

# Técnico Superior en Desarrollo de Aplicaciones Multimedia

Curso Académico 2021/2022 - 2º DAMS

Acceso a Datos: Actividad Tarea Online Unidad 5

Bases de datos XML

Alumno: José Enrique Jordán Moreno

## Tabla de contenido

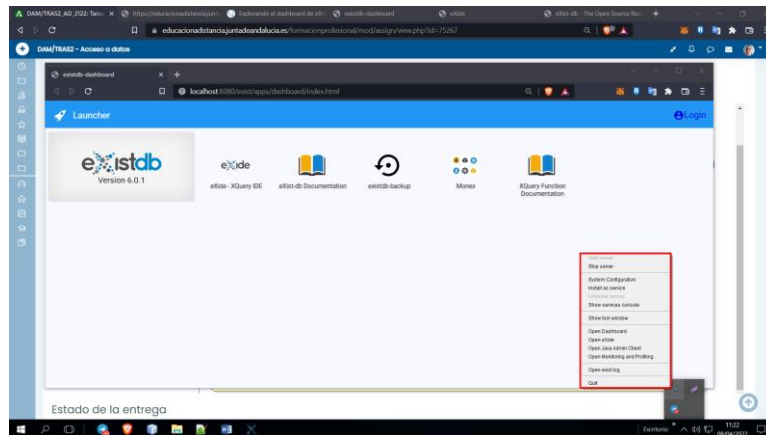
<b>EJERCICIO 1 .....</b>	<b>2</b>
Utilizando la base de datos colecciones.xml, extrae las colecciones 'Libros', 'Prestamos' y 'Autores' .....	2
<b>EJERCICIO 2. ....</b>	<b>8</b>
Realiza estas consultas a la BD colecciones.xml y ejecútalas desde una aplicación Java. Guardar cada consulta en un script.xq, que llamarás desde tu programa.....	8
1. Título de los libros que se hayan publicado en 2002, y ordénalos alfabéticamente.....	8
2. Título de los libros que posean más de 1 autor.....	9
3. Título, primer autor y páginas de los libros prestados. ....	10
4. Nombre de cada lector y suma total de páginas de libros que le han prestado.....	11
<b>EJERCICIO 3. ....</b>	<b>12</b>
Crea estas consultas de actualización .xq para la BD colecciones.xml y comprueba su resultado desde una aplicación Java:.....	12
Insertar el nodo evaluación y cuyo contenido sea 10 como último nodo en la publicacion=2007 .....	12
Sustituir el valor del nodo paginas con 700 en la publicacion=2005.....	13
Modificar el nombre del nodo inicio de cada documento de la colección Préstamos por fechainicio.....	13
Eliminar el nodo dirección de la tabla Préstamos.....	14
Código Java.....	14
Referencias.....	14

## EJERCICIO 1

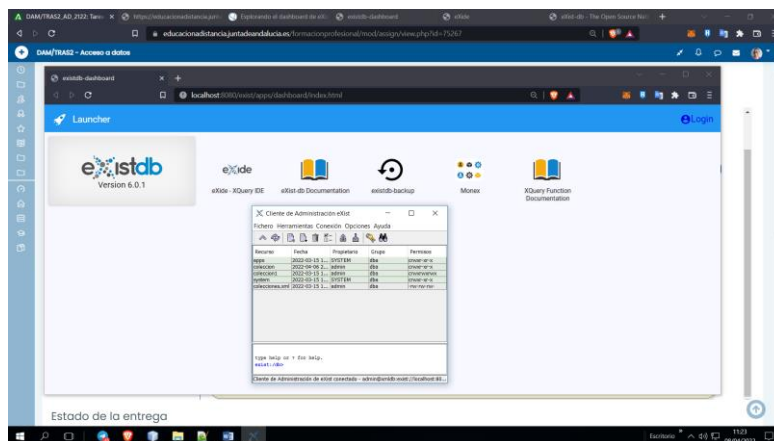
Utilizando la base de datos colecciones.xml, extrae las colecciones 'Libros', 'Prestamos' y 'Autores'

Comenzamos el ejercicio instalando **Exist-db** desde su web oficial:

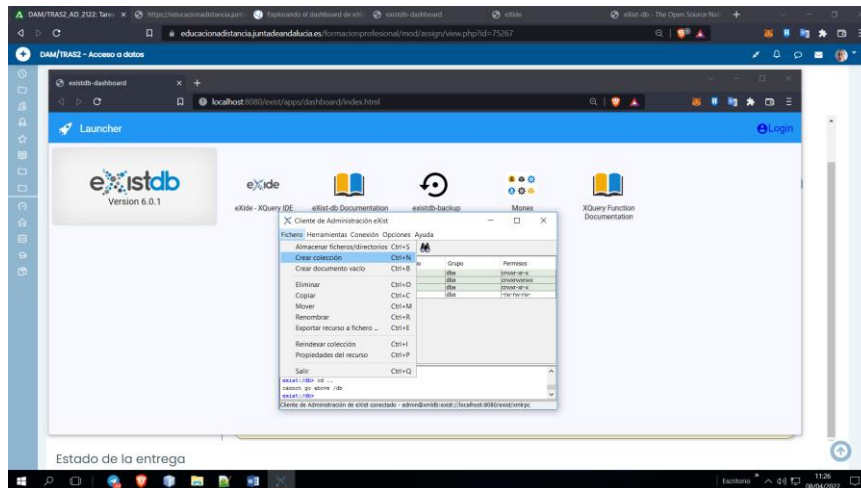
<http://exist-db.org/exist/apps/homepage/index.html>



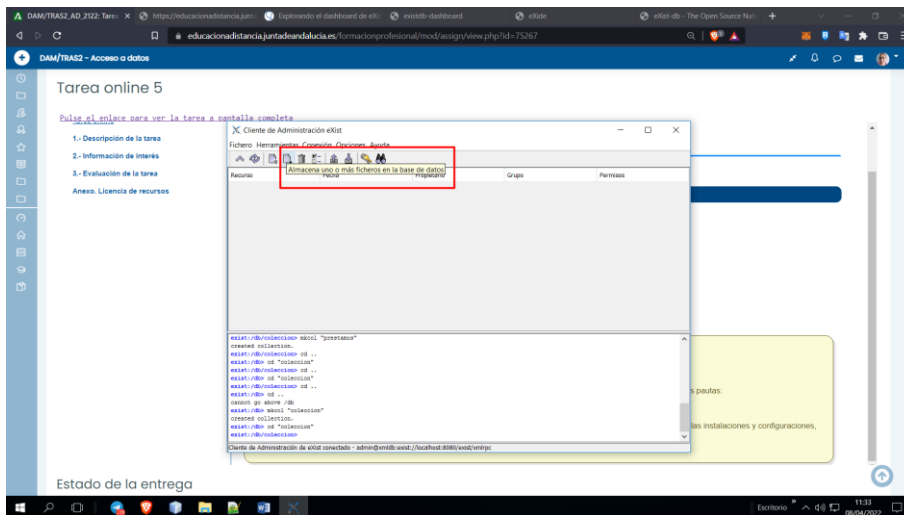
Una vez que tenemos instalado **Exist-db** en la barra y en el Escritorio tenemos su launcher donde podremos trabajar con distintas herramientas, el Dashboard, Exide y Java Admin Client.



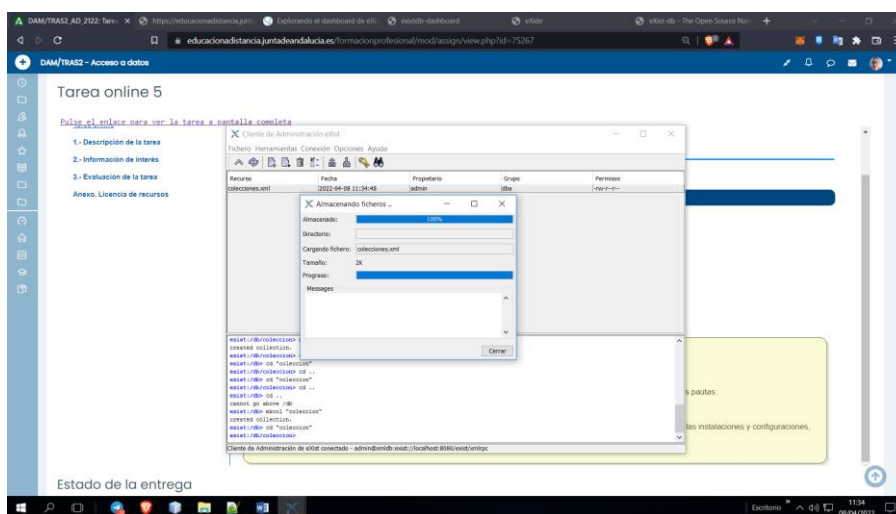
Si abrimos **Java Admin Client**, iniciamos el entorno con nuestro usuario, contraseña y URL. Al abrir el cliente de Administración eXist podemos escoger el recurso que vamos a utilizar.



Vamos a crear una nueva colección llamada **“colección”**.

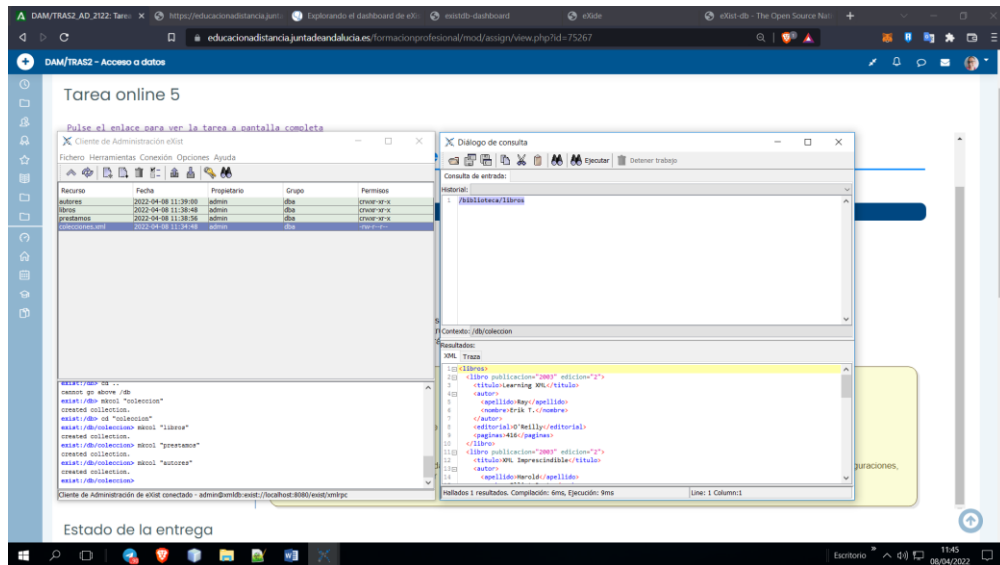


Posteriormente vamos almacenar nuestro fichero **“colecciones.xml”** pulsando la opción **“Almacenar uno o más ficheros en la base de datos”**

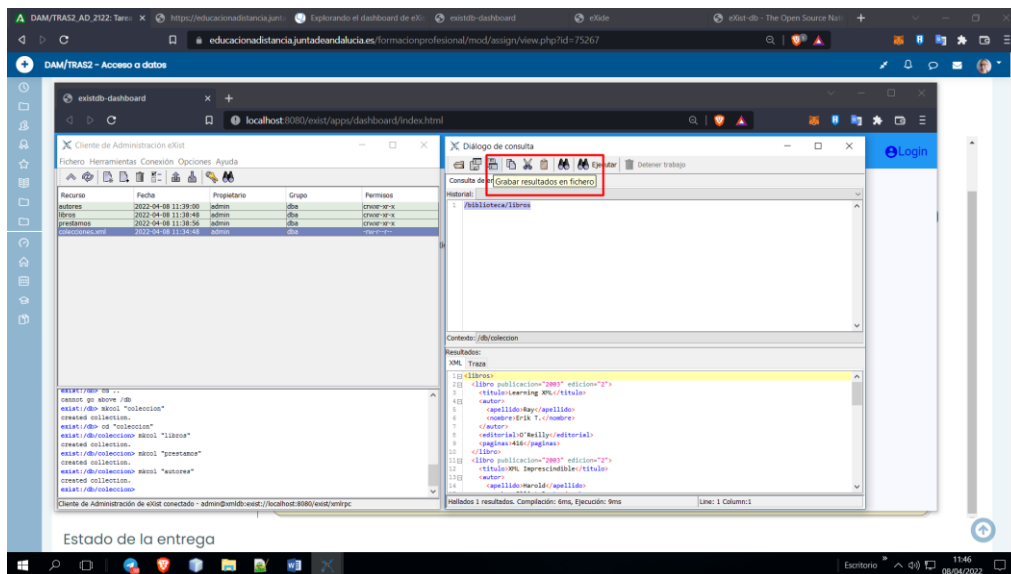




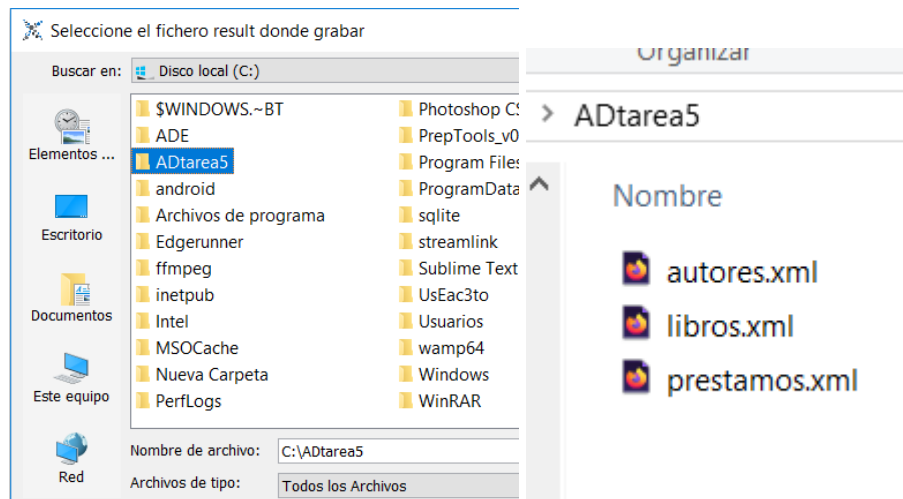
Abrimos las consultas pulsando en los prismáticos. De esta forma vamos a poder hacer consultas XPath en nuestra colección.



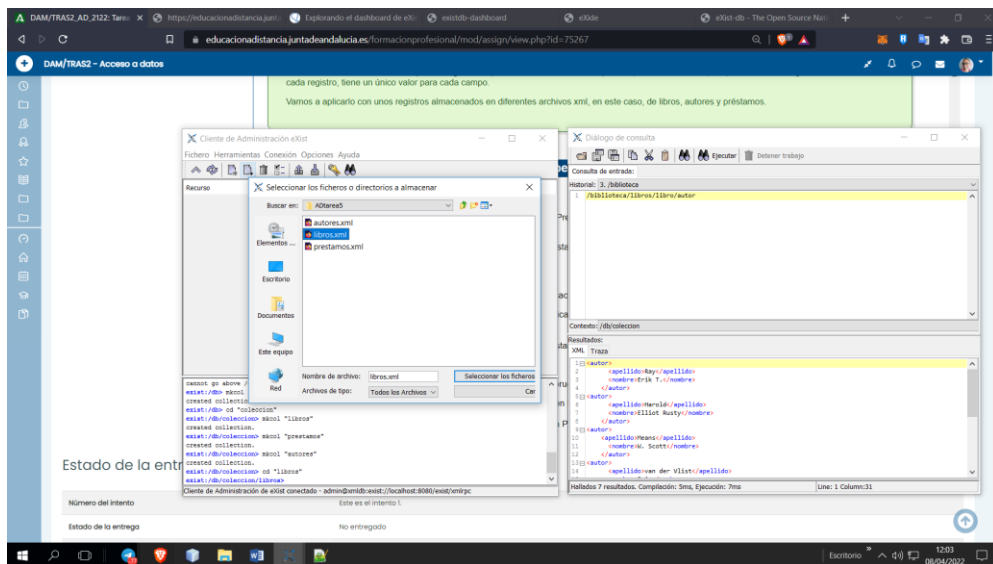
Consultamos los libros que tenemos en nuestra colección. **Consulta:**  
**/biblioteca/libros**



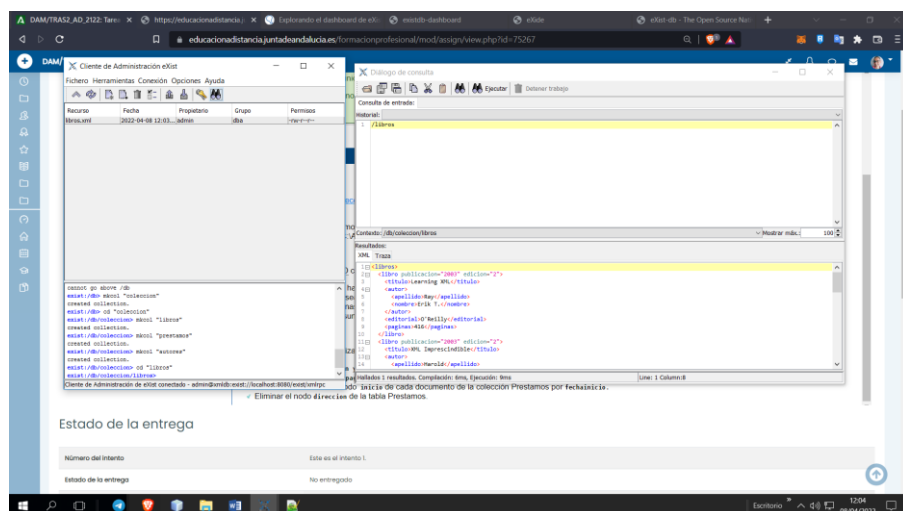
Guardamos los resultados en un fichero **libros.xml** . Procedemos a hacer lo mismo con autores: **Consulta: /biblioteca/libros/libro/autor** . Guardamos el resultado de la consulta **autores.xml** En la carpeta C:\ADTarea5 . Lo mismo con préstamos: **Consulta: /biblioteca/prestamos**. Lo guardamos como **prestamos.xml** .



En la carpeta C:\ADTarea5, los ficheros autores.xml, libros.xml, prestamos.xml

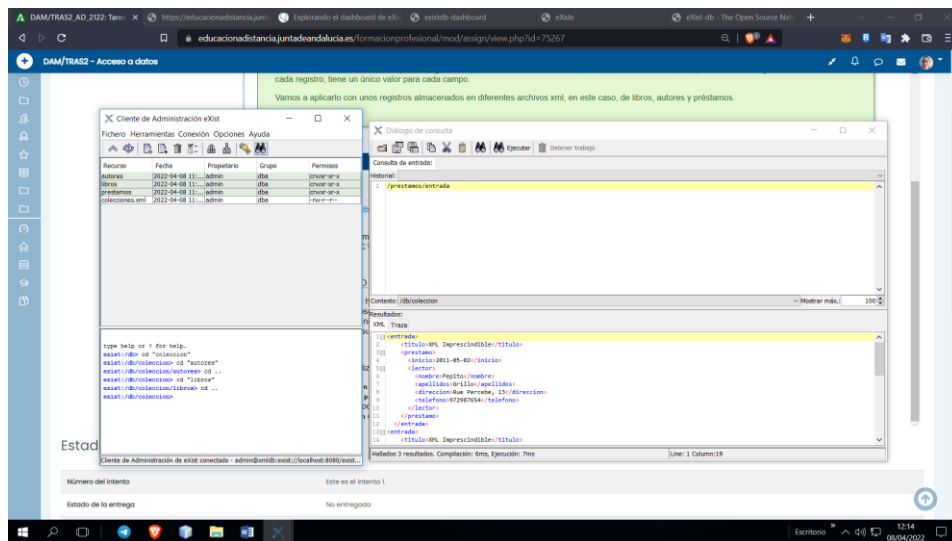


Almacenamos libros.xml dentro de nuestra colección Libros.



Hacemos lo mismo con cada uno de nuestros ficheros , prestamos.xml, autores.xml.

Los almacenamos dentro de su colección.

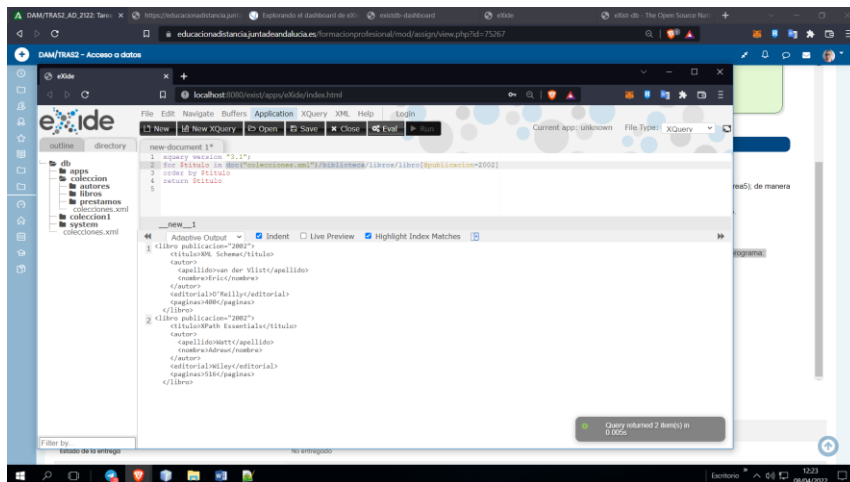


El resultado es que tenemos una colección “colecciones” dentro de la cual tenemos las colecciones “libros”, “prestamos” y “autores”.

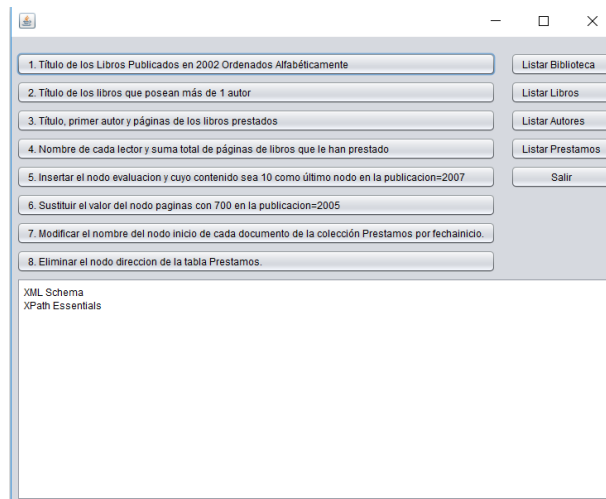
## EJERCICIO 2.

Realiza estas consultas a la BD colecciones.xml y ejecútalas desde una aplicación Java. Guardar cada consulta en un script.xq, que llamarás desde tu programa.

1. Título de los libros que se hayan publicado en 2002, y ordénalos alfabéticamente.



## Consulta en Exide



## Consulta en Java

### Consulta Xquery:

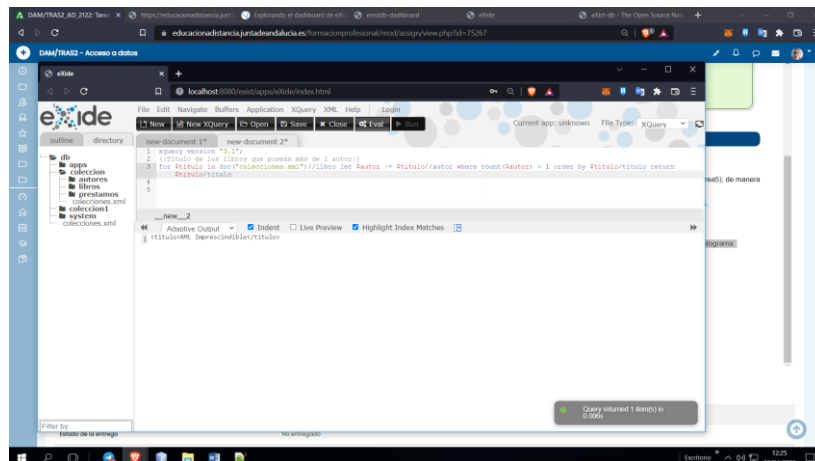
```

        for $titulo in
doc("colecciones.xml")/biblioteca/libros/libro[@publicacion=2002]
        order by $titulo
        return $titulo

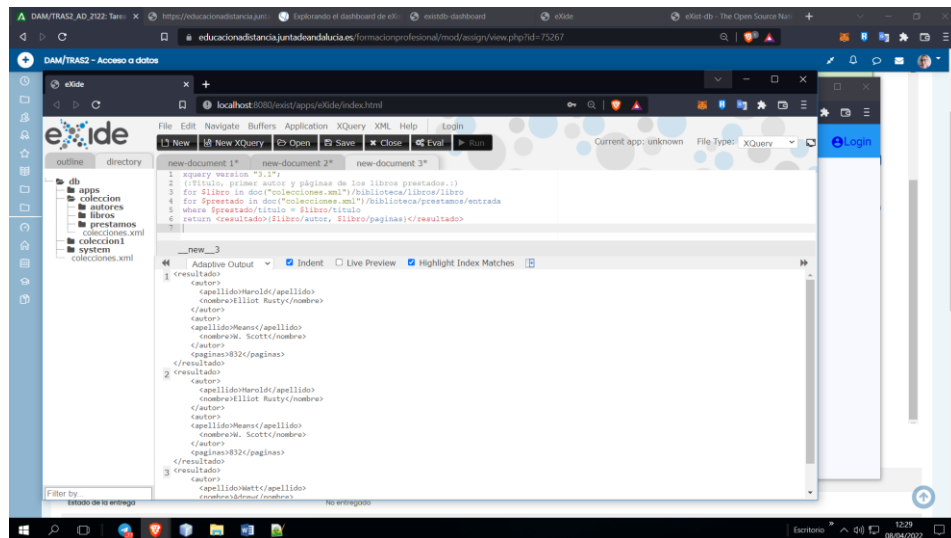
```



## 2. Título de los libros que posean más de 1 autor.



### 3. Título, primer autor y páginas de los libros prestados.



### Consulta Exide

1. Título de los Libros Publicados en 2002 Ordenados Alfabéticamente	Listar Biblioteca
2. Título de los libros que posean más de 1 autor	Listar Libros
3. Título, primer autor y páginas de los libros prestados	Listar Autores
4. Nombre de cada lector y suma total de páginas de libros que le han prestado	Listar Prestamos
5. Insertar el nodo evaluacion y cuyo contenido sea 10 como último nodo en la publicacion=2007	Salir
6. Sustituir el valor del nodo paginas con 700 en la publicacion=2005	
7. Modificar el nombre del nodo inicio de cada documento de la colección Prestamos por fechainicio.	
8. Eliminar el nodo direccion de la tabla Prestamos.	

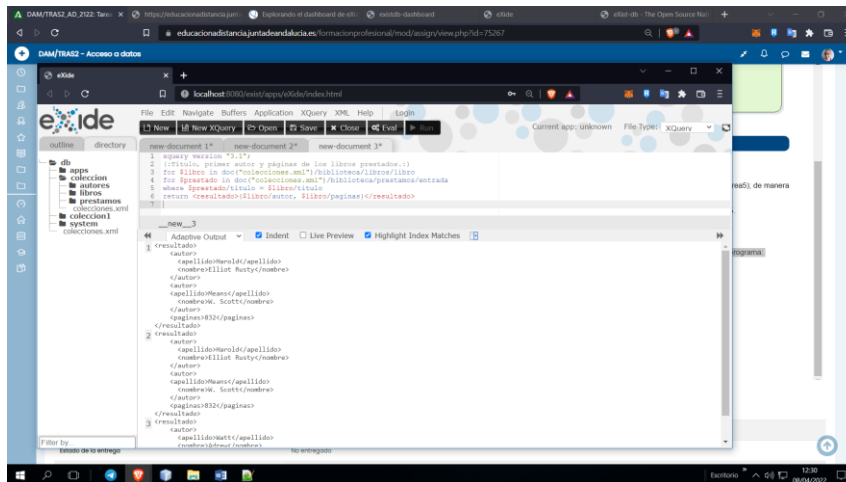
```
<resultado><autor>
  <apellido>Harold</apellido>
  <nombre>Elliot Rusty</nombre>
</autor><autor>
  <apellido>Means</apellido>
  <nombre>W. Scott</nombre>
</autor><paginas>832</paginas></resultado>
<resultado><autor>
  <apellido>Harold</apellido>
  <nombre>Elliot Rusty</nombre>
</autor><autor>
  <apellido>Means</apellido>
  <nombre>W. Scott</nombre>
</autor><paginas>832</paginas></resultado>
<resultado><autor>
  <apellido>Watt</apellido>
```

### Consulta Java

### Consulta Xquery:

```
for $libro in doc("coleccion.xml")/biblioteca/libros/libro
for $prestado in doc("coleccion.xml")/biblioteca/prestamos/entrada
where $prestado/titulo = $libro/titulo
return <resultado>{$libro/autor, $libro/paginas}</resultado>
```

- Nombre de cada lector y suma total de páginas de libros que le han prestado.



### Consulta Exide

1. Título de los Libros Publicados en 2002 Ordenados Alfabéticamente
2. Título de los libros que posean más de 1 autor
3. Título, primer autor y páginas de los libros prestados
4. Nombre de cada lector y suma total de páginas de libros que le han prestado
5. Insertar el nodo evaluacion y cuyo contenido sea 10 como último nodo en la publicacion=2007
6. Sustituir el valor del nodo paginas con 700 en la publicacion=2005
7. Modificar el nombre del nodo inicio de cada documento de la colección Prestamos por fechainicio.
8. Eliminar el nodo direccion de la tabla Prestamos.

Listar Biblioteca  
Listar Libros  
Listar Autores  
Listar Prestamos  
Salir

<resultado><nombre>Jose</nombre>832</resultado>  
<resultado><nombre>Pepito</nombre><nombre>Pepito</nombre>1348</resultado>

### Consulta Java

### Consulta Xquery:

```

for $libro in doc("coleccion.xml")/biblioteca/libros/libro
for $prestado in doc("coleccion.xml")/biblioteca/prestamos/entrada
where $prestado/titulo = $libro/titulo
let $nombre := $prestado//nombre
group by $nombre
return <resultado>
    <nombre>{$prestado/prestamo/lector/nombre/text()}</nombre>
    <paginas>{sum($libro/paginas)}</paginas>
</resultado>

```



## Sustituir el valor del nodo paginas con 700 en la publicacion=2005

The screenshot shows a Java application window with a list of tasks on the left and a code editor on the right. The tasks are:

1. Título de los Libros Publicados en 2002 Ordenados Alfabéticamente
2. Título de los libros que posean más de 1 autor
3. Título, primer autor y páginas de los libros prestados
4. Nombre de cada lector y suma total de páginas de libros que le han prestado
5. Insertar el nodo evaluación y cuyo contenido sea 10 como último nodo en la publicacion=2007
6. Sustituir el valor del nodo paginas con 700 en la publicacion=2005
7. Modificar el nombre del nodo inicio de cada documento de la colección Prestamos por fechainicio.
8. Eliminar el nodo direccion de la tabla Prestamos.

Buttons on the right: Listar Biblioteca, Listar Libros, Listar Autores, Listar Prestamos, Salir.

Code in the editor:

```
<libro publicacion="2005">
  <título> Beginning XSLT 2.0: Form Novice to Professional</título>
  <autor>
    <apellido>Tennison</apellido>
    <nombre>Jeni</nombre>
  </autor>
  <editorial>Apress</editorial>
  <paginas>700</paginas>
</libro>
```

**Consulta Java**

**Consulta XQuery**

```
update replace //libro[@publicacion=2005]/paginas with
  <paginas>700</paginas>
```

## Modificar el nombre del nodo inicio de cada documento de la colección Préstamos por fechainicio.

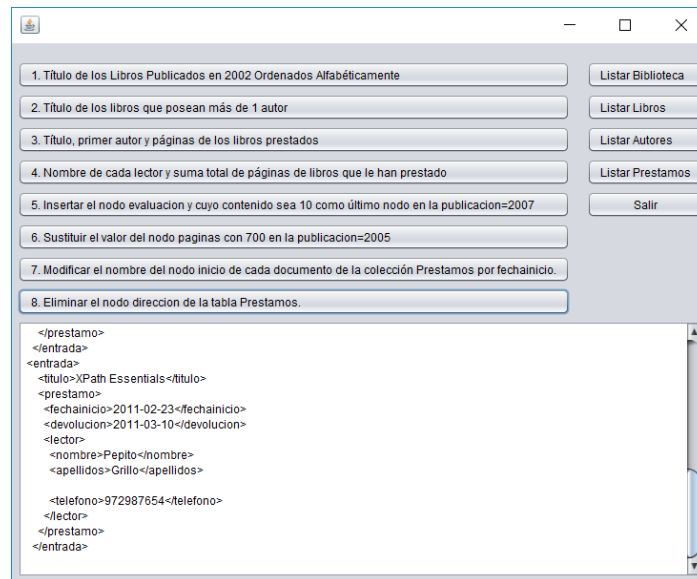
The screenshot shows the same Java application window, but the code in the editor is different:

```
<fechainicio>2011-05-02</fechainicio>
<fechainicio>2011-02-12</fechainicio>
<fechainicio>2011-02-23</fechainicio>
```

**Consulta XQuery :**

```
for $dato in /biblioteca/prestamos/entrada/prestamo/inicio return
  update rename $dato as 'fechainicio''
```

## Eliminar el nodo dirección de la tabla Préstamos.



### Consulta XQuery:

```
for $direccion in
/biblioteca/prestamos/entrada/prestamo/lector/direccion return update
delete $direccion"
```

## Código Java

Para la realización del ejercicio se han creado varias clases y métodos. En la clase Métodos tenemos todo lo relacionado con la base de datos.

Podemos destacar varios parámetros esenciales para conectar con nuestra BD.

```
public static String CONSULTA8 = "for $direccion in /biblioteca/prestamos/entrada/prestamo/lector/direccion return update delete $direccion";
public static String DRIVER = "org.exist.xmlldb.DatabaseImpl"; //Driver de Exist-DB
public static Database DATABASE;
public static Collection COLLECTION;
public static String URI = "xmlldb:exist://localhost:8080/exist/xmlrpc/db/coleccion"; // URI DE NUESTRA COLECCION
public static String USER = "admin"; // Nombre de usuario de la BD
public static String PASSWORD = "1234"; // Contraseña
public static XPathQueryService SERVICE;
```

Tenemos configurados variables estáticas de la mayor parte de los parámetros, contiene el **Driver** de **EXist-DB**, **Database**, **Collection**, **URI** de nuestra colección, **usuario**, **contraseña** y **servicio**.

```
public static void conexion() throws InstantiationException, XMLDBException, ClassNotFoundException, IllegalAccessException {
    Class cl = Class.forName(DRIVER); //Cargar el driver
    DATABASE = (Database) cl.newInstance(); //Instancia de la BD
    DATABASE.setProperty("create-database", "true");
    DatabaseManager.registerDatabase(DATABASE); //Registrar la BD
    COLLECTION = DatabaseManager.getCollection(URI, USER, PASSWORD); //Accedemos a la colección
    SERVICE = (XPathQueryService) COLLECTION.getService("XPathQueryService", "1.0");
    SERVICE.setProperty("pretty", "true");
    SERVICE.setProperty("encoding", "ISO-8859-1");
}
```

El método conexión encargado de crear la mayor parte de parámetros relacionados con la base de datos.

```
public static File FICHERO1 = new File("./consultas/consulta1.xq");
public static File FICHERO2 = new File("./consultas/consulta2.xq");
public static File FICHERO3 = new File("./consultas/consulta3.xq");
public static File FICHERO4 = new File("./consultas/consulta4.xq");
public static File FICHERO5 = new File("./consultas/consulta5.xq");
public static File FICHERO6 = new File("./consultas/consulta6.xq");
public static File FICHERO7 = new File("./consultas/consulta7.xq");
public static File FICHERO8 = new File("./consultas/consulta8.xq");
```

Tenemos las variables estáticas de acceso a los ficheros .xq para las consultas XQuery.

```
// Listar Prestamos
public static String CONSULTAPRESTAMOS = "/biblioteca/prestamos";
// 1. Título de los libros que se hayan publicado en 2002, y ordenalos alfabéticamente.
public static String CONSULTA1 = "for $a in /biblioteca/libros/libro[@publicacion=2002] order by $a/titulo return $a/titulo";
// 2. Título de los libros que posean más de 1 autor.
public static String CONSULTA2 = "for $titulo in /biblioteca/libros/libro let $autor := $titulo/autor where count($autor) > 1 return $titulo";
// 3. Título, primer autor y páginas de los libros prestados.
public static String CONSULTA3 = "for $libro in doc(\"coleccion.xml\")/biblioteca/libros/libro
+ \"for $prestado in doc(\"coleccion.xml\")/biblioteca/prestamos/entrada
+ \"where $prestado/titulo = $libro/titulo
+ \"return <resultado>{$libro/autor, $libro/paginas}</resultado>";
// 4. Nombre de cada lector y suma total de páginas de libros que le han prestado.
public static String CONSULTA4 = "for $libro in doc(\"coleccion.xml\")/biblioteca/libros/libro
+ \"for $prestado in doc(\"coleccion.xml\")/biblioteca/prestamos/entrada
+ \"where $prestado/titulo = $libro/titulo
+ \"let $nombre := $prestado/lector
+ \"group by $nombre
+ \"return <resultado>{$prestado/prestamo/lector/nombre, sum($libro/paginas) }</resultado>";
```

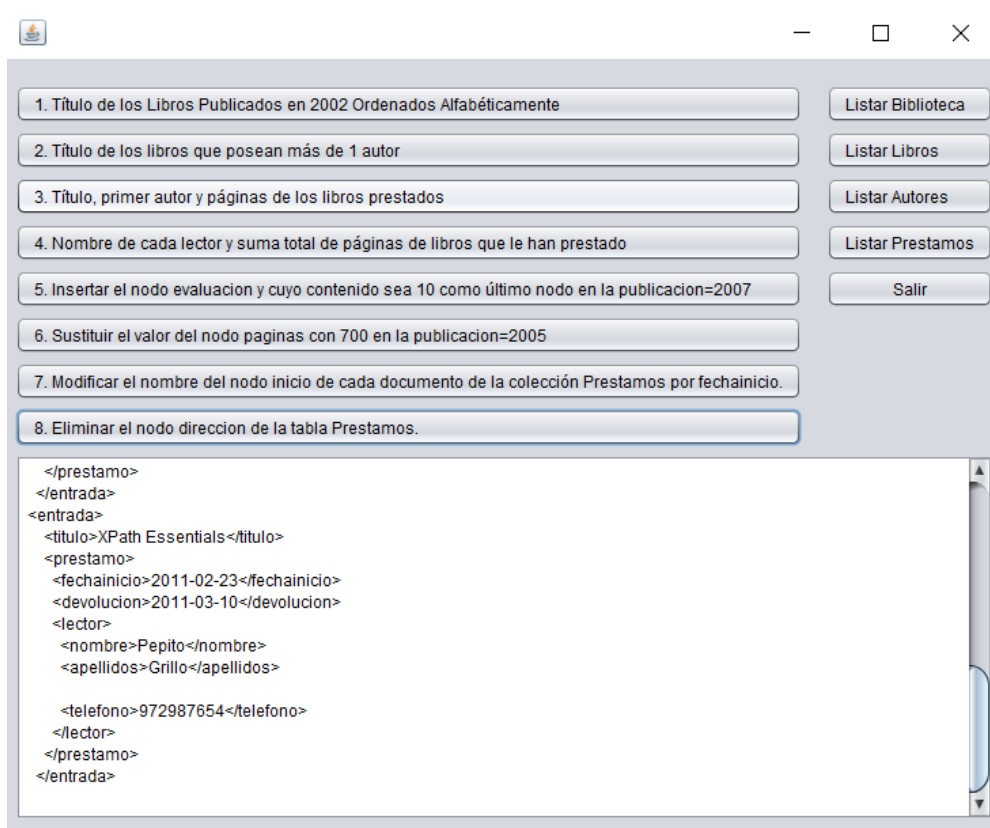
Aunque no era necesario también se han configurado variables y métodos para hacer las consultas desde una "**String**" directamente, aunque finalmente no se han implementado en la aplicación.

```
public static ResultSet consultaAutores() throws ClassNotFoundException, InstantiationException, IllegalAccessException, XMLDBException {
    conexion();
    ResultSet result = SERVICE.query(CONSULTAAUTORES); // Consulta a lanzar
    return result;
}
```

Podemos formular las consultas de varias formas . Para realizar las consultas hemos hecho uso de métodos estaticos que devuelven **ResultSet** , los cuales se conectan con la base de datos con el método estatico **Conexión()** , hace la consulta a la BD y nos devuelve un objeto **ResultSet** el cual va a ser tratado posteriormente en nuestra vista (JFrame).

```
public void imprimirPantalla(ResultSet result) throws XMLDBException {
    txtArea.setText(""); // Limpia la pantalla
    ResourceIterator i = result.getIterator();
    while (i.hasMoreResources()) { // Procesamos el resultado
        Resource r = i.nextResource();
        String cadena = (String) r.getContent();
        txtArea.append(cadena + "\n");
    }
}
```

En nuestra vista JFrame tenemos creado un método **imprimirPantalla** que será el encargado de recoger el **ResultSet** creado anteriormente y mostrarlo en pantalla. El método básicamente muestra el resultado en un **TextArea**.



Se han añadido adicionalmente varios botones a la derecha para listar y salir de la aplicación.

## Referencias

### Ejercicios XQuery

<https://www.ticarte.com/contenido/ejercicios-practicos-de-xquery>

<https://blogsaverroes.juntadeandalucia.es/luismaroto/ut-13-consulta-y-manipulacion-en-xmlxquery/ejemplos-de-uso-de-xquery-ticarte/>

<https://www.cartagena99.com/recursos/alumnos/apuntes/XQuery.pdf>

<http://www.lsi.us.es/docs/informes/LSI-2005-02.pdf>

<http://www.sociedadelainformacion.com/35/xquery.pdf>

### Subconsultas

[https://www.youtube.com/watch?v=Q\\_13zmFnM8&list=PLYwQpNMBCzlkVnlq-HyymWBB8IIXO7Q&index=34&t=1s](https://www.youtube.com/watch?v=Q_13zmFnM8&list=PLYwQpNMBCzlkVnlq-HyymWBB8IIXO7Q&index=34&t=1s)

[https://www.youtube.com/watch?v=V47SzhsActg&list=PLU6TMJPhl\\_cbgneZRbLDbE\\_gaJ-nq74uX&index=6&t=7s](https://www.youtube.com/watch?v=V47SzhsActg&list=PLU6TMJPhl_cbgneZRbLDbE_gaJ-nq74uX&index=6&t=7s)



