**Práctica 9. Tablas anidadas2**

**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.**

**Crear un tipo objeto llamado telefono con dos atributos**

**Tipo varchar(30)**

**Numero number**

**CREATE OR REPLACE TYPE TELEFONO AS OBJECT(tipo varchar2(30), numero number);**

**/**

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

**CREATE OR REPLACE TYPE listin AS TABLE OF TELEFONO;**

**/**

**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.**

**CREATE TABLE clientes(**

**id\_cliente number,**

**nombre varchar2(30),**

**apellido varchar2(30),**

**direccion varchar2(30),**

**poblacion varchar2(30),**

**provincia varchar2(30),**

**telefonos listin) nested table telefonos store as tel\_tab;**

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

**INSERT INTO clientes VALUES(1,'Paco','Perez','sol','madrid','madrid',listin(TELEFONO('MOVIL',633883394),TELEFONO('FIJO',938392034),TELEFONO('MOVIL',647409234)));**

**INSERT INTO clientes VALUES(2,'Ramirez','Kevin','sol','madrid','madrid', listin(TELEFONO('MOVIL',633865394),TELEFONO('FIJO',932342034),TELEFONO('MOVIL',647429394)));**

**INSERT INTO clientes VALUES(3,'Jesus','Garcia','sol','madrid','madrid', listin(TELEFONO('MOVIL',633834394),TELEFONO('FIJO',938309434),TELEFONO('MOVIL',640939494)));**

**Selecciona todos los clientes**

**SELECT \* FROM clientes;**

**Consulta las estructuras de almacenamiento que usa oracle para**

**almacenar los objetos**

**select object\_name,object\_type,status from dba\_objects where object\_name like '%TELEFONO%';**

**Consulta los objetos de la base de datos**

**SELECT SEGMENT\_NAME,SEGMENT\_TYPE FROM DBA\_SEGMENTS WHERE SEGMENT\_NAME LIKE '%TELEFONO%';**

**Lista la vista user\_nested\_tables**

**select \* from user\_nested\_tables;**

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

**select cli.numero, cli.tipo from clientes c, TABLE(c.telefonos) cli where c.id\_cliente=3;**

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

**'fijo',934444444**

**'movil personal',65555555**

**'movilempresa',644444444**

**update clientes set telefonos = listin(TELEFONO('FIJO',934444444),TELEFONO('MOVIL PERSONAL',655555555),TELEFONO('MOVIL EMPRESA',644444444)) where id\_cliente=1;**

**Visualizar todos los teléfonos de todos los clientes**

**SELECT cli.numero, cli.tipo from clientes c, TABLE(c.telefonos) cli;**

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

**SELECT id\_cliente, nombre, cli.numero, cli.tipo from clientes c, TABLE(c.telefonos) cli;**