



# Bases de Datos

**Dr. Federico DEL RAZO LÓPEZ**

**Universidad Autónoma del Estado de México**  
Maestría en Informática

Feb-Jun / 2010

# Contenido

## Plan

### Antecedentes

Introducción  
Modelo Relacional  
Leng. Formales

### DBMS

Selección DBMS  
Oracle

### SQL

SQL-DDL  
SQL-DML

- 1 **Antecedentes**
  - Introducción
  - Modelo Relacional
  - Lenguajes Formales
    - Álgebra relacional
    - Cálculo Relacional
- 2 Sistema Manejador de Bases de Datos
- 3 Lenguaje Estructurado de Consulta - SQL

## Plan

### Antecedentes

#### **Introducción**

#### Modelo Relacional

#### Leng. Formales

### DBMS

#### Selección DBMS

#### Oracle

### SQL

#### SQL-DDL

#### SQL-DML

Datos → Información → Conocimiento

Inicialmente los datos eran tratados por las empresas como **datos crudos (raw data)**<sup>1</sup> para realizar sus tareas cotidianas.

## Ejemplos

Registros de asistencias.

Pago de salarios.

Administración de pedidos.

etc.

En este contexto las **Bases de datos** han permitido el tratamiento de datos de manera eficiente y coherente.

---

<sup>1</sup>Datos que no han sido procesados para ser utilizados.

# Información

## Plan

Antecedentes  
**Introducción**  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

Después las empresas se apoyaron en los datos para obtener información<sup>2</sup>.

Las aplicaciones fueron reagrupadas bajo el término de **informética operacional** privilegiando el desempeño de tareas repetitivas de: lectura, modificación, eliminación o actualización sobre los datos.

## Aplicación operacional

Es capaz de dar una respuesta precisa a preguntas simples del tipo:

Quién?, Cómo?, Cuándo? ó Dónde?

---

<sup>2</sup>A diferencia de los datos crudos que por ellos mismos carecen de sentido, la información es definida como *information = data + meaning*

# Información...

## Plan

### Antecedentes

#### Introducción

#### Modelo Relacional

#### Leng. Formales

#### DBMS

#### Selección DBMS

#### Oracle

#### SQL

#### SQL-DDL

#### SQL-DML

Motivados por diversas razones:

legales,

seguridad,

seguimiento de clientes o de mercados,

etc.

las empresas decidieron almacenar el conjunto histórico de sus datos (**repositorios**).

Pronto se dieron cuenta que estos repositorios contenían **conocimientos**<sup>3</sup> potencialmente interesantes.

---

<sup>3</sup>Knowledge

# Conocimiento

## Plan

### Antecedentes

#### Introducción

#### Modelo Relacional

#### Leng. Formales

#### DBMS

#### Selección DBMS

#### Oracle

#### SQL

#### SQL-DDL

#### SQL-DML

La arquitectura **cliente-servidor** y los avances en el almacenamiento de datos, hicieron posible la construcción de **repositorios de datos**.

Estos repositorios fueron utilizados para la generación de reportes, mismos que representaban una ayuda a la **toma de decisiones (informática decisional)**.

Paralelamente las comunidades estadísticas e informáticas, desarrollaron métodos para:

- La extracción de **conocimiento**,
- El aprendizaje automático,
- El análisis de datos y
- La minería de datos.

Este **conocimiento** busca dar respuesta preguntas más abiertas y complejas cuya solución no puede ser dada por los lenguajes de consulta tradicionales (e.g. SQL).

# Bases de datos (BD)

## Plan

### Antecedentes

#### **Introducción**

#### Modelo Relacional

#### Leng. Formales

### DBMS

#### Selección DBMS

#### Oracle

### SQL

#### SQL-DDL

#### SQL-DML

## Aplicaciones de las Bases de Datos:

En las compras en un supermercado.

Compras con tarjeta de crédito.

Reservación de vuelos en una agencia de viajes.

Uso de bibliotecas virtuales.

Alquiler de vídeos.

Administración de alumnos.



## Base de datos (BD)

Una colección compartida de datos lógicamente relacionados, junto con una descripción de estos datos, que están diseñados para satisfacer las necesidades de información de una organización.

## Sistema de gestión de base de datos (DBMS)

Un sistema (software) que permite a los usuarios definir, crear, mantener y controlar el acceso a la base de datos.

# DBMS: componentes

## Plan

### Antecedentes

#### Introducción

#### Modelo Relacional

#### Leng. Formales

#### DBMS

#### Selección DBMS

#### Oracle

#### SQL

#### SQL-DDL

#### SQL-DML

Hardware (*e.g.* arquitectura cliente-servidor)

Software (*e.g.* Java, PHP, SQL)

Datos + Metadatos

Procedimientos (*e.g.* inicial, detener, respaldar la BD)

Personas

Administradores,

Diseñadores,

Desarrolladores de aplicaciones,

Usuarios finales,

# DBMS: ventajas

## Plan

### Antecedentes

#### Introducción

#### Modelo Relacional

#### Leng. Formales

### DBMS

#### Selección DBMS

#### Oracle

#### SQL

#### SQL-DDL

#### SQL-DML

Control de redundancia,  
Coherencia de los datos,  
Más información con la misma cantidad de datos,  
Compartición de datos,  
Mayor integridad en los datos,  
Mayor seguridad,  
...

# DBMS: desventajas

## Plan

### Antecedentes

#### Introducción

#### Modelo Relacional

#### Leng. Formales

### DBMS

#### Selección DBMS

#### Oracle

### SQL

#### SQL-DDL

#### SQL-DML

Complejidad,  
Incremento del tamaño,  
Costo del DBMS's,  
Mayor impacto de las fallas,  
...

# Modelo Relacional

## Plan

Antecedentes  
Introducción  
**Modelo Relacional**  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

Modelo lógico

Modelo simple

Relación (tabla)

Atributo (columna)

Definido por **Edgar F. Codd** en 1970; premio Turing en 1986. Desarrollado por **IBM**.

Soporte teórico muy sólido

Actualmente es empleado por muchos SGDB comerciales: *Oracle, Informix, DB2, Ingres, Sybase, Access, ...*

## Plan

Antecedentes

Introducción

**Modelo Relacional**

Leng. Formales

DBMS

Selección DBMS

Oracle

SQL

SQL-DDL

SQL-DML

## Modelo formal:

relación, atributo, tupla, identificador, ...

normalización

álgebra relacional

cálculo relacional

## Plan

Antecedentes  
Introducción  
**Modelo Relacional**  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

## Modelo lógico, implementación: SQL

tabla, columna, línea, llave primaria, ...

SQL-DDL definición y modificación de esquemas

SQL-DML inserción y actualización de datos

SQL consultas

## Relación

Los objetos y las relaciones del ***mundo real*** son representadas por las relaciones. Una relación es una matriz comúnmente llamada **tabla**.



## Alumnos

ID	NOMBRE	DIRECCION
1	Juan	Independencia 10
2	Pedro	Revolución #43
3	Claudia	Juárez #29



# Modelo Relacional

## Plan

Antecedentes  
Introducción  
**Modelo Relacional**  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

## Restricciones de integridad

Integridad de entidades,  
Integridad referencial,  
Datos requeridos,  
Restricciones de dominio  
Restricciones generales,

# Modelo Relacional

## Plan

Antecedentes  
Introducción  
**Modelo Relacional**  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

**Relación Base** Es una relación nominada correspondiente a una entidad del esquema conceptual y cuyas tuplas están almacenadas físicamente en una base de datos.

Dan origen a relaciones **virtuales** o **derivadas**.

## Vistas

Es el resultado dinámico de una o más operaciones relacionales que operan sobre las **relaciones base** para producir otra relación. No existen en la base de datos; son generadas al momento de la solicitud.

# Modelo Relacional ...

## Terminología

### Plan

Antecedentes  
Introducción  
**Modelo Relacional**  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

**Relación** Es una tabla con columnas y filas.

**Atributo** Es una columna nominada de una relación.

**Dominio** Es un conjunto de valores permitidos para uno o más atributos.

**Tupla** Es una fila de una relación.

**Grado** Es el número de atributos que contiene una relación.

**Cardinalidad** Es el número de tuplas que contiene una relación.

### Términos alternativos:

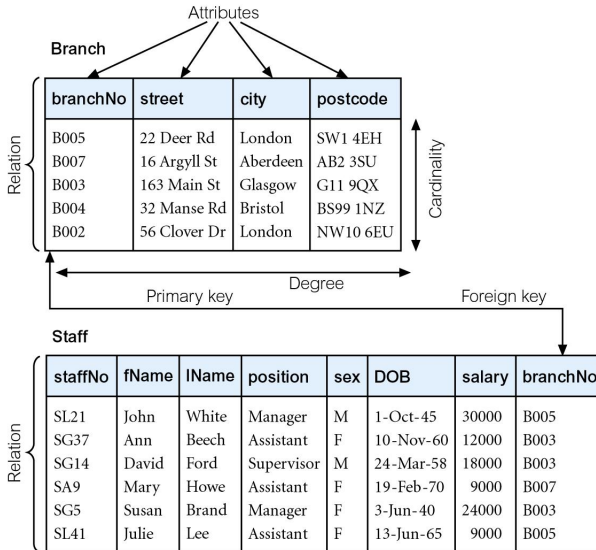
Relación	Tabla	Archivo
Tupla	Fila	Registro
Atributo	Columna	Campo

# Identificación de Relaciones

## Plan

Antecedentes  
Introducción  
**Modelo Relacional**  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML



# Esquema Relacional

## Plan

Antecedentes  
Introducción  
**Modelo Relacional**  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

## Base de datos relacional (BDR)

Es una colección de relaciones normalizadas en la que cada relación tiene un nombre distinto.

Esquema de una base de datos relacional (conjunto de relaciones con nombres distintos)

$$BDR = (R_1, R_2, \dots, R_n)$$

Esquema de una relación (conjunto de **atributos** y **dominios**)

$$R = (a_1 : d_1, a_2 : d_2, \dots, a_m : d_m)$$

ó simplemente

$$R = (a_1, a_2, \dots, a_m)$$

# Caso de estudio: *DreamHome*

## Plan

Antecedentes  
Introducción  
**Modelo Relacional**  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

La primera sucursal de *DreamHome* fue abierta en 1992 en Glasgow, en el Reino Unido. Desde entonces, la empresa ha crecido en forma continua y ahora tiene diversas sucursales en la mayoría de las principales ciudades del Reino Unido.

La empresa es ahora tan grande que cada vez se requiere más personal administrativo para procesar la cantidad, cada vez mayor, de papeles e informes. Además, la comunicación y compartición de la información entre sucursales, incluso dentro de la misma ciudad, es bastante deficiente.

# Caso de estudio: *DreamHome*

## Plan

Antecedentes  
Introducción  
**Modelo Relacional**  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

La directora de la empresa, Sally Mellweadows, cree que se están cometiendo demasiados errores y que el éxito de la empresa puede ser muy efímero si no se hace algo para remediar la situación.

Sabe que una base de datos podría ayudar en parte a resolver el problema por lo cual ha solicitado que se desarrolle un sistema de base de datos para soportar las operaciones de *DreamHome*. ...

# Caso de estudio: *DreamHome*

## Plan

Antecedentes  
Introducción  
**Modelo Relacional**  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

## Esquema de la Base de Datos:

Branch	( <u>branchNo</u> , street, city, postcode)
Staff	( <u>staffNo</u> , fName, lName, position, sex, DOB, salary, branchNo)
PropertyForRent	(propertyNo, street, city, postcode, type, rooms, ownerNo, staffNo, branchNo )
PrivateOwner	( <u>ownerNo</u> , fName, lName, address, telNo)
Client	( <u>clientNo</u> , fName, lName, telNo, prefType, maxRent)
Viewing	( <u>clientNo</u> , propertyNo, viewDate, comment)
Registration	( <u>clientNo</u> , branchNo, staffNo, dateJoined)

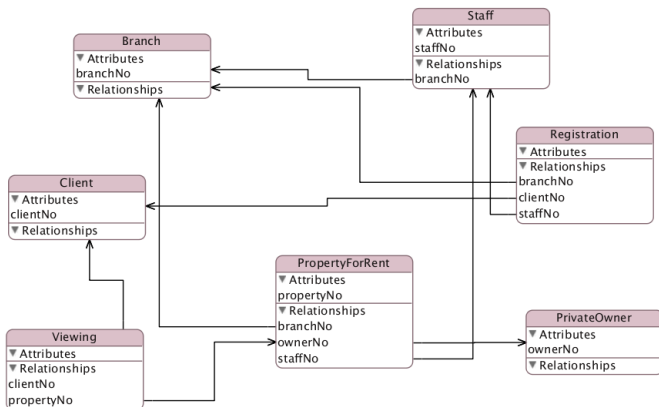


# DreamHome, Diagrama de Relaciones

## Plan

Antecedentes  
Introducción  
**Modelo Relacional**  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

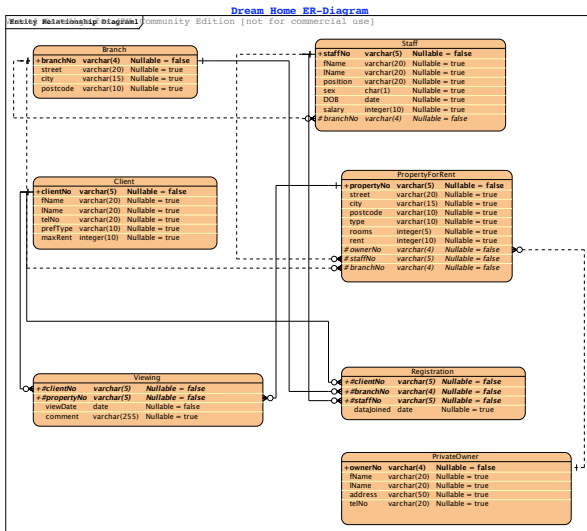


# DreamHome, Diagrama E-R

## Plan

Antecedentes  
Introducción  
**Modelo Relacional**  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML



# Lenguajes Formales

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
**Leng. Formales**

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

## Lenguaje de consulta

Permite extraer y actualizar los datos subyacentes a un modelo relacional (DML, *Data Manipulation Language*).

### Deben:

Ser independientes de la organización interna de las relaciones.

Permitir a los usuarios "lo que ellos quieren buscar en la base de datos", sin importar como lo hagan.

Inicialmente el modelo relacional fue propuesto con dos Lenguajes **formales** de Manipulación de Datos

[[Codd-70](#)]

Álgebra relacional

Cálculo relacional

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

Es un conjunto de operadores que, a partir de una o dos relaciones existentes, dan como resultado una nueva relación temporal.

La relación resultante tiene exactamente las mismas características que una relación de la base de datos y por lo tanto, también, puede ser manipulada por los operadores del álgebra.

Informalmente, el álgebra relacional es un lenguaje (de alto nivel) procedimental<sup>4</sup>.

---

<sup>4</sup>Definen como obtener el resultado deseado.

**Plan**

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

## **Operaciones básicas:**

Selección

Proyección

Producto cartesiano

Unión

Diferencia de Conjuntos

Las anteriores pueden efectuar la mayoría de operaciones para recuperar información.

## **Operaciones complejas:**

Combinación (*Join*)

Intersección

División

# Álgebra Relacional...

## Operaciones básicas

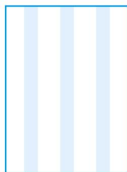
### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

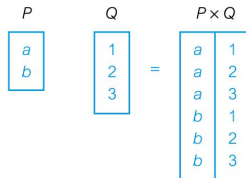
DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML



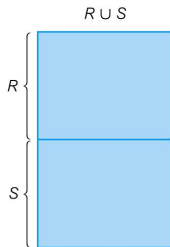
(a) Selection



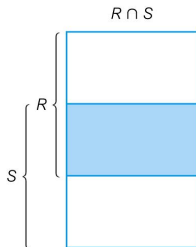
(b) Projection



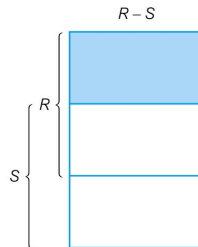
(c) Cartesian product



(d) Union



(e) Intersection



(f) Set difference

# Álgebra Relacional...

## Selección $\sigma$

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

$\sigma_{predicado}(R)$  Se aplica sobre una relación simple  $R$  y define una nueva relación que contiene aquellas tuplas (renglones) de  $R$  que satisfacen la condición especificada en *predicado*.

**Ejemplo:** Listar al personal cuyo salario sea mayor a 10,000.

staffNo	fName	lName	position	sex	DOB	salary	branchNo
SL21	John	White	Manager	M	1-Oct-45	30000	B005
SG37	Ann	Beech	Assistant	F	10-Nov-60	12000	B003
SG14	David	Ford	Supervisor	M	24- Mar-58	18000	B003
SG5	Susan	Brand	Manager	F	3-Jun-40	24000	B003

# Álgebra Relacional...

## Selección $\sigma$ ...

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

**Ejemplo:** Listar al personal cuyo salario sea mayor a 10,000.

staffNo	fName	lName	position	sex	DOB	salary	branchNo
SL21	John	White	Manager	M	1-Oct-45	30000	B005
SG37	Ann	Beech	Assistant	F	10-Nov-60	12000	B003
SG14	David	Ford	Supervisor	M	24- Mar-58	18000	B003
SG5	Susan	Brand	Manager	F	3-Jun-40	24000	B003

Algebra relacional

$\sigma_{salary > 10000}(Staff)$

SQL

```
SELECT *  
FROM Staff  
WHERE salary > 1000;
```



# Álgebra Relacional...

## Proyección $\Pi$

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

$\Pi_{col_1, \dots, col_n}(R)$  Se aplica sobre una relación simple  $R$  y define una relación que contiene un subconjunto vertical de  $R$ , extrayendo los valores de los atributos especificados y eliminando duplicados.

**Ejemplo:** Lista del salarios para todo el personal, mostrando solo: *staffNo*, *fName*, *lName* y *salary*

staffNo	fName	lName	salary
SL21	John	White	30000
SG37	Ann	Beech	12000
SG14	David	Ford	18000
SA9	Mary	Howe	9000
SG5	Susan	Brand	24000
SL41	Julie	Lee	9000

# Álgebra Relacional...

## Proyección $\Pi$

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

**Ejemplo:** Lista del salarios para todo el personal, mostrando solo: *staffNo*, *fName*, *IName* y *salary*

staffNo	fName	IName	salary
SL21	John	White	30000
SG37	Ann	Beech	12000
SG14	David	Ford	18000
SA9	Mary	Howe	9000
SG5	Susan	Brand	24000
SL41	Julie	Lee	9000

Algebra relacional

$\Pi_{staffNo, fName, IName, salary}(Staff)$

SQL

```
SELECT staffNo , fName, IName, salary
FROM Staff;
```

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

$R \cup S$  La unión de dos relaciones  $R$  y  $S$  define una relación que contiene todas las tuplas de  $R$  o  $S$ , o ambas  $R$  y  $S$ , eliminando la tuplas duplicadas.

Si  $R$  y  $S$  tienen  $n$  y  $m$  tuplas respectivamente, la unión es obtenida por la concatenación de ellas en una relación de con  $(n + m)$  tuplas como máximo.

**Ejemplo:** Ciudades que tienen una sucursal o una propiedad en renta.

city
London
Aberdeen
Glasgow
Bristol

**Ejemplo:** Ciudades que tienen una sucursal o una propiedad en renta.

city
London
Aberdeen
Glasgow
Bristol

Algebra relacional

$$\Pi_{city}(Branch) \cup \Pi_{city}(PropertyForRent)$$

SQL

```
SELECT city
FROM Branch
UNION
SELECT city
FROM PropertyForRent;
```

# Diferencia de conjuntos —

**Plan**

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

$R - S$  Define una relación compuesta por las tuplas que se encuentran en la relación  $R$ , pero no en  $S$ .

$R$  y  $S$  deben ser compatibles con respecto a la unión.

**Ejemplo:** Listado de las ciudades que tienen una sucursal pero **NO** propiedades en renta.

city
Bristol

# Diferencia de conjuntos —

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

**Ejemplo:** Listado de las ciudades que tienen una sucursal pero **NO** propiedades en renta.

city
Bristol

Algebra relacional

$$\Pi_{city}(Branch) - \Pi_{city}(PropertyForRent)$$

SQL

```
SELECT city  
FROM Branch  
MINUS  
SELECT city  
FROM PropertyForRent;
```

# Álgebra Relacional...

## Intersección $\cap$

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

$R \cap S$  Define una relación compuesta por las tuplas que se encuentran en ambas relaciones  $R$  y  $S$ .

$R$  y  $S$  deben ser compatibles con respecto a la unión.

$$R \cap S = R - (R - S)$$

**Ejemplo:** Listado de las ciudades en las que exista tanto una sucursal como al menos una propiedad en renta.

city
Aberdeen
London
Glasgow

# Álgebra Relacional...

## Intersección $\cap$

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

**Ejemplo:** Listado de las ciudades en las que exista tanto una sucursal como al menos una propiedad en renta.

city
Aberdeen
London
Glasgow

Algebra relacional

$$\Pi_{city}(Branch) \cap \Pi_{city}(PropertyForRent)$$

SQL

```
SELECT city
FROM Branch
INTERSECTION
SELECT city
FROM PropertyForRent;
```



# Producto cartesiano $RXS$

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

$RXS$  Define una relación formada por la concatenación de cada tupla de la relación  $R$  con cada tupla de la relación  $S$ .

**Ejemplo:** Enumerar los nombres y comentarios de todos los clientes que hayan visitado un inmueble en alquiler.

client.clientNo	fName	lName	Viewing.clientNo	propertyNo	comment
CR76	John	Kay	CR56	PA14	too small
CR76	John	Kay	CR76	PG4	too remote
CR76	John	Kay	CR56	PG4	
CR76	John	Kay	CR62	PA14	no dining room
CR76	John	Kay	CR56	PG36	
CR56	Aline	Stewart	CR56	PA14	too small
CR56	Aline	Stewart	CR76	PG4	too remote
CR56	Aline	Stewart	CR56	PG4	
CR56	Aline	Stewart	CR62	PA14	no dining room
CR56	Aline	Stewart	CR56	PG36	
CR74	Mike	Ritchie	CR56	PA14	too small
CR74	Mike	Ritchie	CR76	PG4	too remote
CR74	Mike	Ritchie	CR56	PG4	
CR74	Mike	Ritchie	CR62	PA14	no dining room
CR74	Mike	Ritchie	CR56	PG36	
CR62	Mary	Tregear	CR56	PA14	too small
CR62	Mary	Tregear	CR76	PG4	too remote
CR62	Mary	Tregear	CR56	PG4	
CR62	Mary	Tregear	CR62	PA14	no dining room
CR62	Mary	Tregear	CR56	PG36	

# Producto cartesiano $R \times S$

Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

**Ejemplo:** Enumerar los nombres y comentarios de todos los clientes que hayan visitado un inmueble en alquiler.

client.clientNo	fName	lName	viewing.clientNo	propertyNo	comment
CR76	John	Kay	CR56	PA14	too small
CR76	John	Kay	CR76	PG4	too remote
CR76	John	Kay	CR56	PG4	
CR76	John	Kay	CR62	PA14	no dining room
CR76	John	Kay	CR56	PG36	
CR56	Aline	Stewart	CR56	PA14	too small
CR56	Aline	Stewart	CR76	PG4	too remote
CR56	Aline	Stewart	CR56	PG4	
CR56	Aline	Stewart	CR62	PA14	no dining room
CR56	Aline	Stewart	CR56	PG36	
CR74	Mike	Ritchie	CR56	PA14	too small
CR74	Mike	Ritchie	CR76	PG4	too remote
CR74	Mike	Ritchie	CR56	PG4	
CR74	Mike	Ritchie	CR62	PA14	no dining room
CR74	Mike	Ritchie	CR56	PG36	
CR62	Mary	Tregear	CR56	PA14	too small
CR62	Mary	Tregear	CR76	PG4	too remote
CR62	Mary	Tregear	CR56	PG4	
CR62	Mary	Tregear	CR62	PA14	no dining room
CR62	Mary	Tregear	CR56	PG36	

Algebra relacional

$\Pi_{clientNo, fName, lName}(Client) \times \Pi_{clientNo, propertyNo, comment}(Viewing)$

SQL

```
SELECT c.clientNo , c.fName , c.lName ,
       v.clientNo , v.propertyNo , v.comments
FROM   Client c, Viewing v;
```

## Álgebra Relacional...

# Operaciones complejas

- Antecedentes
- Introducción
- Modelo Relacional
- Leng. Formales

- DBMS
  - Selección DBMS
  - Oracle
- SQL
  - SQL-DDL
  - SQL-DML

$U$	
$B$	$C$
1	$x$
1	$y$
3	$z$

$T \triangleright_B U$	
$A$	$B$
$a$	1

$$T \propto_c U$$

$A$	$B$	$C$
$a$	1	$x$
$a$	1	$y$
$b$	2	

(i) Left Outer join

S

$R \div S$

$V$	
$A$	$B$
$a$	1
$a$	2
$b$	1
$b$	2
$c$	1

$W$	$B$
	1
	2

$A$
$a$ $b$

### Example of division

# Operaciones complejas ...

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

## Combinación

Combinación Theta ( $\theta$ —combinación)

Equi-combinación

Combinación natural

Combinación externa

Semi-combinación

## División

## Agregación y agrupamiento

# Combinación Theta ( $\theta$ -join) ...

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

$R \bowtie_F S$  Define una relación que contiene tuplas satisfaciendo el predicado  $F$ , a partir del **producto cartesiano** de  $R$  y  $S$ .

El predicado  $F$  tiene la forma  $R.a_i \theta S.b_j$ , donde  $\theta$  puede ser cualquiera de los operadores de comparación ( $<, \leq, > \geq, =, \neq$ ).

Este operador puede ser reescrito usando las operaciones básicas de **Selección** y **Producto cartesiano**.

$$R \bowtie_F S = \sigma_F(R \times S)$$

El grado de una **Combinación Theta** es la suma de los grados de  $R$  y  $S$ .

Si el predicado  $F$  contiene sólo una igualdad ( $=$ ), entonces se utiliza el término **Equi-combinación**.

Combinación Theta ( $\theta$ -join) ...

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

**Ejemplo:** Enumerar los nombres y comentarios de todos los clientes que hayan visitado un inmueble en alquiler.

client.clientNo	fName	lName	Viewing.clientNo	propertyNo	comment
CR76	John	Kay	CR76	PG4	too remote
CR56	Aline	Stewart	CR56	PA14	too small
CR56	Aline	Stewart	CR56	PG4	
CR56	Aline	Stewart	CR56	PG36	
CR62	Mary	Tregear	CR62	PA14	no dining room

Algebra relacional

$(\Pi_{clientNo, fName, lName}(Client)) \bowtie_{Client.clientNo = Viewing.clientNo}$

$(\Pi_{clientNo, propertyNo, comment}(Viewing))$

Combinación Theta ( $\theta$ -join) ...

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

**Ejemplo:** Enumerar los nombres y comentarios de todos los clientes que hayan visitado un inmueble en alquiler.

client.clientNo	fName	lName	Viewing.clientNo	propertyNo	comment
CR76	John	Kay	CR76	PG4	too remote
CR56	Aline	Stewart	CR56	PA14	too small
CR56	Aline	Stewart	CR56	PG4	
CR56	Aline	Stewart	CR56	PG36	
CR62	Mary	Tregear	CR62	PA14	no dining room

## SQL

```
SELECT c.clientNo , c.fName, c.lName,
       v.clientNo , v.propertyNo , v.comments
FROM   Client c, Viewing v
WHERE  c.clientNo = v.clientNo ;
```

# Álgebra Relacional...

## Combinación natural

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

$R \bowtie S$  Es una **Equi-combinación** de dos relaciones  $R$  y  $S$  sobre todos los atributos comunes  $x$ . Una ocurrencia de cada atributo común es eliminada del resultado.

**Ejemplo:** Enumerar los nombres y comentarios de todos los clientes que hayan visitado un inmueble en alquiler.

client.clientNo	fName	lName	Viewing.clientNo	propertyNo	comment
CR76	John	Kay	CR76	PG4	too remote too small
CR56	Aline	Stewart	CR56	PA14	
CR56	Aline	Stewart	CR56	PG4	
CR56	Aline	Stewart	CR56	PG36	
CR62	Mary	Tregear	CR62	PA14	no dining room



# Álgebra Relacional...

## Combinación natural

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

**Ejemplo:** Enumerar los nombres y comentarios de todos los clientes que hayan visitado un inmueble en alquiler.

client.clientNo	fName	lName	Viewing.clientNo	propertyNo	comment
CR76	John	Kay	CR76	PG4	too remote
CR56	Aline	Stewart	CR56	PA14	too small
CR56	Aline	Stewart	CR56	PG4	
CR56	Aline	Stewart	CR56	PG36	
CR62	Mary	Tregear	CR62	PA14	no dining room

## Álgebra Relacional

$$(\pi_{clientNo, fName, lName}(Client)) \bowtie$$
$$(\pi_{clientNo, propertyNo, comment}(Viewing))$$

## SQL

```
SELECT c.clientNo , c.fName , c.lName ,  
       v.clientNo , v.propertyNo , v.comments  
FROM   Client c, Viewing v  
WHERE  c.clientNo = v.clientNo ;
```

## Álgebra Relacional...

# Combinación externa

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

Se utiliza para mostrar los renglones en el resultado, que no tienen un valor correspondiente en la columna de combinación (join).

$R \bowtie S$  La combinación externa a la izquierda, combina las tuplas de  $R$  que no tienen valores correspondientes en las columnas comunes de  $S$  y las muestra en la relación resultante.

**Ejemplo:** Mostrar un reporte sobre las visitas a las propiedades.

propertyNo	street	city	clientNo	viewDate	comment
PA14	16 Holhead	Aberdeen	CR56	24-May-01	too small
PA14	16 Holhead	Aberdeen	CR62	14-May-01	no dining room
PL94	6 Argyll St	London	null	null	null
PG4	6 Lawrence St	Glasgow	CR76	20-Apr-01	too remote
PG4	6 Lawrence St	Glasgow	CR56	26-May-01	
PG36	2 Manor Rd	Glasgow	CR56	28-Apr-01	
PG21	18 Dale Rd	Glasgow	null	null	null
PG16	5 Novar Dr	Glasgow	null	null	null

Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

Las consultas de cálculo relacional especifican **qué hay que extraer**, en lugar de **como extraerlo**.

Se basa en el cálculo de predicados (lógica de primer orden). Un predicado es una función booleana con argumentos.

- Cálculo relacional de tuplas
- Cálculo relacional de dominios

Cuando se asignan valores a los argumentos, la función nos proporciona una expresión, denominada **proposición** que puede ser verdadera o falsa.

# Lenguajes Formales ...

## Cálculo Relacional...

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

Si un predicado contiene una variable (*e.g.* "x es un empleado"), entonces debe existir un **rango** asociado para  $x$ .

Cuando se sustituye  $x$  por algunos valores de este rango, entonces, la proposición puede ser cierta; para otros valores la proposición será falsa.  
*e.g.* "**John White** es empleado." <sup>es</sup> verdadera.

Si  $P$  es un predicado, entonces podemos escribir:  
El conjunto de todos los  $x$  tales que  $P$  es verdadero

$$\{x | P(x)\}$$

**Nota:** Es posible conectar unos predicados con otros mediante los conectores  $\wedge$  (AND),  $\vee$  (OR) y  $\sim$  (NOT) para formar predicados compuestos.

Uno se interesa en localizar las tuplas para las que un cierto predicado sea verdadero.

Este cálculo esta basado en el uso de **variables de tuplas**<sup>5</sup>.

*e.g.* Para especificar el rango de una variable de tupla  $S$  de forma que el rango sea la relación *Staff*, se escribirá.

$$Staff(S)$$

*e.g.* Extraer todo el conjunto de tuplas  $S$  tales que  $F(S)$  sea verdadera.

$$\{S | F(S)\}$$

---

<sup>5</sup>Variables que toman sus "valores" de una determinada relación

# Cálculo Relacional de tuplas ...

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

e.g. Extraer los valores de **staffNo**, **fName**, **lName**, **position**, **sex**, **DOB**, **salary** y **branchNo** para los empleados que ganes más de 10,000 dls.

$$\{S \mid Staff(S) \wedge S.salary > 10000\}$$

e.g. Extraer un atributo concreto de los empleados que ganes más de 10,000 dls.

$$\{S.salary \mid Staff(S) \wedge S.salary > 10000\}$$

# Cálculo Relacional de tuplas ...

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

## Cuantificadores:

### Existencial $\exists$ "existe".

e.g. Existe una tupla de **branch** que tiene el mismo valor de **branchNo** que el valor de **branchNo** correspondiente a la tupla actual de **Staff** *S* y cuya ciudad correspondiente es "Londres".

$$\{ Staff(S) \wedge (\exists B)(Branch(B)) \wedge (B.branchNo = S.branchNo) \wedge B.city = "London" \}$$

### Universal $\forall$ "para todo".

e.g. Para todas las tuplas de **Branch**, la dirección no corresponde a "Paris".

$$(\forall B)(B.city \neq "Paris")\}$$

## Plan

- Antecedentes
- Introducción
- Modelo Relacional
- Leng. Formales

- DBMS
- Selección DBMS
- Oracle
- SQL
- SQL-DDL
- SQL-DML

## Expresiones inseguras

Algunas expresiones pueden generar un conjunto infinito.

*e.g.*

$$\{S \mid \sim Staff(S)\}$$

Representa al conjunto de todas las tuplas que no se encuentran en la relación de **Staff**.



Para evitar las expresiones **inseguras**, se debe cuidar que todos los valores que aparezcan en el resultado, deben ser valores contenidos en el **dominio** de una expresión  $E$ .

$dom(E)$  conjunto de todos los valores que aparecen explícitamente en  $E$  o que aparecen en una o mas relaciones cuyos nombre aparecen en  $E$ .

Una expresión es **segura** si todos los valores que aparecen en el resultado son valores pertenecientes al dominio de la expresión.

# Cálculo Relacional de dominios...

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

En el cálculo relacional de dominios se utilizan variables que toman sus valores de los **dominios** de los atributos.

Para toda expresión de álgebra relacional, existe una expresión equivalente en el cálculo relacional.

Para toda tupla o expresión del cálculo relacional de dominios, existe una expresión equivalente en el álgebra relacional.

# Contenido

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

## DBMS

Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

- 1 Antecedentes
- 2 **Sistema Manejador de Bases de Datos**  
Selección de un DBMS  
Oracle
- 3 Lenguaje Estructurado de Consulta - SQL

# Selección de un DBMS

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

## Debe permitir:

Diseño **conceptual/lógico** de la base de datos.

Diseño **físico** de la base de datos.

## Características a considerar:

Definición de datos: PK, FK, vistas, dominios, tipos de datos, ...

Definición física: Requisitos de memoria y disco, estructura de archivos, ...

Accesibilidad: compatible con: **SQL2/SQL:2003/ODMG**, interfaz con lenguajes **3GL**, ...

Gestión de transacciones: copias de seguridad y recuperación, registro de actividades, ...

Utilidades: Optimización, carga, descarga, administración, ...

Herramientas de desarrollo: **CASE**, Herramientas **4GL/5GL**, desarrollo Web, ...

Otras: actualización, documentación, soporte técnico, costo, soporte XML, ...

## Oracle Server

Es un Sistema de Gestión de Bases de Datos Objeto-Relacional que provee de forma abierta y entendible, el manejo de información.

### Componentes:

**Instancia:** Es una combinación de procesos de *background* y de estructuras de memoria.

**Base de Datos:** Es una colección de datos que son tratados como una **unidad**.

Está formada por:

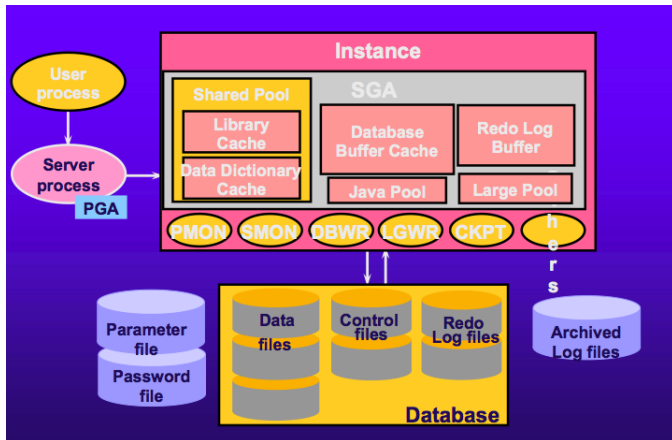
una estructura **lógica** y  
una estructura **física**

# DBMS Oracle arquitectura

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
**Oracle**  
SQL  
SQL-DDL  
SQL-DML



NOTA: La **instancia** es la única forma que tienen los usuarios para acceder a la Base de Datos.

# DBMS Oracle, creación de la Base de Datos

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
**Oracle**  
SQL  
SQL-DDL  
SQL-DML

## 1 Instalación de Oracle Server, usando el **Oracle Universal Installer**

Start -> Programs -> Oracle Installation Products -> **Universal Installer**

## 2 Crear la Base de Datos con el **Configuration & Migration Tools -> Database Assistant**.



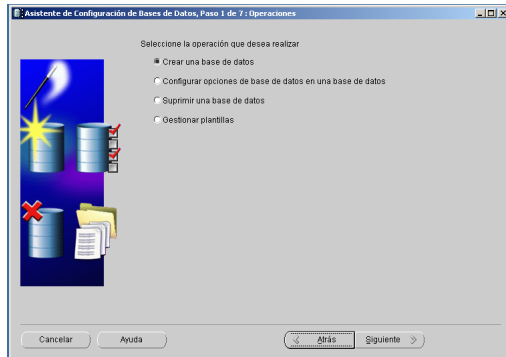
# DBMS Oracle, creación de la Base de Datos

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
**Oracle**  
SQL  
SQL-DDL  
SQL-DML

¿Crear, borrar, configurar una BD?





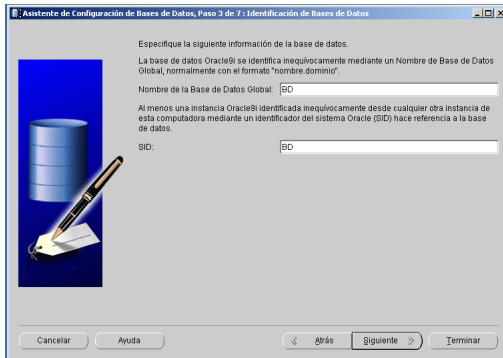
# DBMS Oracle, creación de la Base de Datos

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
**Oracle**  
SQL  
SQL-DDL  
SQL-DML

Dar un nombre a la **Instancia** y a la BD.



# DBMS Oracle, creación de la Base de Datos

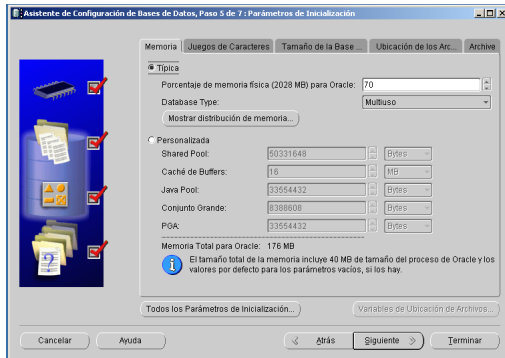
## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
**Oracle**  
SQL  
SQL-DDL  
SQL-DML

Seleccionar los parámetros típicos

Al terminar de crear la BD, Oracle crea los usuarios: **SYS** y **SYSTEM**, a los cuales se les debe asignar una contraseña.



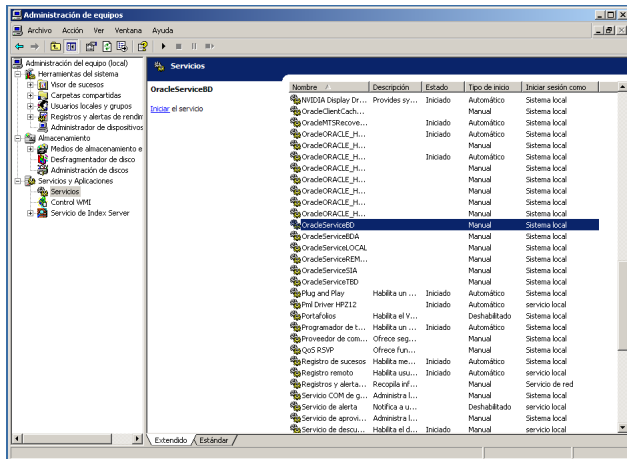
# DBMS Oracle, creación de la Base de Datos

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

ir a **Mi PC** -> **Administrar**. Se ha creado un servicio asociado a la **instancia**. Se puede **iniciar** o **parar** la instancia.



# DBMS Oracle, creación de la Base de Datos

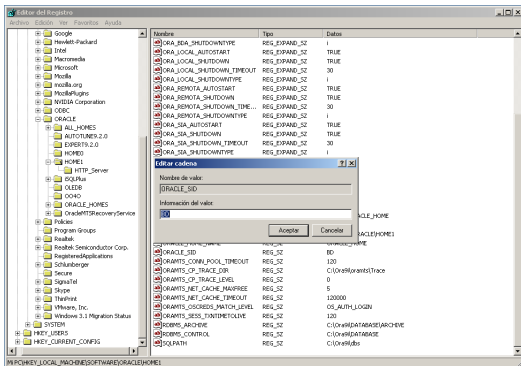
### Plan

- Antecedentes
- Introducción
- Modelo Relacional
- Leng. Formales

- DBMS
  - Selección DBMS
  - Oracle**
- SQL
  - SQL-DDL
  - SQL-DML

ir a **Inicio** -> **Ejecutar** -> **regedit**. Asignar el nombre de la **Instancia** creada a la variable de ambiente **ORACLE\_SID**.

HKEY\_LOCAL\_MACHINE -> SOFTWARE -> ORACLE -> HOME  
-> ORACLE SID.



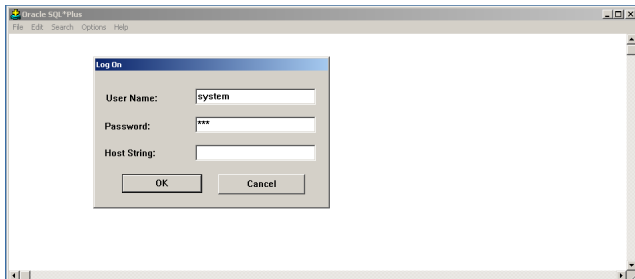
# DBMS Oracle, creación de la Base de Datos

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
**Oracle**  
SQL  
SQL-DDL  
SQL-DML

Conectarse a la BD, usando el programa **sqlplus**.

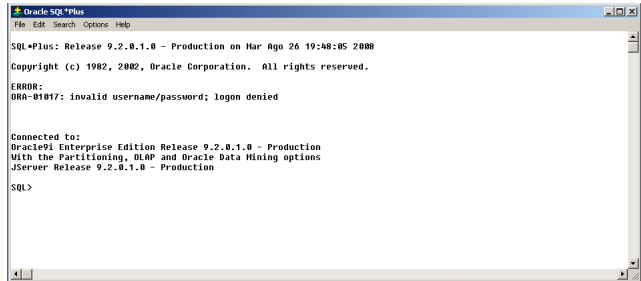


# DBMS Oracle, creación de la Base de Datos

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
**Oracle**  
SQL  
SQL-DDL  
SQL-DML



```
Oracle SQL*Plus
File Edit Search Options Help

SQL*Plus: Release 9.2.0.1.0 - Production on Mar Ago 26 19:48:05 2008

Copyright (c) 1982, 2002, Oracle Corporation. All rights reserved.

ERROR:
ORA-01017: invalid username/password; login denied

Connected to:
Oracle9i Enterprise Edition Release 9.2.0.1.0 - Production
With the Partitioning, OLAP and Oracle Data Mining options
JServer Release 9.2.0.1.0 - Production

SQL>
```

```
SHOW USER;
CREATE USER clase IDENTIFIED BY clase;
GRANT CONNECT, CREATE SESSION, RESOURCE TO clase;

CONNECT clase/clase;
SHOW USER;
```

# DBMS Oracle, creación de la Base de Datos

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
**Oracle**  
SQL  
SQL-DDL  
SQL-DML

## sqlplus



```
SQL*Plus: Release 9.2.0.1.0 - Production on Mon Apr 26 16:16:40 2004
Copyright (c) 1982, 2002, Oracle Corporation. All rights reserved.

USER
SQL>

Connected to:
Oracle9i Enterprise Edition Release 9.2.0.1.0 - Production
With the Partitioning, SQL and Data Warehouse options
Copyright (c) 2002, Oracle Corporation. All rights reserved.

SQL>
```

## Establecer el tamaño de página

```
SET LINESIZE 100
SET PAGESIZE 100
```

**I**, lista la última sentencia en el buffer.

**/**, ejecuta la última sentencia en el buffer.

**@ /.../archivo.sql**, ejecuta un script.

**SPOOL /.../archivo.sql**, envía la salida de **sqlplus** a un archivo.

**SPOOL OFF**, cierra el archivo de bitácora.

# DBMS Oracle, 'Cadena' de conexión

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
**Oracle**  
SQL  
SQL-DDL  
SQL-DML

Incluir en: '**TNSNAMES.ora**'  
(configuración en Windows).

**A una BD local:** (Versión 8i o sup.)

```
LOCAL =  
  (DESCRIPTION =  
    (ADDRESS = (PROTOCOL = TCP)  
      (HOST = localhost)(PORT = 1521))  
    (CONNECT_DATA =  
      (SERVICE_NAME = BD)  
    ) )
```



# DBMS Oracle, 'Cadena' de conexión ...

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
**Oracle**  
SQL  
SQL-DDL  
SQL-DML

Incluir en: '**TNSNAMES.ora**'  
(configuración en Windows).

**A una BD remota:**  
(con protocolo TCP/IP)

```
REMOTA =  
  (DESCRIPTION =  
    (ADDRESS_LIST =  
      (ADDRESS = (PROTOCOL = TCP)  
        (HOST = 192.168.30.2)(PORT = 1521))  
    )  
    (CONNECT_DATA =  
      (SID = BD)  
    ) )
```

# Contenido

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle

**SQL**  
SQL-DDL  
SQL-DML

- 1 Antecedentes
- 2 Sistema Manejador de Bases de Datos
- 3 **Lenguaje Estructurado de Consulta - SQL**  
Lenguaje de Definición de Datos (SQL-DDL)  
Lenguaje de Manipulación de Datos (SQL-DML)

# Lenguaje Estructurado de Consulta (SQL)

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
**SQL**  
SQL-DDL  
SQL-DML

Un lenguaje de Bases de Datos que permite:

Crear una Bases de Datos y sus relaciones

Ejecutar consultas simples y complejas

Insertar, actualizar o borrar datos en las relaciones.

Se especifica *¿qué? información se quiere*, en lugar de  
*¿Cómo se obtiene?*.

## Componentes:

Lenguaje de Definición de Datos (DDL)

Lenguaje de Manipulación de Datos (DML)

# Lenguaje Estructurado de Consulta (SQL)

## Historia

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
**SQL**  
SQL-DDL  
SQL-DML

**1974**, D. Chamberlin introduce el lenguaje llamado *Structured English Query Language* - *SEQUEL*.

**1976**, Sale una versión revisada, SEQUEL/2, cuyo nombre se cambia a **SQL** por razones legales.

**1970**, IBM produce un prototipo de SMBD, llamado *System R*, que utiliza SQL.

**Finales 70's** Oracle aparece, tal vez, como el primer SMBDR comercial basado en SQL.

**1987**, ANSI e ISO publican el estándar inicial para SQL.

# Lenguaje Estructurado de Consulta (SQL)

## Historia

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle

SQL  
SQL-DDL  
SQL-DML

- 1989**, ANSI publica el apéndice *integrity Enhancement Feature*.
- 1992**, Ocurre la primera revisión mayor al estándar ISO, referida como SQL2 o SQL/92.
- 1999**, Se libera SQL:1999 con soporte para el manejo de datos **orientados a objetos**.
- 2003** SQL:2003 versión actual.

## Uso de la notación BNF (Backus-Naur Form)

**MAYÚSCULAS**, representan las palabras reservadas.

**minúsculas**, representan palabras definidas por el usuario.

| indica una elección entre diferentes alternativas.

{ } se utilizan para indicar elementos requeridos.

[ ] se utilizan para indicar elementos opcionales.

... indican repetición de elementos (0 o más).

```
SELECT      [DISTINCT | ALL]
             { * | [columnExpression [AS newName]] [ , ... ] }
FROM         TableName [alias] [ , ... ]
[WHERE      condition]
[GROUP BY   columnList] [HAVING condition]
[ORDER BY   columnList]
```

# Lenguaje Estructurado de Consulta (SQL)

## Literales

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle

**SQL**  
SQL-DDL  
SQL-DML

Son las constantes usadas en las sentencias SQL.

**Alfanuméricas**, deben ser encerradas entre comillas simples (*e.g.* 'Toluca').

**Numéricas**, no se encierran entre comillas (*e.g.* 12583).

# Lenguaje de Definición de Datos

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

Permite **crear** y **destruir** objetos de la base de datos tales como: esquemas, tablas, vistas, dominios, etc.

## Principales sentencias:

**CREATE / DROP SCHEMA**

**CREATE / ALTER / DROP DOMAIN**

**CREATE / ALTER / DROP TABLE**

**CREATE / REPLACE / DROP VIEW**

**CREATE / DROP INDEX**



# Lenguaje de Definición de Datos (SQL-DDL)

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

## Tipos de Datos ISO - SQL

Data type	Declarations		
boolean	BOOLEAN		
character	CHAR	VARCHAR	
bit	BIT	BIT VARYING	
exact numeric	NUMERIC	DECIMAL	INTEGER
approximate numeric	FLOAT	REAL	DOUBLE PRECISION
datetime	DATE	TIME	TIMESTAMP
interval	INTERVAL		
large objects	CHARACTER LARGE OBJECT		BINARY LARGE OBJECT

# Creación de una Base de Datos ...

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

## Esquema

Es una colección nominada de objetos de la base de datos que están relacionados entre sí (e.g tablas, vistas, dominios, constraints, ...). Todos los objetos de un **esquema** tienen el mismo propietario.

```
CREATE SCHEMA [Name | AUTHORIZATION CreatorId ]
```

```
DROP SCHEMA Name [RESTRICT | CASCADE ]
```

```
CREATE SCHEMA test AUTHORIZATION smith;
```

# Lenguaje de Definición de Datos (SQL-DDL)

## Restricciones de integridad

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle

SQL  
**SQL-DDL**  
SQL-DML

Integridad de entidades,  
Integridad referencial,  
Datos requeridos,  
Restricciones de dominio  
Restricciones generales,

# Restricciones de integridad ...

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

## Integridad de entidades

La llave primaria (PK) de una tabla debe contener un valor **ÚNICO** y **NO-NULO** por cada tupla.

El estándar **ISO** soporta la cláusula **PRIMARY KEY** en las sentencias **CREATE** y **ALTER TABLE**:

```
PRIMARY KEY( staffNo )  
PRIMARY KEY( clientNo , propertyNo )
```

Sólo puede existir una cláusula **PRIMARY KEY** por tabla. Puede garantizarse la unicidad para la llaves alternas usando **UNIQUE**.

```
UNIQUE( telNo )
```

# Restricciones de integridad ...

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

## Integridad referencial

La llave foránea (FK) es una columna o un conjunto de columnas que ligán a un registro de una tabla (**hija**) conteniendo una FK, hacia un renglón de otra tabla (**padre**) que contiene la PK correspondiente.

La integridad referencial significa que, si FK contiene un valor, dicho valor debe referir a un registro existente en la tabla padre.

El estándar ISO soporta la definición de FK's mediante la sentencia FOREIGN KEY en las sentencias CREATE y ALTER TABLE.

```
FOREIGN KEY (branchNo) REFERENCES branch (branchNo);
```

# Restricciones de integridad ...

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

## Integridad referencial

Cualquier INSERT / UPDATE que intente crear/modificar un valor FK en una tabla hija, sin que exista una PK en la tala padre, es rechazada

En caso de DELETE / UPDATE sobre la PK de una tabla padre se pueden efectuar las siguientes operaciones:

CASCADE  
SET NULL  
SET DEFAULT  
NO ACTION

```
FOREIGN KEY (staffNo) REFERENCES Staff ON DELETE SET NULL;
```

```
FOREIGN KEY (ownerNo) REFERENCES Owner ON UPDATE CASCADE;
```

# Lenguaje de Definición de Datos (SQL-DDL)

## Restricciones de integridad ...

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

### Datos requeridos

```
puesto VARCHAR(10) NOT NULL;
```

### Restricciones de dominio

#### CHECK

```
sex CHAR NOT NULL  
CHECK(sex IN ('M', 'F'))
```

#### CREATE DOMAIN

```
CREATE DOMAIN SexType AS CHAR  
CHECK(VALUE IN ('M', 'F'));  
  
sex SexType NOT NULL;
```

# Lenguaje de Definición de Datos (SQL-DDL)

## Restricciones de integridad ...

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

## Restricciones de dominio ...

### Búsqueda de una condición

```
CREATE DOMAIN CarreraNo AS CHAR(4)
CHECK (VALUE IN (SELECT id
                  FROM Carreras));
```

### Los dominios pueden ser removidos CHECK

```
DROP DOMAIN DomainName
[RESTRICT | CASCADE]
```



# Lenguaje de Definición de Datos (SQL-DDL)

## Restricciones de integridad ...

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

## Restricciones de generales ...

### ASSERTION

```
CREATE ASSERTION AssertionName  
CHECK (searchCondition)
```

```
CREATE ASSERTION StaffNotHandlingTooMuch  
CHECK (NOT EXISTS (SELECT staffNo  
FROM PropertyForRent  
GROUP BY staffNo  
HAVING COUNT(*) > 100))
```

# Restricciones de integridad ...

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

## Restricciones de generales ...

Son reglas adicionales especificadas por los usuarios o los administradores de la base de datos para definir o restringir algunos aspectos relacionados con la empresa.

```
CREATE or REPLACE TRIGGER alumnos_BI  
BEFORE INSERT ON Alumnos FOR EACH ROW  
DECLARE  
    ...  
BEGIN  
    — Validar la fecha de nacimiento  
    ...  
EXCEPTION  
    WHEN ...  
    — exception handling  
END;
```

# CREATE TABLE ...

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

Crea una tabla con una o más columnas de un **tipo de datos** especificado.

## Sintaxis:

```
CREATE TABLE TableName
{ (colName dataType [NOT NULL] [UNIQUE]
  [DEFAULT defaultOption]
  [CHECK searchCondition] [ , ... ]
  [PRIMARY KEY (listOfColumns),]
  [{UNIQUE (listOfColumns),} [ , ... ],]
  [{FOREIGN KEY (listOfFKColumns)
    REFERENCES ParentTableName [(listOfCKColumns)],
    [ON UPDATE referentialAction]
    [ON DELETE referentialAction ] [ , ... ]
  } [CHECK (searchCondition)] [ , ... ] ) }
```

## CREATE TABLE ...

### Plan

- Antecedentes
- Introducción
- Modelo Relacional
- Leng. Formales

- DBMS
  - Selección DBMS
  - Oracle
- SQL
  - SQL-DDL**
  - SQL-DML

### Creación de Dominios:

```
CREATE DOMAIN OwnerNumber AS VARCHAR(5)
    CHECK (VALUE IN (SELECT ownerNo FROM PrivateOwner));
CREATE DOMAIN StaffNumber AS VARCHAR(5)
    CHECK (VALUE IN (SELECT staffNo FROM Staff));
CREATE DOMAIN PNumber AS VARCHAR(5);
CREATE DOMAIN PRooms AS SMALLINT;
    CHECK(VALUE BETWEEN 1 AND 15);
CREATE DOMAIN PRent AS DECIMAL(6,2)
    CHECK(VALUE BETWEEN 0 AND 9999.99);
```

# CREATE TABLE ...

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

## Creación de Tabla:

```
CREATE TABLE PropertyForRent (
  propertyNo      PNumber      NOT NULL,
  rooms           PRooms        NOT NULL DEFAULT 4,
  rent            PRent          NOT NULL DEFAULT 600,
  ownerNo         OwnerNumber   NOT NULL,
  staffNo         StaffNumber
                  CONSTRAINT StaffNotHandlingTooMuch,
  branchNo        BranchNumber NOT NULL,
  PRIMARY KEY (propertyNo),
  FOREIGN KEY (staffNo) REFERENCES Staff
  ON DELETE SET NULL ON UPDATE CASCADE ...);
```

## ALTER TABLE ...

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL

**SQL-DDL**  
SQL-DML

**ADD COLUMN** Agrega una nueva columna

**DROP COLUMN** Elimina una columna

**ADD CONSTRAINT** Agrega una restricción

**SET** Establece un valor por defecto en una columna

## ALTER TABLE ...

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

### Modificación de una Tabla:

```
ALTER TABLE Staff ALTER position DROP DEFAULT;
```

```
ALTER TABLE Staff ALTER sex SET DEFAULT 'F';
```

```
ALTER TABLE PropertyForRent  
DROP CONSTRAINT StaffNotHandlingTooMuch;
```

```
ALTER TABLE Client  
ADD prefNoRooms PRooms;
```

## DROP TABLE ...

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

### Sintáxis:

```
DROP TABLE TableName [RESTRICT | CASCADE]
```

*e.g.*

```
DROP TABLE PropertyForRent ;
```



# INDEX

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

Es una estructura que permite acelerar el acceso a las filas de una tabla, basándose en los valores de una o más columnas.

Pueden mejorar la velocidad de las consultas, pero una carga adicional de trabajo para el SDBD cada que se actualizan las tablas subyacentes.

```
CREATE [UNIQUE] INDEX indexName  
      ON tableName (columnName [ASC | DESC][...])
```

*e.g.*

```
CREATE UNIQUE INDEX StaffNo_idx ON Staff(staffNo);  
  
CREATE INDEX rent_idx ON PropertyForRent(city , rent);  
  
DROP INDEX rent_idx;
```

# CREATE VIEW

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

## Propósito de las Vistas:

Proporcionan un mecanismo potente y flexible de seguridad al ocultar partes de la base de datos a ciertos usuarios.

Permite a los usuarios acceder a los datos en forma personalizada.

Pueden simplificar las operaciones complejas sobre las relaciones base.

```
CREATE VIEW ViewName [ (newColumnName [ ,...]) ]  
  AS subselect  
  [WITH [CASCADED | LOCAL] CHECK OPTION]
```

# CREATE VIEW

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

**Vista Horizontal:** Crear una vista de manera que el gerente de la sucursal **B003** pueda ver sólo los detalles de los empleados de su sucursal.

```
CREATE VIEW Manager3Staff AS  
  SELECT *  
  FROM Staff  
  WHERE branchNo = 'B003';
```

## CREATE VIEW

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

**Vista Vertical:** Crear una vista de los empleados de la sucursal **B003**, excluyendo sus **salarios**.

```
CREATE VIEW Manager3Staff AS
  SELECT staffNo , fName, lName, position , sex
  FROM   Staff
  WHERE  branchNo = 'B003' ;
```

# CREATE VIEW

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

**Vistas Agrupadas y Combinadas:** Crear una vista de los empleados que gestionan inmuebles para alquilar, en los que se incluya el **número de la sucursal** en la que trabaja el empleado, el **número del empleado** y el **número de inmuebles que gestiona**.

```
CREATE VIEW StaffPropCnt (branchNo , staffNo , cnt) AS
SELECT s.branchNo , s.staffNo , COUNT(*)
FROM Staff s , PropertyForRent p
WHERE s.staffNo = p.staffNo
GROUP BY s.branchNo , s.staffNo ;
```

# CREATE VIEW ... WITH CHECK OPTION

**Plan**

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

```
CREATE VIEW Manager3Staff AS  
SELECT *  
FROM Staff  
WHERE branchNo = 'B003'  
WITH CHECK OPTION;
```

No permite actualizar el número de sucursal de  
'B003' a 'B002'

No permite insertar en la vista una sucursal  
diferente a 'B003'

# DROP VIEW

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

## Elimina una vista de la Base de Datos

```
DROP VIEW viewName [RESTRICT | CASCADE]
```

**CASCADE** Borra todos los objetos dependientes relacionados con la vista.

**RESTRICT** Si existe un objeto, cuya existencia dependa de la vista a borrar, el comando será abortado.

```
DROP VIEW Manager3Staff;
```

## Transacciones

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

### Transacción

Es una unidad lógica de trabajo compuesta por una o más sentencias SQL-DML.

**COMMIT** Hace que la transacción termine con éxito y los cambios realizados en la base de datos serán permanentes. Después de COMMIT se iniciará una nueva transacción.

**ROLLBACK** Aborta la transacción, deshaciendo cualquier cambio efectuado en la base de datos.



# Privilegios, GRANT

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

## Privilegio

Acciones que se le permiten llevar a cabo a un usuario sobre una base de datos u objeto de la base de datos.

```
GRANT    { PrivilegeList | ALL PRIVILEGES }  
ON      ObjectName  
TO      { AuthorizationIdList | PUBLIC }  
[WITH GRANT OPTION]
```

SELECT

DELETE

INSERT [(columnName [,...])]

UPDATE [(columnName [,...])]

REFERENCES [(columnName [,...])]

USAGE

# GRANT ...

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

Otorgar al usuario identificado por **Manager** todos los privilegios sobre la tabla **Staff**.

```
GRANT ALL PRIVILEGES  
ON Staff  
TO Manager WITH GRANT OPTION;
```

Dar a los usuarios **Personal** y **Director** los privilegios de SELECT y UPDATE sobre la columna **salary** de **Staff**.

```
GRANT SELECT, UPDATE (salary)  
ON Staff TO Personnel, Director;
```

Conceder a todos los usuarios, el privilegio SELECT sobre la tabla **Branch**

```
GRANT SELECT ON Branch TO PUBLIC;
```

## REVOKE

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

Quita los privilegios concedidos con la instrucción GRANT.

```
REVOKE [GRANT OPTION FOR]
      { PrivilegeList | ALL PRIVILEGES }
ON ObjectName
FROM { AuthorizationIdList | PUBLIC }
     [RESTRICT | CASCADE]
```

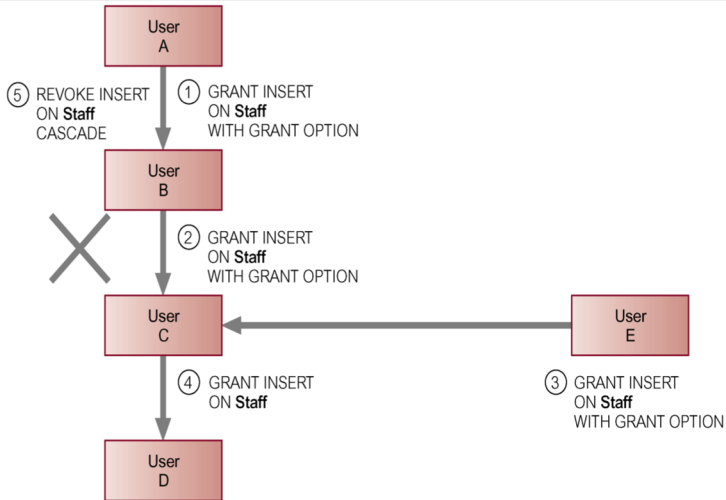
# Lenguaje de Definición de Datos (SQL-DDL)

## REVOKE...

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML



## REVOKE...

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

Revocar el privilegio **SELECT** a todos los usuarios sobre la tabla **Branch**.

```
REVOKE SELECT ON Branch FROM PUBLIC;
```

Revocar todos los privilegios dados al **Director** sobre la tabla **Staff**.

```
REVOKE ALL PRIVILEGES ON Staff FROM Director;
```

# SEGURIDAD

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

Crear un usuario identificado como **Manager3** que solo tenga acceso a la información de los empleados asignados a la sucursal **B003**, sin que se muestren los detalles de su salario.

## Conectarse como **System**

```
CREATE USER Manager3 IDENTIFIED BY Manager3;  
GRANT CONNECT, CREATE SESSION;
```

## Conectarse como **dreamHome**

```
GRANT SELECT ON Manager3Staff TO Manager3;
```

## Conectarse como **Manager3**

```
SELECT *  
FROM dreamHome.Manager3Staff;
```

# SYNONYM

**Plan**

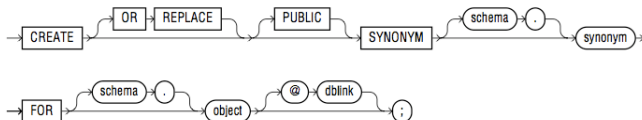
Antecedentes  
 Introducción  
 Modelo Relacional  
 Leng. Formales

DBMS  
 Selección DBMS  
 Oracle  
 SQL  
**SQL-DDL**  
 SQL-DML

Es un nombre alternativo para un objeto de la base de datos (e.g. tabla, vista, secuencia, ...)

## Sintaxis:

**create\_synonym::=**



## SYNONYM ...

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

### Conectarse como **Manager3**

```
CREATE SYNONYM Staff FOR dreamHome.Manager3Staff;  
  
SELECT *  
FROM Staff;
```

Los sinónimos permiten establecer la compatibilidad entre los esquemas y las aplicaciones.



# DATABASE LINK's ...

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
**SQL-DDL**  
SQL-DML

Conectarse de forma remota a una BD usando una 'Cadena de conexión'.

```
CREATE DATABASE LINK bd_remota
CONNECT TO user IDENTIFIED BY pass USING 'remota';

SELECT *
FROM Staff@bd_remota;
```

# Lenguajes de Manipulación de Datos ...

# SQL

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

**SELECT**, **consulta** información a partir de una o más tablas.

**INSERT**, **inserta** nuevas filas de datos a una tabla.

**UPDATE**, **modifica** los datos existentes en una tabla.

**DELETE**, **elimina** filas de datos de una tabla.

# SQL SELECT

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

```
SELECT      [DISTINCT | ALL]
            { * | [columnExpression [AS newName]] [, ...] }
FROM        TableName [alias] [, ...]
[WHERE      condition]
[GROUP BY  columnList] [HAVING condition]
[ORDER BY  columnList]
```

**SELECT** Especifica las columnas que aparecerán en la salida.

**FROM** Especifica la(s) tabla(s) utilizadas.

**WHERE** Filtra los renglones.

**GROUP BY** Forma grupos de renglones que tienen el mismo valor de columna(s).

**HAVING** Filtra grupos sujetos a una condición.

**ORDER BY** Especifica el orden de salida.

## Todas las columnas, todos los renglones.

Listar los detalles completos de todos los empleados.

```
SELECT  staffNo , fName, lName, address ,  
        position , sex, DOB, salary , branchNo  
FROM    Staff;
```

## Método abreviado:

```
SELECT  *  
FROM    Staff;
```

## Especificar las columnas y todos los renglones.

Lista de salarios de todos los empleados, mostrando sólo: **número de personal, nombre y apellidos y salario.**

```
SELECT staffNo , fName, lName, salary  
FROM Staff;
```

## Uso de DISTINCT.

Listar el identificador de todas las propiedades en renta que han sido visitadas.

```
SELECT propertyNo  
FROM Viewing;
```

## Eliminando duplicados

```
SELECT DISTINCT propertyNo  
FROM Viewing;
```

## Campos calculados.

Listado de los **salarios mensuales** de todos los empleados, mostrando: **número de empleado, nombre y apellidos y salario mensual.**

```
SELECT staffNo , fName, lName, salary/12  
FROM Staff;
```

## Asignando nombre a las columnas con la cláusula AS:

```
SELECT staffNo , fName, lName, salary/12 AS monthlySalary  
FROM Staff;
```

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

## Búsqueda con una condición. (WHERE)

Listar a todos los empleados con un salario mayor a 10,000.

```
SELECT staffNo, fName, lName, position, salary  
FROM Staff  
WHERE salary > 10000;
```



## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

## Búsquedas con condiciones compuestas. (AND, OR, NOT)

Listar las direcciones de todas las sucursales en  
**London o Glasgow.**

```
SELECT *  
FROM Branch  
WHERE city = 'London' OR city = 'Glasgow';
```

## Búsqueda dentro de un rango. (BETWEEN)

Listar a todo el personal cuyo salario esté entre 20,000 y 30,000.

```
SELECT staffNo , fName, lName, position , salary  
FROM Staff  
WHERE salary BETWEEN 20000 AND 30000;
```

**BETWEEN:** incluye los límites del rango.

O bien a través de rangos de valores.

```
SELECT staffNo, fName, lName, position, salary  
FROM Staff  
WHERE salary >= 20000 AND salary <= 30000;
```

Una versión *negada* con NOT BETWEEN

## Correspondencia con un conjunto de datos. (IN)

Listado de todos los **gerentes** y **supervisores**.

```
SELECT staffNo , fName, lName, position
FROM Staff
WHERE position IN ( 'Manager', 'Supervisor' );
```

O bien:

```
SELECT staffNo , fName, lName, position
FROM Staff
WHERE position='Manager' OR position='Supervisor' ;
```

## Búsqueda de *patrones*. (LIKE)

Encontrar a los propietarios con la palabra *Glasgow* incluida en su dirección.

```
SELECT ownerNo, fName, lName, address, telNo  
FROM PrivateOwner  
WHERE address LIKE ' %Glasgow % ;
```

% representa una secuencia de cero o más caracteres.

## Búsqueda de contenidos NULOS. (IS NULL)

Listar todos los registros de visitas a la propiedad  
**PG4** que **NO TENGAN** comentarios registrados.

```
SELECT clientNo , viewDate  
FROM Viewing  
WHERE propertyNo = 'PG4' AND comment IS NULL;
```

Listar todos los registros de visitas a la propiedad  
**PG4** que **TENGAN** comentarios registrados.

```
SELECT clientNo , viewDate  
FROM Viewing  
WHERE propertyNo = 'PG4' AND comment IS NOT NULL;
```

## Ordenamientos. (ORDER BY)

Listar los salarios de todos los empleados, ordenados en forma descendente.

```
SELECT staffNo, fName, lName, salary  
FROM Staff  
ORDER BY salary DESC;
```

Listado de propiedades ordenadas por **tipo** y **precio de renta**.

```
SELECT propertyNo, type, rooms, rent  
FROM PropertyForRent  
ORDER BY type, rent DESC;
```

## Funciones de Agregación.<sup>6</sup>

**COUNT** Regresa el número de valores en una columna especificada.

**SUM** Retorna la suma de valores en una columna especificada.

**AVG** Retorna el promedio de valores en una columna especificada.

**MIN** Retorna el valor más pequeño en una columna especificada.

**MAX** Retorna el valor más grande en una columna especificada.

---

<sup>6</sup>Definidas por el estándar ISO



## Funciones de Agregación.

Cada función opera sobre una sola columna.

**COUNT**, **MIN** y **MAX** se aplican en columnas numéricas y no-numéricas, mientras que **SUM** y **AVG** pueden ser usadas en campos numéricos.

Excepto para **COUNT(\*)**, cada función primero elimina los **NULOS** y opera sobre los valores **NO-NULOS** restantes.

**COUNT(\*)** cuenta todos los registros de la tabla, sin importar que existan valores **NULOS** o **DUPLICADOS**.

**DISTINCT** no tiene efecto sobre **MIN/MAX**, pero sí lo puede tener para **SUM/AVG**.

## Funciones de Agregación.

Las funciones de agregación sólo pueden ser usadas en la lista **SELECT** y en la cláusula **HAVING**.

Si la lista **SELECT** incluye una función de agregación y no existe la cláusula **GROUP BY**, entonces ésta es ilegal.

```
SELECT staffNo , COUNT(salary)
FROM Staff;
```

## Uso de COUNT(\*)

¿Cuántas propiedades cuestan más \$350 de renta?

```
SELECT COUNT(*) AS myCount
FROM PropertyForRent
WHERE rent > 350;
```

## Uso de COUNT(DISTINCT)

¿Cuántas propiedades diferentes fueron visitadas en Mayo '04?

```
SELECT COUNT(DISTINCT propertyNo) AS myCount
FROM Viewing
WHERE viewDate BETWEEN TO_DATE('1-May-04', 'dd-mon-yy')
AND TO_DATE('31-May-04', 'dd-mon-yy');
```

## Uso de COUNT y SUM

Encontrar el número de gerentes y la suma de sus salarios.

```
SELECT COUNT(staffNo) AS myCount,  
       SUM(salary) AS mySum  
FROM Staff  
WHERE position = 'Manager';
```

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

## Uso de MIN, MAX y AVG

Encontrar el mínimo, máximo y promedio de los salarios de los empleados.;

```
SELECT MIN(salary) AS myMin,  
       MAX(salary) AS myMax,  
       AVG(salary) AS myAvg  
FROM Staff;
```

## Uso de la cláusula GROUP BY para obtener sub-totales

Las cláusulas **SELECT** y **GROUP BY** deben estar integradas:

Cada registro en la lista **SELECT** debe ser **"mono-valuado por grupo"**, y la cláusula **SELECT** sólo puede contener:

- Nombres de columnas
- Funciones de agregación
- Constantes
- Expresiones involucrando la combinación de los elementos anteriores.

## Uso de la cláusula GROUP BY para obtener sub-totales ...

Todos los nombres de las columnas en la lista **SELECT** deben aparecer en la cláusula **GROUP BY** a menos que el nombre sea usado por una función de agregación.

Si se utiliza **WHERE** con un **GROUP BY**, se aplica primero la cláusula **WHERE** y luego se forman los grupos a partir de las filas restantes que satisfacen la condición de búsqueda.

## Uso de la cláusula GROUP BY para obtener sub-totales ...

Encontrar el número de empleados que trabajan en cada sucursal y la suma de sus salarios.

```
SELECT  branchNo,  
        COUNT(staffNo) AS myCount,  
        SUM(salary) AS mySum  
FROM    Staff  
GROUP BY branchNo  
ORDER BY branchNo;
```



## Uso de HAVING

La cláusula **HAVING** ha sido diseñada para ser utilizada con **GROUP BY** a fin de restringir los grupos que aparecen en los resultados.

Es similar a **WHERE**, sin embargo **WHERE** filtra **regiones**, mientras que **HAVING** filtra **grupos**.

Los nombres de las columnas en la cláusula **HAVING** deben aparecer la lista **GROUP BY**, o estar contenidos en una función de agregación.

## Uso de HAVING ...

Para cada sucursal que **tenga más de un empleado**, averiguar el número de empleados que trabajan en cada sucursal y la suma de sus salarios.

```
SELECT    branchNo,  
          COUNT(staffNo) AS myCount,  
          SUM(salary) AS mySum  
FROM      Staff  
GROUP BY  branchNo  
HAVING    COUNT(staffNo) > 1  
ORDER BY  branchNo;
```

## Sub-consultas

Algunas sentencias **SQL** pueden incluir una cláusula **SELECT** dentro de otro **SELECT**.

Puede realizarse una sub-consulta en las cláusulas **WHERE** y **HAVING** de un **SELECT** externo, en cuyo caso a ésta selección se le llama **sub-consulta** o **consulta anidada**.

Las **sub-consultas** pueden aparecer también en las sentencias **INSERT**, **UPDATE** y **DELETE**.

## Sub-consultas ...

Listar a todos los empleados de la sucursal ubicada en '163 Main St'

```
SELECT staffNo, fName, lName, position
FROM Staff
WHERE branchNo = (SELECT branchNo
                  FROM Branch
                  WHERE street LIKE ' %163_Main_St %');
```

## Subconsultas ...

La **sub-consulta** encuentra el número de sucursal ubicada en '163 Main St' → ('B003').

La consulta externa recupera entonces los detalles de los empleados que trabajan en esta sucursal, quedando la consulta externa como sigue:

```
SELECT staffNo , fName, lName, position
FROM   Staff
WHERE  branchNo = 'B003' ;
```

## Sub-consulta con una función de agregación

Listar a todos los empleados cuyo salario sea mayor que el salario promedio, indicando cual es la diferencia en cada caso.

```
SELECT staffNo, fName, lName, position,  
       salary - (SELECT AVG(salary) FROM Staff) As SalDiff  
FROM   Staff  
WHERE  salary >  
       (SELECT AVG(salary) FROM Staff);
```

No se puede escribir **'WHERE salary > AVG(salary)**  
Las subconsultas convierten la sentencia anterior en:

```
SELECT staffNo, fName, lName, position,  
       salary - 17000 As salDiff  
FROM   Staff  
WHERE  salary > 17000;
```

## Sub-consultas

No se puede usar **ORDER BY** en las sub-consultas.

La lista **SELECT** de una sub-consulta debe tener sólo un nombre de columna o expresión, excepto para las consultas que usen la cláusula **EXISTS**

Cuando se utilice una sub-consulta como un operando en una comparación, la sub-consulta debe aparecer en el lado derecho.

## Consultas anidadas

Listar las propiedades que son gestionadas por el personal que labora en '163 Main St.'

```
SELECT propertyNo, street, city, postcode, type, rooms, rent
FROM PropertyForRent
WHERE staffNo IN
      (SELECT staffNo
       FROM Staff
        WHERE branchNo =
              (SELECT branchNo
               FROM Branch
                WHERE street LIKE ' %163_Main_St % ));
```



## ANY, ALL

**ANY** y **ALL** pueden utilizarse en sub-consultas que generen un única columna de números.

Con **ALL**, la condición sólo será **verdadera** si ésta es satisfecha por todos los valores producidos por la sub-consulta.

Con **ANY**, la condición será **verdadera** si se ve satisfecha por cualquiera de los valores producidos por la sub-consulta.

Si la sub-consulta es **vacía**, **ALL** regresa **verdadero** y **ANY** regresa **falso**.

**SOME**, se puede usar en lugar de **ANY**.

## Uso de ANY/SOME

Encontrar a los empleados cuyo salario sea superior al salario de al menos uno de los empleados de la sucursal 'B003'.

```
SELECT staffNo, fName, lName, position, salary
FROM Staff
WHERE salary > SOME
      (SELECT salary
       FROM Staff
       WHERE branchNo = 'B003');
```

## Uso de ALL

Encontrar a los empleados cuyo salario sea mayor que el salario de cada uno de los miembros de la sucursal 'B003'.

```
SELECT staffNo, fName, lName, position, salary
FROM Staff
WHERE salary > ALL
      (SELECT salary
       FROM Staff
       WHERE branchNo = 'B003');
```

## Consultas multi-tablas

Si el resultado viene de más de una tabla, se debe usar una combinación (**join**).

En la cláusula **FROM** deben incluirse más de una tabla, separadas por comas (,). Se pueden usar **alias**.

Los **alias** ayudan a resolver ambigüedades en los nombres de las tablas.

Se debe usar la cláusula **WHERE** para especificar la combinación.

## Combinación simple, (equi-join)

Listar los nombres de todos los clientes que han visitado un inmueble, incluyendo los comentarios realizados sobre el mismo.

```
SELECT c.clientNo , fName, lName,  
        propertyNo , comment  
FROM   Client c, Viewing v  
WHERE  c.clientNo = v.clientNo ;
```

## Alternativas para construir una combinación (join)

En cada caso, la cláusula **FROM** reemplaza al **FROM - WHERE** originales. La primera produce una relación con dos nombres idénticos: **clientNo**.

```
FROM Client c JOIN Viewing v ON c.clientNo = v.clientNo
```

```
FROM Client JOIN Viewing USING clientNo
```

```
FROM Client NATURAL JOIN Viewing
```

## Ordenando una combinación

Para cada sucursal, indicar los números y los nombres de los empleados que gestionan inmuebles, así como los inmuebles que gestionan.

```
SELECT  s.branchNo, s.staffNo, fName, lName,
         propertyNo
FROM    Staff s, PropertyForRent p
WHERE    s.staffNo = p.staffNo
ORDER BY s.branchNo, s.staffNo, propertyNo;
```

## Combinación de tres tablas

Por cada sucursal, listar los números y nombres de los empleados que gestionan inmuebles, incluyendo la ciudad en que está ubicada la sucursal y los inmuebles que el empleado gestiona.

```
SELECT b.branchNo, b.city , s.staffNo , fName, lName, propertyNo  
FROM Branch b, Staff s, PropertyForRent p  
WHERE b.branchNo = s.branchNo AND  
        s.staffNo = p.staffNo  
ORDER BY b.branchNo, s.staffNo , propertyNo;
```

Alternativamente quedaría:

```
FROM (Branch b JOIN Staff s USING branchNo) AS bs  
      JOIN PropertyForRent p USING staffNo
```



## Múltiples columnas de agrupamiento

Determinar el número de inmuebles gestionados por cada empleado.

```
SELECT s.branchNo, s.staffNo, COUNT(*) AS myCount
FROM   Staff s, PropertyForRent p
WHERE  s.staffNo = p.staffNo
GROUP BY s.branchNo, s.staffNo
ORDER BY s.branchNo, s.staffNo;
```

## Combinaciones externas (LEFT-OUTER-JOIN)

Listar todas las sucursales y propiedades que están en la misma ciudad, mostrando los también las sucursales que no tengan propiedades en la misma ciudad.

```
SELECT b.*, p.*  
FROM Branch1 b LEFT JOIN  
PropertyForRent1 p ON b.bCity = p.pCity;
```

## Combinaciones externas (RIGHT-OUTER-JOIN)

```
SELECT b.*, p.*  
FROM   Branch1 b RIGHT JOIN  
       PropertyForRent1 p ON b.bCity = p.pCity;
```

## Combinaciones externas (FULL-OUTER-JOIN)

```
SELECT b.*, p.*  
FROM   Branch1 b FULL JOIN  
       PropertyForRent1 p ON b.bCity = p.pCity;
```

## EXISTS / NOT EXISTS

Sólo pueden ser usadas en las subconsultas

Producen un solo resultado (**cierto/falso**)

Verdadero sólo si existe al menos un registro en el resultado regresado por la subconsulta.

Falso si la subconsulta regresa **vacío**.

**NOT EXISTS** es la negación de **EXISTS**.

## EXISTS

Encontrar a todos los empleados que trabajan en la sucursal de 'London'

```
SELECT staffNo, fName, lName, position
FROM Staff s
WHERE EXISTS
  (SELECT *
   FROM Branch b
   WHERE s.branchNo = b.branchNo AND
        city = 'London');
```

## UNION, INTERSECT, DIFFERENCE

Son operaciones que se usan para combinar el resultado de dos o más consultas en una única tabla de resultados.

De la **UNIÓN** de dos tablas  $A$  y  $B$  resulta una tabla conteniendo todos los registros de  $A$  y  $B$ .

La **INTERSECCIÓN** es una tabla que contiene lo registros comunes a ambas tablas.

La **DIFERENCIA** contiene los registro que se encuentran en  $A$  y pero no en  $B$ .

## UNION, INTERSECT, DIFFERENCE

El formato para cada operador en todos los casos es:

```
op [ALL] [CORRESPONDING [BY {column1 [, ...]}]]
```

donde,

Si se especifica la palabra **CORRESPONDING BY**, la operación se aplica sobre la columnas designadas.  
Si se incluye la palabra **CORRESPONDING** pero NO la cláusula **BY**, la operación se realiza sobre la columnas que sean comunes a ambas tablas.  
Si se especifica **ALL**, el resultado puede incluir registros duplicados.

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

## UNION

Listar a todas las ciudades donde exista una sucursal u una propiedad en renta.

```
(SELECT city  
FROM Branch  
WHERE city IS NOT NULL)  
UNION  
(SELECT city  
FROM PropertyForRent  
WHERE city IS NOT NULL);
```



## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

## UNION

Listar a todas las ciudades donde exista una sucursal u una propiedad en renta.

```
(SELECT *  
  FROM Branch  
 WHERE city IS NOT NULL)  
UNION CORRESPONDING BY city  
(SELECT *  
  FROM PropertyForRent  
 WHERE city IS NOT NULL);
```

## Uso de INTERSECT

Listar todas las ciudades donde exista una sucursal  
**Y** al menos una propiedad en renta.

```
(SELECT city FROM Branch)  
INTERSECT  
(SELECT city FROM PropertyForRent);
```

```
(SELECT * FROM Branch)  
INTERSECT CORRESPONDING BY city  
(SELECT * FROM PropertyForRent);
```

## Uso de EXCEPT

Listar todas las ciudades donde exista una sucursal pero que no existan propiedades en renta.

```
(SELECT city FROM Branch)  
EXCEPT  
(SELECT city FROM PropertyForRent);
```

```
(SELECT * FROM Branch)  
EXCEPT CORRESPONDING BY city  
(SELECT * FROM PropertyForRent);
```

### Sintaxis:

```
INSERT INTO TableName [ (columnList) ]  
VALUES (dataValueList)
```

**columnList** es opcional; si se omite SQL asume la lista de todas las columnas de la tabla en el orden que fueron creadas (**CREATE TABLE**).

la lista **dataValueList** debe corresponder con la **columnList** de manera que:

- el número de elementos sea el mismo.

- debe existir una correspondencia en la posición de los elementos de las dos listas.

- los tipos de datos de cada elemento en

- dataValueList** debe ser compatible con el tipo de dato de su columna correspondiente.

## INSERT ... VALUES

Insertar un nuevo registro en la tabla **Staff**,  
suministrando datos para todas las columnas.

```
INSERT INTO Staff
VALUES ( 'SG16', 'Alan', 'Brown', 'Assistant', 'M',
        TO_DATE( '1957-05-25', 'yyyy-mm-dd' ), 8300, 'B003' );
```

## INSERT ... usando valores por omisión

Insertar un nuevo registro en la tabla **Staff**,  
suministrando solo para las columnas obligatorias.

```
INSERT INTO Staff (staffNo , fName, lName,  
                  position , salary , branchNo)  
VALUES ( 'SG44' , 'Anne' , 'Jones' ,  
        'Assistant' , 8100 , 'B003' );
```

```
INSERT INTO Staff  
VALUES ( 'SG44' , 'Anne' , 'Jones' , 'Assistant' , NULL,  
        NULL , 8100 , 'B003' );
```

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
**SQL-DML**

## INSERT ... SELECT

Los datos son insertados a partir de una o más tablas.

```
INSERT INTO TableName [ (columnList) ]  
SELECT ...
```

## INSERT ... SELECT

Considere la siguiente tabla llamada **StaffPropCount** que contiene los nombres de los empleados y el numero de propiedades que gestionan.

```
StaffPropCount(staffNo , fName, lName, propCnt)
```

### Insertar datos a partir de Staff y PropertyForRent:

```
INSERT INTO StaffPropCount
  (SELECT s.staffNo, fName, lName, COUNT(*)
   FROM Staff s, PropertyForRent p
   WHERE s.staffNo = p.staffNo
   GROUP BY s.staffNo, fName, lName)
 UNION
  (SELECT staffNo, fName, lName, 0
   FROM Staff
   WHERE staffNo NOT IN
     (SELECT DISTINCT staffNo
      FROM PropertyForRent));
```



## Sintaxis:

```
UPDATE TableName
      SET      columnName1 = dataValue1
            [, columnName2 = dataValue2 ...]
[WHERE searchCondition]
```

**TableName** puede ser el nombre de una tabla o de una vista que pueda ser **actualizada**.

La cláusula **SET** especifica los nombres de una o más columnas a ser actualizadas.

La cláusula **WHERE** es opcional:

Si se omite, las columnas nombradas son actualizadas para todos los registros de la tabla.

Si se especifica, sólo aquellos registros que satisfacen la condición, son actualizados.

Los nuevos datos deben ser compatibles con los tipos de datos de las columnas.

## Actualizando todos los registros

Otorgar un incremento del 3 % al salario de todos los empleados.

```
UPDATE Staff  
SET salary = salary*1.03;
```

Dar a todos los 'Managers' un incremento del 5 %;

```
UPDATE Staff  
SET salary = salary*1.05  
WHERE position = 'Manager';
```

## Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

## Actualizando múltiples columnas

Promover a 'David Ford' (staffNo='SG14') como 'Manager' y cambiar su salario a 18,000.

```
UPDATE Staff  
  SET position = 'Manager', salary = 18000  
 WHERE staffNo = 'SG14';
```

### Sintaxis:

```
DELETE FROM TableName  
[WHERE searchCondition];
```

**TableName** puede ser el nombre de una tabla o de una vista que pueda ser **actualizada**.

La cláusula **SET** especifica los nombres de una o más columnas a ser actualizadas.

La cláusula **WHERE** es opcional:

Si se omite, todos los registros de la tabla son borrados.

Si se especifica, sólo aquellos registros que satisfacen la condición son borrados.

### Plan

Antecedentes  
Introducción  
Modelo Relacional  
Leng. Formales

DBMS  
Selección DBMS  
Oracle  
SQL  
SQL-DDL  
SQL-DML

## Borrando renglones específicos

Borrar todas las visitas relacionadas con la propiedad 'PG4'

```
DELETE FROM Viewing  
      WHERE propertyNo = 'PG4';
```

Eliminar todos los registros de la tabla 'Viewing'

```
DELETE FROM Viewing;
```

# Contenido

Plan

Planeación BD

Planeación de Bases de

Datos

DreamHome

DreamHome

Referencias cruzadas de  
vistas de usuario

## 4 Planeación de Bases de Datos

### Planeación de Bases de Datos

## 5 DreamHome

# Planeación de Bases de Datos

Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

## Sistema de información - SI

Los recursos que permiten la recopilación, gestión, control y diseminación de la información en una determinada organización.

Las bases de datos son un componente fundamental de los SI y su desarrollo y utilización deben contemplarse desde la perspectiva de los requisitos globales de la organización.

# El ciclo de vida del desarrollo de sistemas de

Plan

Planeación BD  
Planeación de Bases de

Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

Planificación de la base de datos

Definición del sistema

Recopilación y análisis de requisitos

Diseño de la base de datos

Selección de SMBD

Diseño de la aplicación

Prototipo

Implementación

Conversión y carga de los datos

Pruebas

Mantenimiento operativo



# Planificación de la base de datos

Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

Las actividades de gestión que permiten llevar a cabo las distintas etapas del ciclo de vida del desarrollo de sistemas de base de datos de la forma más eficiente y efectiva posible.

**La planeación debe estar integrada con la estrategia global de sistemas de información de la organización:**

Identificando los planes y objetivos de la empresa  
Evaluando los sistemas de información actuales para determinar las fortalezas y debilidades existentes

Aprovechando las oportunidades tecnológicas de la información que puedan proporcionar una ventaja competitiva

# Planificación de la base de datos...

Plan

Planeación BD  
Planeación de Bases de

Datos

DreamHome

DreamHome

Referencias cruzadas de  
vistas de usuario

También se deben definir estándares que regulen como:

- Recopilar los datos

- Especificar formatos

- Que documentación hace falta y como procesarla

- Cómo proceder al diseño y la implementación

# Definición del sistema

Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

Describe el ámbito y los límites de la aplicación de las bases de datos y las principales vistas de los usuarios.

## Vistas de usuario

Define qué es lo que se requiere de la base de datos desde la perspectiva de

Un determinado rol de la organización (*e.g.* Gerente o Supervisor)

Un área de aplicación empresarial (*e.g.* marketing, recursos humanos, almacén)

# Vistas

Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

Un sistema de base de datos puede tener una o más vistas de usuario

La identificación de vistas ayuda a garantizar que no se omita a ninguno de los usuarios principales de la base de datos a la hora de desarrollar los requisitos del nuevo sistema.

Las vistas ayudan también a construir sistemas de bases de datos complejos, al permitir descomponer los requisitos en piezas más manejables.

# Recopilación y análisis de requisito

Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

Este proceso consiste en recopilar y analizar la información acerca de la parte de la organización a la que el sistema de base de datos tenga que dar soporte, y utilizar ésta información para identificar los requisitos relativos al nuevo sistema.

La información es obtenida a partir de las vistas de los usuarios principales, incluyendo:

- Una descripción de los datos utilizados o generados

- Los detalles acerca de cómo se deben utilizar o generar los datos

- Cualquier otro requisito que sean aplicables al nuevo sistema de base de datos

# Diseño de la base de datos

## Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

Es un proceso de creación de un diseño, que dé soporte a la misión y objetivos de la misión de la empresa, para el sistema de base de datos requerido.

## Técnicas

Top-down

Bottom-up

Inside-out

Mixed.

## **Propósitos principales:**

Ayudar a comprender el significado (*i.e.* semántica) de los datos;

Facilitar la comunicación de los requisitos de información.

## **Con un modelo de datos comprendemos:**

La perspectiva que cada usuario tiene de los datos;

La naturaleza de los propios datos,  
independientemente de su representación física;

La utilización de los datos en distintas vistas de usuario.

# Fases

Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

**Diseño conceptual** Construcción de un modelo de datos utilizado en una empresa, de forma independiente de todas las consideraciones físicas.

**Diseño lógico** Construcción de un modelo de datos utilizados en una empresa basándose en un modelo de datos específico, pero de forma independiente de un SMBD concreto y de cualquier otra consideración física.

**Diseño físico** Genera una descripción de la implementación de la base de datos en el almacenamiento secundario; describe las relaciones base, la organización de los archivos y los índices utilizados para conseguir un acceso eficiente a los datos. Describe también las medidas de seguridad y restricciones de integridad asociadas.



# Selección de SMBD

Plan

Planeación BD  
Planeación de Bases de

Datos  
DreamHome

DreamHome

Referencias cruzadas de  
vistas de usuario

Selección de un SMBD apropiado para soportar el sistema de base de datos.

## Características a evaluar:

- Definición de datos

  - Tipos de datos disponibles

  - Control de integridad

  - Mecanismos de vistas

  - Compresión de datos

  - ...

- Accesibilidad

  - Lenguajes de consulta

  - Multiusuario

  - Seguridad

- Utilidades

  - Optimización

  - Facilidades de carga/descarga

  - Soporte para la administración

# Selección de SMBD...

Plan

Planeación BD  
Planeación de Bases de

Datos  
DreamHome

DreamHome

Referencias cruzadas de  
vistas de usuario

## Características a evaluar:...

Otras características

Costo

Actualizaciones

Plataformas

Escalabilidad

Documentación

Soporte técnico

...

# Diseño de la aplicación

Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

Diseño de la interfaz de usuario y de los programas de aplicación que permitan utilizar y procesar la base de datos.

**Nota:** El diseño de la base de datos y el diseño de la aplicación son actividades que transcurren en paralelo dentro del ciclo de vida de desarrollo.

Diseño de transacciones

Diseño de la interfaz de usuario

# Diseño de transacciones

## Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

Una acción o serie de acciones llevadas a cabo por único usuario o programa de aplicación y que acceden al contenido de la base de datos o lo modifican.

**Propósitos** Definir y documentar:

- los datos utilizados en la transacción;
- las características funcionales de la transacción;
- la salida de la transacción;
- la frecuencia esperada de uso;
- la importancia para los usuarios;

# Diseño de la interfaz de usuario

## Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

## Recomendaciones:

Título significativo

Instrucciones comprensibles

Agrupamiento y secuenciamiento lógico de los campos

Diseño visual atractivo del formulario

Etiquetas familiares para los campos

Terminología y abreviaturas coherentes

Utilización coherente del color

Espacios y límites visibles para los campos

Mensajes de error

...

# Prototipo

Plan

Planeación BD  
Planeación de Bases de

Datos

DreamHome

DreamHome

Referencias cruzadas de  
vistas de usuario

Construcción de un modelo operativo del sistema de base de datos.

## Propósitos:

- Identificar las características del sistema que trabajan bien o que sean inadecuadas;

- Sugerir mejoras o nuevas características

- Clarificar los requerimientos de los usuarios;

- Evaluar la factibilidad de un diseño concreto.

Los prototipos deben ser **relativamente baratos y rápidos de construir**.

# Implementación

## Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome  
Referencias cruzadas de  
vistas de usuario

Realización física del diseño de la base de datos y del diseño de las aplicaciones.

Utilización de un DDL para crear el esquema de la base de datos (vacía).

Uso de un DDL para las vistas de usuario requeridas

Uso de un 3GL o 4GL para crear los programas de la aplicación. Se pueden incluir las transacciones a la base de datos implementadas con un DML (posiblemente integrado a un lenguaje de programación HOST).

# Conversión y carga de los datos

## Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

Transferencia de los datos existentes a la nueva base de datos y conversión de las aplicaciones existentes para que se ejecuten con la nueva base de datos.

**Nota:** Es requerida solo cuando el nuevo sistema de base de datos reemplaza a un sistema anterior.

Los SMBD normalmente incluyen una utilidad para cargar los datos existentes en la nueva base de datos.



# Pruebas

## Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

Consisten en operar el sistema de base de datos a fin de localizar posibles errores.

**Nota:** Demostrar que la base de datos y los programas de la aplicación trabajan de acuerdo a los requerimientos.

### A evaluar:

- Facilidad de aprendizaje

- Prestaciones

- Robustez

- Capacidad de recuperación

- Adaptabilidad

# Mantenimiento operativo

Plan

Planeación BD  
Planeación de Bases de

Datos  
DreamHome

DreamHome

Referencias cruzadas de  
vistas de usuario

Consiste en monitorear y mantener el sistema de base de datos después de la instalación.

Monitorear las prestaciones del sistema.

Administrador de la base de datos - DBA

Mantenimiento y actualización del sistema.

Desarrolladores.

# Contenido

## Plan

Planeación BD  
Planeación de Bases de

Datos  
**DreamHome**

DreamHome

Referencias cruzadas de  
vistas de usuario

## 4 Planeación de Bases de Datos

## 5 DreamHome

DreamHome

Planificación de la base de datos

Referencias cruzadas de vistas de usuario

# DreamHome [Connolly-06]

## Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome  
Referencias cruzadas de  
vistas de usuario

La primera sucursal de *DreamHome* fue abierta en 1992 en Glasgow, en el Reino Unido. Desde entonces, la empresa ha crecido en forma continua y ahora tiene diversas sucursales en la mayoría de las principales ciudades del Reino Unido.

La empresa es ahora tan grande que cada vez se requiere más personal administrativo para procesar la cantidad, cada vez mayor, de papeles e informes.

Además, la comunicación y compartición de la información entre sucursales, incluso dentro de la misma ciudad, es bastante deficiente.

La directora de la empresa, **Sally Mellweadows**, cree que se están cometiendo demasiados errores y que el éxito de la empresa puede ser muy efímero si no se hace algo para remediar la situación.

Sabe que una base de datos podría ayudar en parte a resolver el problema por lo cual ha solicitado que se desarrolle un sistema de base de datos para soportar las operaciones de *DreamHome*.

La directora ha proporcionado la siguiente breve descripción de la forma en la que *DreamHome* actualmente opera.

*DreamHome* está especializada en la gestión de inmuebles, actuando de intermediario entre los propietarios que quieren alquilar sus inmuebles, ya amueblados, y los clientes de *DreamHome* que necesitan alquilar esos inmuebles durante un periodo fijo de tiempo.

*DreamHome* cuenta actualmente con unos 2000 empleados que trabajan en cien sucursales. Cuando un nuevo empleado se incorpora a la empresa, se utiliza el formulario de registro de empleado de *DreamHome*.

# DreamHome ...

Plan

Planeación BD  
Planeación de Bases de

Datos  
DreamHome

DreamHome

Referencias cruzadas de  
vistas de usuario

DreamHome Staff Registration Form	
<b>Staff Number</b> <u>SG5</u>	<b>Branch Number</b> <u>B003</u>
<b>Full Name</b> <u>Susan Brand</u>	<b>Branch Address</b> <u>163 Main St, Glasgow</u>
<b>Sex</b> <u>F</u> <b>DOB</b> <u>3-Jun-40</u>	<b>Telephone Number(s)</b> <u>0141-339-2178 / 0141-339-4439</u>
<b>Position</b> <u>Manager</u>	
<b>Salary</b> <u>24000</u>	
<b>Enter details where applicable</b>	<b>Manager Start Date</b> <u>01-Jun-90</u>
<b>Supervisor Name</b> _____	<b>Manager Bonus</b> <u>2350</u>

Cada sucursal tiene asignados una serie de empleados, de los tipos adecuados, incluyendo un gerente, supervisores y ayudantes.

El gerente (Manager) es responsable de la operación diaria de la sucursal. Cada supervisor es responsable de controlar a un grupo de empleados que tiene la categoría de ayudantes(Assitants).



# DreamHome ...

Plan

Planeación BD  
Planeación de Bases de

Datos  
DreamHome

**DreamHome**

Referencias cruzadas de  
vistas de usuario

## DreamHome Staff Listing

Branch Number B003

Branch Address

163 Main St, Glasgow

Telephone Number(s)

G11 9QX

0141-339-2178 / 0141-339-4439

Staff Number	Name	Position
SG5	Susan Brand	Manager
SG14	David Ford	Supervisor
SG37	Ann Beech	Assistant
SG112	Annet Longhorn	Supervisor
SG126	Chris Lawrence	Assistant
SG132	Sofie Walters	Assistant

Cada sucursal ofrece una serie de inmuebles en alquiler. Para ofrecer un inmueble a través de *DreamHome*, el propietario normalmente contacta a la sucursal de *DreamHome* que esté situada más cerca del inmueble en alquiler.

El propietario suministra los detalles del inmueble y acuerda un precio de alquiler adecuado junto con el gerente de la sucursal.

# DreamHome ...

Plan

Planeación BD  
Planeación de Bases de

Datos  
DreamHome

DreamHome

Referencias cruzadas de  
vistas de usuario

<b>DreamHome</b> <b>Property Registration Form</b>	
<b>Property Number</b> <u>PG16</u>  <b>Type</b> <u>Flat</u> <b>Rooms</b> <u>4</u>  <b>Rent</b> <u>450</u>  <b>Address</b> <u>5 Novar Drive,</u> <u>Glasgow, G12 9AX</u> <u></u> <u></u>	<b>Owner Number</b> <u>C093</u> (If known)  <b>Person/Business Name</b> <u>Tony Shaw</u>  <b>Address</b> <u>12 Park Pl,</u> <u>Glasgow G4 0QR</u>  <b>Tel No</b> <u>0141-225-7025</u>
	<b>Enter details where applicable</b>  <b>Type of business</b> _____  <b>Contact Name</b> _____
<b>Managed by staff</b> <u>David Ford</u>	<b>Registered at branch</b> <u>163 Main St, Glasgow</u>

# Planificación de la base de datos

## Definición de la misión

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome  
Referencias cruzadas de  
vistas de usuario

## Entrevista a la directora de *DreamHome*

### *¿Cuál es el propósito de la empresa?*

Ofrecemos un amplio rango de inmuebles de alto standing para clientes registrados en nuestras sucursales repartidas por todo el Reino Unido.

Nuestra capacidad para ofrecer inmuebles de gran calidad depende de, por supuesto, los servicios que proporcionamos a los propietarios. Les proporcionamos un servicio muy profesional para garantizar un beneficio máximo a la hora de alquilar los inmuebles.

# Planificación de la base de datos

## Definición de la misión

### Plan

Planeación BD  
Planeación de Bases de

Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

### Entrevista a la directora de *DreamHome*...

#### *¿Por qué cree que necesita una base de datos?*

Para ser honestos, somos incapaces de adaptarnos a nuestro propio éxito. En los últimos años hemos abierto numerosas sucursales en la mayoría de las principales ciudades del Reino Unido y en cada sucursal ofrecemos ahora un mayor número de inmuebles a un conjunto también mayor de clientes.

# Planificación de la base de datos

## Definición de la misión

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome  
Referencias cruzadas de  
vistas de usuario

## Entrevista a la directora de *DreamHome*...

### ***¿Por qué cree que necesita una base de datos?***

Sin embargo, este éxito ha estado acompañado por una serie de problemas crecientes de gestión de los datos, lo que significa que el nivel de servicio que proporcionamos esté cayendo.

Asimismo, hay una falta de cooperación y de compartición de información entre las sucursales, lo que representa una evolución muy preocupante.

# Planificación de la base de datos

## Definición de la misión

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

### Entrevista a la directora de *DreamHome*...

***¿Cómo sabe que una base de datos resolverá su problema?***

Todo lo que sé es que estamos inundados de papeles. Necesitamos algo que acelere nuestra manera de trabajar automatizando buena parte de las tareas cotidianas que parecen alargarse indefinidamente en estos últimos meses.

Asimismo, quiero que las sucursales comiencen a cooperar unas con otras. Una base de datos ayudaría a conseguir este objetivo, no cree?

El propósito del sistema de base de datos de  
*DreamHome* es:

**Mantener los datos que se utilizan y generan**  
para soportar el negocio de alquiler de inmuebles  
con el que se presta servicio a nuestros clientes y a  
los propietarios, y

**Facilitar la cooperación y compartición** de  
información entre sucursales.



# Planificación de la base de datos

## Objetivos de la misión

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

### Estrategia:

Se plantean una serie de preguntas a la directora y a diversos empleados que tienen las categorías de gerente, supervisor y ayudante.

De ser necesario, las preguntas pueden ser adaptadas, dependiendo a quién se esté entrevistando.

# Planificación de la base de datos

## Objetivos de la misión

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

## Entrevista a la directora de *DreamHome*...

### *¿Qué papel juega en la compañía?*

Controlo las operaciones de la empresa para garantizar que continuemos proporcionando el mejor servicio posible de alquiler de inmuebles tanto a nuestros clientes como a los propietarios de los mismos.

# Planificación de la base de datos

## Objetivos de la misión

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

## Entrevista a la directora de *DreamHome*...

### ***¿Qué tipo de tareas realiza en un día normal?***

Monitorizo la forma en que cada uno de nuestros gerentes dirige su sucursal. Trato de garantizar que las sucursales colaboren y compartan información importante acerca de los inmuebles y de los clientes.

Normalmente trato de mantener un estrecho control de los gerentes, llamando a cada sucursal al menos una o dos veces al mes.

# Planificación de la base de datos

## Objetivos de la misión

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome  
Referencias cruzadas de  
vistas de usuario

## Entrevista a la directora de *DreamHome*...

### *¿Con qué tipo de datos trabaja?*

Necesito poder acceder a todo, o por lo menos a un resumen de los datos utilizados o generados por *DreamHome*. Eso incluye datos sobre el personal de las sucursales, sobre todos los inmuebles y sus propietarios, sobre todos los clientes y sobre todos los contratos.

También me gusta controlar el grado con el que cada sucursal anuncia los inmuebles en los periódicos.

# Planificación de la base de datos

## Objetivos de la misión

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

## Entrevista a la directora de *DreamHome*...

### *¿Qué tipo de informes utiliza?*

Necesito saber que es lo que sucede en cada sucursal y hay un montón de ellas. Dedico buena parte del día a analizar largos informes sobre todos los aspectos de *DreamHome*.

Necesito informes que sean fáciles de acceder y que me proporcionen una buena panorámica de lo que esta sucediendo en una determinada sucursal y en el conjunto de todas ellas.

# Planificación de la base de datos

## Objetivos de la misión

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

### Entrevista a la directora de *DreamHome*...

***¿Qué tipo de cosas necesita controlar?***

Como he dicho antes, necesito obtener una panorámica de todo, necesito poder ver una imagen global.

***¿Qué servicio proporciona su empresa a los clientes?***

Tratamos de proporcionar el mejor servicio posible de alquiler de inmuebles en el Reino Unido.

# Planificación de la base de datos

## Objetivos de la misión

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome  
Referencias cruzadas de  
vistas de usuario

### Entrevista el gerente de *DreamHome*...

***¿Cual es el cometido de su puesto de trabajo?***

...

***¿Qué tipo de tareas realiza en un día normal?***

...

***¿Con qué tipo de datos trabaja?***

Fundamentalmente trabajo con los datos sobre los inmuebles ofrecidos en mi sucursal y sobre los propietarios, clientes y contratos de alquiler.

# Planificación de la base de datos

## Objetivos de la misión

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

### Entrevista el gerente de *DreamHome*...

También necesito saber cuándo un inmueble está resultando difícil de alquilar, para poder disponer lo necesario para que se anuncie en los periódicos. Necesito controlar este aspecto del negocio porque los anuncios pueden resultar muy costosos.

También necesito acceder a los datos sobre el personal que trabaja en mi sucursal y el personal de otras sucursales locales. Esto se debe a que a veces tengo que contactar con otras sucursales para concertar reuniones de gestión o para pedir prestado personal a otras sucursales de forma temporal con el fin de cubrir bajas por enfermedad o durante los periodos de vacaciones.



# Planificación de la base de datos

## Objetivos de la misión

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome  
Referencias cruzadas de  
vistas de usuario

### Entrevista el gerente de *DreamHome*...

Este intercambio de empleados entre las sucursales locales es informal y no suele tener lugar muy a menudo, afortunadamente.

Además de los datos sobre el personal, resultaría bastante útil poder ver otros tipos de datos relativos a las otras sucursales, como por ejemplo los datos sobre los inmuebles, los propietarios, los clientes y los contratos, fundamentalmente para efectuar comparaciones.

# Planificación de la base de datos

## Objetivos de la misión

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome  
Referencias cruzadas de  
vistas de usuario

### Entrevista el gerente de *DreamHome*...

De hecho, creo que la directora confía en que este proyecto de base de datos ayude a promover la cooperación y la compartición de información entre sucursales. Sin embargo, algunos gerentes que conozco no van a estar muy de acuerdo con esto, por que piensan que las distintas sucursales debemos competir unas con otras.

Parte del problema es que un cierto porcentaje del salario de los gerentes está compuesto por un bono que está relacionado con el número de inmuebles que somos capaces de alquilar.

# Planificación de la base de datos

## Objetivos de la misión

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

## Entrevista el gerente de *DreamHome*...

***¿Qué tipo de informes utiliza?***

...

***¿Qué tipo de cosas necesita controlar?***

...

***¿Qué servicio proporciona su empresa a los clientes?***

...

# Planificación de la base de datos

## Objetivos de la misión

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

## Entrevista el supervisor de *DreamHome*...

...

***¿Con qué tipo de datos trabaja?***

Trabajo con datos relativos a los empleados de mi sucursal, inmuebles, propietarios, clientes, visitas a los inmuebles y contratos de alquiler.

...

# Planificación de la base de datos

## Objetivos de la misión

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

## Entrevista a el ayudante de *DreamHome*...

...

***¿Con qué tipo de datos trabaja?***

Trabajo con los datos sobre los inmuebles y las visitas a los inmuebles por parte de los clientes, y en ocasiones también sobre los contratos de alquiler.

...

# Planificación de la base de datos

## Objetivos de la misión

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

### Introducir, actualizar y borrar los datos sobre:

Sucursales,  
Personal,  
Inmuebles en alquiler,  
Propietarios de inmuebles,  
Clientes,  
Visitas a los inmuebles,  
Contratos de alquiler,  
Anuncios en los periódicos,

# Planificación de la base de datos

## Objetivos de la misión

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

### Realizar búsquedas de información sobre:

Sucursales,  
Personal,  
Inmuebles en alquiler,  
Propietarios de inmuebles,  
Clientes,  
Visitas a los inmuebles,  
Contratos de alquiler,  
Anuncios en los periódicos,

# Planificación de la base de datos

## Objetivos de la misión

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

### Controlar el estado de los:

Inmuebles en alquiler,

Clientes que desean alquilar un inmueble,

Contratos de alquiler



# Planificación de la base de datos

## Objetivos de la misión

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

### Generar informes sobre:

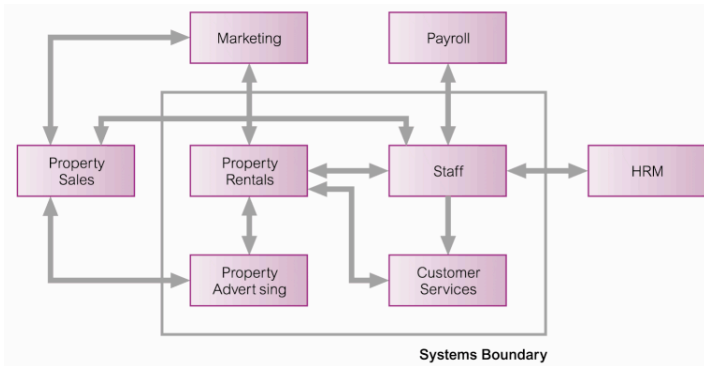
Sucursales,  
Personal,  
Inmuebles en alquiler,  
Propietarios de inmuebles,  
Clientes,  
Visitas a los inmuebles,  
Contratos de alquiler,  
Anuncios en los periódicos

# Límites del sistema de base de datos

## Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario



# Planificación de la base de datos

## Vistas de usuario

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome  
Referencias cruzadas de  
vistas de usuario

La mayoría de los datos acerca de las vistas de usuario fueron recopilados mediante entrevistas con la **directora** y con diversos miembros del personal que tenían las categorías de **gerente**, **supervisor** y **ayudante**.

# Planificación de la base de datos

## Vistas de usuario...

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

Data	Access Type	Director	Manager	Supervisor	Assistant
All Branches	Maintain				
	Query	X	X		
	Report	X	X		
Single Branch	Maintain		X		
	Query		X		
	Report		X		
All Staff	Maintain				
	Query	X	X		
	Report	X	X		
Branch Staff	Maintain		X		
	Query		X	X	
	Report		X	X	
All Property	Maintain				
	Query	X			
	Report	X	X		
Branch Property	Maintain		X	X	
	Query		X	X	X
	Report		X	X	X
All Owners	Maintain				
	Query	X			
	Report	X	X		
Branch Owners	Maintain		X	X	
	Query		X	X	X
	Report		X		
All Clients	Maintain				
	Query	X			
	Report	X	X		
Branch Clients	Maintain		X	X	
	Query		X	X	X
	Report		X		

# Recopilación y análisis de requisitos

## Plan

Planeación BD  
Planeación de Bases de

Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

## Entrevista el gerente de *DreamHome*...

***¿Qué tipo de datos necesita almacenar sobre los empleados?***

Los datos almacenados sobre un empleado son el nombre completo, su categoría, el sexo, la fecha de nacimiento y el salario.

# Recopilación y análisis de requisitos

Plan

Planeación BD  
Planeación de Bases deDatos  
DreamHome  
DreamHomeReferencias cruzadas de  
vistas de usuario

## Entrevista el gerente de *DreamHome*...

***¿Qué tipo de cosas hace con los datos sobre los empleados?***

Tengo que poder introducir los detalles de los nuevos empleados y borrar los detalles de aquellos que dejan la empresa. Necesito mantener actualizados los detalles sobre los empleados e imprimir informes que indiquen nombre completo, su categoría y su salario, para cada empleado de mi sucursal. Tengo que poder asignar personal a los supervisores.

Algunas veces, cuando necesito comunicarme con otras sucursales, necesito averiguar los nombres y números de teléfono de los gerentes de esas otras sucursales.

# Recopilación y análisis de requisitos

## Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

Recopilación de información de carácter más general sobre los requisitos del sistema.

***e.g.***

¿Qué transacciones se ejecutan de manera frecuente en la base de datos?

¿Qué transacciones son críticas para la operación de las organizaciones?

¿Cuándo se ejecutan las transacciones críticas?

¿Cuáles son los periodos de carga de trabajo baja, normal y alta para las transacciones críticas?

¿Qué tipo de seguridad se desea para el sistema de la base de datos?

# Recopilación y análisis de requisitos

## Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome  
Referencias cruzadas de  
vistas de usuario

Recopilación de información de carácter más general sobre los requisitos del sistema.

***e.g.***

¿Qué Hay algún dato altamente confidencial al que sólo debieran acceder ciertos empleados?

¿Qué datos históricos se quieren conservar?

¿Cuales son los requisitos de interconexión por red y acceso compartido para el sistema de base de datos?

¿Qué tipo de protección frente a fallos o pérdidas de datos debe tener el sistema de base de datos?



# Recopilación y análisis de requisitos

## Plan

Planeación BD  
Planeación de Bases deDatos  
DreamHome  
DreamHomeReferencias cruzadas de  
vistas de usuario

## Entrevista el gerente de *DreamHome*...

***¿Qué transacciones se ejecutan de manera frecuente en la base de datos?***

Frecuentemente recibimos solicitudes por teléfono o a través de clientes que llaman a la sucursal para buscar un tipo particular de inmueble en un área concreta de la ciudad y que tenga un alquiler no superior a una determinada cantidad.

También necesitamos información actualizada sobre los inmuebles y los clientes para poder generar informes que muestren los inmuebles actualmente disponibles para alquilar y los clientes que actualmente están buscando inmueble.

# Recopilación y análisis de requisitos

Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

## Entrevista a la gerente de *DreamHome*...

***¿Qué transacciones son críticas para la operación de las organizaciones?***

De nuevo, las transacciones críticas incluyen ser capaz de buscar inmuebles concretos e imprimir informes con listas actualizadas de inmuebles disponibles para alquilar.

Nuestros clientes se irían con la competencia si no pudiéramos proporcionar este servicio básico.

***¿Cuándo se ejecutan las transacciones críticas?***

Todos los días.

# Recopilación y análisis de requisitos

## Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

## Entrevista a la gerente de *DreamHome*...

***¿Cuáles son los periodos de carga de trabajo baja, normal y alta para las transacciones críticas?***

Abrimos seis días a la semana; En general, se tiende a estar más o menos tranquilos por la mañanas y a medida que avanza el día se tiene más y más trabajo.

Sin embargo, las horas más críticas cada día, en lo que se refiere al trato con los clientes, son entre las 12 y las 2 y entre las 5 y las 7.

# Recopilación y análisis de requisitos

## Entrevista a la directora de *DreamHome*...

***¿Qué tipo de seguridad se desea para el sistema de la base de datos?***

No creo que una base de datos que almacene información para una empresa de alquiler de inmuebles contenga datos muy confidenciales, pero no me gustaría que ninguno de nuestros competidores viera los datos sobre los inmuebles, propietarios, clientes y contratos de alquiler.

Los empleados solo deben ver los datos que sean necesarios para hacer su trabajo en la forma que resulte adecuada en cada momento. Por ejemplo, aunque es necesario que los supervisores y ayudantes vean los detalles de los clientes, los registros de clientes sólo deberían visualizarse de uno en uno y en forma de listado.

# Recopilación y análisis de requisitos

## Plan

Planeación BD  
Planeación de Bases de

Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

## Entrevista a la directora de *DreamHome*...

***¿Qué Hay algún dato altamente confidencial al que sólo debieran acceder ciertos empleados?***

Como ya he dicho antes, los empleados sólo deben ver los datos necesarios para su trabajo. Por ejemplo, aunque los supervisores necesitan poder ver los datos sobre los empleados, no deberían incluirse los detalles salariales.

# Recopilación y análisis de requisitos

## Plan

Planeación BD  
Planeación de Bases deDatos  
DreamHome  
DreamHomeReferencias cruzadas de  
vistas de usuario

## Entrevista a la directora de *DreamHome*...

### *¿Qué datos históricos se quieren conservar?*

Quiero conservar los detalles de los clientes y propietarios durante un par de años después del último contacto que hayan tenido con nosotros, para poder efectuar campañas de marketing por correo a fin de informarles sobre nuestras últimas ofertas y generalmente tratar de atraerles de nuevo a nuestra empresa.

También quiero mantener la información de contratos durante un par de años para poder averiguar qué tipos de inmuebles y áreas de la ciudad son las más populares para el mercado de alquiler de inmuebles.

# Recopilación y análisis de requisitos

## Plan

Planeación BD  
Planeación de Bases deDatos  
DreamHome  
DreamHomeReferencias cruzadas de  
vistas de usuario

## Entrevista a la directora de *DreamHome*...

***¿Cuales son los requisitos de interconexión por red y acceso compartido para el sistema de base de datos?***

Quiero que todas las sucursales estén conectadas con nuestra oficina principal en Glasgow, con el fin de que el personal pueda acceder al sistema desde cualquier lugar y en el momento en que lo necesite.

En la mayoría de las sucursales, cabe esperar que haya dos o tres empleados accediendo al sistema en cualquier momento concreto, pero recuerde que tenemos unas cien sucursales. La mayor parte del tiempo, los empleados, estarán simplemente accediendo a datos relativos a la sucursal local.

# Recopilación y análisis de requisitos

## Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

## Entrevista a la directora de *DreamHome*...

***¿Qué tipo de protección frente a fallos o pérdidas de datos debe tener el sistema de base de datos?***

La mejor que exista, por supuesto, Vamos a llevar el negocio utilizando una base de datos, por lo que si ésta se detiene, estamos perdidos.

Hablando en serio, creo que probablemente deberíamos hacer una copia de seguridad de los datos todas las tardes, después de cerrar la sucursal.



# Referencias cruzadas de vistas de usuario

## Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

## Referencias cruzadas de vistas de usuario

	Director	Manager	Supervisor	Assistant
branch	X	X		
staff	X	X	X	
property for rent	X	X	X	X
owner	X	X	X	X
client	X	X	X	X
property viewing			X	X
lease	X	X	X	X
newspaper	X	X		

# Especificaciones para la base de datos

## Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

## Tamaño inicial de la base de datos

Hay aproximadamente 2,000 empleados trabajando en más de 100 sucursales. Hay un promedio de 20 y un máximo de 40 empleados en cada sucursal.

Hay aproximadamente 100,000 inmuebles disponibles en todas las sucursales. Hay un promedio de 1,000 y un máximo de 3,000 inmuebles en cada sucursal.

Hay aproximadamente 60,000 propietarios de inmuebles. Hay una media de 600 y un máximo de 1,000 propietarios en cada sucursal.

Hay aproximadamente 100,000 clientes registrados en todas las sucursales. Hay un promedio de 1,000 y un máximo de 1,500 clientes registrados en cada sucursal.

# Especificaciones para la base de datos

## Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

## Tamaño inicial de la base de datos

Hay aproximadamente 4,000,000 de visitas registradas en todas las sucursales. Hay una media de 40,000 y un máximo de 100,000 visitas en cada sucursal.

Hay aproximadamente 400,000 contratos de alquiler entre todas las sucursales. Hay un promedio de 4,000 y un máximo de 10,000 contratos de alquiler en cada sucursal.

Hay aproximadamente 50,000 anuncios de periódico en 100 periódicos entre todas las sucursales.

# Especificaciones para la base de datos

**Plan**

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

## Tasa de crecimiento de la base de datos

Cada mes se añaden a la base de datos aproximadamente 500 nuevos inmuebles y 200 nuevos propietarios.

Cuando un inmueble deja de estar disponible para alquilar, el registro correspondiente se borra de la base de datos. Cada mes se borran aproximadamente 100 registros de inmuebles.

Si un propietario no proporciona ningún inmueble en alquiler a lo largo de un periodo de dos años, su registro se borra. Cada mes se borran aproximadamente 100 propietarios de inmuebles.

...

# Tipos de búsquedas de registros

Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

Búsqueda de la información relativa a una sucursal:  
aproximadamente 10 por día.

Búsqueda de la información relativa a un empleado  
de una sucursal: aproximadamente 20 por día.

Búsqueda de la información relativa a un cierto  
inmueble: aproximadamente 5,000 por día (de lunes  
a jueves), aproximadamente 10,000 por día (viernes  
y sábados). Las horas pico de carga de trabajo son  
entre 12:00 - 14:00 y 17:00 - 19:00 diariamente.

...

# Interconexión por red y acceso compartido

## Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

Todas las sucursales deben estar conectadas por red de manera que segura por una base de datos centralizada ubicada en la oficina principal de *DreamHome* en Glasgow.

El sistema debe permitir que al menos dos o tres personas accedan concurrentemente al sistema desde cada sucursal.

Es preciso tener en cuenta los requisitos de licencia para este número de accesos concurrentes.

## Planificación de la base de datos

# Prestaciones

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

Durante las horas de apertura, aunque no durante las horas pico, debe poder esperarse un tiempo de respuesta inferior a un segundo para todas las búsquedas de un único registro. Durante las horas pico, puede esperarse un periodo de respuesta inferior a 5 segundos por cada búsqueda.

Durante las horas de apertura, aunque no durante las horas pico, debe poder esperarse un tiempo de respuesta inferior a 5 segundos para todas las búsquedas de múltiples registros. Durante las horas pico, puede esperarse un periodo de respuesta inferior a 10 segundos para cada búsqueda múltiple de registros.

...

# Planificación de la base de datos

## Seguridad

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

La base de datos debe estar protegida mediante contraseñas.

A cada empleado deben asignárseles los privilegios apropiados de acceso a la base de datos que se correspondan con una vista de usuario concreta, la de Director, Gerente, Supervisor o Ayudante.

Cada empleado sólo debe poder ver los datos necesarios para hacer su tarea de una forma que resulte adecuada para lo que está haciendo.



# Copia de seguridad y recuperación

## Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

Debe hacerse una copia de seguridad de la base de datos diariamente a las doce de la noche.

# Planificación de la base de datos

## Cuestiones legales

### Plan

Planeación BD  
Planeación de Bases de  
Datos  
DreamHome  
DreamHome

Referencias cruzadas de  
vistas de usuario

Ya que la base de datos de *DreamHome* contiene datos sobre los empleados, clientes y propietarios de inmuebles, será preciso destinar e implementar todas las normas legales con las que haya que cumplir (e.g. protección a la vida privada, derechos sobre el acceso y modificación de la información almacenada en medios digitales).

# Contenido

Plan

**Casos**

Biblioteca

Referencias

## 6 Casos de Estudio

### Biblioteca

La biblioteca de una comunidad está formada por 5 centros de préstamo. Estos centros de préstamo disponen de computadoras personales interconectadas que permiten la gestión de los préstamos. Una entrevista con los bibliotecarios permite determinar los siguientes hechos.

Cualquier persona que se inscriba en la biblioteca, debe dejar un depósito. Dependiendo del depósito entregado, las personas tendrán el derecho de efectuar al mismo tiempo de 1 a 10 préstamos.

Los prestamos duran máximo 15 días.

Un libro es caracterizado por: un número único (identificador), título, editor, y su(s) autor(es).

Se desea poder obtener, por cada usuario registrado, los prestamos que él ha efectuado (nombre, número y título del libro, fecha de préstamo) en el transcurso de los últimos tres meses.

Todas las semanas se edita la lista de usuarios con prestamos pendientes: nombre y dirección de usuario, fecha de préstamo, número(s) y título(s) del (los) libros concerniente(s).

Se desea poder conocer por cada libro: su fecha de compra, su estado, y si está disponible en qué centro.

# Referencias

Plan

Casos  
Biblioteca  
Referencias

Thomas M. Connolly and Carolyn E. Begg.  
*Sistemas de Bases de Datos: Un Enfoque Práctico  
Para Diseño, Implementación y Gestión*, Pearson  
Education. 2006

Oracle  
*SQL- Reference*, Oracle Corporation. 2003

Oracle  
*PL/SQL- User Guide and Reference*, Oracle  
Corporation. 2003

Oracle  
*Administrator's Guide*, Oracle Corporation. 2003

Edgar F. Codd  
*A Relational Model of Data for Large Shared Data  
Banks*, In *Communication of the ACM*, volume 13,  
pages 377-387. 1970