuas
- La presencia de representaciones no únicas o no singulares

Resumen

- Se explicó qué es la normalización
- Se presentó la importancia de la normalización en las bases de datos
- Se presentó la necesidad de la normalización en las bases de datos
- Se explicaron las diferentes clases de formas normales

Unidad 4:

Laboratorio de Diseño de Base de Datos

Unidad 5:

Laboratorio de Modelado de Base de Datos