**PRÁCTICA 11.**

**Modelo relacional y objeto relacional, para la gestión de ventas.**

**Veremos cómo Oracle permite proporcionar una solución relacional y otra objeto-relacional.**

**Modelo lógico para una base de datos relacional**

**CLIENTE** (idcliente,nombre,dirección,población,cdopostal,provincia,nif,tel1,tel2,tel3)

**PRODUCTOS(**idproducto,descripción,pvp,stockactual)

**VENTAS(**idventas, idcliente,fechaventa)

**LINEASVENTAS(**idventa,numerolinea,idproducto,cantidad)

1. **Diseña el script para el modelo relacional**

**Modelo lógico para una base de datos objeto-relacional**

1. **Diseña el modelo objeto-relacional**

Tipo **tip\_telefonos** será un **varray** de 3 elementos varchar(15).

CREATE OR REPLACE TYPE tip\_telefonos AS VARRAY(3) OF VARCHAR(15);

/

Tipo **tip\_direccion**

calle varchar(50)

poblacion varchar(50)

codpos varchar(20)

provincia varchar(40)

CREATE OR REPLACE TYPE tip\_direccion AS OBJECT(

calle varchar(50),

poblacion varchar(50),

codpos varchar(20),

provincia varchar(40)

);

/

Tipo **tip\_cliente**

idcliente number

nombre varchar(50)

direc tip\_dirección

nif varchar(9)

telef tip\_telefonos

CREATE OR REPLACE TYPE tip\_cliente AS OBJECT(

idcliente number,

nombre varchar(50),

direc tip\_direccion,

nif varchar(9),

telef tip\_telefonos

);

/

Tipo **tip\_producto**

idproducto number

descripción varchar(80)

pvp number

stockactual number

CREATE OR REPLACE TYPE tip\_producto AS OBJECT(

idproducto number,

descripcion varchar(80),

pvp number,

stockactual number

);

Tipo **tip\_linea\_venta**

numerolinea number

idproducto REF tip\_producto

cantidad number

CREATE OR REPLACE TYPE tip\_lineaventa AS OBJECT(

numerolinea number,

idproducto REF tip\_producto,

cantidad number

);

/

Creamos un tipo tabla anidada para contener la líneas de una venta:

Tipo tabla anidada **tip\_lineas\_venta** de tipo tip\_lineaventa.

CREATE OR REPLACE TYPE tip\_linea\_venta AS TABLE OF tip\_lineaventa;

/

Creamos un tipo venta **tip\_venta** para los datos de las ventas, cada venta tendrá un atributo **Lineas** del tipo tabla anidada definida anteriormente.

CREATE OR REPLACE TYPE tip\_venta AS OBJECT(

idventa number,

idcliente ref tip\_cliente,

fechaventa date,

lineas tip\_linea\_venta,

member function total\_venta return number);

/

En el tipo venta se definirá una función miembro **total\_venta** que calcula el total de la venta de las líneas de venta que forman parte de una venta. ( líneas.COUNT devuelve el número de líneas de una venta)

Tipo **tip\_venta**

Idventa number

Idcliente REF tip\_cliente,

Fechaventa date,

**Líneas** **tip\_lineas\_venta**,

Member function total\_venta retorna number

1. **Crea el cuerpo para la función total\_venta,**que calcula el total de la venta de las líneas de venta que forman parte de una venta.

( Líneas.COUNT devuelve el número de líneas de una venta)

(Al ser una función miembro trabaja sobre objetos y no sobre las tablas, por lo que los select que se necesiten se harán del tipo from dual)

CREATE OR REPLACE TYPE BODY tip\_venta AS

MEMBER FUNCTION total\_venta return NUMBER IS

TOTAL NUMBER:=0;

LINEA TIP\_LINEAVENTA;

PRODUCT TIP\_PRODUCTO;

BEGIN

FOR I IN 1..LINEAS.COUNT LOOP

LINEA:=LINEAS(I);

SELECT DEREF(LINEA.IDPRODUCTO) INTO PRODUCT FROM DUAL;

TOTAL:=TOTAL+LINEA.CANTIDAD\*PRODUCT.PVP;

END LOOP;

return TOTAL;

END;

END;

/

1. **Persiste los objetos**

**Crear las tablas:**

**Tabla\_clientes** de tipo tip\_clientes con una clave primaria idcliente

CREATE TABLE tabla\_clientes OF tip\_cliente(

idcliente PRIMARY KEY

);

**Tabla\_productos** de tipo tip\_producto con una clave primaria idproducto

CREATE TABLE tabla\_productos OF tip\_producto(

idproducto PRIMARY KEY

);

**Tabla\_ventas** de tipo tip\_ventascon una clave primaria idventa

CREATE TABLE tabla\_ventas OF tip\_venta(

idventa PRIMARY KEY

) NESTED TABLE LINEAS STORE AS TABLA\_LINEAS;

1. **Inserta datos en las tablas**

**Inserta estos datos 2 clientes y 5 productos:**

(1,'Luis Garcia’, tip\_direccion('calle Las Flores,23’,’Guadalajara’,’19003’,’Guadalajara’),’34343434L’,tip\_telefonos(‘949876655’,’949876655’))

(2,'ana Serrano, tip\_direccion('calle Galiana,6,’Guadalajara’,’19004’,’Guadalajara’),’76767667F’,tip\_telefonos(‘94980009’))

(1, 'caja de cristal de murano',100,5)

(2, 'bicicleta city',120,15)

(3, '100 lapices de colores',20,5)

(4, 'ipad',600,5)

(5, 'ordenador portatil',400,10)

INSERT INTO tabla\_clientes VALUES(1,'Luis Garcia',tip\_direccion('calle Las Flores, 23','Guadalajara','19003','Guadalajara'),'34343434L', tip\_telefonos('94986665','94986666'));

INSERT INTO tabla\_clientes VALUES(2,'Ana Serrano',tip\_direccion('calle Galiana, 6','Guadalajara','19004','Guadalajara'),'76767676F', tip\_telefonos('94980009'));

INSERT INTO tabla\_productos VALUES(1,'Caja de cristal de murano', 100, 5);

INSERT INTO tabla\_productos VALUES(2,'Bicicleta City', 120, 15);

INSERT INTO tabla\_productos VALUES(3,'100 lapices de colores', 20, 5);

INSERT INTO tabla\_productos VALUES(4,'IPad', 600, 5);

INSERT INTO tabla\_productos VALUES(5,'Ordenador portatil', 400, 10);

En la tabla tabla\_ventas se define una tabla anidada para el atributo LINEAS de tipo tip\_venta, contendrá las líneas de venta.

Se puede hacer de dos formas, por pasos o de una sola vez:

En dos pasos:

**Inserto la venta con idventa 1 para el cliente idcliente 1:**

INSERT INTO TABLA\_VENTAS SELECT 1, REF(C),SYSDATE, TIP\_LINEAS\_VENTA() FROM TABLA\_CLIENTES C WHERE C.IDCLIENTE=1;

**Inserto en la tabla\_ventas dos líneas de venta para el idventa 1, para los productos de línea 1 la cantidad es 1 y para los productos de línea 2 la cantidad es 2**

**Líneas para la venta 1 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

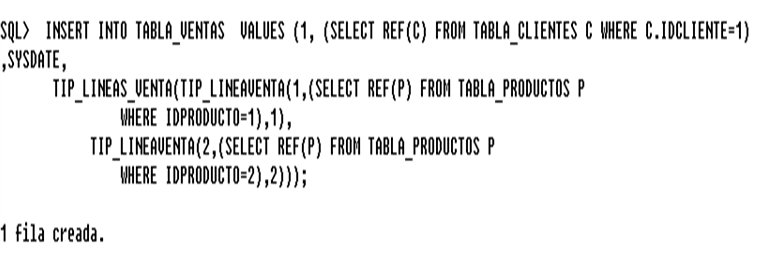
**Linea1**

INSERT INTO TABLE (SELECT V.LINEAS FROM TABLA\_VENTAS V WHERE V.IDVENTA=1)(SELECT 1,REF(P),1 FROM TABLA\_PRODUCTOS P WHERE IDPRODUCTO=1);

**Línea 2**

INSERT INTO TABLE (SELECT V.LINEAS FROM TABLA\_VENTAS V WHERE V.IDVENTA=1)(SELECT 2,REF(P),2 FROM TABLA\_PRODUCTOS P WHERE IDPRODUCTO=2);

En un solo paso:



**Inserto la venta con idventa 2 para el cliente idcliente 1:**

En dos pasos:

INSERT INTO TABLA\_VENTAS SELECT 2, REF(C),SYSDATE, TIP\_LINEAS\_VENTA() FROM TABLA\_CLIENTES C WHERE C.IDCLIENTE=1;

**Líneas para la venta 2 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

Linea1

INSERT INTO TABLE (SELECT V.LINEAS FROM TABLA\_VENTAS V WHERE V.IDVENTA=2)(SELECT 1,REF(P),2 FROM TABLA\_PRODUCTOS P WHERE IDPRODUCTO=4);

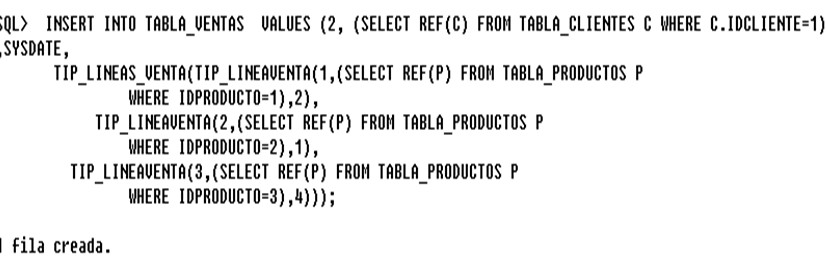
Línea 2

INSERT INTO TABLE (SELECT V.LINEAS FROM TABLA\_VENTAS V WHERE V.IDVENTA=2)(SELECT 2,REF(P),1 FROM TABLA\_PRODUCTOS P WHERE IDPRODUCTO=1);

Línea 3

INSERT INTO TABLE (SELECT V.LINEAS FROM TABLA\_VENTAS V WHERE V.IDVENTA=2)(SELECT 3,REF(P),4 FROM TABLA\_PRODUCTOS P WHERE IDPRODUCTO=5);

En un solo paso:



INSERT INTO tabla\_ventas values(1,(select ref (C) from tabla\_clientes C where C.idcliente=1),sysdate,tip\_linea\_venta(tip\_lineaventa(1,(select ref(P) from tabla\_productos P where idproducto=1),1),tip\_lineaventa(2,(select ref(P) from tabla\_productos P where idproducto=2),2)));

INSERT INTO tabla\_ventas values(2,(select ref (C) from tabla\_clientes C where C.idcliente=1),sysdate,tip\_linea\_venta(tip\_lineaventa(1,(select ref(P) from tabla\_productos P where idproducto=1),2),tip\_lineaventa(2,(select ref(P) from tabla\_productos P where idproducto=2),1),tip\_lineaventa(3,(select ref(P) from tabla\_productos P where idproducto=3),4)));

1. **Consultas**

**6.1 Visualizar todas las líneas de venta para la venta id 2.**

**Haz la consulta de otra forma sin sacar el nombre del objeto, usando TABLE.**

SELECT v.lineas FROM tabla\_ventas v where v.idventa=2;

select lin.\* from tabla\_ventas v, table(v.lineas) lin where v.idventa=2;

* 1. **Visualizar todas las líneas de venta para la venta id 2, obteniendo los productos en vez de su oid.**

select lin.numerolinea,lin.cantidad,deref(lin.idproducto) from tabla\_ventas v,table(v.lineas) lin where v.idventa=2;

* 1. **Visualizar todas las líneas de venta de todas las ventas.**

select idventa,lin.\* from tabla\_ventas v, table(v.lineas) lin;

* 1. **Consulta el nombre del cliente idcliente 2**

select nombre from tabla\_clientes where idcliente=2;

* 1. **Modifica el nombre del cliente 2 por Rosa Serrano**

update tabla\_clientes set nombre='Rosa Serrano' where idcliente=2;

* 1. **Consulta la dirección del cliente 2 y modifica la calle por calle Estopa,34**

select c.direc from tabla\_clientes c where idcliente=2;

update tabla\_clientes set direc=tip\_direccion('calle Estopa, 34', 'Guadalajara','19004','Guadalajara') where idcliente=2;

--update tabla\_clientes set direc.calle='calle Estopa, 34' where direc.calle='calle Galiana, 6';

* 1. **Consulta todos los datos del cliente 1 y añade un nuevo teléfono a su lista de teléfonos.**

**Haz la consulta de otra forma usando value**

select \* from tabla\_clientes where idcliente=1;

select value(c) from tabla\_clientes c where idcliente=1;

update tabla\_clientes set telef=tip\_telefonos('94986665','94986666','90000000') where idcliente=1;

* 1. **Visualiza el nombre del cliente que ha realizado la venta 2.**

**Haz la misma consulta usando DEREF**

select v.idcliente.nombre from tabla\_ventas v where v.idventa=2;

select DEREF(v.idcliente).nombre from tabla\_ventas v where v.idventa=2;

* 1. **Visualiza todos los datos del cliente anterior, que ha realizado la venta 2**

select DEREF(v.idcliente) from tabla\_ventas v where v.idventa=2;

* 1. **Visualizar el numero de venta y el total de ventas hechas por el cliente 1**

select v.idventa, v.total\_venta() from tabla\_ventas v where v.idcliente.idcliente=1;

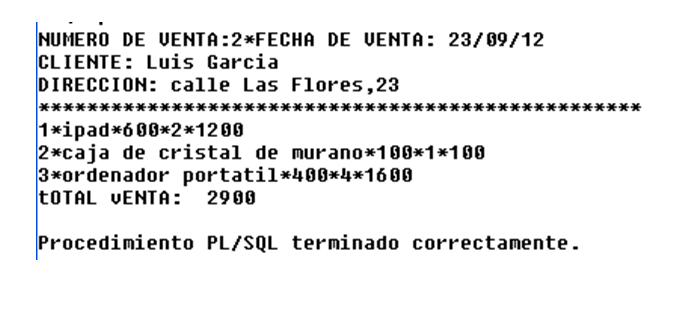
select v.idventa,v.total\_venta() from tabla\_ventas v where deref(v.idcliente).idcliente=1;

* 1. **Visualiza las ventas de todos los clientes**

SELECT v.idventa, deref(idcliente), v.fechaventa from tabla\_ventas v;

* 1. **Crea un procedimiento que reciba como parámetro un id de venta y visualice los datos de la venta cuyo identificador recibe.**

**La salida será como la siguiente:**



CREATE OR REPLACE PROCEDURE VER\_VENTA(ID NUMBER) AS

LIN NUMBER;

CANT NUMBER;

IMPORTE NUMBER;

TOTAL\_V NUMBER;

PRODUC TIP\_PRODUCTO:=TIP\_PRODUCTO(NULL,NULL,NULL,NULL);

CLI TIP\_CLIENTE:=TIP\_CLIENTE(NULL,NULL,NULL,NULL,NULL);

DIR TIP\_DIRECCION:=TIP\_DIRECCION(NULL,NULL,NULL,NULL);

FEC DATE;

CURSOR C1 IS SELECT lin.numerolinea, DEREF(lin.idproducto), lin.cantidad

from tabla\_ventas v, TABLE(v.lineas) lin WHERE v.idventa=ID;

BEGIN

SELECT DEREF(idcliente),fechaventa, v.total\_venta() into CLI, FEC, TOTAL\_V

FROM tabla\_ventas v where idventa=ID;

DIR:=CLI.DIREC;

DBMS\_OUTPUT.PUT\_LINE('NUMERO DE VENTA:'||ID||'\*FECHA DE VENTEA: '||FEC);

DBMS\_OUTPUT.PUT\_LINE('CLIENTE: '||CLI.NOMBRE);

DBMS\_OUTPUT.PUT\_LINE('DIRECCION: '||DIR.CALLE);

DBMS\_OUTPUT.PUT\_LINE('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

OPEN C1;

FETCH C1 INTO LIN, PRODUC, CANT;

WHILE C1%FOUND LOOP

IMPORTE:=CANT\*PRODUC.PVP;

DBMS\_OUTPUT.PUT\_LINE(LIN||'\*'||PRODUC.DESCRIPCION||'\*'||PRODUC.PVP||'\*'||CANT||'\*'||IMPORTE);

FETCH C1 INTO LIN,PRODUC,CANT;

END LOOP;

CLOSE C1;

DBMS\_OUTPUT.PUT\_LINE('TOTAL VENTA: '||TOTAL\_V);

END;

/

BEGIN

VER\_VENTA(2);

END;

/