



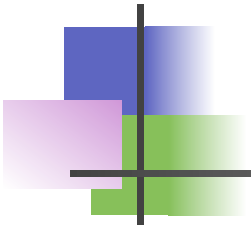
**Departamento de
Ciencias de la Computación y Tecnologías de Información
Universidad del Bío-Bío
Sede Chillán**

Bases de Datos

Lenguaje SQL

M^a Angélica Caro Gutiérrez

<http://www.face.ubiobio.cl/~mcaro/>
mcaro@ubiobio.cl

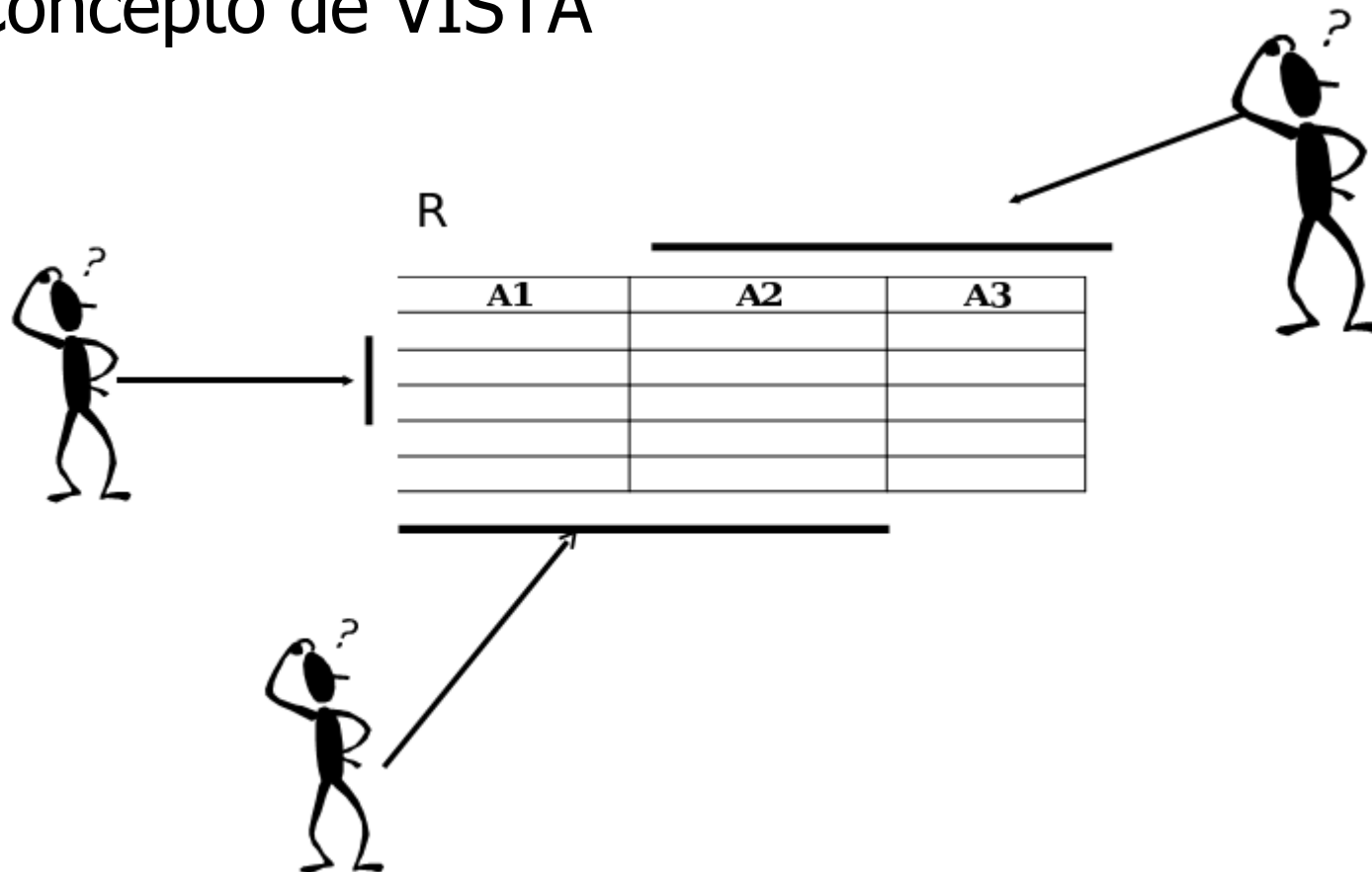


Lenguaje SQL

- Introducción
- Conceptos básicos
- Consultas básicas en SQL
- Consultas complejas en SQL
- ➔ ■ Vistas (tablas virtuales) en SQL

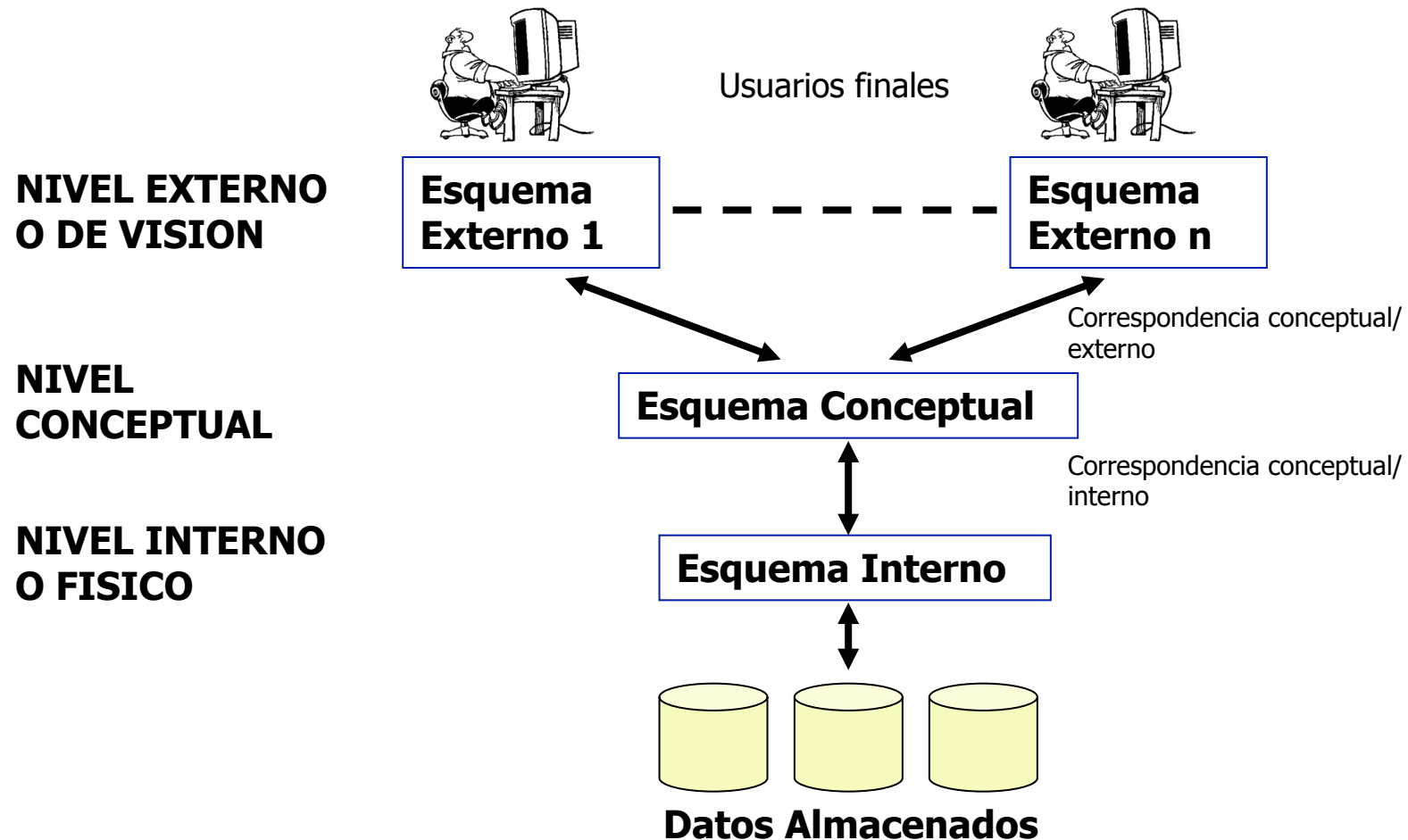
Vistas (Tablas virtuales) en SQL

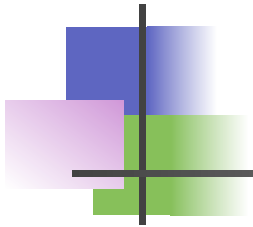
- Concepto de VISTA



Vistas (Tablas virtuales) en SQL

- Arquitectura y Niveles de abstracción

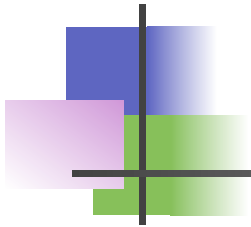




Vistas (Tablas virtuales) en SQL

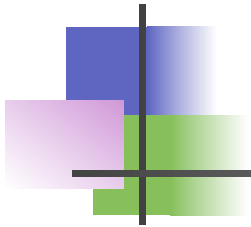
- Concepto de VISTA

- Una vista (VIEW) es una **tabla lógica** (no física) que se basa en otras tablas o vistas.
- Funciona **como una ventana** a través de la cual pueden visualizarse o modificarse datos de las tablas.
- Una vista **no contiene datos** en si misma.
- Las tablas sobre las que se define una vista se llaman tablas base.
- La vista se almacena como una sentencia SELECT nominada en el diccionario de datos.



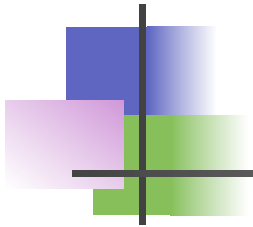
Vistas (Tablas virtuales) en SQL

- ¿Por qué utilizar vistas?
 - Conseguir un **acceso restringido** a la base de datos, ya que la vista puede mostrar sólo una porción específica.
 - Sustituir **consultas complejas** por varias consultas **más simples** que emplean vistas.
 - Por ejemplo, una selección sobre una vista definida sobre varias tablas será más fácil de realizar que si tuviéramos que utilizar la combinación (JOIN) de esas tablas para realizarla.
 - Proveer **independencia de datos**.
 - Proveer **distintas maneras de ver** los mismos datos, adaptados a cada usuario o aplicación.
 - Permitir el **acceso a grupos de usuarios** de acuerdo con unos criterios concretos.



Vistas (Tablas virtuales) en SQL

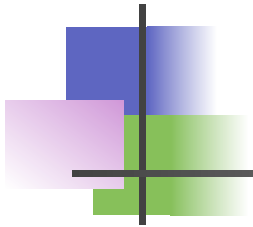
- Vistas Simples vs Complejas
 - En función de las operaciones de manipulación de datos (DML), es decir, INSERT, UPDATE y DELETE, que se pueden realizar a través de ellas, existen dos categorías de vistas:
 - **Simple:**
 - Extraen los datos de **una sola tabla**
 - No contienen funciones ni grupos de datos (GROUP BY)
 - Siempre pueden realizarse operaciones DML a través de ellas.
 - **Complejas:**
 - Extraen los datos de **múltiples tablas**
 - Contienen funciones o grupos de datos (GROUP BY)
 - Es frecuente que no permitan operaciones DML.



Vistas (Tablas virtuales) en SQL

- La cláusula CREATE VIEW permite la creación de vistas. La cláusula asigna un nombre a la vista y permite especificar la consulta que la define.
- Creando una Vista – sintaxis (i):

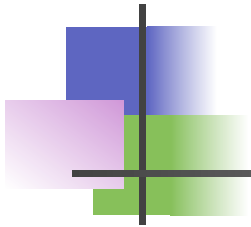
```
CREATE [OR REPLACE] [FORCE|NOFORCE] VIEW <vista>  
[( <alias>[, <alias>] ... )]  
AS <subconsulta>  
[WITH CHECK OPTION [CONSTRAINT <restricción>]]  
[WITH READ ONLY [CONSTRAINT <restricción>]];
```

Vistas (Tablas virtuales) en SQL

- Creando una Vista – sintaxis (ii):

OR REPLACE	Se utiliza por si la vista ya estuviera creada anteriormente. En ese caso, la sustituye por la nueva definición.
FORCE	Crea la vista sin comprobar si las tablas base existen.
NO FORCE	Crea la vista sólo si las tablas base de donde se extraen los datos existen realmente (es la opción por defecto).
<vista>	Es el nombre de la vista.
<alias>	Especifica alias para las expresiones/columnas seleccionadas por la subconsulta. El número de alias debe coincidir con el número de expresiones seleccionadas por la vista.



Vistas (Tablas virtuales) en SQL

- Creando una Vista – sintaxis (iii):

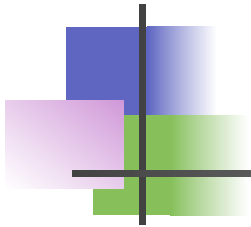
<subconsulta>	Es una sentencia SELECT completa. Se pueden emplear alias para las columnas en la lista que sigue al SELECT.
WITH CHECK OPTION	Especifica si la vista pueden actualizarse. El SGBD comprobará si la vista puede actualizarse.
<restricción>	Nombre asignado a la restricción CHECK OPTION o a la restricción READ ONLY.
WITH READ ONLY	Asegura que no podrán ejecutarse operaciones de DML a través de la vista. La vista sólo permite consultas.



Vistas (Tablas virtuales) en SQL

- Creando una Vista – Ejemplo:
 - Crear una vista que contiene los apellidos y nombre de los empleados del departamento “Ventas”. Asegurarse que a través de ella sólo pueden modificarse, eliminarse o añadirse los empleados de dicho departamento.

```
CREATE VIEW EmpDepVentas
AS SELECT apellidos, nombre
FROM Empleados
WHERE dep="Ventas"
WITH CHECK OPTION;
```



Vistas (Tablas virtuales) en SQL

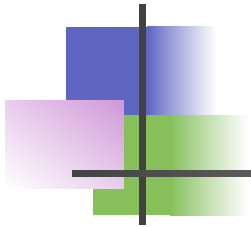
- Creando una Vista – Ejemplo:

- “Los empleados del departamento de ‘Investigación’ ”

```
CREATE VIEW INVESTIGACION AS
SELECT E.DNI, E.NOMBRE, E.APELLIDO1, E.SUELDO,E.DIRECCION
FROM (EMPLEADO E JOIN DEPARTAMENTO
      ON E.DNO = NUMERODPTO)
WHERE NOMBREDPTO = 'Investigacion'
ORDER BY E.APELLIDO1,E.NOMBRE;
```

- Ahora podemos consultar directamente la vista INVESTIGACION

```
SELECT * FROM INVESTIGACION;
```



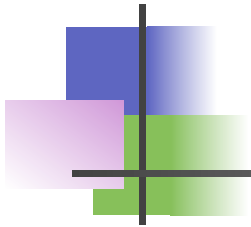
Vistas (Tablas virtuales) en SQL

- Creando una Vista – Ejemplo:
 - Por cada departamento queremos mantener un resumen con el total de empleados y el sueldo total

```
CREATE VIEW RESUMEN (NOMBRE_DEPTO, TOTAL_EMPLEADOS,  
TOTAL_SUELDO) AS  
SELECT NOMBREDPTO, COUNT(*), SUM(SUELDO)  
FROM (EMPLEADO JOIN DEPARTAMENTO ON DNO=NUMERODPTO)  
GROUP BY NOMBREDPTO  
ORDER BY NOMBREDPTO;
```

- Ahora

```
SELECT * FROM RESUMEN;
```



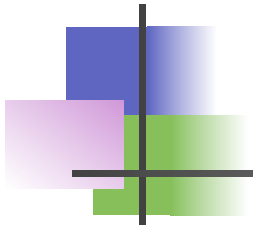
Vistas (Tablas virtuales) en SQL

■ Creando una Vista – Empleo de Alias

- Los alias se pueden definir dentro de la subconsulta o fuera de ella, justo a continuación del nombre la vista.
- Crear una vista que contiene los apellidos y nombre de los empleados del departamento "Ventas" renombrando dichas propiedades como "Last_name" y "First_name".
 - a)

```
CREATE VIEW EmpDepVentas
AS SELECT apellidos Last_name, nombre First_name
FROM Empleados
WHERE dep="Ventas";
```
 - b)

```
CREATE VIEW EmpDepVentas (Last_name, First_name)
AS SELECT apellidos, nombre
FROM Empleados
WHERE dep="Ventas";
```



Vistas (Tablas virtuales) en SQL

- Otros operadores:

- Visualizar la estructura de una vista:

DESCRIBE <vista>;

donde:

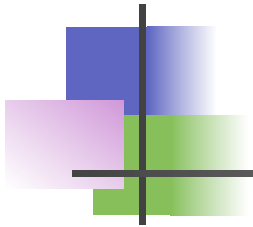
<vista> Es el nombre de la vista.

- Listar las vistas existentes:

SELECT * FROM USER_VIEWS;

- Eliminación de una vista:

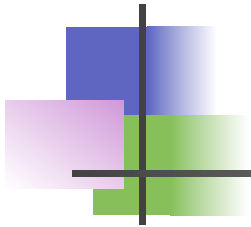
- DROP VIEW <nombre de la vista>



Vistas (Tablas virtuales) en SQL

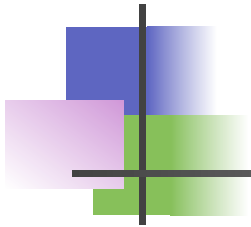
- Recuperando datos desde una Vista
 - Una Vista se puede emplear exactamente igual que una tabla base para recuperar datos de ella mediante consultas (SELECT).
 - Cuando se ejecuta una vista, el DBMS recupera la definición de la vista y ejecuta la subconsulta correspondiente.
 - Ejemplo: Listar los apellidos y nombre de los empleados del departamento "Ventas" ordenados alfabéticamente.

```
SELECT Last_name, First_name  
FROM EmpDepVentas  
ORDER BY Last_name, First_name;
```

Vistas (Tablas virtuales) en SQL

- Implementación de Vistas
 - Modificación de consulta. Convertir la consulta sobre las tablas de base. Su principal desventaja es el tiempo de ejecución.
 - Materialización de vistas. Crear físicamente la tabla de vista cuando se consulta la vista por primera vez y mantener la tabla para aprovecharla en consultas posteriores. Esto implica contar con estrategias para mantener actualizada la vista siendo una de ella la actualización incremental



Vistas (Tablas virtuales) en SQL

- Actualización de Vistas
 - Muy complicada y ambigua.
 - Una vista con una sola tabla base es actualizable si los atributos de la vista contienen la clave primaria o alguna otra clave candidata de la relación base.
 - En general las vistas definidas sobre múltiples tablas por medio de join no son actualizables.
 - Las vistas definidas mediante agrupación y funciones de agregación no son actualizables.

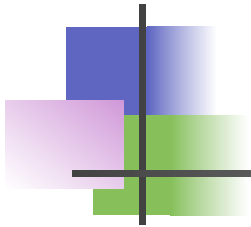


Vistas (Tablas virtuales) en SQL

■ Tipos de Vistas:

Horizontales o por Restricción:

- Son las que restringen el acceso de un usuario a únicamente un conjunto de filas de una tabla.
- Son adecuadas cuando la tabla sobre la que se definen contiene datos que relacionan a varias organizaciones o usuarios.
- Proporcionan una “tabla privada” para cada organización/usuario compuesta solo por las filas necesarias para ese usuario en concreto.

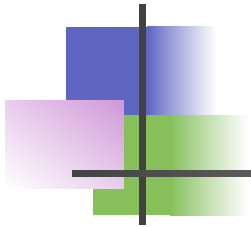


Vistas (Tablas virtuales) en SQL

- Tipos de Vistas:

- Horizontales o por Restricción, Ejemplo:
- “Los empleados del departamento de ‘Administración’ ”

```
CREATE VIEW ADMINISTRACION AS  
SELECT *  
FROM EMPLEADO  
WHERE DNO IN (SELECT NumeroDpto  
              FROM DEPARTAMENTO  
              WHERE NombreDpto = 'Administración' );
```

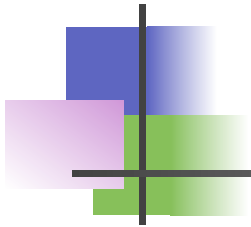


Vistas (Tablas virtuales) en SQL

- Tipos de Vistas:

- Verticales o por Proyección:

- Son las que restringen el acceso de un usuario a un cierto conjunto de columnas de una tabla.
 - Son adecuadas cuando los datos de una tabla son compartidos por varias organizaciones o usuarios, pero solo de algunos atributos.

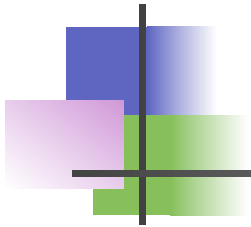


Vistas (Tablas virtuales) en SQL

- Tipos de Vistas:

- Verticales o por Proyección, Ejemplo:
- "Los Dni y Sueldo de los empleados de la empresa "

```
CREATE VIEW INFOSUELDOS (Identificacion, Sueldo)  
AS SELECT Dni, Sueldo  
FROM EMPLEADO;
```

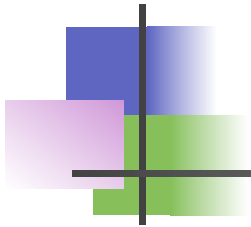


Vistas (Tablas virtuales) en SQL

- Tipos de Vistas:

- Subconjunto Fila/columna:

- Son las que se crean con parte de las filas y parte de las columnas de una tabla.
 - Suele ser una de las formas más habituales de creación de vistas.

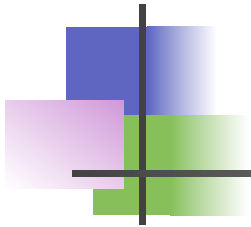


Vistas (Tablas virtuales) en SQL

- Tipos de Vistas:

- Subconjunto Fila/columna, Ejemplo:
- “Los Dni y Sueldo de los empleados del departamento de Administración”

```
CREATE VIEW INFOSUELDOS (Identificacion, Sueldo)
AS SELECT Dni, Sueldo
FROM EMPLEADO
WHERE DNO IN (SELECT NumeroDpto
              FROM DEPARTAMENTO
              WHERE NombreDpto = 'Administración' );
```

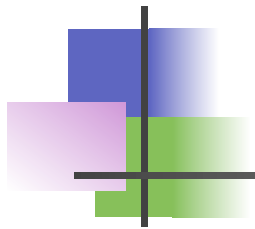
Vistas (Tablas virtuales) en SQL

- Tipos de Vistas:

- Agrupadas:

- Son vistas que incluyen una cláusula GROUP BY en la consulta especificada.
 - Ejemplo: "Una vista con el numero de empleados por departamento y el promedio de su sueldo"

```
CREATE VIEW RESUMEN (NUM_DEPTO, TOTAL_EMPLEADOS,  
PROM_SUELDO) AS  
SELECT DNO, COUNT(*), AVG(SUELDO)  
FROM EMPLEADO  
GROUP BY DNO;
```



Vistas (Tablas virtuales) en SQL

- Tipos de Vistas:

- Compuestas:

- Son vistas creadas a partir de los datos de dos o más tablas.
 - Presentan los datos como una única tabla virtual.
 - Se utilizan frecuentemente para facilitar el manejo de consultas complejas de uso frecuente en la BD.



Vistas (Tablas virtuales) en SQL

- Tipos de Vistas:

- Compuestas, Ejemplo:

- “Los departamentos con sus nombres, total empleados y promedio de sueldos”

```
CREATE VIEW RESUMEN (NOMBRE_DEPTO, TOTAL_EMPLEADOS,  
PROM_SUELDO) AS  
SELECT NOMBREDPTO, COUNT(*), AVG(SUELDO)  
FROM (EMPLEADO JOIN DEPARTAMENTO ON  
DNO=NUMERODPTO)  
GROUP BY NOMBREDPTO;
```