-- ETAPA 1: CREACION DE TRIGGER

-- crearemos una venta

insert into venta values(20,0,0,0,'10/10/2016',1,4);

-- crearemos las entradas al cine

insert into ticket values(44,40,'D',0,4,6);

insert into ticket values(45,41,'D',0,4,6);

insert into ticket values(46,42,'D',0,4,6);

-- comprarlos boletos: el trigger debe detectar que se compro

-- un boleto en DETALLE y recuperar el valor de la PELICULA

-- y el DESCUENTO (SI TIENE) por el precio de la pelicula

-- a) crear una funcion que permita retornar el valor de la

-- PELICULA, tomando el IDVENTA recuperare el valor de la

-- PELICULA

create or replace function fn\_obt\_precio(p\_idventa number)

return number

is

v\_precio number;

v\_sql varchar2(255);

begin

v\_sql:='select p.precio

from venta v inner join peliculas p

on v.idpelicula=p.idpelicula where v.idventa = :p1';

execute immediate v\_sql into v\_precio using p\_idventa;

return v\_precio;

end;

----- probar

select v.\*, fn\_obt\_precio(v.idventa) from venta v;

-- b) crear una funcion que retorne el PORCENTAJE de descuento

-- que tiene el PRECIO de la pelicula

create or replace function fn\_obt\_descuento(p\_precio number)

return number

is

v\_valor number;

begin

select porcentaje into v\_valor from descuentoentradas

where p\_precio between valorentrada\_min and valorentrada\_max;

return round(p\_precio \* v\_valor,0);

exception

when others then

return 0;

end;

-- probar

select v.\*,fn\_obt\_precio(v.idventa) as precio\_entrada

,fn\_obt\_descuento( fn\_obt\_precio(v.idventa) ) as descuento

from venta v;

-- c) creacion de TRIGGER

create or replace trigger trg\_detalle

after insert or delete

on detalle

for each row

declare

v\_precio number;

v\_descuento number;

v\_total number;

begin

if inserting then

v\_precio:= fn\_obt\_precio(:new.idventa);

if :new.adulto='N' then

v\_precio:=round(v\_precio/2,0);

end if;

v\_descuento:= fn\_obt\_descuento(v\_precio);

v\_total:=v\_precio-v\_descuento;

update venta set

subtotal=subtotal+v\_precio,

descuento=descuento+v\_descuento,

total=total+v\_total

where idventa= :new.idventa;

end if;

if deleting then

v\_precio:= fn\_obt\_precio(:old.idventa);

if :old.adulto='N' then

v\_precio:=round(v\_precio/2,0);

end if;

v\_descuento:= fn\_obt\_descuento(v\_precio);

v\_total:=v\_precio-v\_descuento;

update venta set

subtotal=subtotal-v\_precio,

descuento=descuento-v\_descuento,

total=total-v\_total

where idventa= :old.idventa;

end if;

end;

select \* from venta where idventa=20;

insert into detalle values(44,20,'N');

insert into detalle values(45,20,'S');

insert into detalle values(46,20,'S');

delete from detalle where idticket=44;

------------------------------------------------

-- ETAPA 2:

-- recalcular las ventas

declare

v\_precio number;

v\_descuento number;

v\_total number;

begin

for x in (select \* from detalle)

loop

v\_precio:= fn\_obt\_precio(x.idventa);

if x.adulto='N' then

v\_precio:=round(v\_precio/2,0);

end if;

v\_descuento:= fn\_obt\_descuento(v\_precio);

v\_total:=v\_precio-v\_descuento;

update venta set

subtotal=subtotal+v\_precio,

descuento=descuento+v\_descuento,

total=total+v\_total

where idventa= x.idventa;

end loop;

end;

select \* from venta;

-- resumenes

select

c.nombre,p.titulo,p.idioma,count(t.idticket) as entradas

,sum(v.total) as recaudacion

from detalle d

join venta v on v.idventa = d.idventa

join peliculas p on v.idpelicula = p.idpelicula

join ticket t on t.idticket = d.idticket

join cine c on c.idcine=t.cine\_idcine

where v.fecha BETWEEN '01/01/2016' and '31/12/2016'

group by c.nombre, p.titulo, p.idioma

order by recaudacion desc;

create sequence sec\_resumen\_cine;

create or replace procedure sp\_informe\_resumen\_cine(p\_fecha date,p\_fi date,p\_ft date)

is

cursor cur\_resumen is select

c.nombre,p.titulo,p.idioma,count(t.idticket) as entradas

,sum(v.total) as recaudacion

from detalle d

join venta v on v.idventa = d.idventa

join peliculas p on v.idpelicula = p.idpelicula

join ticket t on t.idticket = d.idticket

join cine c on c.idcine=t.cine\_idcine

where v.fecha BETWEEN p\_fi and p\_ft

group by c.nombre, p.titulo, p.idioma

order by recaudacion desc;

begin

execute immediate 'truncate table resumen\_cine';

for x in cur\_resumen

loop

insert into resumen\_cine values(sec\_resumen\_cine.nextval,p\_fecha,x.nombre,x.titulo,

x.idioma,x.entradas, x.recaudacion);

end loop;

end;

execute sp\_informe\_resumen\_cine(sysdate,'01/01/2016','31/12/2016');

select \* from resumen\_cine;

----------------------------------------------------------------------

-- RESUMEN 2: resumen por pelicula

create or replace function fn\_adulto\_pelicula(p\_idpelicula number)

return number

is

v\_cantidad number;

begin

select count(\*) into v\_cantidad

from detalle d join venta v on v.idventa = d.idventa

where v.idpelicula=p\_idpelicula and d.adulto='S';

return v\_cantidad;

end;

create or replace function fn\_nino\_pelicula(p\_idpelicula number)

return number

is

v\_cantidad number;

begin

select count(\*) into v\_cantidad

from detalle d join venta v on v.idventa = d.idventa

where v.idpelicula=p\_idpelicula and d.adulto='N';

return v\_cantidad;

end;

create or replace procedure sp\_resumen\_pelicula

is

cursor cur\_pelicula is select p.idpelicula, p.titulo,p.idioma,

c.descripcion,sum(v.total) as recaudacion

from detalle d

join venta v on v.idventa = d.idventa

join peliculas p on v.idpelicula = p.idpelicula

join categoria c on p.idcategoria = c.idcategoria

group by p.titulo, p.idioma, c.descripcion, p.idpelicula

order by p.idpelicula;

v\_adulto number;

v\_nino number;

begin

execute immediate 'truncate table resumen\_peliculas';

for x in cur\_pelicula

loop

v\_adulto:= fn\_adulto\_pelicula(x.idpelicula);

v\_nino:= fn\_nino\_pelicula(x.idpelicula);

insert into resumen\_peliculas values(x.idpelicula,x.titulo,x.idioma,

x.descripcion,v\_adulto,v\_nino, x.recaudacion);

end loop;

sp\_pelicula\_sin\_ver;

end;

execute sp\_resumen\_pelicula;

select \* from resumen\_peliculas;

-----------------------------------------------

create or replace procedure sp\_pelicula\_sin\_ver

is

cursor cur\_peli\_sin\_ver is select p.titulo,v.total

from peliculas p

left join venta v on v.idpelicula = p.idpelicula

where v.total is null;

begin

for x in cur\_peli\_sin\_ver

loop

insert into errores values(sq\_error.nextval,0,

'película ' || x.titulo ||' no tuvo movimiento de entradas',sysdate);

end loop;

end;

---------------------------------------------------------

create or replace procedure sp\_errores(p\_codigo number,p\_desc varchar2)

is

v\_sql varchar2(255);

begin

v\_sql:='insert into errores values(sq\_error.nextval,:p1,:p2,sysdate)';

execute immediate v\_sql using p\_codigo,p\_desc;

end;

----------------------------------------------------------

-- paquete

create or replace PACKAGE pkg\_cine

is

function fn\_obt\_precio(p\_idventa number)return number;

function fn\_obt\_descuento(p\_precio number)return number;

procedure sp\_errores(p\_codigo number,p\_desc varchar2);

end;

create or replace package body pkg\_cine

is

function fn\_obt\_precio(p\_idventa number) return number

is

v\_precio number;

v\_sql varchar2(255);

begin

v\_sql:='select p.precio

from venta v inner join peliculas p

on v.idpelicula=p.idpelicula where v.idventa = :p1';

execute immediate v\_sql into v\_precio using p\_idventa;

return v\_precio;

end;

function fn\_obt\_descuento(p\_precio number) return number

is

v\_valor number;

begin

select porcentaje into v\_valor from descuentoentradas

where p\_precio between valorentrada\_min and valorentrada\_max;

return round(p\_precio \* v\_valor,0);

exception

when others then

return 0;

end;

procedure sp\_errores(p\_codigo number,p\_desc varchar2)

is

v\_sql varchar2(255);

begin

v\_sql:='insert into errores values(sq\_error.nextval,:p1,:p2,sysdate)';

execute immediate v\_sql using p\_codigo,p\_desc;

end;

end;