

Examen XQUERY

1. Creamos en BaseX la base de datos de neptuno.

The screenshot shows the BaseX 9.0.1 interface. The left pane displays the file explorer with 'neptunoExamen.xml' selected. The main editor shows the XML structure of the database, including tables like 'articulos', 'clientes', and 'proveedores'. The 'Result' pane shows the command: 'CREATE DB neptunoExamen "/home/dam/DAM1/PrimerosUD/examen query/neptunoExamen.xml";' and the result: 'Database "neptunoExamen" created in 118.98 ms.' The 'Info' pane shows a tree view of the database structure.

Ejercicios

1) Referencia del artículo, referencia del proveedor, Nombre, categoría, precio unidad, y unidades en existencia de todos los artículos que pertenezcan a la categoría bebidas

The screenshot shows the BaseX 9.0.1 interface with the 'Editor' pane active. The query being executed is:

```
1 for $articulo in /neptuno/articulos/articulo
2 where $articulo/NombreCategoria = 'Bebidas'
3 return
4   <articulo>
5     <RefArticulo>{data($articulo/RefArticulo)}</RefArticulo>
6     <ReferenciaProveedor>{data($articulo/ReferenciaProveedor)}</
ReferenciaProveedor>
7     <NombreArticulo>{data($articulo/NombreArticulo)}</
NombreArticulo>
8     <NombreCategoria>{data($articulo/NombreCategoria)}</
NombreCategoria>
9     <PrecioUnidad>{data($articulo/PrecioUnidad)}</PrecioUnidad>
10    <UnidadesExistencia>{data($articulo/UnidadesExistencia)}</
UnidadesExistencia>
11  </articulo>
```

The status bar at the bottom shows 'OK' and '11 : 14'.

The screenshot shows the 'Result' pane of the BaseX 9.0.1 interface. The output of the XQuery query is displayed as XML:

```
<articulo>
  <RefArticulo>30</RefArticulo>
  <ReferenciaProveedor>130</ReferenciaProveedor>
  <NombreArticulo>Fanta</NombreArticulo>
  <NombreCategoria>Bebidas</NombreCategoria>
  <PrecioUnidad>50</PrecioUnidad>
  <UnidadesExistencia>50</UnidadesExistencia>
</articulo>
```

2) Referencia del proveedor, nombre del proveedor, dirección, código postal, país y teléfono de todos los proveedores que sean de Francia

The screenshot shows the BaseX IDE interface. On the left, a file explorer displays the contents of the `/home/dam/basex` directory, including files like `data`, `.basex`, `.basexgui`, `2.xq`, `3.xq`, `asd.xq`, `baile.xml`, and `basex.zip`. The main editor window, titled "Editor", shows the XSLT script `ejercicio1.xq` with the following code:

```
1 for $proveedor in /neptuno/proveedores/proveedor
2 where $proveedor/PaisProveedor = 'Francia'
3 return
4   <proveedor>
5     <ReferenciaProveedor>{data($proveedor/ReferenciaProveedor)}<
6   /ReferenciaProveedor>
7     <NombreProveedor>{data($proveedor/NombreProveedor)}</
8   NombreProveedor>
9     <Direccion>{data($proveedor/DireccionProveedor)}</Direccion>
10    <CodigoPostal>{data($proveedor/CodigoPostal)}</CodigoPostal>
11    <Pais>{data($proveedor/PaisProveedor)}</Pais>
12    <Telefono>{data($proveedor/TelefonoProveedor)}</Telefono>
13  </proveedor>
```

Below the editor, the "Result" pane displays the output XML:

```
<proveedor>
  <ReferenciaProveedor>130</ReferenciaProveedor>
  <NombreProveedor>David</NombreProveedor>
  <Direccion>Les Enfants, 30</Direccion>
  <CodigoPostal>123456</CodigoPostal>
  <Pais>Francia</Pais>
  <Telefono>9876543</Telefono>
</proveedor>
```

3) Referencia del cliente, nombre del cliente, dirección y ciudad de todos los clientes que sean de la ciudad de Madrid

The screenshot shows the BaseX IDE interface. On the left, the file explorer shows the same directory structure as the previous screenshot. The main editor window, titled "Editor", shows the XSLT script `ejercicio2.xq` with the following code:

```
1 for $cliente in /neptuno/clientes/cliente
2 where $cliente/CiudadCliente = 'Madrid'
3 return
4   <cliente>
5     <RefCliente>{data($cliente/RefCliente)}</RefCliente>
6     <NombreCliente>{data($cliente/NombreCliente)}</NombreCliente>
7   >
8     <Direccion>{data($cliente/DireccionCliente)}</Direccion>
9     <Ciudad>{data($cliente/CiudadCliente)}</Ciudad>
10  </cliente>
```

Below the editor, the "Result" pane displays the output XML:

```
<cliente>
  <RefCliente>2</RefCliente>
  <NombreCliente>Pedro</NombreCliente>
  <Direccion>C/ El Laurel, 2</Direccion>
  <Ciudad>Madrid</Ciudad>
</cliente>
```

4)Referencia de los pedidos, referencia del cliente, referencia del artículo y fecha de los pedidos realizados por el cliente con referencia PICCO

5)Referencia de los artículo, nombre de los artículo, referencia del proveedor y nombre del proveedor de todos los productos que sean de la categoría bebidas

The screenshot shows the NetBeans IDE interface. On the left is the 'Project Explorer' showing the file structure of the 'basex' project, including a 'data' directory and various XML and XQuery files. The main 'Editor' window displays the content of 'ejercicio3.xq'. The code is an XQuery that filters articles by category 'Bebidas' and returns product details. The 'Result' window at the bottom shows the output XML, which is a single product entry for 'Fanta'.

```

1 for $articulo in /neptuno/articulos/articulo[NombreCategoria = '
  Bebidas']
2 let $proveedor_ref := $articulo/ReferenciaProveedor
3 let $proveedor := /neptuno/proveedores/proveedor[
  ReferenciaProveedor = $proveedor_ref]
4 return
5   <producto>
6     <RefArticulo>{data($articulo/RefArticulo)}</RefArticulo>
7     <NombreArticulo>{data($articulo/NombreArticulo)}</
  NombreArticulo>
8     <RefProveedor>{data($proveedor/ReferenciaProveedor)}</
  RefProveedor>
9     <NombreProveedor>{data($proveedor/NombreProveedor)}</
  NombreProveedor>
10    </producto>
11

```

The 'Result' window shows the output XML:

```

<producto>
  <RefArticulo>30</RefArticulo>
  <NombreArticulo>Fanta</NombreArticulo>
  <RefProveedor>130</RefProveedor>
  <NombreProveedor>David</NombreProveedor>
</producto>

```

