**PARTE 1 - (c) QUERY COMPLEJO**

SELECT c1.lname+','+c1.fname AS CLIENTE,

COALESCE(SUM(i1.total\_price\*i1.quantity),0) AS totalCompra,

ref.lname + ', ' + ref.fname AS Referido,

ref.totalCompraReferido \* 0.05 AS totalComision

FROM customer c1 LEFT JOIN orders o1 ON (c1.customer\_num=o1.customer\_num)

LEFT JOIN items i1 ON (o1.order\_num=i1.order\_num)

LEFT JOIN (SELECT c2.customer\_num,

c2.lname,

c2.fname,

SUM(i2.total\_price \* i2.quantity) AS totalCompraReferido,

c2.customer\_num\_referedBy

FROM customer c2 JOIN orders o2 ON (c2.customer\_num = o2.customer\_num)

JOIN items i2 ON (i2.order\_num=o2.order\_num)

WHERE c2.customer\_num\_referedBy IS NOT NULL

AND i2.stock\_num IN (1,4,5,6,9)

GROUP BY c2.customer\_num,

c2.lname,

c2.fname,

c2.customer\_num\_referedBy

) AS Ref ON (Ref.customer\_num\_referedBy=c1.customer\_num)

GROUP BY c1.customer\_num, c1.lname, c1.fname,

Ref.customer\_num, Ref.lname, Ref.fname,

Ref.totalCompraReferido

ORDER BY 1

**PARTE 2 - (e) TRIGGER**

CREATE TRIGGER dbo.customerTr

ON dbo.v\_Productos

INSTEAD OF INSERT AS

BEGIN

DECLARE @codCliente SMALLINT,

@nombre VARCHAR(15),

@apellido VARCHAR(15),

@codProvincia CHAR(2),

@fechaLlamado DATETIME,

@usuarioId CHAR(32),

@codTipoLlamada CHAR(1),

@descrLlamada VARCHAR(40),

@descrTipoLlamada VARCHAR(30)

DECLARE contactos\_cur CURSOR FOR

SELECT \* FROM inserted

OPEN contactos\_cur

FETCH NEXT FROM contactos\_cur

INTO @codCliente, @nombre, @apellido, @codProvincia, @fechaLlamado, @usuarioId, @codTipoLlamada, @descrLlamada, @descrTipoLlamada;

WHILE @@FETCH\_STATUS = 0 BEGIN

BEGIN TRY

BEGIN TRANSACTION

IF @codCliente IS NULL OR @codProvincia IS NULL OR @codTipoLlamada IS NULL OR @fechaLlamado IS NULL BEGIN

RAISERROR ('El código de cliente, la provincia, el tipo de llamada y la fecha de llamado son OBLIGATORIAS.', -- Message text. 6, -- Severity. 1 -- State.);

END

IF @codTipollamada NOT IN ('B','D','I','L','O') BEGIN

RAISERROR ('Codigo de Llamada inválido.', 16, 1)

END

IF NOT EXISTS (SELECT 1 FROM dbo.state where code = @codProvincia) BEGIN

RAISERROR ('Codigo de Provincia inexistente.', 16, 1);

END

IF NOT EXISTS (SELECT 1 FROM dbo.customer c WHERE c.customer\_num = @codCliente) BEGIN

INSERT INTO CUSTOMER (customer\_num, fname, lname, state)

VALUES (@codCliente, @nombre, @apellido, @codProvincia)

END

IF NOT EXISTS (SELECT 1 FROM call\_type ct WHERE ct.call\_code = @codTipoLlamada) BEGIN

INSERT INTO call\_type (call\_code, code\_descr)

VALUES (@codTipoLlamada, @descrTipoLlamada);

END

INSERT INTO cust\_calls (customer\_num, call\_dtime, user\_id, call\_code, call\_descr)

VALUES (@codCliente, @fechaLlamado, @usuarioId, @codTipoLlamada, @descrLlamada)

COMMIT TRANSACTION

END TRY

BEGIN CATCH

ROLLBACK TRANSACTION

PRINT 'hubo un error' -- esto es para debug

END CATCH

FETCH NEXT FROM contactos\_cur

INTO @codCliente, @nombre, @apellido, @codProvincia, @fechaLlamado, @usuarioId, @codTipoLlamada, @descrLlamada, @descrTipoLlamada;

END

CLOSE contactos\_cur

DEALLOCATE contactos\_cur

END