

# **BASES DE DATOS I**

## **VISTAS**

*curso 2008*

# Vistas: conceptos

---

- Una VISTA es una **tabla virtual** derivada de otras tablas (que pueden ser tablas base o también otras vistas).
- Sus tuplas no se almacenan sino que se generan a partir de las tablas de las que depende (no necesariamente existen en forma física, por ello se las considera **tablas virtuales**)
- Son útiles para usar, como si fueran tablas, consultas que se efectúan frecuentemente, y también para cuestiones de **seguridad**.

# Vistas: Sintaxis

---

## ***SQL3:***

```
CREATE VIEW <name> [ ( <column> [, ...] ) ]  
AS <query >  
[ WITH [ CASCADE | LOCAL ] CHECK OPTION ] ;
```

## ***En Oracle:***

```
CREATE [OR REPLACE] VIEW  
[FORCE | NOFORCE ] <name> [( <column> [, ...] )]  
AS <query>  
[WITH CHECK OPTION [CONSTRAINT constraint]]  
[WITH READ ONLY [CONSTRAINT constraint]];
```

# Definición - ejemplos

---

**PROVEEDOR** (id\_prov, nombreAp, situación, ciudad)

**PRODUCTO** (id\_prod, nombre, color, peso, ciudad)

**ENVIO** (id\_prov, id\_prod, cant)

Crear una vista de los proveedores cuya situación sea superior a 15.

**CREATE VIEW** Buenos\_proveedores (prov, sit, ciudad ) **AS**

SELECT id\_prov, situación, ciudad

FROM PROVEEDORES

WHERE situación > 15;

Crear una vista que corresponda a los proveedores cuya situación sea superior a 15 pero que no sean de Capital Federal.

**CREATE VIEW** Buenos\_proveedores\_noCapital **AS**

SELECT \*

FROM Buenos\_proveedores

WHERE ciudad <> 'Capital Federal';

# Definición - ejemplos

---

Los nombres de columna deberán especificarse explícitamente si:

- Cualquier columna se deriva de una función, una expresión operacional o un literal.
- Dos o mas columnas de la vista reciben el mismo nombre

```
CREATE VIEW EnvioProveedor (id_prov, cantTotal ) AS  
  SELECT id_prov, SUM (cant)  
  FROM ENVIO  
  GROUP BY id_prov;
```

```
CREATE VIEW Pareja_de_ciudades (ciudadProv, ciudadProd) AS  
  SELECT DISTINCT PROVEEDORES.ciudad, PRODUCTO.ciudad  
  FROM PROVEEDORES, PRODUCTO, ENVIO  
  WHERE PROVEEDORES.id_prov = ENVIO.id_prov AND  
  ENVIO.id_prod = PRODUCTO.id_prod ;
```

# Consulta y eliminación

---

Consulta: como si fuera una tabla base, por ejemplo

```
SELECT *  
FROM Buenos_proveedores  
WHERE prov < 30  
ORDER BY ciudad;
```

DROP VIEW <Nombre-vista > [CASCADE | RESTRICT ];

Ejemplo: DROP VIEW Buenos\_proveedores;

# Actualización de Vistas

---

Si la vista es actualizable y está definida con:

- **WITH CHECK OPTION:** todos los INSERT y UPDATE sobre la vista son chequeados para asegurar que los datos satisfacen la condición de definición de la Vista.
- **WITH LOCAL CHECK OPTION:** chequea la integridad sobre la vista.
- **WITH CASCADE CHECK OPTION :** chequea la integridad sobre la vista y cualquier vista dependiente.

# Ejemplos

---

```
CREATE VIEW Buenos_proveedores (prov, sit, ciudad ) AS  
  SELECT id_prov, situación, ciudad  
  FROM PROVEEDORES  
  WHERE situación > 15  
  WITH CHECK OPTION ;
```

Insert into Buenos\_proveedores (prov, sit, ciudad ) values (10, 20, 'Paris'); ✓<sub>n</sub>

Update Buenos\_proveedores set situacion = 5 where prov = 20; **X**

Insert into Buenos\_proveedores (prov, sit, ciudad ) values (8, 5, 'Roma'); **X**



# Vistas actualizables

---

No toda vista es actualizable.

Es SQL-actualizable si:

- Está definida a partir de una única tabla
- Conserva la clave (primaria o alternativa)
- No está definida mediante agrupación o funciones de agregación
- No incluye la cláusula Distinct
- No incluye subconsultas
- No se define mediante unión, intersección, diferencia

# Ejemplos - Vistas actualizables

---

No todas las vistas son actualizables:

```
CREATE VIEW Proveedor_ciudad (id_prov, ciudad )AS  
  SELECT id_prov, ciudad  
  FROM PROVEEDORES;
```

**ACTUALIZABLE**

```
CREATE VIEW Situación_ciudad (situación, ciudad )AS  
  SELECT situación, ciudad  
  FROM PROVEEDORES;
```

**NO  
ACTUALIZABLE**

Insert into Proveedor\_ciudad values (1,'Roma'); ✓ n

Insert into Situación\_ciudad values (40,'Roma'); X

# Ejemplos - Vistas actualizables

---

**CREATE VIEW** Proveedor\_Londres **AS**

*SELECT* id\_prov, nombreAp, situacion, ciudad  
*FROM* PROVEEDORES

*WHERE* ciudad = 'Londres';

**ACTUALIZABLE**

Insert into Proveedor\_Londres values (15,'Martin Lopez', 5, 'Tandil'); ✓<sub>n</sub>

**CREATE VIEW** EnvioProveedor (id\_prov, cantTotal ) **AS**

*SELECT* id\_prov, SUM (cant)

*FROM* ENVIO

*GROUP BY* id\_prov;

**NO**

**ACTUALIZABLE**

Insert into EnvioProveedor (1,100); **X**

# Actualización de Vistas en Oracle

---

## Motivos por los que una vista no es actualizable en Oracle:

- Contiene operadores de conjuntos (UNION, INTERSECT,...)
- Contiene el operador DISTINCT.
- Contiene funciones agregadas (SUM, AVG, ..).
- Contiene una cláusula GROUP BY
- Se define mediante una subconsulta en una lista SELECT
- Se define mediante JOIN (con excepción de *key-preserved table* \*)

\* Una tabla base o una vista es considerada una ***key-preserved table*** si cada valor clave (primary-key o unique-key) en cada tabla base es también clave en el resultado de la Join View.