

# 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,cdpostal,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

2. Diseña el modelo objeto-relacional

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

Tipo tip\_direccion

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

Tipo tip\_producto

idproducto number

descripción varchar(80)

pvp number

stockactual number

Tipo tip\_linea\_venta

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.

Creamos un tipo venta tip\_venta para los datos de las ventas,

cada

venta tendrá un atributo Lineas del tipo tabla anidada definida anteriormente.

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

3. 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)

4. Persiste los objetos

Crear las tablas:

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

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

Tabla\_ventas de tipo tip\_ventas con una clave primaria idventa

5. 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)
```

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 \*\*\*\*\*

Línea 1

```
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);
```

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

## 6. 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.

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

6.3 Visualizar todas las líneas de venta de todas las ventas.

6.4 Consulta el nombre del cliente idcliente 2

6.5 Modifica el nombre del cliente 2 por Rosa Serrano

6.6 Consulta la dirección del cliente 2 y modifica la calle por calle Estopa,34

6.7 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

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

Haz la misma consulta usando Deref

6.9 Visualiza todos los datos del cliente anterior, que ha realizado la venta 2

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

6.11 Visualiza las ventas de todos los clientes

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

```

1 CLEAR SCR;
2 --Creación tablas
3
4 DROP TABLE LINEASVENTA;
5 DROP TABLE VENTAS;
6 DROP TABLE CLIENTE;
7 DROP TABLE PRODUCTOS;
8
9 CREATE TABLE CLIENTE (
10 ID_CLIENTE NUMBER(6) PRIMARY KEY,
11 NOMBRE VARCHAR(20),
12 DIRECCION VARCHAR2(30),
13 POBLACION VARCHAR(20),
14 CP VARCHAR2(20),
15 PROVINCIA VARCHAR2(20),
16 NIF VARCHAR2(20),
17 TELEFONO1 NUMBER(10),
18 TELEFONO2 NUMBER(10),
19 TELEFONO3 NUMBER(10)
20 );
21
22 CREATE TABLE PRODUCTOS(
23 ID_PRODUCTO NUMBER(6) PRIMARY KEY,
24 DESCRIPCION VARCHAR2(30),
25 PVP NUMBER(9),
26 STOCK_ACTUAL NUMBER(9)
27 );
28

```

```

28
29 CREATE TABLE VENTAS (
30 ID_VENTAS NUMBER(6),
31 ID_CLIENTE NUMBER(6),
32 FECHA_VENTA DATE,
33 CONSTRAINT FK_ID_CLIENTE FOREIGN KEY (ID_CLIENTE)
34 REFERENCES CLIENTE (ID_CLIENTE)
35 );
36
37 CREATE TABLE LINEASVENTA (
38 ID_VENTA NUMBER(6),
39 NUMERO_LINEA NUMBER(6),
40 ID_PRODUCTO VARCHAR2(6),
41 CANTIDAD NUMBER(9),
42 CONSTRAINT PK_VENTA_NUMEROLINEA PRIMARY KEY (ID_VENTA,NUMERO_LINEA),
43 CONSTRAINT FK_NUMERO_LINEA FOREIGN KEY (NUMERO_LINEA)
44 REFERENCES PRODUCTOS(ID_PRODUCTO)
45 );
46
47

```

```

48 --Diseño modelo objeto - relacional
49
50 DROP TABLE TABLA_VENTAS;
51 DROP TABLE TABLA_PRODUCTOS;
52 DROP TABLE TABLA_CLIENTES;
53 DROP TYPE TIP_VENTA;
54 DROP TYPE TIP_LINEAS_VENTA;
55 DROP TYPE TIP_LINEA_VENTA;
56 DROP TYPE TIP_PRODUCTO;
57 DROP TYPE TIP_CLIENTE;
58 DROP TYPE TIP_DIRECCION;
59 DROP TYPE TIP_TELEFONOS;
60
61 CREATE TYPE TIP_TELEFONOS AS VARRAY(3) OF VARCHAR2(15);
62 /
63
64 CREATE TYPE TIP_DIRECCION AS OBJECT (
65 CALLE VARCHAR2(50),
66 POBLACION VARCHAR2(50),
67 COD_POS VARCHAR2(20),
68 PROVINCIA VARCHAR2(40)
69 );
70 /

```

```

72 CREATE TYPE TIP_CLIENTE AS OBJECT (
73 ID_CLIENTE NUMBER(6),
74 NOMBRE VARCHAR2(50),
75 DIREC TIP_DIRECCION,
76 NIF VARCHAR2(9),
77 TELEF TIP_TELEFONOS
78 );
79 /
80
81 CREATE TYPE TIP_PRODUCTO AS OBJECT (
82 ID_PRODUCTO NUMBER(6),
83 DESCRIPCION VARCHAR2(80),
84 PVP NUMBER(9),
85 STOCK_ACTUAL NUMBER(9)
86 );
87 /
88
89 CREATE TYPE TIP_LINEA_VENTA AS OBJECT (
90 NUMERO_LINEA NUMBER(9),
91 ID_PRODUCTO REF TIP_PRODUCTO,
92 CANTIDAD NUMBER(9)
93 );
94 /
95

```

```

96 --CREACION DE TABLAS OBJETO - TYPES
97 CLEAR SCR;
98
99 CREATE TYPE TIP_LINEAS_VENTA AS TABLE OF TIP_LINEA_VENTA;
100 /
101
102 CREATE TYPE TIP_VENTA AS OBJECT (
103 ID_VENTA NUMBER(6),
104 ID_CLIENTE REF TIP_CLIENTE,
105 FECHA_VENTA DATE,
106 LINEAS TIP_LINEAS_VENTA,
107 MEMBER FUNCTION TOTAL_VENTA RETURN NUMBER
108 )
109 /
110
111 CREATE OR REPLACE TYPE BODY TIP_VENTA AS
112 MEMBER FUNCTION TOTAL_VENTA RETURN NUMBER IS
113
114     TOTAL NUMBER:=0;
115     LINEA TIP_LINEA_VENTA;
116     PRODUCTO TIP_PRODUCTO;
117
118 BEGIN
119
120     FOR I IN 1..LINEAS.COUNT LOOP
121         LINEA:=LINEAS(I);
122         SELECT Deref(LINEA.ID_PRODUCTO) INTO PRODUCTO FROM DUAL;
123         TOTAL:= TOTAL + LINEA.CANTIDAD * PRODUCTO.PVP;
124     END LOOP;
125     RETURN TOTAL;
126 END;
127 END;
128 /

```

```

130 --Creación de tablas de objetos
131 CL SCR;
132
133 CREATE TABLE TABLA_CLIENTES OF TIP_CLIENTE(
134 ID_CLIENTE PRIMARY KEY
135 );
136
137 CREATE TABLE TABLA_PRODUCTOS OF TIP_PRODUCTO(
138 ID_PRODUCTO PRIMARY KEY
139 );
140
141 CREATE TABLE TABLA_VENTAS OF TIP_VENTA(
142 ID_VENTA PRIMARY KEY
143 ) NESTED TABLE LINEAS STORE AS TIP_VENTAS;
144

```

```

145 --INSERTS
146 CL SCR;
147
148 INSERT INTO TABLA_CLIENTES VALUES (1,'LUIS GARCÍA',TIP_DIRECCION
149                                     (
150                                     'CALLE LAS FLORES,23',
151                                     'GUADALAJARA',
152                                     '82738',
153                                     'GUADALAJARA'
154                                     ),
155                                     '11111111C',
156                                     TIP_TELEFONOS
157                                     (
158                                     '11111111',
159                                     '22222221'
160                                     ));
161
162 INSERT INTO TABLA_CLIENTES VALUES (2,'QUE CHAVAL',TIP_DIRECCION
163                                     (
164                                     'CALLE LAS CAOBA,23',
165                                     'MADRID',
166                                     '82738',
167                                     'MADRID'
168                                     ),
169                                     '11111112C',
170                                     TIP_TELEFONOS
171                                     (
172                                     '11111112',
173                                     '22222222'
174                                     ));

```

```

176 INSERT INTO TABLA_PRODUCTOS VALUES (1,'CAJA DE CRISTAL DE MURANO',100,5);
177 INSERT INTO TABLA_PRODUCTOS VALUES (2,'BICICLETA',120,15);
178 INSERT INTO TABLA_PRODUCTOS VALUES (3,'LAPICES',20,5);
179 INSERT INTO TABLA_PRODUCTOS VALUES (4,'REPRODUCTOR',600,5);
180 INSERT INTO TABLA_PRODUCTOS VALUES (5,'LAPTOP',400,10);
181
182 --TABLA ANIDADA
183
184 INSERT INTO TABLA_VENTAS SELECT 1,REF(C),SYSDATE,TIP_LINEAS_VENTA() FROM TABLA_CLIENTES C WHERE C.ID_CLIENTE=1;
185
186 INSERT INTO TABLA_VENTAS VALUES(2,(SELECT REF(C) FROM TABLA_CLIENTES C
187 WHERE C.ID_CLIENTE=1),SYSDATE, TIP_LINEAS_VENTA(
188 TIP_LINEA_VENTA(1,(SELECT REF(P) FROM TABLA_PRODUCTOS P WHERE ID_PRODUCTO=1),1),
189 TIP_LINEA_VENTA(2,(SELECT REF(P) FROM TABLA_PRODUCTOS P WHERE ID_PRODUCTO=2),2)));
190
191 INSERT INTO TABLA_VENTAS VALUES(3,(SELECT REF(C) FROM TABLA_CLIENTES C
192 WHERE C.ID_CLIENTE=1),SYSDATE, TIP_LINEAS_VENTA(
193 TIP_LINEA_VENTA(1,(SELECT REF(P) FROM TABLA_PRODUCTOS P WHERE ID_PRODUCTO=1),2),
194 TIP_LINEA_VENTA(2,(SELECT REF(P) FROM TABLA_PRODUCTOS P WHERE ID_PRODUCTO=2),1),
195 TIP_LINEA_VENTA(3,(SELECT REF(P) FROM TABLA_PRODUCTOS P WHERE ID_PRODUCTO=3),4)));
196

```

```

197 --Consultas
198 CL SCR;
199
200 --Consulta 6.1
201 SELECT DIR.NUMERO_LINEA FROM TABLA_VENTAS X, TABLE(X.LINEAS) DIR WHERE ID_VENTA=2;
202
203 --Consulta 6.2
204 SELECT DIR.NUMERO_LINEA,DEREF(DIR.ID_PRODUCTO) FROM TABLA_VENTAS X, TABLE(X.LINEAS) DIR WHERE ID_VENTA=2;
205
206 --Consulta 6.3
207 SELECT ID_VENTA,DIR.NUMERO_LINEA FROM TABLA_VENTAS X, TABLE(X.LINEAS) DIR;
208
209 --Consulta 6.4
210 SELECT NOMBRE FROM TABLA_CLIENTES WHERE ID_CLIENTE=2;
211
212 --Consulta 6.5
213 UPDATE TABLA_CLIENTES SET NOMBRE='ROSA SERRANO' WHERE ID_CLIENTE=2;
214
215 --UPDATE 6.6
216 UPDATE TABLA_CLIENTES SET DIREC=TIP_DIRECCION('CALLE ESTOPA, 23','MADRID','282888','MADRID') WHERE ID_CLIENTE=2;
217
218 --UPDATE 6.7
219 SELECT * FROM TABLA_CLIENTES WHERE ID_CLIENTE=1;
220 UPDATE TABLA_CLIENTES SET TELEF= TIP_TÉLEFONOS('9284728472','111111112','222222222') WHERE ID_CLIENTE=1;
221 SELECT VALUE(C) FROM TABLA_CLIENTES C WHERE ID_CLIENTE=1;
222
223 --Consulta 6.8
224 SELECT Deref(ID_CLIENTE).NOMBRE FROM TABLA_VENTAS X WHERE ID_VENTA=2;
225
226 --Consulta 6.9
227 SELECT Deref(ID_CLIENTE) FROM TABLA_VENTAS X WHERE ID_VENTA=2;

```

```

229 --Consulta 6.10
230 SELECT X.ID_VENTA,X.TOTAL_VENTA() FROM TABLA_VENTAS X WHERE Deref(X.ID_CLIENTE).ID_CLIENTE=1;
231
232 --Consulta 6.11
233 SELECT Deref(ID_CLIENTE).NOMBRE,ID_VENTA FROM TABLA_VENTAS X;
234

```

```

235 --PROCEDIMIENTO 6.12
236 CREATE OR REPLACE PROCEDURE PROCE12
237 (V_ID_VENTA IN TABLA_VENTAS.ID_VENTA%TYPE)
238 IS
239
240     V_VENTAS TIP_VENTA;
241     V_CLIENTE TIP_CLIENTE;
242     CURSOR C1 IS SELECT DIR.NUMERO_LINEA,DEREF(DIR.ID_PRODUCTO).DESCRIPCION,
243     Deref(DIR.ID_PRODUCTO).PVP,DIR.CANTIDAD
244     FROM TABLA_VENTAS X, TABLE(X.LINEAS) DIR
245     WHERE ID_VENTA=V_ID_VENTA;
246     V_NUMEROLINEA NUMBER;
247     V_DESCRIPCION VARCHAR2(30);
248     V_PVP NUMBER;
249     V_CANTIDAD NUMBER;
250

```

```

251 BEGIN
252     OPEN C1;
253     SELECT VALUE(E) INTO V_VENTAS FROM TABLA_VENTAS E WHERE ID_VENTA=V_ID_VENTA;
254     SELECT Deref(ID_CLIENTE) INTO V_CLIENTE FROM TABLA_VENTAS WHERE ID_VENTA=V_ID_VENTA;
255
256
257     DBMS_OUTPUT.PUT_LINE('NUMERO DE VENTA: '||V_VENTAS.ID_VENTA||' FECHA DE VENTA: '||V_VENTAS.FECHA_VENTA);
258     DBMS_OUTPUT.PUT_LINE('CLIENTE: '||V_CLIENTE.NOMBRE);
259     DBMS_OUTPUT.PUT_LINE('CLIENTE: '||V_CLIENTE.DIREC.CALLE);
260     DBMS_OUTPUT.PUT_LINE('*****');
261     FETCH C1 INTO V_NUMEROLINEA,V_DESCRIPCION,V_PVP,V_CANTIDAD;
262     WHILE (C1%FOUND) LOOP
263         DBMS_OUTPUT.PUT_LINE(V_NUMEROLINEA||' - '||V_DESCRIPCION||' - '
264         ||V_PVP||' - '||V_CANTIDAD||' - '||
265         (V_CANTIDAD*V_PVP));
266     FETCH C1 INTO V_NUMEROLINEA,V_DESCRIPCION,V_PVP,V_CANTIDAD;
267     END LOOP;
268     DBMS_OUTPUT.PUT_LINE('TOTAL VENTA: '||V_VENTAS.TOTAL_VENTA());
269 END;
270 /
271
272 EXECUTE PROCE12(3);

```