

## Unidad 6

# BASE DE DATOS



**UAIOnline**  
**Ultra**»»



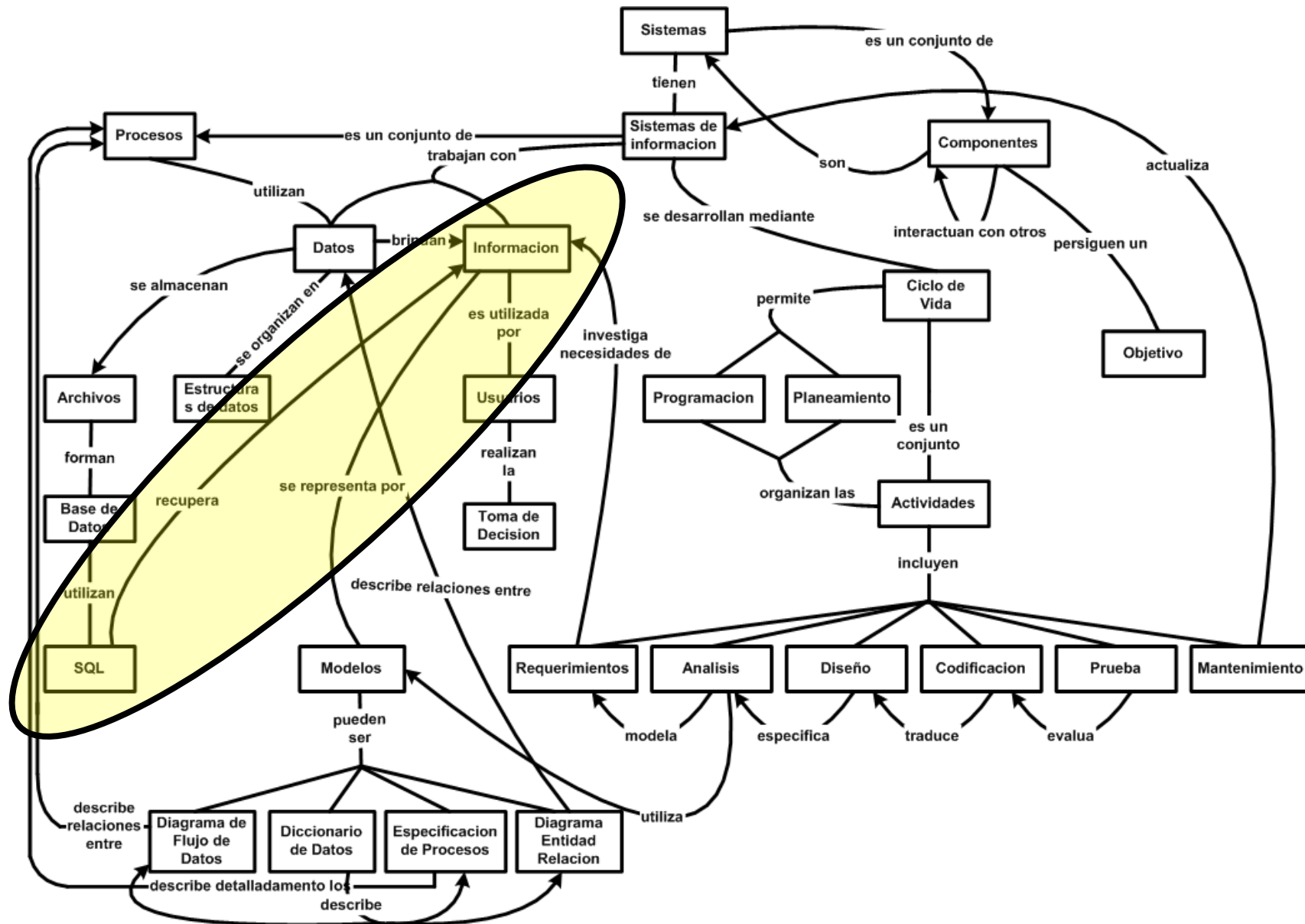
# Base de Datos

## Unidad 6

### ■ OBJETIVOS

- Realizar operaciones de búsqueda de información sobre una estructura de datos utilizando el lenguaje de consulta estructurado.





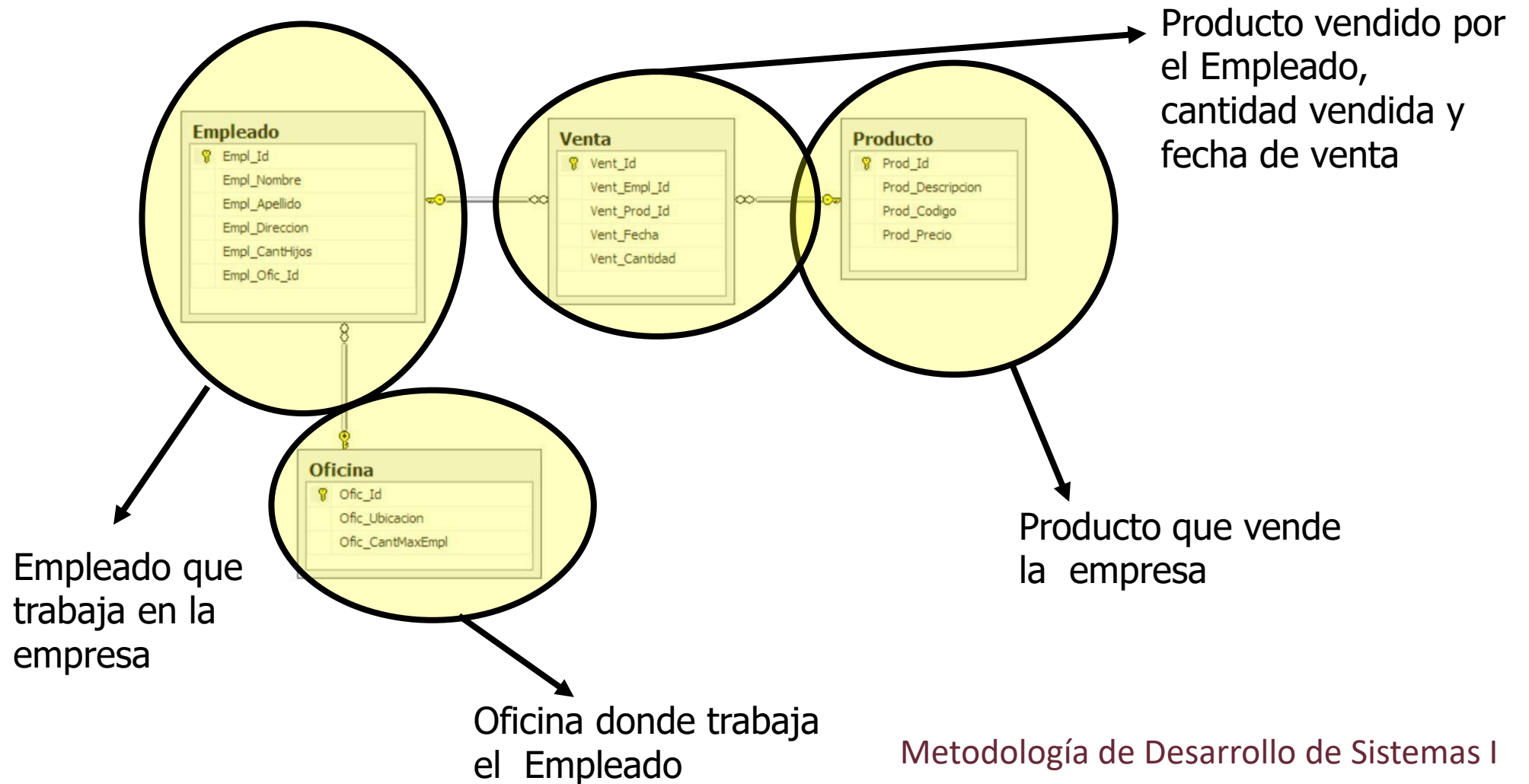
# SQL



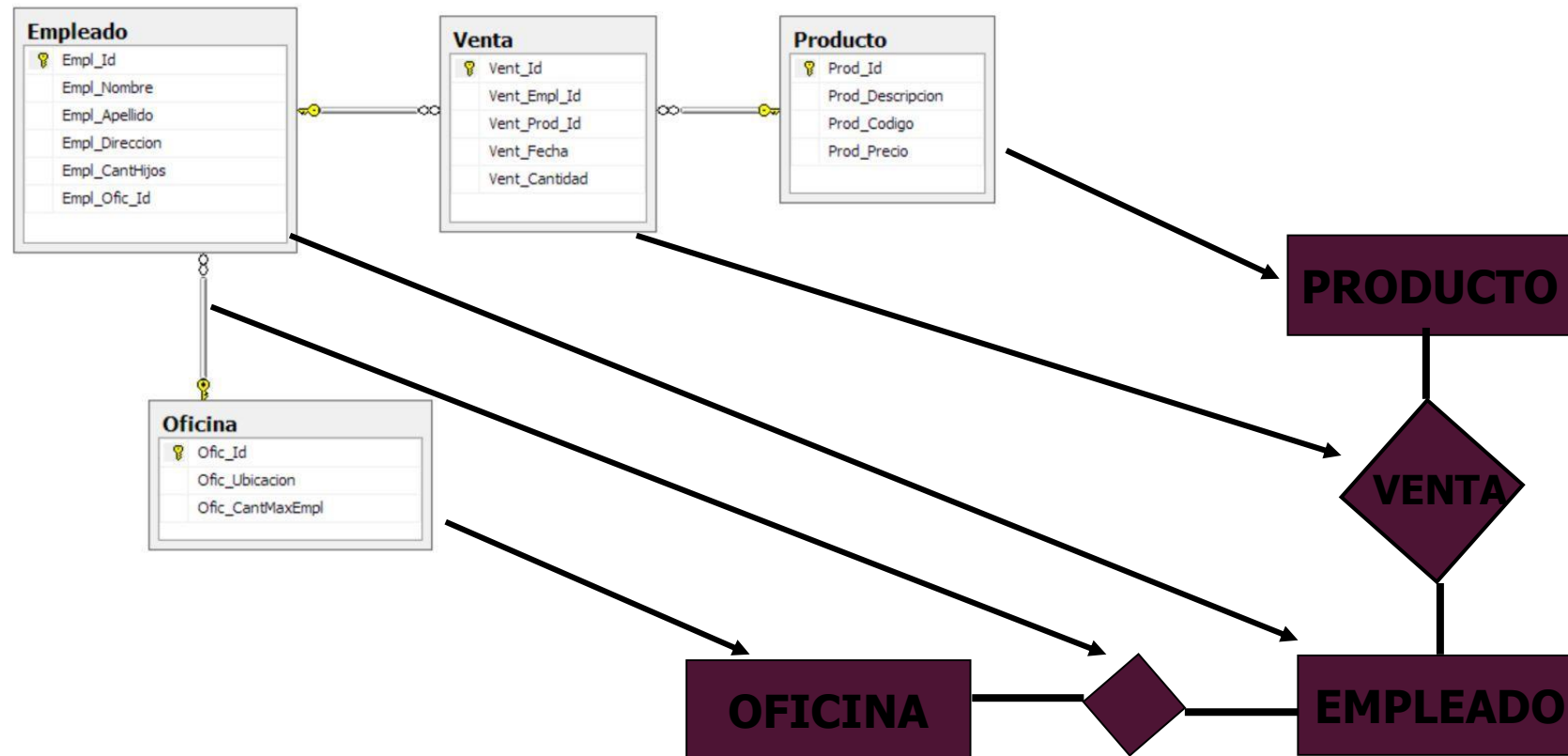
**UAIOnline**  
**Ultra**»»



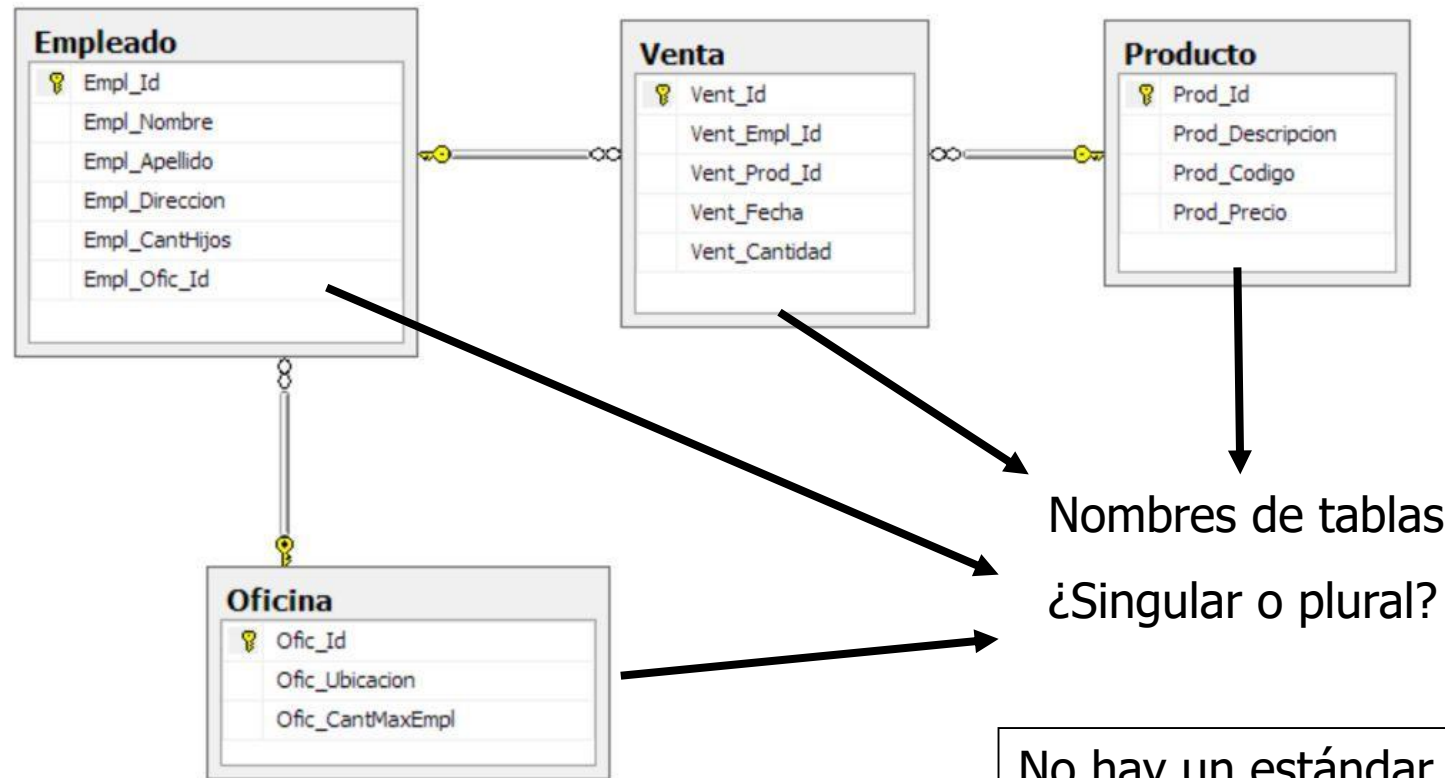
# EJEMPLO



# UNA APROXIMACIÓN AL MODELO ENTIDAD INTERRELACIÓN

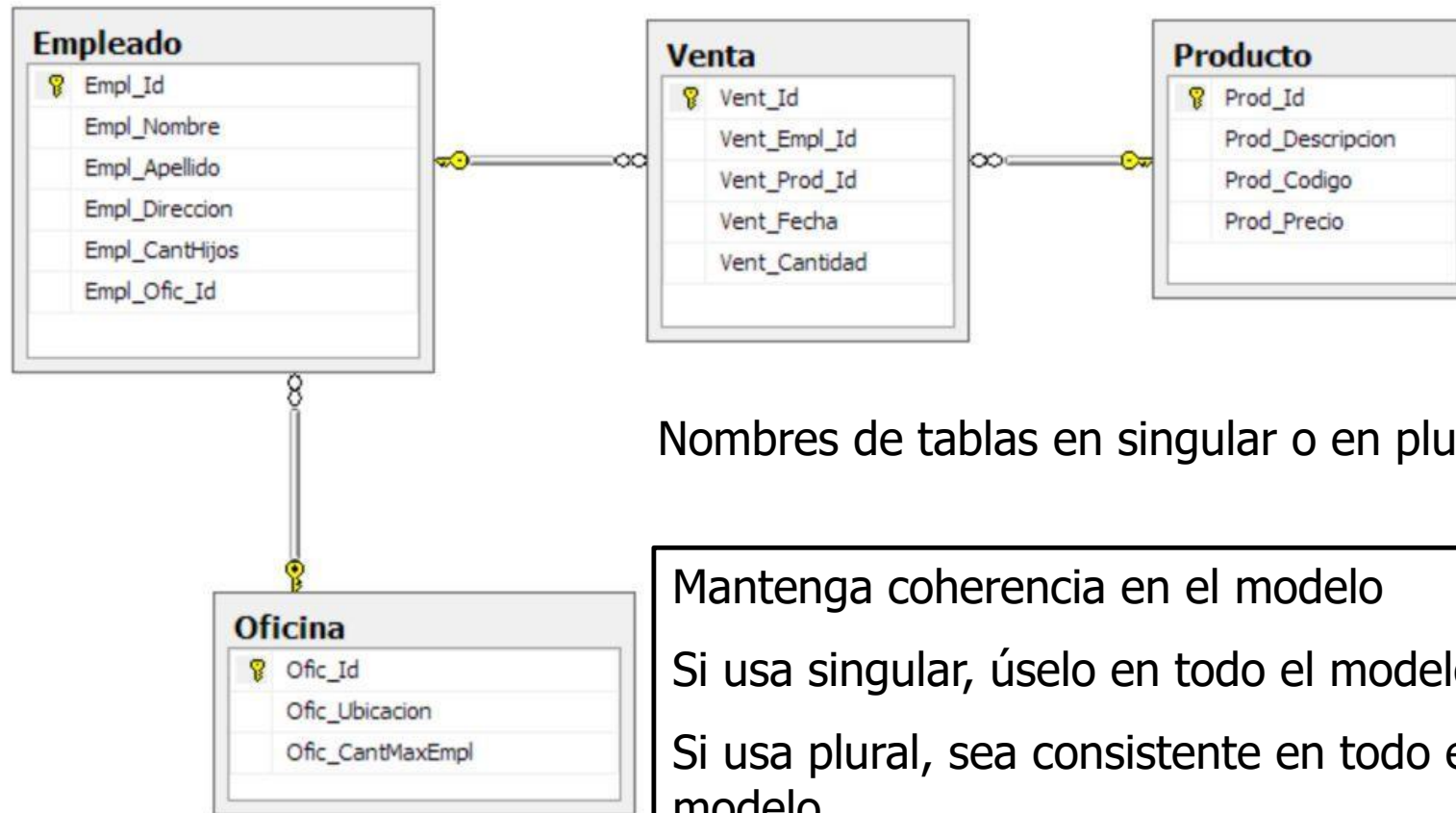


# BUENAS PRÁCTICAS



No hay un estándar. La comunidad establece buenas prácticas.

# BUENAS PRÁCTICAS



Nombres de tablas en singular o en plural

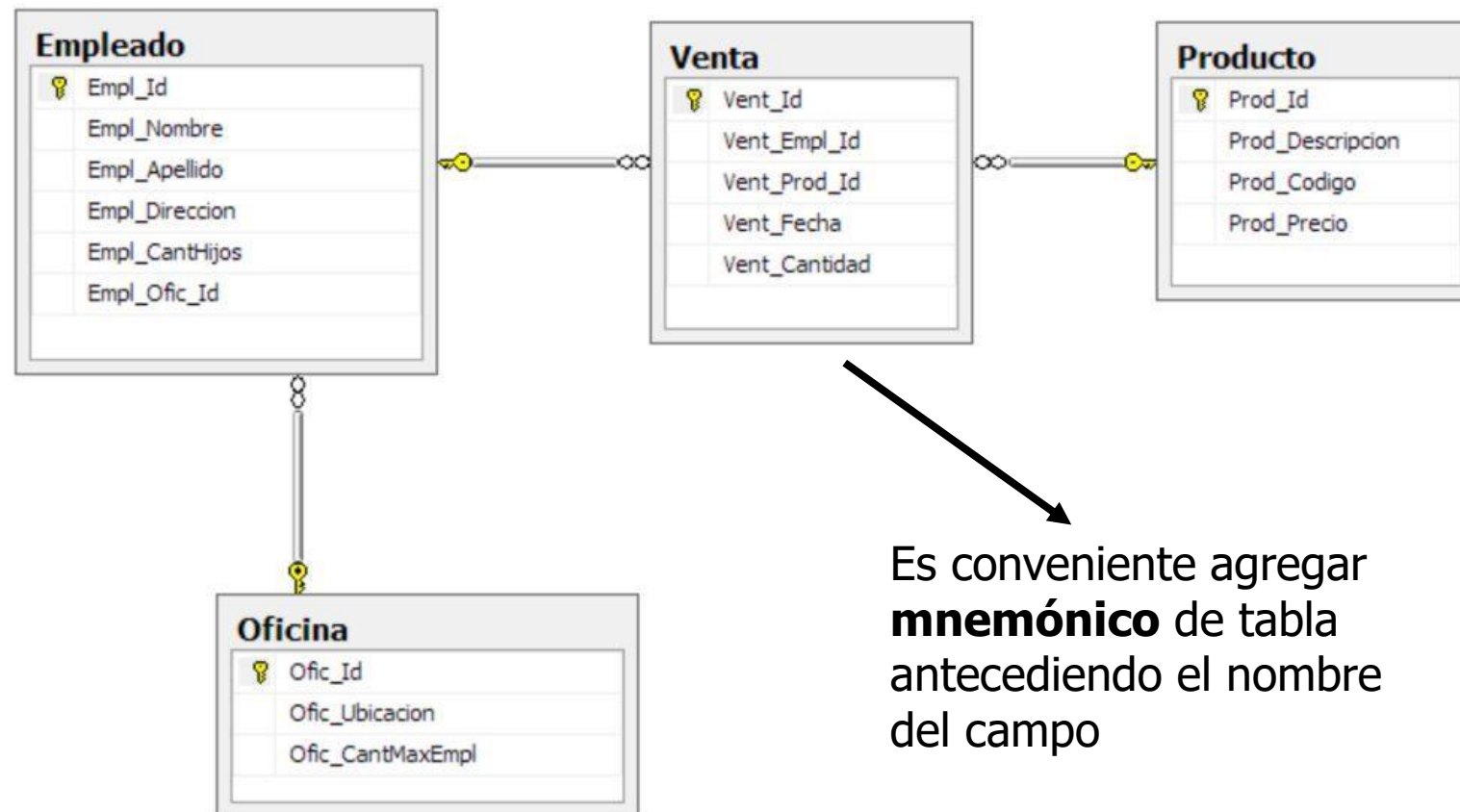
Mantenga coherencia en el modelo

Si usa singular, úselo en todo el modelo

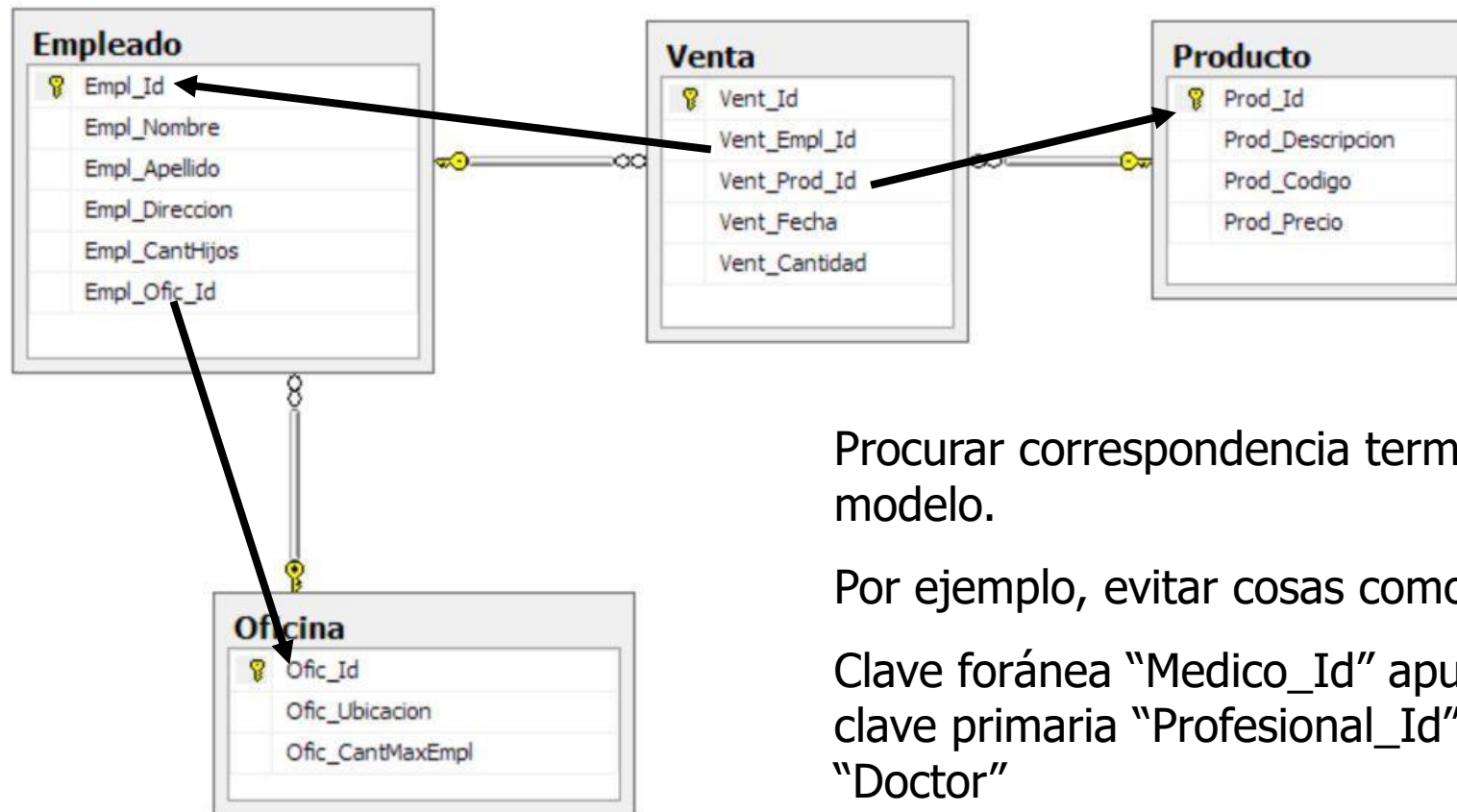
Si usa plural, sea consistente en todo el modelo.



# BUENAS PRÁCTICAS



# BUENAS PRÁCTICAS

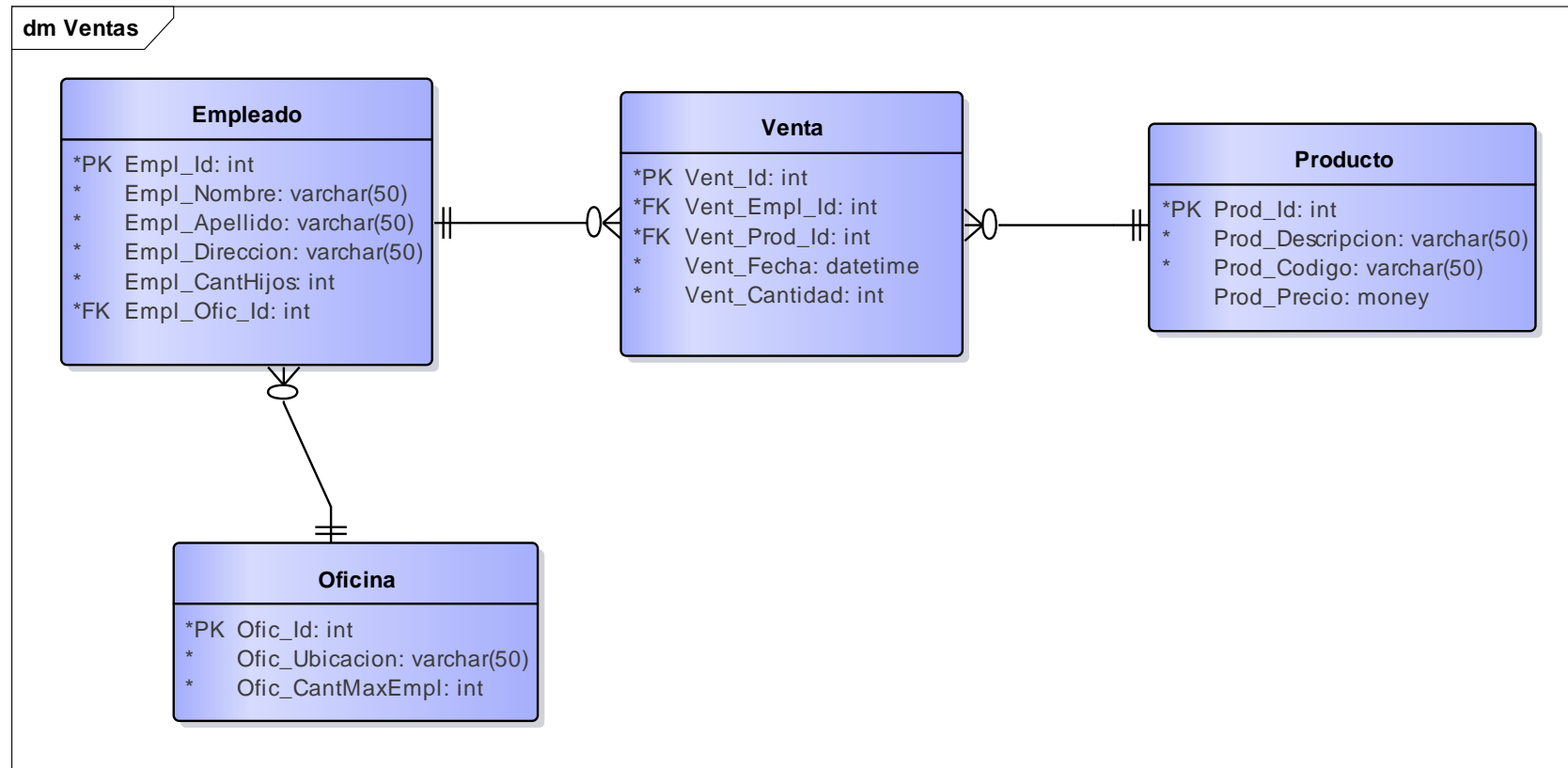


Procurar correspondencia terminológica en el modelo.

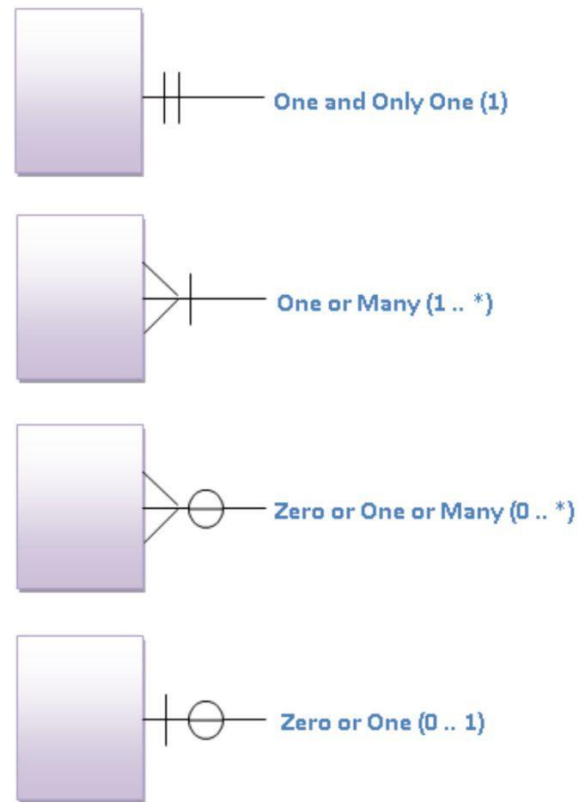
Por ejemplo, evitar cosas como:

Clave foránea "Medico\_Id" apuntando a la clave primaria "Profesional\_Id" de la tabla "Doctor"

# NOTACIÓN MARTIN (O PATA DE GALLO)



# NOTACIÓN MARTIN (O PATA DE GALLO)



# OTRAS NOTACIONES

Bachman



IDEF1X



# OTRAS NOTACIONES

Crow's Foot



Barker's



# OTRAS NOTACIONES

Chen



Min-Max/ISO

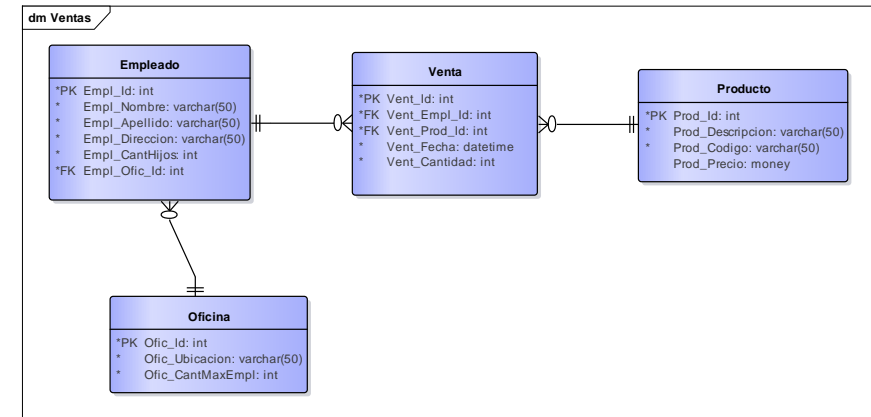


# OTRAS NOTACIONES

Ventajas de notación Martin (Pata de gallo):

Ampliamente usada.

Muestra al mismo tiempo opcionalidad y cardinalidad máxima.





# SQL

SQL (lenguaje de consulta estructurado)

LDD (lenguaje de definición de datos)

permite crear la estructuras, modificarlas y borrarlas

LMD (lenguaje de manipulación de datos)

permite hacer consultas, altas, bajas y modificaciones del contenido de las tablas

# SQL SINTAXIS BÁSICA

SELECT campo(s)

FROM tabla(s)

WHERE <condición>

GROUP BY tabla(s)

HAVING <condición>

ORDER BY campo(s)

Seleccione estos campos, de las siguientes tablas, que cumplan estas condiciones, agrúpelos bajo estas condiciones y ordénelos con estos campos

# SQL CONSULTAS BÁSICAS / I

## Consultas de recuperación de campos

**Selecciones todos los campos de la tabla Empleado**

```
SELECT *  
FROM Empleado;
```

**Seleccione el nombre y el apellido del empleado**

```
SELECT nombre, apellido  
FROM Empleado;
```

**Seleccione el nombre y el apellido de los empleados que vivan en Morón**

```
SELECT nombre, apellido  
FROM Empleado  
WHERE direccion ='Moron'
```

# SQL CONSULTAS BÁSICAS /2

Consultas con operadores lógicos (>, >=, <, <=, <>, AND, OR, NOT)

**Seleccione el nombre y el apellido de los empleados que vivan en Morón y tengas más de 3 hijos**

```
SELECT nombre, apellido  
FROM Empleado  
WHERE direccion ='Moron' AND cant_hijos > 3;
```

Consultas con ordenamientos

**Seleccione el nombre y el apellido y ordénelos por apellido y nombre**

```
SELECT nombre, apellido  
FROM Empleado  
ORDER BY apellido, nombre;
```

# SQL CONSULTAS BÁSICAS /3

Consultas usando operadores comodín (\*, ?)

**Seleccione el nombre y el apellido de los empleados cuyo apellido termine con 's' y el nombre tengan una 'm' como tercera letra**

```
SELECT nombre, apellido  
FROM Empleado  
WHERE apellido LIKE '%s' AND nombre LIKE '__m%';
```

# SQL CONSULTAS BÁSICAS /4

## Funciones de agrupación

Count(), Count (\*), Sum(), Avg(), Max(), Min()

### Determine la cantidad de empleados

```
SELECT count(*)  
FROM Empleado;
```

### Determine la cantidad total de hijos que tienen los empleados en toda la empresa

```
SELECT sum(cant_hijos)  
FROM Empleado;
```

# SQL CONSULTAS BÁSICAS /5

## Consultas con más de una tabla

Listar el nombre y apellido de cada empleado y la ubicación de la oficina que ocupa

```
SELECT nombre, apellido, ubicacion  
FROM empleado, oficina  
WHERE empleado.oficina = oficina.cod_of
```

## Usando inner join

```
SELECT nombre, apellido, ubicación  
FROM empleado inner join oficina  
on empleado.oficina = oficina.cod_of
```

nombre	apellido	ubicacion
jorge	perez	1º piso
raul	martinez	1º piso
maria	fernandez	1º piso
rosa	lima	1º piso
pedro	suarez	2º piso

# COMO SE ARMA LA CONSULTA

empleado

cod_emp	nombre	apellido	direccion	cant_hijos	oficina
01	jorge	perez	merlo	4	01
02	raul	martinez	padua	2	01
03	maria	fernandez	moron	3	02
04	rosa	lima	moreno	1	02
05	pedro	suarez	merlo	0	03

oficina

cod_of	ubicacion	cant_max_emp
01	1º piso	20
02	1º piso	25
03	2º piso	10
04	3º piso	15

```
SELECT nombre, apellido, ubicacion  
FROM empleado, oficina  
WHERE empleado.oficina = oficina.cod_of
```

Resultado de la consulta

nombre	apellido	ubicacion
jorge	perez	1º piso
raul	martinez	1º piso
maria	fernandez	1º piso
rosa	lima	1º piso
pedro	suarez	2º piso



# CONSULTA DE AGRUPAMIENTO

## Determinar la cantidad de ventas por empleado

```
SELECT empleado.cod_emp, empleado.nombre, Sum(venta.cantidad)
FROM empleado INNER JOIN venta ON empleado.cod_emp = venta.cod_emp
GROUP BY empleado.cod_emp, empleado.nombre;
```

cod_emp	nombre	cantidad
01	jorge	30
02	raul	45
03	maria	14

# COMO SE ARMA LA CONSULTA

empleado

cod_emp	nombre	apellido	direccion	cant_hijos	oficina
01	jorge	perez	merlo	4	01
02	raul	martinez	padua	2	01
03	maria	fernandez	moron	3	02
04	rosa	lima	moreno	1	02
05	pedro	suarez	merlo	0	03

venta

cod_emp	cod_prod	fecha	cantidad
01	01	19/02/2006	10
01	02	20/02/2006	15
01	03	21/02/2006	5
02	01	17/03/2006	15
02	02	18/03/2006	30
03	01	25/04/2006	4
03	02	26/04/2006	10

resultado de la consulta

cod_emp	nombre	cantidad
01	jorge	30
02	raul	45
03	maria	14

```
SELECT empleado.cod_emp, empleado.nombre, Sum(venta.cantidad)
FROM empleado INNER JOIN venta ON empleado.cod_emp = venta.cod_emp
GROUP BY empleado.cod_emp, empleado.nombre;
```

# AUTO EVALUACIÓN

Comprendí los conceptos más importantes de la unidad 2.3 y 2.4 si:

- Comprendo cómo influye la estructura de almacenamiento en el proceso de actualización del contenido de la misma
- Entiendo la diferencia entre dato e información en el contexto de las consultas a una base de datos
- Comprendo porqué las consultas no producen anomalías de actualización
- Entiendo la diferencia entre estructura de almacenamiento y el contenido de dicha estructura

# GUÍA DE APRENDIZAJE

## ALUMNO

COD\_ALU TEXTO  
NOM\_ALU  
APE\_ALU  
CALLE\_ALU  
NRO\_ALU  
LOC\_ALU  
FECH\_NAC  
FECH\_ING  
COD\_DIV TEXTO

## PROFESOR

COD\_PRO TEXTO  
NOM\_PRO  
APE\_PRO  
CALLE\_PRO  
NRO\_PRO  
LOC\_PRO

## MATERIAS

COD\_MAT TEXTO  
DES\_MAT  
AÑO\_MAT

## DIVISION

COD\_DIV TEXTO  
AÑO\_DIV  
TUR\_DIV

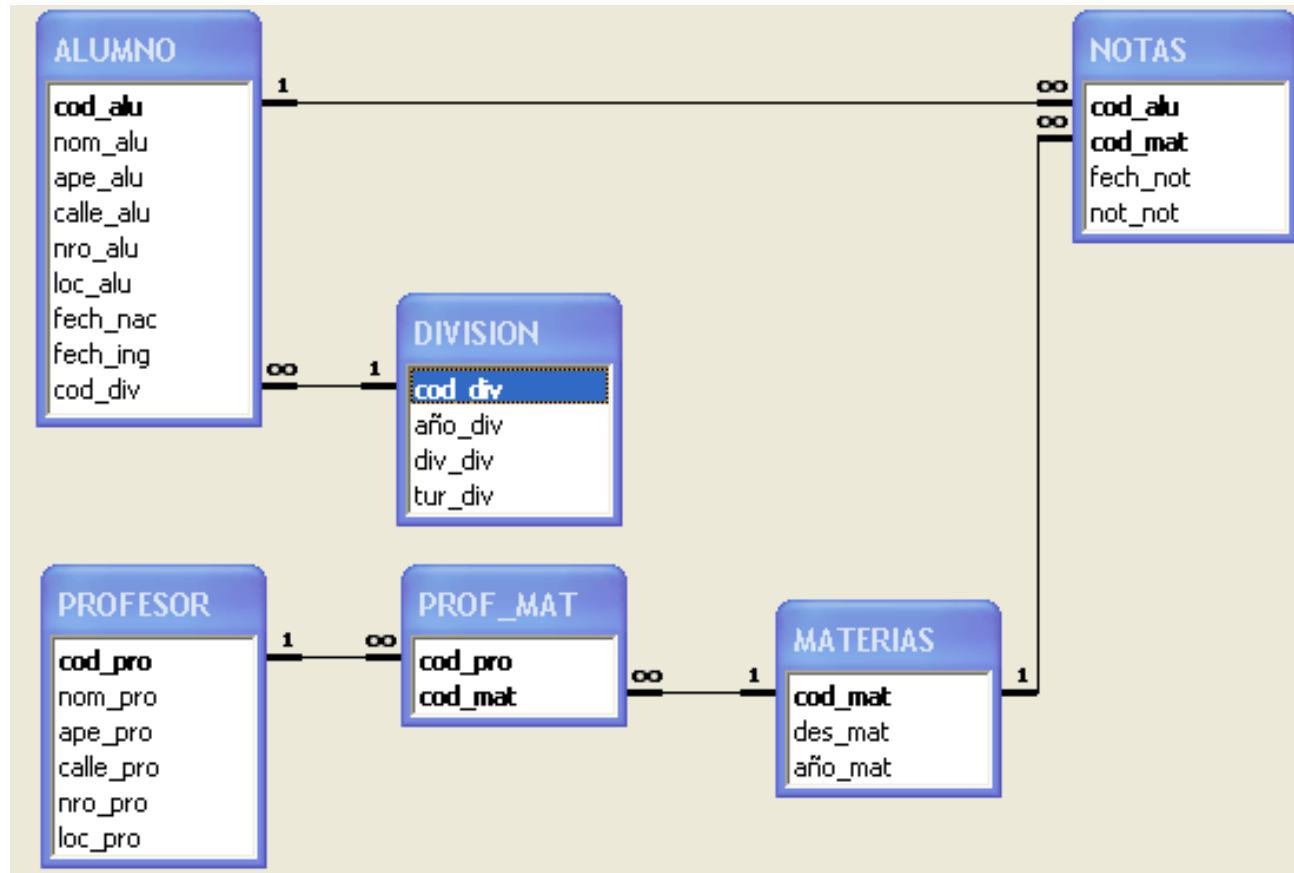
## PROF\_MAT

COD\_PRO TEXTO  
COD\_MAT TEXTO

## NOTAS

COD\_ALU TEXTO  
COD\_MAT TEXTO  
FECH\_NOT  
NOT\_NOT

# GUIA DE APRENDIZAJE





Fin de la clase



**UAI**

**Universidad Abierta  
Interamericana**