Vamos a mostrar un ejemplo de uso de triggers y procedimientos almacenados paso a paso.

Nuestro sistema tendrá dos tablas, una con ventas y la otra con comisiones. En la primera se almacenarán cada venta que se realiza en un comercio, y en la segunda las comisiones que le corresponden a cada vendedor en el momento.

CREATE TABLE `ventas` (

`vendedor` int(11),

`producto` int(11),

`importe` float

)

CREATE TABLE `comisiones` (

`vendedor` int(11),

`comision` float

)

Las comisiones se calculan de una forma especial, le corresponde un porcentaje de las ventas según el tipo de producto, y es importante para los vendedores el que se sepa en cada momento qué comisiones lleva ganadas (esto es una justificación para no usar un *cron* o algo parecido).

Para calcular qué comisión le corresponde a un vendedor, calcularemos los porcentajes para cada tipo de producto vendido (tres tipos de producto) y luego lo añadiremos/actualizaremos en la tabla de comisiones. Todo se realizará en un procedimiento almacenado.

DROP PROCEDURE IF EXISTS comisiones;

delimiter ##

CREATE PROCEDURE comisiones (IN pvendedor INT)

BEGIN

DECLARE total\_1, total\_2, total\_3 INT;

DECLARE existe BOOL;

DECLARE comision\_1, comision\_2, comision\_3, total\_comision INT DEFAULT 0;

select IFNULL(sum(importe),0) into total\_1 from ventas where producto = '1' and vendedor = pvendedor;

SET comision\_1 = total\_1 \* 0.15;

select IFNULL(sum(importe),0) into total\_2 from ventas where producto = '2' and vendedor = pvendedor;

SET comision\_2 = total\_2 \* 0.1;

select IFNULL(sum(importe),0) into total\_3 from ventas where producto = '3' and vendedor = pvendedor;

SET comision\_3 = total\_3 \* 0.2;

SET total\_comision=comision\_1+comision\_2+comision\_3;

select count(vendedor)>0 into existe from comisiones where vendedor = pvendedor;

if existe then

UPDATE comisiones set comision = total\_comision where vendedor = pvendedor;

else

insert into comisiones (vendedor, comision) values (pvendedor, total\_comision);

end if;

END##

Ahora, para actualizar los datos de las comisiones usaremos un trigguer, así cuando se haga una venta (insert en la tabla ventas), se llamará al procedimiento almacenado (*sp\_comisiones*), que recalculará la comisión para ese vendedor.

delimiter ##

DROP TRIGGER IF EXISTS ventas\_insert ##

CREATE TRIGGER ventas\_insert AFTER INSERT on ventas

FOR EACH ROW BEGIN

call comisiones(new.vendedor);

END##

INSERT INTO VENTAS VALUES('1', '2',CURDATE(),1080);

INSERT INTO VENTAS VALUES('2', '2',CURDATE()-1,800);

INSERT INTO VENTAS VALUES('3', '3',CURDATE()+2,120);

INSERT INTO VENTAS VALUES('2', '3',CURDATE()-5,100);

INSERT INTO VENTAS VALUES('3', '1',CURDATE()-9,650);

INSERT INTO VENTAS VALUES('1', '3',CURDATE()+6,798);

SELECT \* FROM COMISIONES;