

VISTAS

En este anexo vamos a ver algunos ejemplos sobre utilización de vistas para aclarar los conceptos, para ello vamos a utilizar las tablas que aparecen a continuación.

Uso de las vistas:

- **Como medida de seguridad a la hora de asignar permisos a los diferentes usuarios que hagan uso de la base de datos. Creamos vistas y permitimos a los diferentes usuarios que usen las vistas en lugar de las tablas directamente.**
- **Para resolver consultas de un grado de dificultad alta. Nos permite dividir una consulta en dos partes de forma que es más fácil de realizar.**

Una empresa almacena la información de sus empleados en dos tablas llamadas "empleados" y "secciones".

- Creamos las tablas:

```
create table secciones(  
  codigo number(2),  
  nombre varchar2(20),  
  sueldo number(5,2)  
  constraint CK_secciones_sueldo check (sueldo>=0),  
  constraint PK_secciones primary key (codigo)  
);
```

```
create table empleados(  
  legajo number(5),  
  documento char(8),  
  sexo char(1)  
  constraint CK_empleados_sexo check (sexo in ('f','m')),  
  apellido varchar2(20),  
  nombre varchar2(20),  
  domicilio varchar2(30),  
  seccion number(2) not null,  
  cantidadhijos number(2)  
  constraint CK_empleados_hijos check (cantidadhijos>=0),  
  estadocivil char(10)  
  constraint CK_empleados_estadocivil check (estadocivil in  
('casado','divorciado','soltero','viudo')),  
  fechaingreso date,  
  constraint PK_empleados primary key (legajo),
```

```

constraint FK_empleados_seccion
foreign key (seccion)
references secciones(codigo),
constraint UQ_empleados_documento
unique(documento)
);

```

- Insertar algunos registros:

```

insert into secciones values(1,'Administracion',300);
insert into secciones values(2,'Contaduría',400);
insert into secciones values(3,'Sistemas',500);

```

```

insert into empleados values(100,'22222222','f','Lopez','Ana','Colon
123',1,2,'casado','10/10/1990');
insert into empleados values(102,'23333333','m','Lopez','Luis','Sucre
235',1,0,'soltero','02/10/1990');
insert into empleados
values(103,'24444444','m','Garcia','Marcos','Sarmiento
1234',2,3,'divorciado','12/07/1998');
insert into empleados
values(104,'25555555','m','Gomez','Pablo','Bulnes
321',3,2,'casado','10/09/1998');
insert into empleados values(105,'26666666','f','Perez','Laura','Peru
1254',3,3,'casado','05/09/2000');

```

- Creamos la vista "vista_empleados", que almacena nombre y apellido como nombre, sexo, sección y cantidad de hijos de todos los empleados de la empresa, el resultado, con nuestros datos, de una combinación en la cual se muestran 5 campos:

```

create view vista_empleados as
select (apellido||' '||e.nombre) as nombre,sexo,
s.nombre as seccion, cantidadhijos
from empleados e
join secciones s
on codigo=seccion;

```

- Para ver la información que almacena la vista, utilizaremos una select como si se tratase de una tabla:

```

select *from vista_empleados;

```

- Realizamos una consulta a la vista como si se tratara de una tabla:

```

select seccion,count(*) as cantidad
from vista_empleados
group by seccion;

```

- Creamos otra vista de "empleados" denominada "vista_empleados_ingreso" que almacena la cantidad de empleados por año:

```
create view vista_empleados_ingreso (fecha,cantidad)
as
select extract(year from fechaingreso),count(*)
from empleados
group by extract(year from fechaingreso);--año de la fecha de ingreso
```

- Vemos la información:

```
select *from vista_empleados_ingreso;
```

- Los registros resultantes de una vista no se almacena en la base de datos, sino la definición de la vista, por lo tanto, al modificar las tablas referenciadas por la vista, el resultado de la vista cambia.
- Modificamos una fecha en la tabla "empleados" y luego consultamos la vista para verificar que está actualizada:

```
update empleados set fechaingreso='10/09/2000' where
fechaingreso='10/09/1998';

select *from vista_empleados_ingreso;
```

MODIFICAR DATOS DE UNA TABLA A TRAVÉS DE UNA VISTA

Podemos modificar, eliminar o insertar datos pertenecientes a una vista, contenidos en una tabla, pero no en todos los casos porque siempre tiene preferencia conservar la integridad de la base de datos.

- Vamos a modificar los contenidos y la estructura de las tablas utilizadas

```
delete from secciones;
delete from empleados;
```

Modificamos la estructura de las tablas

```
Alter table secciones drop (sueldo);
Alter table empleados add (sueldo number (6,2));
```

- Insertar algunos registros:

```
insert into secciones values(1,'Administracion');
insert into secciones values(2,'Contaduría');
insert into secciones values(3,'Sistemas');
```

```

insert into empleados values(100,'22222222','f','Lopez','Ana','Colon
123',1,2,'casado','10/10/1990',600);

insert into empleados values(101,'23333333','m','Lopez','Luis','Sucre
235',1,0,'soltero','02/10/1990',650);

insert into empleados values(103,'24444444','m','Garcia','Marcos',
'Sarmiento 1234',2,3,'divorciado','07/12/1998',800);

insert into empleados
values(104,'25555555','m','Gomez','Pablo','Bulnes
321',3,2,'casado','10/09/1998',900);

insert into empleados values(105,'26666666','f','Perez','Laura','Peru
1254',3,3,'casado','05/09/2000',700);

```

- Vemos la información contenida en la vista que teníamos creada desde el principio:

```
select *from vista_empleados;
```

- Creamos otra vista de "empleados" denominada "vista_empleados2" que consulta solamente la tabla "empleados":

```

create view vista_empleados2
as
select legajo,nombre,apellido,fechaingreso,seccion,sueldo
from empleados
where sueldo>=600;

```

Consultamos la vista:

```
select *from vista_empleados2;
```

- No podemos insertar un registro en la vista "vista_empleados" porque tal vista tiene campos calculados ("nombre", que es una concatenación de "apellido" y "nombre"), además afecta a 2 tablas ("empleados" y "secciones") y hay campos no accesibles desde la vista que no admiten valores nulos. Si ejecutamos el siguiente "insert", Oracle mostrará un mensaje de error:

```
insert into vista_empleados values('Pedro Perez','m','Sistemas',2);
```

- Podemos insertar un registro en la vista "vista_empleados2" porque tal vista afecta a una sola tabla y los campos de ""empleados" no accesibles desde la vista admiten valores nulos:

```

insert into vista_empleados2
values(200,'Pedro','Perez','10/10/2000',2,800);

```

- Vemos la tabla "empleados" para comprobar que el nuevo registro insertado desde la vista está presente en "empleados", los campos para los cuales no se insertaron datos, almacenan el valor "null":

```
select *from empleados;
```

- Actualizamos el campo "nombre" de un registro de la vista "vista_empleados2":

```
update vista_empleados2 set nombre='Beatriz' where nombre='Ana';
```

- Verificamos que se actualizó la tabla:

```
select *from empleados;
```

- Si intentamos actualizar el campo "nombre" de un empleado a través de la vista "vista_empleados", Oracle no lo permite porque es una columna calculada (concatenación de dos campos):

```
update vista_empleados set nombre='Lopez Carmen' where nombre='Lopez Beatriz';
```

- Si podemos actualizar otros campos, por ejemplo, el campo "cantidadhijos" de un empleado a través de la vista "vista_empleados":

```
update vista_empleados set cantidadhijos=3 where nombre='Lopez Beatriz';
```

Verificamos que se actualizó la tabla:

```
select *from empleados;
```

- Eliminamos un registro de "empleados" a través de la vista "vista_empleados2":

```
delete from vista_empleados2 where apellido='Lopez' and nombre='Beatriz';
```

- Verificamos que se eliminó tal registro de la tabla "empleados":

```
select *from empleados;
```

- Podemos eliminar registros de empleados a través de la vista "vista_empleados":

```
delete from vista_empleados where seccion='Administracion';
```

- Verificamos que no hay registros en "empleados" de la sección "1" ("Administracion"):

```
select *from empleados;
```

READ ONLY

Vamos a utilizar una nueva tabla empleados2

- Creamos la tabla:

```
create table empleados2(  
  documento char(8),  
  sexo char(1)  
  constraint CK_empleados2_sexo check (sexo in ('f','m')),  
  apellido varchar2(20),  
  nombre varchar2(20),  
  domicilio varchar2(30),  
  seccion varchar2(30),
```

```

cantidadhijos number(2),
constraint CK_empleados2_hijos check (cantidadhijos>=0),
estadocivil char(10)
        constraint CK_empleados2_estadocivil check (estadocivil in
('casado','divorciado','soltero','viudo')),
fechaingreso date
);

```

Insertamos algunos registros:

```

insert into empleados2 values('22222222','f','Lopez','Ana','Colon
123','Administracion',2,'casado','10/10/1990');
insert into empleados2 values('23333333','m','Lopez','Luis','Sucre
235','Administracion',0,'soltero','02/10/1990');
insert into empleados2
values('24444444','m','Garcia','Marcos','Sarmiento
1234','Contaduria',3,'divorciado','07/12/1998');
insert into empleados2 values('25555555','m','Gomez','Pablo','Bulnes
321','Contaduria',2,'casado','10/09/1998');
insert into empleados2 values('26666666','f','Perez','Laura','Peru
1254','Sistemas',3,'casado','05/09/2000');

```

- Reemplazamos, modificamos, la vista "vista_empleados", que teníamos inicialmente, esto lo hacemos añadiendo "or replace" en el momento de creación de la vista, de esta manera si la vista no existe la crea y si ya existe una vista con este nombre la modifica, la reemplaza.

```

create or replace view vista_empleados
as
select apellido, nombre, sexo, seccion
from empleados2;

```

- Creamos la vista "vista_empleados2", igual que "vista_empleados", pero ahora colocamos **"with read only"** para impedir que puedan ejecutarse "insert", "update" y "delete" sobre esta vista:

```

create or replace view vista_empleados2
as
select apellido, nombre, sexo, seccion
from empleados2
with read only;

```

- Actualizamos el nombre de un empleado a través de la vista "vista_empleados2":

```

update vista_empleados set nombre='Beatriz' where nombre='Ana';

```

- Veamos si la modificación se realizó en la tabla:

```

select *from empleados2;

```

- Intentamos actualizar el nombre de un empleado a través de la vista "vista_empleados2":

```
update vista_empleados2 set nombre='Pedro' where nombre='Marcos';
```

No lo permite.

- Insertamos un registro en la tabla "empleados" a través de la vista "vista_empleados":

```
insert into vista_empleados values('Juarez','Juan','m','Sistemas');
```

Oracle acepta la inserción.

- Verificamos que la inserción se realizó en la tabla:

```
select *from empleados2;
```

Intentamos insertar un registro a través de la vista "vista_empleados2":

```
insert into vista_empleados2
values('Gimenez','Julieta','f','Sistemas');
```

Oracle no lo permite porque la vista fue definida con "with read only".

- Eliminamos un registro en la tabla "empleados" a través de la vista "vista_empleados":

```
delete from vista_empleados where apellido='Juarez';
```

Oracle acepta la eliminación.

- Verificamos que la eliminación se realizó en la tabla:

```
select *from empleados2;
```

- Intentamos eliminar registros a través de la vista "vista_empleados2":

```
delete from vista_empleados2 where apellido='Lopez';
```

Oracle no lo permite porque la vista fue definida con "with read only".

WITH CHECK OPTION

Antes de empezar borramos el contenido de la tabla empleados2, eliminamos dos campos de su estructura e insertar registros nuevos a la tabla.

```
delete from empleados2 ;
alter table empleados2 drop (domicilio, fechaingreso);
```

```
insert into empleados2
values('22222222','f','Lopez','Ana','Administracion',2,'casado');

insert into empleados2
values('23333333','m','Lopez','Luis','Administracion',0,'soltero');
```

```

insert into empleados2
values('24444444','m','Garcia','Marcos','Sistemas',3,'divorciado');

insert into empleados2
values('25555555','m','Gomez','Pablo','Sistemas',2,'casado');

insert into empleados2
values('26666666','f','Perez','Laura','Contaduria',3,'casado');

```

- Creamos o reemplazamos (si existe) la vista "vista_empleados", para que muestre el nombre, apellido, sexo y sección de todos los empleados de "Administración" agregando la cláusula "**with check option**" para evitar que se modifique la sección de tales empleados a través de la vista y que se inserten empleados de otra sección:

```

create or replace view vista_empleados
as
select apellido, nombre, sexo, seccion
from empleados2
where seccion='Administracion'
with check option;

```

- Creamos otra vista igual que la anterior pero sin CHECK OPTION para ver la diferencia de funcionamiento.

```

create or replace view vista_empleados_Sin AS
select apellido, nombre, sexo, seccion
from empleados2
where seccion='Administracion';

```

- Consultamos las vistas:

```

select *from vista_empleados;
select *from vista_empleados_sin;

```

- Actualizamos el nombre de un empleado a través de la vista:

```

update vista_empleados set nombre='Beatriz' where nombre='Ana';

```

Oracle aceptó la actualización porque la vista está creada con la opción "with check option" que impide que desde la vista modifiquemos lo referente al atributo sección de los pertenecientes a la vista, no nos permitirá cambiar a un empleado de sección Administración a otra, ni insertar empleados que no pertenezcan a la sección Administración.

- Veamos si la modificación se realizó en la tabla:

```

select *from empleados2;

```

- Intentamos actualizar la sección de un empleado a través de la vista con CHECK OPTION:


```
update vista_empleados set seccion='Sistemas' where nombre='Beatriz';
```

Oracle no aceptó la actualización porque supone un cambio de sección de Beatriz.

- Intentamos actualizar la sección de un empleado a través de la vista sin CHECK OPTION:

```
update vista_empleados_sin set seccion='Sistemas' where nombre='Beatriz';
```

Oracle acepta el cambio desde esta vista sin check option

- Este mismo cambio tampoco hay ningún problema si lo hago desde la tabla directamente, la restricción solo se aplica a la vista.

```
update empleados2 set seccion='Sistemas' where nombre='Beatriz';
```

- Insertamos un registro mediante la vista (ver que este empleado pertenece al departamento de administración, que es el que contiene la vista:

```
insert into vista_empleados  
values('Gomez', 'Gabriela', 'f', 'Administracion');
```

Oracle acepta la inserción porque insertamos un valor para "sección" que incluirá el registro en la vista.

- Intentamos insertar un empleado de otra sección:

```
insert into vista_empleados values('Torres', 'Tatiana', 'f', 'Sistemas');
```

Oracle no acepta la inserción porque insertamos un valor para que no pertenecerá a la vista. Por supuesto si esta inserción la hacemos directamente a la tabla no hay problema.

```
insert into vista_empleados_sin  
values('Torres', 'Tatiana', 'f', 'Sistemas');
```

- Comprobamos cambios

```
select * from empleados2;
```