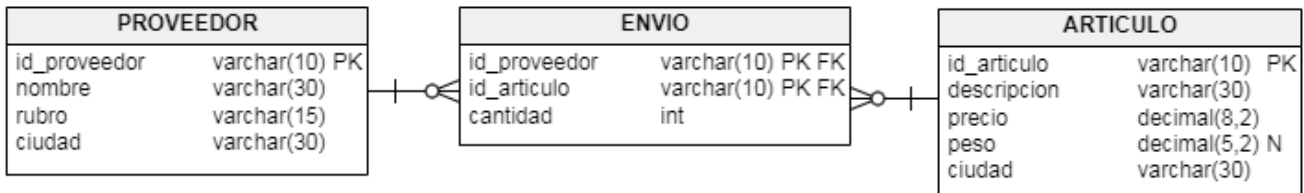


Vistas

Ejercicio 1

Considere el siguiente DERExt ([esq_articulos](#)):



- a) Defina las siguientes vistas mediante sentencias SQL:
- a.1) *ENVIOS500* con los envíos de 500 o más unidades (a partir de ENVIO)
 - a.2) *ENVIOS500-M* con los envíos de entre 500 y 999 unidades (a partir de *ENVIOS500*)
 - a.3) *RUBROS_PROV* con los diferentes rubros que poseen los proveedores ubicados en Tandil
 - a.4) *ENVIOS_PROV* con los diferentes id y nombre de proveedor y la cantidad total de unidades enviadas
- b) Determine si las vistas anteriores son automáticamente actualizables según el estándar SQL o no (en este caso indicar la/s causa/s).
- c) Compruebe si resultan automáticamente actualizables para PostgreSQL, proporcionando sentencias de actualización sobre las vistas en cada caso.

Ejercicio 2

Considere el [esquema de Peliculas](#) (unc_esq_peliculas):

- a) Escriba las sentencias de creación de las vistas solicitadas en cada caso.
- b) Indique si para el estándar SQL dicha vista es actualizable o no. Justifique cada respuesta.
- c) idem b) en PostgreSQL, teniendo en cuenta de construir vistas que resulten automáticamente actualizables, siempre que sea posible.
1. Cree una vista EMPLEADO_DIST_20 que liste el id_empleado, nombre, apellido, sueldo y fecha_nacimiento de los empleados que corresponden al distribuidor con identificador 20.
 2. Sobre la vista anterior defina otra vista EMPLEADO_DIST_2000 con el id, nombre, apellido y sueldo de los empleados con sueldo mayor a 2000.
 3. Sobre la vista EMPLEADO_DIST_20 cree la vista EMPLEADO_DIST_20_70 con aquellos empleados que han nacido en la década del 70 (entre los años 1970 y 1979).
 4. Cree una vista PELICULAS_ENTREGADAS que contenga el código de cada película y la cantidad de unidades entregadas.
 5. Cree una vista DISTRIB_NAC con los el id de las distribuidoras nacionales, nro_incripcion y encargado, con distribuidor mayorista del país AR
 6. Usando la vista anterior, cree la vista DISTRIB_NAC_MAS2EMP con los datos completos de las distribuidoras nacionales cuyos departamentos tengan más de 2 empleados.

Ejercicio 3

A partir del Ejercicio 1)

- a) ¿Qué sucedería con las actualizaciones sobre las vistas definidas si se agrega **WITH CHECK OPTION**?
- b) Determine el efecto de las siguientes operaciones sobre la vista *Envios500*, según si tenga especificada opción de chequeo (WITH CHECK OPTION) o no.

Nota: considere que existen los proveedores P1 y P2 y los artículos A1 y A2

Vistas

- d.1) INSERT INTO *ENVIOS500* VALUES ('P1', 'A1', 500);
- d.2) INSERT INTO *ENVIOS500* VALUES ('P2', 'A2', 300);
- d.3) UPDATE *ENVIOS500* SET cantidad=100 WHERE id_proveedor= 'P1';
- d.4) UPDATE *ENVIOS500* SET cantidad=1000 WHERE id_proveedor= 'P2';
- d.5) INSERT INTO *ENVIOS500* VALUES ('P1', 'A3', 700);

Ejercicio 4

Dadas las siguientes vistas sobre el [esquema de Voluntarios](#) (unc_esq_voluntario):

```
CREATE OR REPLACE VIEW tarea10000hs AS
SELECT * FROM tarea
WHERE max_horas > 10000
WITH LOCAL CHECK OPTION;
```

```
CREATE OR REPLACE VIEW tarea10000rep AS
SELECT * FROM tarea10000hs
WHERE id_tarea LIKE '%REP%'
WITH LOCAL CHECK OPTION;
```

a) ¿Cuál sería el resultado de las siguientes sentencias (ejecutadas en el orden dado)? Justifique.

1. INSERT INTO tarea10000rep (id_tarea, nombre_tarea, min_horas, max_horas)
VALUES ('MGR', 'Org Salud', 18000, 20000);
2. INSERT INTO tarea10000hs (id_tarea, nombre_tarea, min_horas, max_horas)
VALUES ('REPA', 'Organiz Salud', 4000, 5500);
3. INSERT INTO tarea10000rep (id_tarea, nombre_tarea, min_horas, max_horas)
VALUES ('CC_REP', 'Organizacion Salud', 8000, 9000);
4. INSERT INTO tarea10000hs (id_tarea, nombre_tarea, min_horas, max_horas)
VALUES ('ROM', 'Org Salud', 10000, 12000);

b) Luego de ejecutadas las sentencias anteriores, en qué objetos de la BD aparecen insertadas las tuplas?

Ejercicio 5

Analice cuáles serían los controles y el comportamiento ante actualizaciones sobre las vistas EMPLEADO_DIST_20, EMPLEADO_DIST_2000 y EMPLEADO_DIST_20_70 creadas en el ej. 2, si las mismas están definidas con WITH CHECK OPTION, ya sea LOCAL o CASCADE en cada una de ellas. Evalúe diferentes alternativas.

Ejercicio 6

Para las siguientes vistas de ensamble sobre el [esquema de Peliculas](#) (unc_esq_peliculas):

```
CREATE VIEW ciudad_kp_2 AS
SELECT id_ciudad, nombre_ciudad, c.id_pais, nombre_pais
FROM ciudad c NATURAL JOIN pais p;
```

```
CREATE VIEW entregas_kp_3 AS
SELECT nro_entrega, re.codigo_pelicula, cantidad, titulo
FROM renglon_entrega re JOIN pelicula p using (codigo_pelicula);
```

Vistas

- Indique en cada caso cuál es la clave preservada en la vista, de qué tabla proviene y qué atributos de la vista se pueden actualizar de acuerdo al estándar SQL
- Escriba la implementación completa de los **triggers INSTEAD OF** que considere adecuados para permitir actualizaciones en cada una de las vistas en PostgreSQL. Justifique.
- Provea sentencias de actualización sobre cada vista e indique su propagación sobre las tablas base.

Ejercicio 7

Sobre el [esquema de Voluntarios](#), defina una vista que contenga:

Vista1: nro, voluntario, nombre, apellido, horas aportadas y datos completos de la tarea que realiza

Vista2: datos de las instituciones ubicadas en el país US junto con la cantidad de tareas que se han realizado en ellas

Vista3: identificador y nombre de todas las instituciones existentes, con la cantidad de voluntarios que están trabajando en cada una

Para cada una de las vistas, determine si resulta automáticamente actualizable en PostgreSQL. Caso contrario, implemente en PostgreSQL los **triggers INSTEAD OF** necesarios para permitir actualizaciones sobre ellas.

Ejercicio 8

Dadas las siguientes vistas a partir del esquema de Voluntarios:

```
CREATE VIEW voluntario1000hs AS
SELECT v.* FROM voluntario v
WHERE v.horas_aportadas > 1000
WITH LOCAL CHECK OPTION;
```

```
CREATE VIEW voluntario1000mayor AS
SELECT m.* FROM voluntario1000hs m
WHERE AGE(m.fecha_nacimiento) > interval '21 year'
WITH LOCAL CHECK OPTION;
```

- a) ¿Cuál sería el resultado de las siguientes sentencias (ejecutadas en el orden dado)?

Nota: Tener en cuenta que el resto de los valores se completan con datos válidos.

1. INSERT INTO Voluntario1000Hs (nro_voluntario, apellido, fecha_nacimiento, horas_aportadas,)
VALUES (344555, 'Perez', to_date('1986-10-28', 'yyyy-MM-dd'), 6, ...);

2. INSERT INTO Voluntario1000Mayor (nro_voluntario, apellido, fecha_nacimiento, horas_aportadas,)
VALUES (366777, 'Gomez', to_date('1990-11-27', 'yyyy-MM-dd'), 1100, ...);

3. INSERT INTO Voluntario1000Mayor (nro_voluntario, apellido, fecha_nacimiento, horas_aportadas,)
VALUES (366777, 'Gomez', to_date('1990-11-27', 'yyyy-MM-dd'), 60, ...);

4. INSERT INTO Voluntario1000Hs (nro_voluntario, apellido, fecha_nacimiento, horas_aportadas,)
VALUES (366777, 'Gomez', to_date('1990-11-27', 'yyyy-MM-dd'), 2300, ...);

- b) ¿En qué objetos de la BD (tablas/vistas) aparecen las tuplas insertadas?