

FAMILIA PROFESIONAL:

CICLOS FORMATIVOS:

MÓDULO:

Informática y Comunicaciones

Desarrollo de Aplicaciones Multiplataforma

Acceso a Datos

UNIDAD 5: BASES DE DATOS XML

ACTIVIDADES



AUTORES: **Fernando Rodríguez Alonso**
Sonia Pasamar Franco

Este documento está bajo licencia Creative Commons Reconocimiento-NoComercial-SinObraDerivada 4.0 Internacional License.

Para ver una copia de esta licencia, visite <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

Usted es libre de:

- **Compartir** — copiar y redistribuir el material en cualquier medio o formato.

El licenciante no puede revocar estas libertades en tanto usted siga los términos de la licencia.

Bajo los siguientes términos:

- **Atribución** — Usted debe dar crédito de manera adecuada, brindar un enlace a la licencia, e indicar si se han realizado cambios. Puede hacerlo en cualquier forma razonable, pero no de forma tal que sugiera que usted o su uso tienen el apoyo de la licenciante.
- **NoComercial** — Usted no puede hacer uso del material con propósitos comerciales.
- **SinDerivadas** — Si remezcla, transforma o crea a partir del material, no podrá distribuir el material modificado.

No hay restricciones adicionales — No puede aplicar términos legales ni medidas tecnológicas que restrinjan legalmente a otras a hacer cualquier uso permitido por la licencia.

BASE DE DATOS pruebas

ACTIVIDAD 1x01

Codifica una clase **Actividad_1x01** que incluya un programa principal **main**. Este programa gestionará los **productos** de la base de datos **pruebas** mediante el siguiente menú de opciones:

0) **Salir del programa.**

1) **Insertar un producto en la base de datos.**

Leerá por teclado el código, la denominación, el precio, el stock actual, el stock mínimo y el código de zona del producto a insertar.

Se conectará a la base de datos **pruebas**, creada en **exist-db**, ejecutará una sentencia XUpdate de modificación para insertar un nuevo producto (con estos datos leídos) en la base de datos y cerrará la conexión.

Si la inserción se ha realizado correctamente, escribirá en consola el mensaje:

Se ha insertado un producto en la base de datos.

Si la inserción no se ha realizado porque se ha encontrado un producto con el código dado, escribirá en consola el mensaje:

Se ha encontrado un producto con ese código en la base de datos.

2) **Consultar todos los productos de la base de datos.**

Se conectará a la base de datos **pruebas**, creada en **exist-db**, ejecutará una sentencia XQuery de consulta para recuperar todos los productos de la base de datos y cerrará la conexión.

Si los resultados de la consulta no contienen ningún producto, escribirá en consola el mensaje:

No se ha encontrado ningún producto en la base de datos.

Si los resultados de la consulta contienen uno o más productos:

- Escribirá en consola un listado de todos los productos almacenados en la base de datos, indicando para cada producto, un resumen con los nombres y los valores de todos los atributos de dicho producto.
- Escribirá en consola el número de productos consultados de la base de datos.

El siguiente ejemplo muestra todos los productos:

```
Producto [Código = 1010, Precio = 50,00,
          Denominación = Placa Base MSI G41M-P26,
          StockActual = 10, StockMínimo = 3, CódigoZona = 10]
Producto [Código = 1011, Precio = 120,00,
          Denominación = Micro Intel Core i5-2320,
          StockActual = 3, StockMínimo = 5, CódigoZona = 10]
Producto [Código = 1012, Precio = 170,00,
          Denominación = Micro Intel Core i5 2500,
          StockActual = 5, StockMínimo = 6, CódigoZona = 20]
Producto [Código = 1013, Precio = 80,00,
          Denominación = HD Seagate Barracuda 250GB SATA,
          StockActual = 10, StockMínimo = 5, CódigoZona = 20]
Producto [Código = 1014, Precio = 150,00,
          Denominación = HD Caviar Blue 500GB SATA3,
          StockActual = 5, StockMínimo = 6, CódigoZona = 30]
```

```

Producto [Código = 1015, Precio = 40,00,
          Denominación = Tarjeta gráfica Asus GeForce EN210 Silent 1GB,
          StockActual = 10, StockMínimo = 5, CódigoZona = 30]
Producto [Código = 1016, Precio = 50,00,
          Denominación = Tarjeta gráfica Gigabyte GeForce 1GB,
          StockActual = 5, StockMínimo = 6, CódigoZona = 40]
Producto [Código = 1017, Precio = 45,00,
          Denominación = Tarjeta gráfica Nvidia Express 1GB,
          StockActual = 10, StockMínimo = 5, CódigoZona = 30]
Producto [Código = 1018, Precio = 60,00,
          Denominación = Micro Intel Dual Core G620,
          StockActual = 15, StockMínimo = 5, CódigoZona = 40]
Producto [Código = 1019, Precio = 10,00,
          Denominación = Memoria DDR3 G.Skill 2GB,
          StockActual = 5, StockMínimo = 3, CódigoZona = 10]
Producto [Código = 1020, Precio = 30,00,
          Denominación = Memoria DDR3 G.Skill 4GB,
          StockActual = 30, StockMínimo = 10, CódigoZona = 10]
Producto [Código = 1021, Precio = 16,00,
          Denominación = Memoria DDR3 Kingston HyperX 4GB,
          StockActual = 15, StockMínimo = 4, CódigoZona = 20]
Producto [Código = 1022, Precio = 52,00,
          Denominación = Placa Base ASRock G41M-S3 ,
          StockActual = 2, StockMínimo = 2, CódigoZona = 30]
Se han consultado 13 productos de la base de datos.

```

3) Consultar un producto, por código, de la base de datos.

Leerá por teclado el código del producto a consultar.

Se conectará a la base de datos **pruebas**, creada en **eXist-db**, ejecutará una sentencia XQuery de consulta para recuperar un jugador, por código, de la base de datos y cerrará la conexión.

Si el resultado de la consulta no contiene ningún producto, escribirá en consola el mensaje:

No se ha encontrado ningún producto con ese código en la base de datos.

Si el resultado de la consulta contiene un producto, escribirá en consola un resumen con los nombres y los valores de todos los atributos de dicho producto.

El siguiente ejemplo muestra el producto cuyo código es 1012:

```

Producto [Código = 1012, Precio = 170,00,
          Denominación = Micro Intel Core i5 2500,
          StockActual = 5, StockMínimo = 6, CódigoZona = 20]

```

4) Actualizar un producto, por código, de la base de datos.

Leerá por teclado el código, la nueva denominación, el nuevo precio, el nuevo stock actual, el nuevo stock mínimo y el nuevo código de zona del producto a actualizar.

Se conectará a la base de datos **pruebas**, creada en **eXist-db**, ejecutará una sentencia XUpdate de modificación para actualizar un producto, por código (reemplazando la denominación, el precio, el stock actual, el stock mínimo y el código de zona), de la base de datos y cerrará la conexión.

Si la actualización se ha realizado correctamente, escribirá en consola el mensaje:

Se ha actualizado un producto de la base de datos.

Si la actualización no se ha realizado porque no se ha encontrado ningún jugador con el código dado, escribirá en consola el mensaje:

No se ha encontrado ningún producto con ese código en la base de datos.

5) Eliminar un producto, por código, de la base de datos.

Leerá por teclado el código del producto a eliminar.

Se conectará a la base de datos **pruebas**, creada en **eXist-db**, ejecutará una sentencia XUpdate de modificación para eliminar un jugador, por código, de la base de datos y cerrará la conexión.

Si la eliminación se ha realizado correctamente, escribirá en consola el mensaje:

Se ha eliminado un producto de la base de datos.

Si la eliminación no se ha realizado porque no se ha encontrado ningún producto con el código dado, escribirá en consola el mensaje:

No se ha encontrado ningún producto con ese código en la base de datos.

Este programa principal deberá validar que la opción de menú elegida sea válida (comprendida entre 0 y 5). Si no lo es, escribirá en consola el mensaje:

La opción de menú debe estar comprendida entre 0 y 5.

ACTIVIDAD 1x02

Codifica una clase **Actividad_1x02** que incluya un programa principal **main**. Este programa gestionará las **zonas** de la base de datos **pruebas** mediante el siguiente menú de opciones:

0) **Salir del programa.**

1) **Insertar una zona en la base de datos.**

Leerá por teclado el nombre y el director de la zona a insertar.

Se conectará a la base de datos **pruebas**, creada en **eXist-db**, ejecutará una sentencia XUpdate de modificación para insertar una nueva zona (con código generado de forma autoincremental) en la base de datos y cerrará la conexión.

Si la inserción se ha realizado correctamente, escribirá en consola el mensaje:

Se ha insertado una zona en la base de datos.

2) **Consultar todas las zonas de la base de datos.**

Se conectará a la base de datos **pruebas**, creada en **eXist-db**, ejecutará una sentencia XQuery de consulta para recuperar todas las zonas de la base de datos y cerrará la conexión.

Si los resultados de la consulta no contienen ninguna zona, escribirá en consola el mensaje:

No se ha encontrado ninguna zona en la base de datos.

Si los resultados de la consulta contienen una o más zonas:

- Escribirá en consola un listado de todas las zonas almacenadas en la base de datos, indicando para cada zona, un resumen con los nombres y los valores de todos los atributos de dicha zona.
- Escribirá en consola el número de zonas consultadas de la base de datos.

El siguiente ejemplo muestra todas las zonas:

```
Zona [Código = 10, Nombre = Galicia, Asturias, Cantabria y País Vasco,
      Director = Pedro Martín]
Zona [Código = 20, Nombre = Castilla León, La Rioja y Navarra,
      Director = Alicia Pérez]
Zona [Código = 30, Nombre = Aragón, Cataluña, Valencia y Baleares,
      Director = Carlos Grau]
Zona [Código = 40, Nombre = Extremadura, Madrid y Castilla La Mancha,
      Director = Antonio González]
Zona [Código = 50, Nombre = Andalucía y Murcia,
      Director = Rocío Gil]
Zona [Código = 60, Nombre = Canarias,
      Director = Isabel Serrano]
Se han consultado 6 zonas de la base de datos.
```

3) **Consultar una zona, por código, de la base de datos.**

Leerá por teclado el código de la zona a consultar.

Se conectará a la base de datos **pruebas**, creada en **eXist-db**, ejecutará una sentencia XQuery de consulta para recuperar una zona, por código, de la base de datos y cerrará la conexión.

Si el resultado de la consulta no contiene ninguna zona, escribirá en consola el mensaje:

No se ha encontrado ninguna zona con ese código en la base de datos.

Si el resultado de la consulta contiene una zona, escribirá en consola un resumen con los nombres y los valores de todos los atributos de dicha zona.

El siguiente ejemplo muestra la zona cuyo código es 30:

```
Zona [Código = 30, Nombre = Aragón, Cataluña, Valencia y Baleares,
      Director = Carlos Grau]
```

4) Actualizar una zona, por código, de la base de datos.

Leerá por teclado el código, el nuevo nombre y el nuevo director de la zona a actualizar.

Se conectará a la base de datos **pruebas**, creada en **eXist-db**, ejecutará una sentencia XUpdate de modificación para actualizar una zona, por código (reemplazando el nombre y el director), de la base de datos y cerrará la conexión.

Si la actualización se ha realizado correctamente, escribirá en consola el mensaje:

Se ha actualizado una zona de la base de datos.

Si la actualización no se ha realizado porque no se ha encontrado ninguna zona con el código dado, escribirá en consola el mensaje:

No se ha encontrado ninguna zona con ese código en la base de datos.

5) Eliminar una zona, por código, de la base de datos.

Leerá por teclado el código de la zona a eliminar.

Se conectará a la base de datos **pruebas**, creada en **eXist-db**, ejecutará una sentencia XQuery de consulta para recuperar los productos, por código de zona, de la base de datos y cerrará la conexión.

Si los resultados de la consulta contienen uno o más productos:

- Escribirá en consola un mensaje indicando el número de productos relacionados con la zona indicada.
- Escribirá en consola los productos relacionados con la zona indicada, indicando para cada producto, un resumen con los nombres y los valores de todos los atributos de dicho producto.

El siguiente ejemplo muestra los productos relacionados con la zona cuyo código es 30:

Se han encontrado 4 productos relacionados con esa zona en la base de datos:

```
Producto [Código = 1014, Precio = 150,00,
          Denominación = HD Caviar Blue 500GB SATA3,
          StockActual = 5, StockMínimo = 6, CódigoZona = 30]
Producto [Código = 1015, Precio = 40,00,
          Denominación = Tarjeta gráfica Asus GeForce EN210 Silent 1GB,
          StockActual = 10, StockMínimo = 5, CódigoZona = 30]
Producto [Código = 1017, Precio = 45,00,
          Denominación = Tarjeta gráfica Nvidia Express 1GB,
          StockActual = 10, StockMínimo = 5, CódigoZona = 30]
Producto [Código = 1022, Precio = 52,00,
          Denominación = Placa Base ASRock G41M-S3 ,
          StockActual = 2, StockMínimo = 2, CódigoZona = 30]
```

Si los resultados de la consulta no contienen ningún producto:

- Se conectará a la base de datos **pruebas**, creada en **eXist-db**, ejecutará una sentencia XUpdate de modificación para eliminar una zona, por código, de la base de datos y cerrará la conexión.
- Si la eliminación se ha realizado correctamente, escribirá en consola el mensaje:
Se ha eliminado una zona de la base de datos.
- Si la eliminación no se ha realizado porque no se ha encontrado ninguna zona con el código dado, escribirá en consola el mensaje:

No se ha encontrado ninguna zona con ese código en la base de datos.

Este programa principal deberá validar que la opción de menú elegida sea válida (comprendida entre 0 y 5). Si no lo es, visualizará en consola el mensaje:

La opción de menú debe estar comprendida entre 0 y 5.