

Triggers

- a. Se pide: Crear un trigger que valide que ante un insert (de una o más filas) en la tabla `ítems`, realice la siguiente validación.
- Si la orden de compra a la que pertenecen los ítems ingresados corresponde a clientes del estado de California, se deberá validar que las órdenes pueden contar con hasta 5 registros en la tabla `ítem`.
 - Si se insertan más ítems de los definidos, el resto de los ítems se deberán insertar en la tabla `ítems_error` la cuál contiene la misma estructura que la tabla `ítems` más un atributo fecha que deberá contener la fecha del día en que se trató de insertar.

Si por ejemplo la OC cuenta con 3 ítems y se realiza un insert masivo de 3 ítems más, el trigger deberá insertar los 2 primeros en la tabla `ítems` y el restante en la tabla `ítems_error`.

Supuesto: En el caso de un insert masivo los ítems pertenecen siempre a la misma orden.

- b. Dada la siguiente vista

```
CREATE VIEW ProdPorFabricante AS
SELECT m.manu_code, manu_name, COUNT(*)
FROM manufact m INNER JOIN stock s ON (m.manu_code = s.manu_code)
GROUP BY m.manu_code, manu_name
```

Se pide: Crear un trigger que permita ante un insert en la vista `ProdPorFabricante` insertar los datos en la tabla `manufact`.

Observaciones: el atributo `leadtime` deberá insertarse con un valor default 10
El trigger deberá contemplar inserts de varias filas, ante un `INSERT` masivo (`INSERT SELECT`).

- c. Crear un trigger que valide que ante un update (de una o más filas) en la tabla `customer`, realice la siguiente validación.
- La cuota de clientes correspondientes al estado de California es de 20, si se supera dicha cuota se deberán grabar el resto de los clientes en la tabla `customer_update_pend`.
 - Validar que si de los clientes a modificar se modifica el Estado, no se puede superar dicha cuota.

Si por ejemplo el estado de CA cuenta con 18 clientes y se realiza un update masivo de 5 clientes con estrado de CA, el trigger deberá modificar los 2 primeros en la tabla `customer` y los restantes grabarlos en la tabla `customer_updates_pend`. La tabla `customer_updates_pend` tendrá la misma estructura que la tabla `customer` con un atributo adicional fecha que deberá actualizarse con la fecha y hora del día.

- d. Dada la siguiente vista

```
CREATE VIEW ProdPorFabricanteDet AS
SELECT m.manu_code, manu_name, stock_num, description
FROM manufact m LEFT OUTER JOIN stock s ON (m.manu_code = s.manu_code)
```

Se pide: Crear un trigger que permita ante un `DELETE` en la vista `ProdPorFabricante` borrar los datos en la tabla `manufact` pero sólo de los fabricantes cuyo campo `description` sea NULO (o sea que no tienen stock).

El trigger deberá contemplar borrado de varias filas, ante un `DELETE` masivo. En ese caso sólo borrará de la tabla los fabricantes que no tengan productos en stock.