**Desarrollo de aplicaciones multiplataforma**

**Acceso a datos**

**Tema 2-Práctica 10: Tablas anidadas 2**

Supongamos que queremos almacenar la información de clientes, y que hay clientes que tienen 3 móviles (el móvil de su trabajo, su móvil particular y otro móvil de otra compañía que le ofrece buenos descuentos de sms).

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Id\_cli | Nombre | Apellido | Dirección | Población | Provincia | teléfono | Móvil |
| 1 | Francisco | pérez | sol | madrid | madrid | 91345655 | 6564433 |

En el modelo E/R la solución óptima sería crear una relación que sea cliente-movil y que contenga el id\_cliente y su número de móvil.

|  |  |
| --- | --- |
| Id\_cli | móvil |
| 1 | 654555555 |
| 1 | 666543211 |
| 1 | 699767676 |

Las BBDD orientadas a objetos brindan la posibilidad de crear un tipo de objeto.

1. Crear un tipo objeto llamado telefono con dos atributos

Tipo varchar(30)

Numero number

**RESPUESTA:**

CREATE TYPE telefono AS OBJECT (

Tipo VARCHAR2(30),

Numero NUMBER

);

1. Crear una tabla tipo llamada listin basada en el tipo objeto, para añadir la funcionalidad de múltiples valores.

**RESPUESTA:**

CREATE TYPE tabla\_telefonos AS TABLE OF telefono;

CREATE TABLE listin (

Id NUMBER,

Nombre VARCHAR2(50),

Telefonos tabla\_telefonos

) NESTED TABLE Telefonos STORE AS t\_telefonos\_listin;

1. Crear la tabla clientes con los campos

Id\_cli number

Nombre varchar(30)

Apellido varchar(30)

Dirección varchar(30)

Población varchar(30)

Provincia varchar(30)

Telefonos

Siendo teléfonos de tipo listin, añade la tabla anidada teléfonos que se almacena como tel\_tab.

**RESPUESTA:**

-- Crear la tabla clientes con la columna Telefonos de tipo listin

CREATE TABLE clientes (

Id\_cli NUMBER,

Nombre VARCHAR2(30),

Apellido VARCHAR2(30),

Dirección VARCHAR2(30),

Población VARCHAR2(30),

Provincia VARCHAR2(30),

Telefonos tabla\_telefonos

) NESTED TABLE Telefonos STORE AS tel\_tab;

1. Inserta 3 registros, con 3 teléfonos cada uno.

**RESPUESTA:**

-- Insertar el primer cliente

INSERT INTO clientes VALUES (

1,

'Juan',

'Pérez',

'Calle A, 123',

'Ciudad A',

'Provincia A',

tabla\_telefonos(

telefono('Personal', 111111111),

telefono('Trabajo', 222222222),

telefono('Casa', 333333333)

)

);

-- Insertar el segundo cliente

INSERT INTO clientes VALUES (

2,

'María',

'López',

'Calle B, 456',

'Ciudad B',

'Provincia B',

tabla\_telefonos(

telefono('Personal', 444444444),

telefono('Trabajo', 555555555),

telefono('Casa', 666666666)

)

);

-- Insertar el tercer cliente

INSERT INTO clientes VALUES (

3,

'Pedro',

'Gómez',

'Calle C, 789',

'Ciudad C',

'Provincia C',

tabla\_telefonos(

telefono('Personal', 777777777),

telefono('Trabajo', 888888888),

telefono('Casa', 999999999)

)

);

1. Selecciona todos los clientes.

**RESPUESTA:**

SELECT c.Id\_cli, c.Nombre, c.Apellido, c.Dirección, c.Población, c.Provincia,

t.Tipo AS Tipo\_Telefono, t.Numero AS Numero\_Telefono

FROM clientes c, TABLE(c.Telefonos) t;

1. Consulta las estructuras de almacenamiento que usa oracle para almacenar los objetos.

**RESPUESTA:**

SELECT object\_name, object\_type, created, last\_ddl\_time

FROM user\_objects

WHERE object\_name IN ('CLIENTES', 'TEL\_TAB');

1. Consulta los objetos de la base de datos.

**RESPUESTA:**

SELECT object\_name, object\_type, created, last\_ddl\_time

FROM user\_objects;

1. Lista la vista user\_nested\_tables.

**RESPUESTA:**

SELECT object\_name, object\_type, created, last\_ddl\_time

FROM user\_objects

WHERE object\_type = 'TABLE';

1. Lista todos los teléfonos del cliente 3, usando el operador TABLE.

**RESPUESTA:**

SELECT c.Id\_cli, t.Tipo, t.Numero

FROM clientes c, TABLE(c.Telefonos) t

WHERE c.Id\_cli = 3;

1. Actualiza la tabla clientes cambiando los números de teléfono del cliente 1 por :

'fijo',934444444

'movil personal',65555555

'movilempresa',644444444

**RESPUESTA:**

MERGE INTO clientes c

USING (SELECT 1 AS Id\_cli FROM dual) d

ON (c.Id\_cli = d.Id\_cli)

WHEN MATCHED THEN

UPDATE SET c.Telefonos = tabla\_telefonos(

telefono('fijo', 934444444),

telefono('movil personal', 65555555),

telefono('movilempresa', 644444444)

);

1. Visualizar todos los teléfonos de todos los clientes.

**RESPUESTA:**

SELECT c.Id\_cli, t.Tipo, t.Numero

FROM clientes c, TABLE(c.Telefonos) t;

1. Visualizar el nombre, id , tipo de teléfono, número de teléfono de todos los teléfonos de todos los clientes

**RESPUESTA:**

SELECT c.Id\_cli, c.Nombre, t.Tipo, t.Numero

FROM clientes c, TABLE(c.Telefonos) t;