

```

-- -----
--
-- TPI; ESTUDIO DE LOS ABM PRINCIPALES
-- se presentan los siguientes procedimientos que confirman el
estudio de ABM
-- Enzo Manuel Valenzuela,2021
-- -----
--
--
USE `mydb` ;-- se reconoce la base de datos a utilizar
-- -----
--
-- AUTOMOVILES
-- -----
--
--
--
--
-- AltaVehiculo
-- -----
--
CREATE PROCEDURE `altaVehiculo`(IN modeloId INT, IN
PedidoDetalleId INT)
BEGIN
DECLARE ChasisParam VARCHAR(45);
DECLARE FechaInicio datetime;
SELECT LEFT(MD5(RAND()), 8) into ChasisParam;
Insert INTO automovil(Chasis, Eliminado, pedido_detalle_Id,
pedido_detalle_modelo_Id)
VALUES      (ChasisParam,0, PedidoDetalleId, modeloId);
END;
-- -----
--
-- Delete_Automovil
-- -----
--
CREATE PROCEDURE `delete_automovil` (IN Id_Automovil INT)
BEGIN
update automovil SET
Eliminado = 1,
FechaEliminado = now()
where Id = Id_Automovil;
END;
-- -----
--
-- Creacion_Automoviles
-- -----
--
CREATE PROCEDURE `creacion_automoviles`(IN ParamIdDetalle INT)
BEGIN
DECLARE idModeloParametro INT;
DECLARE nCantidadDetalle INT;

```

```

DECLARE finished INT DEFAULT 0;
DECLARE nInsertados INT;
DECLARE curDetallePedido CURSOR
FOR SELECT modelo_Id, Cantidad_modelo FROM pedido_detalle WHERE
pedido_detalle.Id = ParamIdDetalle;
DECLARE CONTINUE HANDLER FOR SQLSTATE '02000'
SET finished = 1;
DECLARE CONTINUE HANDLER FOR SQLSTATE '23000'
SET finished = 1;
OPEN curDetallePedido;
FETCH curDetallePedido INTO idModeloParametro, nCantidadDetalle;
SET nInsertados = 0;
WHILE nInsertados < nCantidadDetalle DO
    call altaVehiculo(idModeloParametro, ParamIdDetalle);
    SET nInsertados = nInsertados +1;
END WHILE;
CLOSE curDetallePedido;

END;
-- -----
--
-- creacion_Automoviles_de_Pedido
-- -----
--
CREATE PROCEDURE `creacion_Automoviles_de_Pedido`(IN
ParamIdPedido INT)
BEGIN
DECLARE idDetallePedido INT;
DECLARE finished INT default 0;
DECLARE curDetallePedido CURSOR FOR
select Id FROM pedido_detalle WHERE pedido_Id = ParamIdPedido;
DECLARE CONTINUE HANDLER FOR NOT FOUND SET finished = 1;
OPEN curDetallePedido;
getModelo: LOOP
FETCH curDetallePedido INTO idDetallePedido;
IF finished = 1 THEN
    LEAVE getModelo;
END IF;
call creacion_automoviles(idDetallePedido);
END LOOP getModelo;
CLOSE curDetallePedido;
END;
-- -----
--
--
--
-- CONSECIONARIO
-- -----
--
--
--
--
-- Alta_Consecionario

```

```

-- -----
---
CREATE DEFINER=`root`@`localhost` PROCEDURE
`alta_consecionario`(IN nombre VARCHAR(45))
BEGIN
if consecionariosRepetidos(nombre) = 0 THEN
    insert into consecionaria(Nombre, Eliminado) VALUES
        (nombre, 0);
    end if;
END;
-- -----
---
-- CantConcesionario
-- -----
---
CREATE DEFINER=`root`@`localhost` PROCEDURE
`CantConsecionarios`(IN nombreC VARCHAR(45), OUT resultado INT)
BEGIN
declare valor int;
DECLARE c CURSOR FOR
select Count(Nombre) from consecionaria
WHERE Nombre = nombreC;
OPEN c;
FETCH c into resultado;
CLOSE c;
END;
-- -----
---
-- ConsecionariosRepetidos
-- -----
---
CREATE DEFINER=`root`@`localhost` FUNCTION
`consecionariosRepetidos`(nombre VARCHAR(45)) RETURNS int(11)
BEGIN
declare b INT;
call CantConsecionarios(nombre,b);
return b;
END;
-- -----
---
-- Delete_Consecionario
-- -----
---
CREATE DEFINER=`root`@`localhost` PROCEDURE
`delete_consecionario`(IN Id_Cons INT)
BEGIN
update consecionaria SET
Eliminado = 1,
FechaEliminado = now()
where Id = Id_Cons;
END;
-- -----

```

```

---
-- Update_Conseccionario
-- -----
---
CREATE DEFINER=`root`@`localhost` PROCEDURE
`update_conseccionario`(id INT ,nombreNuevo VARCHAR(45))
BEGIN
if conseccionariosRepetidos(nombreNuevo) = 0 THEN
    update conseccionaria c
    set Nombre = nombreNuevo
    where c.Id = id;
END IF;
END;
-- -----
-----
----
-- PEDIDO_DETALLE
-- -----
-----
----
-- -----
---
-- Alta_Pedido_Detalle
-- -----
---
CREATE DEFINER=`root`@`localhost` PROCEDURE
`alta_pedido_detalle`(IN id_pedido INT , IN Cantidad INT, IN
modelo_Id INT)
BEGIN

insert into pedido_detalle(modelo_Id, modelo_Id1,
Cantidad_modelo, pedido_Id, Eliminado)
VALUES

                (modelo_Id, modelo_Id, Cantidad, id_pedido, 0)

;

END;
-- -----
---
-- Delete_PedidoDetalle
-- -----
---
CREATE DEFINER=`root`@`localhost` PROCEDURE
`delete_PedidoDetalle`(IN ParamPedido_detalle INT)
BEGIN
DECLARE finished int default 0;
DECLARE Id_Automovil INT;
DECLARE C cursor for
select Id from automovil where pedido_detalle_Id =
ParamPedido_detalle;
DECLARE CONTINUE HANDLER FOR NOT FOUND SET finished = 1;
update pedido_detalle

```

```

SET Eliminado = 1,
FechaEliminado = now()
WHERE Id = ParamPedido_detalle;
OPEN C;
delAuto: LOOP
FETCH C into Id_Automovil;
IF finished = 1 THEN
    LEAVE delAuto;
END IF;
call delete_automovil(Id_Automovil);
END LOOP delAuto;
CLOSE C;
END;

-- -----
-- -----
----
-- PEDIDO
-- -----
-- -----
----
-- -----
---
-- AltaPedido
-- -----
---
CREATE DEFINER=`root`@`localhost` PROCEDURE `altapedido`(IN
Id_Cons INT)
BEGIN
insert into pedido(consecionaria_Id, FechaDeVenta, Eliminado )
values
(Id_Cons, now(), 0);
END;

-- -----
---
-- Delete_Pedido
-- -----
---
CREATE DEFINER=`root`@`localhost` PROCEDURE `delete_Pedido`(IN
ParamPedido INT)
BEGIN
DECLARE finished int default 0;
DECLARE Id_PedidoDetalle INT;
DECLARE C cursor for
select Id from pedido_detalle where pedido_Id = ParamPedido;
DECLARE CONTINUE HANDLER FOR NOT FOUND SET finished = 1;
update pedido
SET Eliminado = 1,
FechaEliminado = now()
WHERE Id = ParamPedido;
OPEN C;
delDetalle: LOOP
FETCH C into Id_PedidoDetalle;

```

```
IF finished = 1 THEN
    LEAVE delDetalle;
END IF;
call delete_PedidoDetalle(Id_PedidoDetalle);
END LOOP delDetalle;
CLOSE C;
END;
-- -----
-- -----
----
```