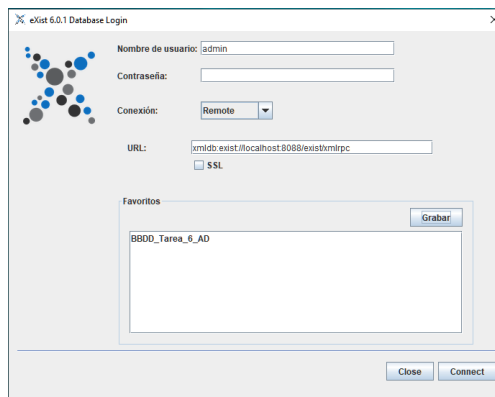
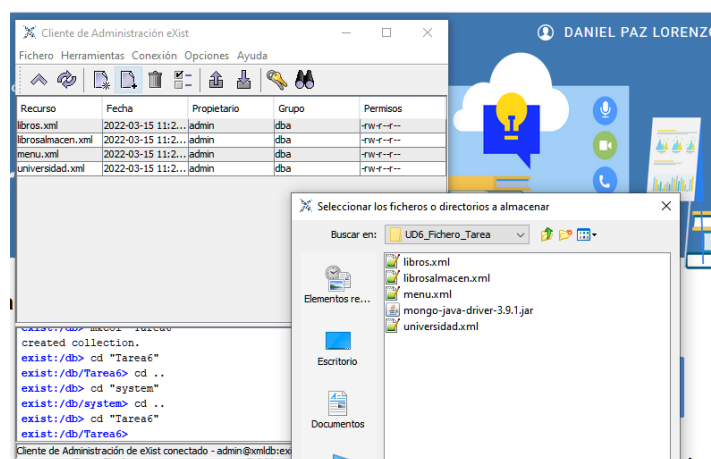


Antes de realizar los ejercicios 1 y 2 me dispongo a instalar en mi ordenador la base de datos “Exist-db” (versión 4.61 como venía en el temario) utilizando como tutorial el punto 3 del temario. Cuando la tenga instalada ejecuto, mediante un lanzador del servicio que se me crea en el escritorio, voy al icono que aparece en la barra de tareas, pincho con botón derecho y selecciono “Open Java admin client”:



Pongo los datos de mi conexión a la bbdd y los grabo para poder reutilizar cada vez que quiera entrar



Ya conectado creo una nueva colección llamada “Tarea 6” y añado los xml propuestos en la tarea (El libros.xml es el actualizado con los “id” y que se ha colgado en el foro de la asignatura)

EJERCICIO 1 – XPATH

Ya tenemos creada la base de datos sobre la que podemos realizar y ejecutar las consultas Xpath que se nos plantean en la tarea. Abrimos un diálogo de consulta y vamos ejecutando una a una cada consulta planteada para que nos muestre el resultado, las consultas de la 21 en adelante las he ejecutado en la página web <http://xpather.com/> porque en Exist no daba ningún error pero no me mostraba resultados :

1. Nombre de la universidad	2. País de la universidad

3. Nombre de las carreras

Diálogo de consulta

Consulta de entrada:

Historial: 1. //universidad/nombre/text()

1 //universidad/carreras/carrera/nombre/text()

Contexto: /db

Resultados:

XML	Traza
1	I.T. Informática
2	Dipl. Empresariales
3	Dipl. Relaciones Laborales
4	Lic. Química
5	Lic. Biología
6	Lic. Humanidades
7	

Hallados 6 resultados. Compilación: 8ms, Ejecución: 26ms Line: 1 Column:1

4. Años de plan de estudio de las carreras

Diálogo de consulta

Consulta de entrada:

Historial: 1. //universidad/nombre/text()

1 //universidad/carreras/carrera/plan/text()

Contexto: /db

Resultados:

XML	Traza
1	2003
2	2001
3	2001
4	2003
5	2001
6	1980
7	

Hallados 6 resultados. Compilación: 5ms, Ejecución: 24ms Line: 1 Column:1

5. Nombres de todos los alumnos

Diálogo de consulta

Consulta de entrada:

Historial: 1. //universidad/nombre/text()

1 //universidad/alumnos/alumno/nombre/text()

Contexto: /db

Resultados:

XML	Traza
1	Víctor Manuel
2	Luisa
3	Fernando
4	María
5	

Hallados 4 resultados. Compilación: 9ms, Ejecución: 30ms Line: 1 Column:1

6. Identificadores de todas las carreras

Diálogo de consulta

Consulta de entrada:

Historial: 9. //universidad/carreras/carrera/@id/data()

1 //universidad/carreras/carrera/@id/data()

Contexto: /db

Resultados:

XML	Traza
1	c01
2	c02
3	c03
4	c04
5	c05
6	c06
7	

Hallados 6 resultados. Compilación: 5ms, Ejecución: 20ms Line: 1 Column:32

7. Datos de la carrera cuyo id es c01

Diálogo de consulta

Consulta de entrada:

Historial: 9. //universidad/carreras/carrera/@id/data()

1 //universidad/carreras/carrera[@id='c01']/data()

Contexto: /db

Resultados:

XML	Traza
1	
2	I.T. Informática
3	2003
4	250
5	Escuela de Informática
6	
7	

Hallados 1 resultados. Compilación: 6ms, Ejecución: 23ms Line: 1 Column:1

8. Centro en que se estudia la carrera cuyo id es c02

Diálogo de consulta

Consulta de entrada:

Historial: 11. //universidad/carreras/carrera[@id='c01']

1 //universidad/carreras/carrera[@id='c02']/centro/data()

Contexto: /db

Resultados:

XML	Traza
1	Facultad de Ciencias Sociales
2	

Hallados 1 resultados. Compilación: 6ms, Ejecución: 22ms Line: 1 Column:49

9. Nombre de las carreras que tengan subdirector

Diálogo de consulta

Consulta de entrada:

Historial: 11. //universidad/carreras/carrera[@id='c01']

1 //universidad/carreras/carrera[subdirector]/nombre/data()

Contexto: /db

Resultados:

XML	Traza
1	Dipl. Relaciones Laborales
2	

Hallados 1 resultados. Compilación: 8ms, Ejecución: 23ms Line: 1 Column:51

10. Nombre de los alumnos que estén haciendo proyecto

Diálogo de consulta

Consulta de entrada:

Historial: 11. //universidad/carreras/carrera[@id='c01']

1 //universidad/alumnos/alumno[estudios/proyecto]/nombre/text()

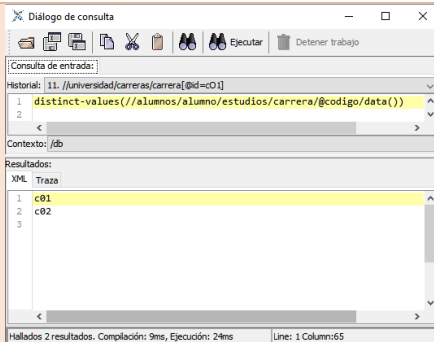
Contexto: /db

Resultados:

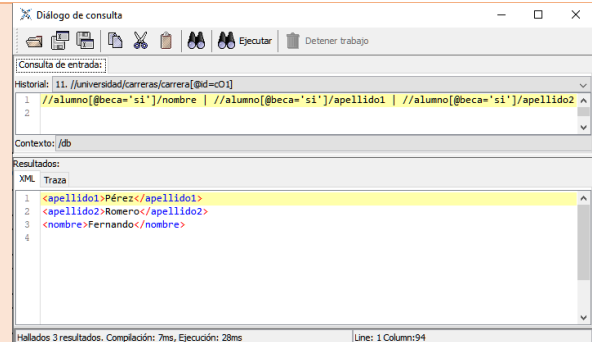
XML	Traza
1	Luisa
2	María
3	

Hallados 2 resultados. Compilación: 8ms, Ejecución: 20ms Line: 1 Column:62

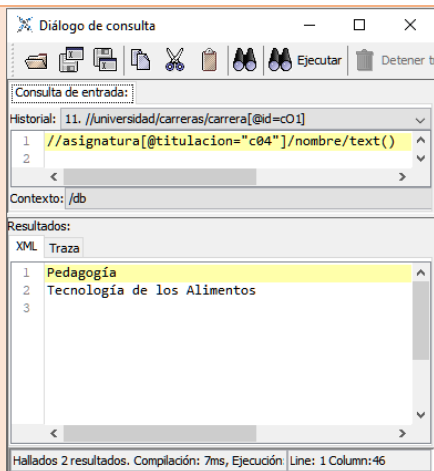
11. Códigos de las carreras en las que hay algún alumno matriculado



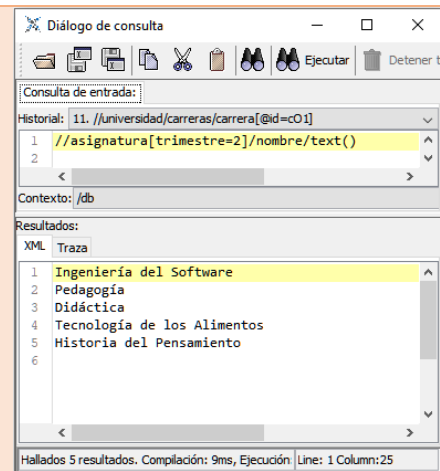
12. Apellidos y Nombre de los alumnos con beca



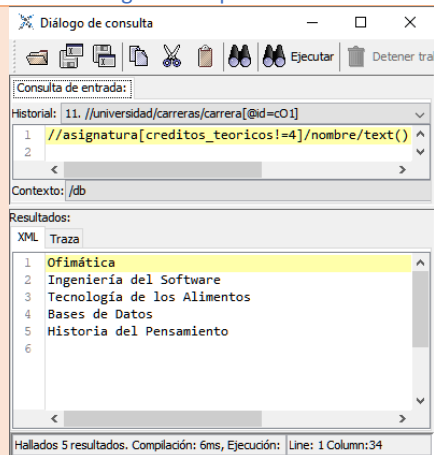
13. Nombre de las asignaturas de la titulación c04



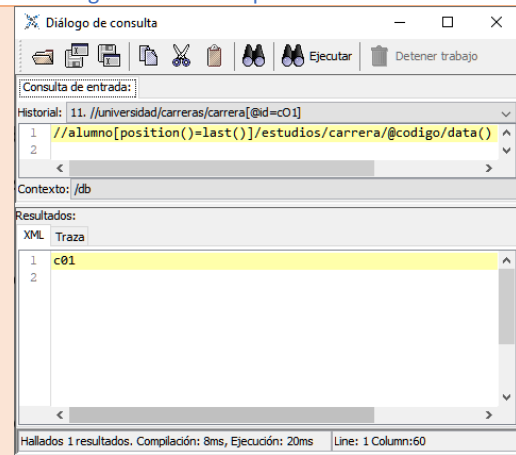
14. Nombre de las asignaturas de segundo trimestre



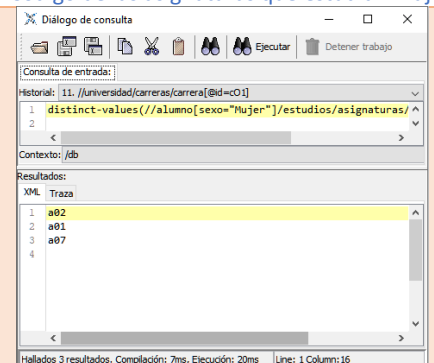
15. Nombre de las asignaturas que no tienen 4 créditos teóricos



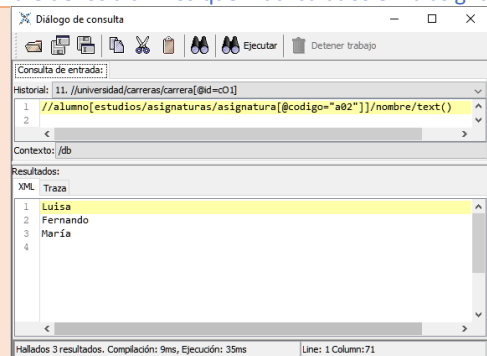
16. Código de la carrera que estudia el último alumno



17. Código de las asignaturas que estudian mujeres



18. Nombre de los alumnos que matriculados en la asignatura a02



19. Códigos de las carreras que estudian los alumnos matriculados en alguna asignatura

20. Apellidos de todos los hombres

21. Nombre de la carrera que estudia Víctor Manuel

22. Nombre de las asignaturas que estudia Luisa

```
//asignatura[&id=//alumno[nombre="Luisa"]/estudios/asignaturas/asignatura[@codigo]nombre]
```

Elements found: 2

OK Cancel

```
<!-- ASIGNATURAS -->

<asignaturas>

  <asignatura id="a01" titulos="01">
    <nombre<oficiante/>nombre>
    <creditos<teoricos></creditos<teoricos>
    <creditos<practicos>1</creditos<practicos>
    <ctsumatorio<ctsumatorio>

  </asignatura>

  <asignatura id="a02" titulos="01">
    <nombre<Ingenieria del Software/>nombre>
    <creditos<teoricos>6</creditos<teoricos>
    <creditos<practicos>1</creditos<practicos>
    <ctsumatorio<ctsumatorio>

  </asignatura>
```

1 Ómnico
2 Ingeniería del Software

23. Primer ejemplo de los alumnos matriculados en Ingeniería del Software

24. Nombre de las carreras que estudian los alumnos matriculados en la asignatura Tecnología de los Alimentos

25. Nombre de los alumnos matriculados en carreras que no tienen subdirector

```
//alumno[/estudios/carrera/@codigo=/carrera[not (subdirector)]]/#id/nombre
```

Elements found: 4

id	nombre
1	Victor Manuel
2	Luisa
3	Fernando
4	María

26. Nombre de los alumnos matriculados en asignaturas con 0 créditos prácticos y que estudien la carrera de I.T. Informática

27. Nombre de los alumnos que estudian carreras cuyos planes son anteriores a 2002

```
//alumno[././carrera[@codigo=//carrera[plan<2002]/@id]]/nombre
```

Elements found: 2

```
<alumno id="e01">
  <apellido1@prase/>apellido1
  <apellido2@searle/>apellido2
  <nombre/>Salazar/nombre
  <sexo/>Mujer/sexo
  <estudiano>
    <carrera codigo="e02"/>
    <signaturas>
      <signatura codigo="e02"/>
      <signatura codigo="e01"/>
    </signaturas>
    <correctivoWeb de IBM.com/>correctivo
  </estudiano>
</alumno>
```

1. Luisa
2. Fernando

EJERCICIO 2 – XQUERY

Como en el apartado anterior voy a ejecutar las consultas que se plantean en la tarea dentro de la base de datos “Exist-db”:

1. Listar el título de todos los libros.

Diálogo de consulta

Consulta de entrada:

```
1 for $libro in //libro return $libro/titulo/text()
```

Historial:

Contexto: /db/Tarea6

Resultados:

XML	Traza
1	TCP/IP Illustrated
2	Advan Programming for Unix environment
3	Data on the Web
4	Economics of Technology for Digital TV
5	

Hallados 4 resultados. Compilación: 6ms, Ejecución: 9ms Line: 1 Column:50

2. Listar año y título de todos los libros, ordenados por el año.

Diálogo de consulta

Consulta de entrada:

```
1 for $libro in //libro order by $libro/año return <libro>{$libro/año}
2 <titulo>{$libro/titulo/text()}</titulo>
3 </libro>
```

Historial:

Contexto: /db/Tarea6

Resultados:

XML	Traza
1	<libro año="1994"><titulo>TCP/IP Illustrated</titulo></libro>
2	<libro año="1992"><titulo>Advan Programming for Unix environment</titulo></libro>
3	<libro año="2000"><titulo>Data on the Web</titulo></libro>
4	<libro año="1999"><titulo> Economics of Technology for Digital TV</titulo></libro>
5	

Hallados 4 resultados. Compilación: 10ms, Ejecución: 12ms Line: 3 Column:27

3. Listar los libros cuyo precio sea 65.95

Diálogo de consulta

Consulta de entrada:

```
1 for $libro in //libro where $libro/precio=65.95 return <libro>{$libro/titulo}</libro>
```

Historial:

Contexto: /db/Tarea6

Resultados:

XML	Traza
1	<libro><titulo>TCP/IP Illustrated</titulo></libro>
2	<libro><titulo>Advan Programming for Unix environment</titulo></libro>
3	

Hallados 2 resultados. Compilación: 8ms, Ejecución: 16ms Line: 1 Column:42

4. Listar los libros publicados antes del año 2000

Diálogo de consulta

Consulta de entrada:

```
1 for $libro in //libro where $libro/@año<2000 return <libro>{$libro/titulo}</libro>
```

Historial:

Contexto: /db/Tarea6

Resultados:

XML	Traza
1	<libro><titulo>TCP/IP Illustrated</titulo></libro>
2	<libro><titulo>Advan Programming for Unix environment</titulo></libro>
3	<libro><titulo> Economics of Technology for Digital TV</titulo></libro>
4	

Hallados 3 resultados. Compilación: 9ms, Ejecución: 8ms Line: 1 Column:45

5. Listar año y título de los libros publicados por Addison-Wesley después del año 1992.

Diálogo de consulta

Consulta de entrada:

```
1 for $libro in //libro where $libro/editorial="Addison-Wesley" and $libro/@año>1992
2 return <libro>{$libro/@año}{ $libro/titulo}</libro>
```

Historial:

Contexto: /db/Tarea6

Resultados:

XML	Traza
1	<libro año="1994"><titulo>TCP/IP Illustrated</titulo></libro>
2	

Hallados 1 resultados. Compilación: 8ms, Ejecución: 9ms Line: 1 Column:62

6. Listar año y título de los libros que tienen más de un autor.

Diálogo de consulta

Consulta de entrada:

```
1 for $libro in //libro where count($libro/autor)>1
2 return <libro>{$libro/@año}{ $libro/titulo}</libro>
```

Historial:

Contexto: /db/Tarea6

Resultados:

XML	Traza
1	<libro año="2000"><titulo>Data on the Web</titulo></libro>
2	

Hallados 1 resultados. Compilación: 6ms, Ejecución: 8ms Line: 1 Column:50

7. Listar año y título de los libros que tienen o no tienen autor.

Diálogo de consulta

Consulta de entrada:

```
1 for $libro in //libro where count($libro/autor)=0
2 return <libro>{$libro/@año}{$libro/titulo}</libro>
```

Contexto: /db/Tarea6

Resultados:

XML	Traza
1	<libro año="1999"><titulo> Economics of Technology for Digital TV</titulo></libro>
2	

Hallados 1 resultados. Compilación: 5ms, Ejecución: 8ms Line: 3 Column: 1

8. Mostrar los apellidos de los autores que aparecen en el documento, sin repeticiones, ordenados alfabéticamente.

Diálogo de consulta

Consulta de entrada:

```
1 for $libro in distinct-values(//libro/autor/apellido) order by $libro ascending
2 return <autor>{$libro}</autor>
```

Contexto: /db/Tarea6

Resultados:

XML	Traza
1	<autor>Abiteboul</autor>
2	<autor>Buneman</autor>
3	<autor>Stevens</autor>
4	<autor>Suciu</autor>
5	

Hallados 4 resultados. Compilación: 8ms, Ejecución: 8ms Line: 1 Column: 1

9. Por cada libro, listar agrupado en un elemento <result> su título y autores

Diálogo de consulta

Consulta de entrada:

```
1 for $libro in //libro
2 return <result>{$libro/titulo}{$libro/autor}</result>
```

Contexto: /db/Tarea6

Resultados:

XML	Traza
1	<result><titulo>TCP/IP Illustrated</titulo><autor>
2	<apellido>Stevens</apellido>
3	<nombre>W.</nombre>
4	</autor></result>
5	<result><titulo>Advan Programming for Unix environment</titulo><autor>
6	<apellido>Stevens</apellido>
7	<nombre>W.</nombre>
8	</autor></result>
9	<result><titulo>Data on the Web</titulo><autor>
10	<apellido>Abiteboul</apellido>
11	<nombre>Serge</nombre>
12	</autor><autor>
13	<apellido>Buneman</apellido>
14	<nombre>Peter</nombre>
15	</autor><autor>
16	<apellido>Suciu</apellido>
17	<nombre>Dan</nombre>
18	</autor></result>
19	<result><titulo> Economics of Technology for Digital TV</titulo></result>
20	

Hallados 4 resultados. Compilación: 22ms, Ejecución: 8ms Line: 1 Column: 1

10. Por cada libro, obtener su título y el número de autores, agrupados en un elemento <libro>

Diálogo de consulta

Consulta de entrada:

```
1 for $libro in //libro
2 return <libro>{$libro/titulo}{count($libro/autor)}</libro>
```

Contexto: /db/Tarea6

Resultados:

XML	Traza
1	<libro><titulo>TCP/IP Illustrated</titulo>1</libro>
2	<libro><titulo>Advan Programming for Unix environment</titulo>1</libro>
3	<libro><titulo>Data on the Web</titulo>3</libro>
4	<libro><titulo> Economics of Technology for Digital TV</titulo>0</libro>
5	

Hallados 4 resultados. Compilación: 6ms, Ejecución: 9ms Line: 1 Column: 1

11. Una lista ordenada alfabéticamente de categorías de libros comprados.

Diálogo de consulta

Consulta de entrada:

```
1 for $libro in //libro/@id, $almacen in //comprados[codigo=$libro]
2 order by $libro/./titulo ascending
3 return <comprados><libro>{$libro/./titulo}</libro></comprados>
```

Contexto: /db/Tarea6

Resultados:

XML	Traza
1	<comprados><libro><titulo>Advan Programming for Unix environment</titulo>
2	<comprados><libro><titulo>TCP/IP Illustrated</titulo></libro></comprados>
3	

Hallados 2 resultados. Compilación: 5ms, Ejecución: 9ms Line: 3 Column: 63

12. Obtener la suma del importe de todos los libros que están pendientes.

Diálogo de consulta

Consulta de entrada:

```
1 <suma_libros_pendientes> {sum(
2 for $libro in //libro/@id, $almacen in //pendientes[codigo=$libro]
3 return $libro/./precio
4 )} </suma_libros_pendientes>
```

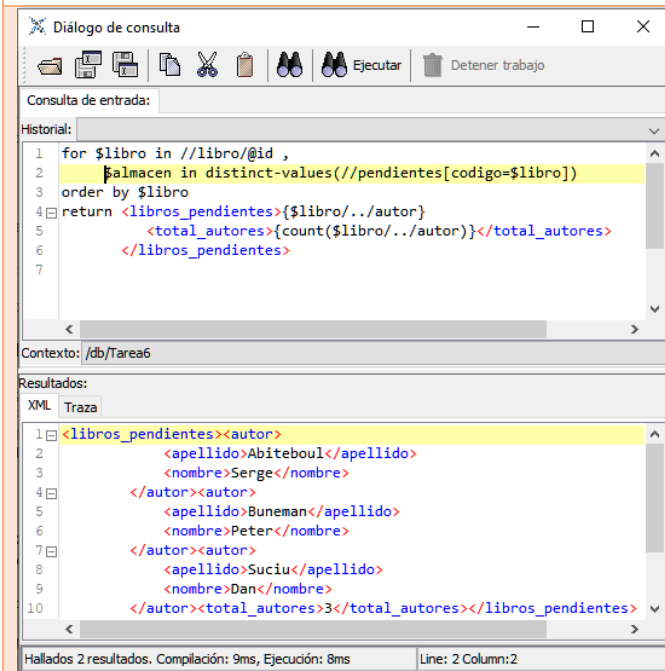
Contexto: /db/Tarea6

Resultados:

XML	Traza
1	<suma_libros_pendientes>169.89999999999998</suma_libros_pendientes>
2	

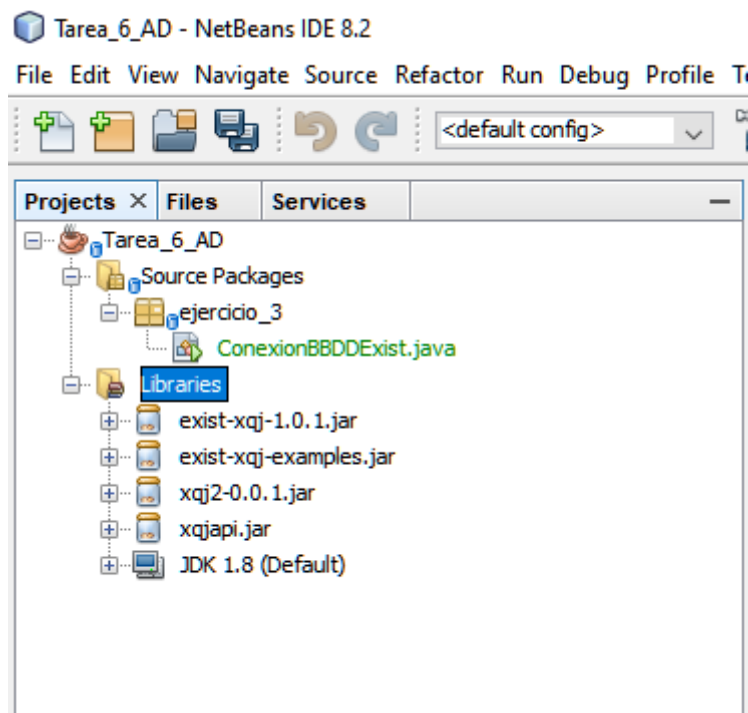
Hallados 1 resultados. Compilación: 14ms, Ejecución: 23ms Line: 5 Column: 1

13. Una lista ordenada de autores que tengan libros pendientes. La última línea contendrá una línea que tenga el total de autores.



EJERCICIO 3

Abrimos NetBeans y creamos un nuevo proyecto (Tarea_6_AD), a continuación le añadimos las librerías de “Exist XQJ API” que contienen las interfaces para comunicar el proyecto con la base de datos de Exist-db que tengo instalada en mi pc:



El siguiente paso es crear el código, para ello he reutilizado el que se nos propone en el apartado 3.7.2 del temario. Cambiando las propiedades de usuario y contraseña, adaptándolo a mi puerto de conexión con la base de datos (8088) y creando una nueva sentencia XQuery para que nos muestre todos los libros contenidos en el fichero “libroActualizado.xml”, el código final quedaría de la siguiente manera:

```

14  */
15  public class ConexionBBDDExist {
16
17      /**
18       * Método principal sin argumentos
19       *
20       * @param args the command line arguments
21       */
22
23      public static void main(String[] args) throws XQException {
24          //Creamos las instancias XQDataSource, XQConnection necesarias para
25          //la conexión con la bbdd
26          XQDataSource ds = new ExistXQDataSource();
27          ds.setProperty("serverName", "localhost");
28          ds.setProperty("port", "8088");
29          ds.setProperty("user", "admin");
30          ds.setProperty("password", "");
31          XQConnection con = ds.getConnection();
32
33          //Creamos la consulta Xquery
34          String query = "for $librosActualizado in //bib/libro return <libro>{$librosActualizado/titulo}</libro>";
35          //Creamos una instancia de XQPreparedExpression con las propiedades de conexión y la consulta
36          XQPreparedExpression expr = con.prepareExpression(query);
37          //Ejecutamos la consulta y recogemos resultado
38          XQSequence result = expr.executeQuery();
39          //Imprime por pantalla el resultado de la consulta
40          System.out.println(result.getSequenceAsString(null));
41          //Cerramos todas las conexiones
42          result.close();
43          expr.close();
44          con.close();
45      }
46  }
47

```

Si lo ejecuto muestra por pantalla lo siguiente:

```

Output X
Asignaturas-2-DAM - D:\Curso DAM\Repositorios GitHub\Asignaturas-2-DAM x Tarea_6_AD (run) x

run:
<libro>
  <titulo xmlns="">TCP/IP Illustrated</titulo>
</libro> <libro>
  <titulo xmlns="">Advan Programming for Unix environment</titulo>
</libro> <libro>
  <titulo xmlns="">Data on the Web</titulo>
</libro> <libro>
  <titulo xmlns=""> Economics of Technology for Digital TV</titulo>
</libro>
BUILD SUCCESSFUL (total time: 0 seconds)

```

Comprobamos que se nos muestran los títulos de los 4 libros contenidos en el fichero “librosActualizado.xml”