

Tarea 4 para AD05_2024-2025

Apertura: miércoles, 29 de enero de 2025, 00:00

Fecha de entrega: jueves, 20 de febrero de 2025, 23:59

Tarea 4 para AD05.

EJERCICIO 1

Crear una base de datos llamada **DBJefeHijo** con DB40 con la siguiente información:

// código para crear el método que crea la base de datos

```
File fichero=new File("baseDatos");
fichero.delete();
ObjectContainer baseDatos=Db4oEmbedded.openFile("BDJefeHijo");
baseDatos.store(new Jefe("Ángel", 5, 53, new Hijo("Gustavo", 7)));
baseDatos.store(new Jefe("Nieves", 3, 45, new Hijo("Iván", 3)));
baseDatos.store(new Jefe("Jesús", 3, 5, new Hijo("Noelia", 3)));
baseDatos.store(new Jefe("Dolores", 5,63, new Hijo("Sergio", 7)));
baseDatos.store(new Jefe("Vicki", 3, 5, null));
baseDatos.store(new Jefe("Fátima", 5, 63, new Hijo("Lidia", 27)));
baseDatos.store(new Jefe("Juan Luís", 3, 5, null));
baseDatos.store(new Jefe("Elena", 1, 42, new Hijo("David", 19)));
baseDatos.store(new Jefe("Miguel", 20,45, new Hijo("Paula", 3)));
baseDatos.store(new Jefe("Jesús", 19, 44, new Hijo("Rubén", 12)));
baseDatos.close();
```

Implementar los siguientes puntos en el *main* (con un menú de opciones) para la realización de las siguientes consultas:

1. Crear base de datos (con el código indicado anteriormente).
2. Visualizar toda la base de datos. (0.8 puntos)
3. Visualizar los jefes que tengan más de 55 años.(0.8 puntos)
4. Modificar la edad de Miguel incrementando su edad un año más.(0.8 puntos)
5. Borrar los jefes que llevan más de 6 años en la empresa.(0.8 puntos)
6. Visualizar todos los jefes que quedan, incluidos sus hijos.(0.8 puntos)

Puntuación total **4 puntos**.

Tienes el archivo **db4o-8.0.224.15975-all-java5.jar** adjunto al final de la tarea.

EJERCICIO 2

Partiremos de una base de datos para gestionar los pedidos de los clientes, y veremos cómo Oracle permite proporcionar una solución relacional y otra objeto-relacional.

Dado el siguiente modelo de datos:

```

CREATE TABLE CLIENTES(
IDCLIENTE NUMBER PRIMARY KEY,
NOMBRE VARCHAR2(50),
DIRECCION VARCHAR2(50),
POBLACION VARCHAR2(50),
CODPOSTAL NUMBER(5),
PROVINCIA VARCHAR2(40),
NIF VARCHAR2(9) UNIQUE,
TELEFON01 VARCHAR2(15),
TELEFON02 VARCHAR2(15),
TELEFON03 VARCHAR2(15)
);

CREATE TABLE PRODUCTOS(
IDPRODUCTO NUMBER PRIMARY KEY,
DESCRIPCION VARCHAR2(80),
PVP NUMBER,
STOCKACTUAL NUMBER
);

CREATE TABLE VENTAS(
IDVENTA NUMBER PRIMARY KEY,
IDCLIENTE NUMBER NOT NULL REFERENCES CLIENTES,
FECHAVENTA DATE
);

CREATE TABLE LINEASVENTAS(
IDVENTA NUMBER,
NUMEROLINEA NUMBER,
IDPRODUCTO NUMBER,
CANTIDAD NUMBER,
FOREIGN KEY (IDVENTA) REFERENCES VENTAS (IDVENTA),
FOREIGN KEY (IDPRODUCTO) REFERENCES PRODUCTOS (IDPRODUCTO),
PRIMARY KEY (IDVENTA, NUMEROLINEA)
);

```

Se adjunta al final de la tarea el script **Ejercicio2_esquema_inicial.sql**.

Implementar los siguientes puntos en SQLDeveloper:

1. Definir un tipo/TYPE varray de dimensión 3 para contener los teléfonos.
2. Crear los tipos **dirección**, **cliente**, **producto** y **línea de venta**.
3. Crear un tipo tabla anidada para contener las líneas de una venta.
4. Crear un tipo venta para los datos de las ventas, cada venta tendrá un atributo **LINEAS** del tipo tabla anidada definida anteriormente.
5. Crea el cuerpo/BODY del tipo anterior, teniendo en cuenta que se definirá la **función miembro TOTAL_VENTA** que calcula el total de la venta de las líneas de venta que forman parte de una venta, contará el número de elementos de una tabla o de un array y devolverá el número de líneas que tiene la venta.
6. Crear las tablas donde almacenar los objetos de la aplicación. Se creará una tabla para **clientes**, otra para **productos** y otra para las **ventas**, en dichas tablas se definirán las oportunas claves primarias.
7. Inserta dos clientes y cinco productos.
8. Insertar en **TABLA_VENTAS** la venta con **IDVENTA 1** para el **IDCLIENTE 1**.
9. Insertar en **TABLA_VENTAS** dos líneas de venta para el **IDVENTA 1** para los productos 1 (la CANTIDAD es 1) y 2 (la CANTIDAD es 2).
10. Insertar en **TABLA_VENTAS** la venta con **IDVENTA 2** para el **IDCLIENTE 1**.
11. Insertar en **TABLA_VENTAS** tres líneas de venta para el **IDVENTA 2** para los productos 1 (la CANTIDAD es 2), 4 (la CANTIDAD es 1) y 5 (la CANTIDAD es 4)
12. Realizar un procedimiento que recibiendo el identificador visualice los datos de la venta.

Puntuación total 6 **puntos** (0.5 cada apartado).

Se tendrá en cuenta que:

- El funcionamiento correcto de lo que se pide.
- Tratamiento adecuado de posibles excepciones.

Puntuación:

Total 10 puntos.



- **Ejercicio 1.** 4 puntos (1 punto cada apartado).
- **Ejercicio 2.** 6 puntos (0.5 puntos cada apartado).

Indicaciones de entrega:

El envío se realizará a través de la plataforma de la forma establecida para ello, y el archivo comprimido en formato **.zip** se nombrará siguiendo las siguientes pautas:

apellido1_apellido2_nombre_SIGxx_Tarea.zip

Asegúrate que el nombre no contenga la letra ñ, tildes ni caracteres especiales extraños. Así por ejemplo la alumna **Begoña Sánchez Mañas para la cuarta unidad del MP de AD**, debería nombrar esta tarea como **sanchez_manas_begona_AD04_Tarea.zip**

-  [db4o-8.0.224.15975-all-java5.jar](#)
-  [Ejercicio2_esquema_inicial.sql](#)

29 de enero de 2025, 18:06

29 de enero de 2025, 18:06

Agregar entrega

Estado de la entrega

Número del intento	Este es el intento 1 (2 intentos permitidos).
Estado de la entrega	Todavía no se han realizado envíos
Estado de la calificación	No calificada
Tiempo restante	21 días 10 horas restante

 [Contactar con el soporte del sitio](#)

Usted se ha identificado como KEVIN ZAMORA AMELA ([Cerrar sesión](#))

Plan de Modernización de la Formación Profesional. La oferta parcial y nueva de la Formación Profesional a Distancia en Aragón durante el curso 22-23 está financiada por el Ministerio de Educación y Formación Profesional y la Unión Europea NextGenerationEu en el marco del PRTR



