


Vamos a instalar Existdb:

<https://github.com/eXist-db/exist/releases/tag/eXist-6.3.0>

 exist-distribution-6.3.0-win.zip	155 MB	Oct 26, 2024
--	--------	--------------

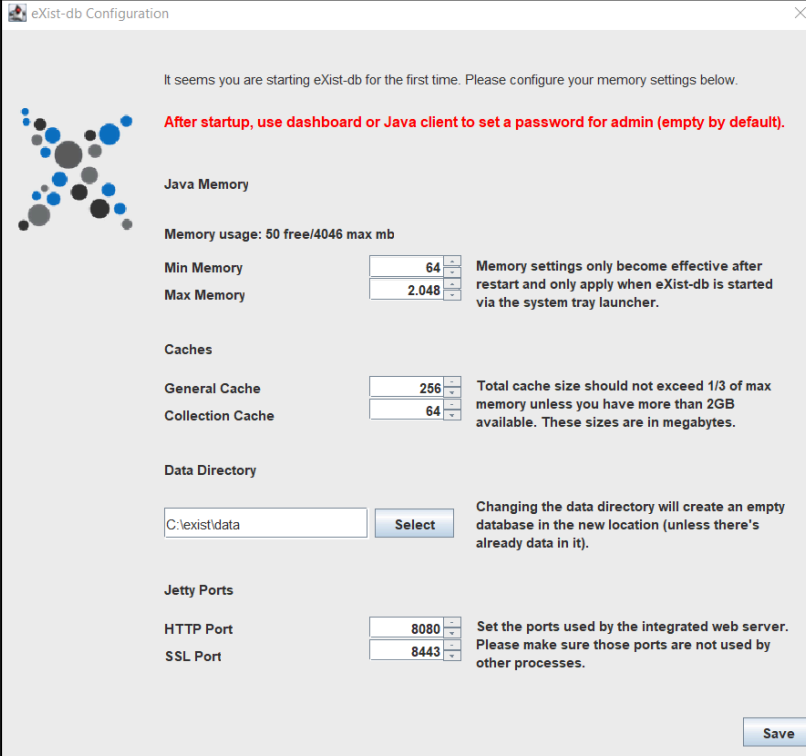
Una vez que nos hayamos descargado la base de datos nos darán un .zip.
Lo extraemos en C: y **le cambiamos de nombre a la carpeta** porque el nombre de la carpeta es demasiado largo, lo dejamos así.

 exist	04/02/2025 10:14	Carpeta de archivos
---	------------------	---------------------

La ruta debe de ser lo más corta posible:

C:\exist

Una vez que tengamos esto hecho vamos a **C:\exist\bin** , hacemos doble click en launcher.bat para que nos aparezca la configuración:



The screenshot shows the 'eXist-db Configuration' window. It contains the following sections and settings:

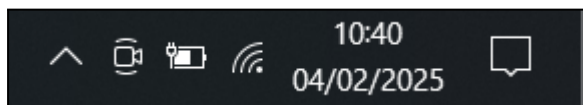
- Java Memory:** Memory usage: 50 free/4046 max mb. Min Memory: 64. Max Memory: 2.048. A note states: 'Memory settings only become effective after restart and only apply when eXist-db is started via the system tray launcher.'
- Caches:** General Cache: 256. Collection Cache: 64. A note states: 'Total cache size should not exceed 1/3 of max memory unless you have more than 2GB available. These sizes are in megabytes.'
- Data Directory:** C:\exist\data. A 'Select' button is next to the text box. A note states: 'Changing the data directory will create an empty database in the new location (unless there's already data in it).'
- Jetty Ports:** HTTP Port: 8080. SSL Port: 8443. A note states: 'Set the ports used by the integrated web server. Please make sure those ports are not used by other processes.'

A 'Save' button is located at the bottom right of the window.

Lo dejamos por defecto y le damos a "Save":



Nos aparecerá esto y se cerrará, pero no se cierra del todo sino que abajo en los iconos ocultos:



Le damos a la flecha para arriba:



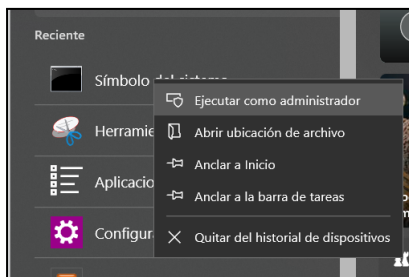
Aparece ahí.

ANTES DE CONTINUAR VAMOS A RESOLVER UN ERROR COMÚN:

Como habréis visto en la configuración el puerto que vamos a utilizar es el 8080, ¿qué pasa con este puerto?

Este puerto se suele utilizar mucho y si por alguna razón lo está utilizando otro programa te puedes volver loco buscando la solución, vamos a ver los puertos que tenemos con el cmd, en el caso de que haya uno usandolo nos lo vamos a cargar y así aseguramos que va perfectamente.

Vamos a abrir el cmd como administrador:



Vamos a poner este comando:

netstat -ano | findstr :8080

Este comando mostrará una lista con los procesos que están utilizando este puerto:

```
C:\WINDOWS\system32>netstat -ano | findstr :8080
TCP    0.0.0.0:8080      0.0.0.0:0        LISTENING       7008
TCP    [::]:8080         [::]:0           LISTENING       7008

C:\WINDOWS\system32>
```

En este caso no hay ninguno usando este puerto pero si en vez de esto apareciera esto otro:

```
C:\WINDOWS\system32>netstat -ano | findstr :8080
TCP    0.0.0.0:8080      0.0.0.0:0        LISTENING       5340
TCP    [::]:8080         [::]:0           LISTENING       5340
TCP    [::1]:8080        [::1]:50384      ESTABLISHED     5340
TCP    [::1]:8080        [::1]:50385      ESTABLISHED     5340
TCP    [::1]:50384       [::1]:8080       ESTABLISHED     15916
TCP    [::1]:50385       [::1]:8080       ESTABLISHED     15916

C:\WINDOWS\system32>
```

Aquí sí que tendríamos que matar el proceso **PID 5340** porque sino no nos dejará arrancar el existdb.

Para matar el proceso ponemos este comando:

taskkill /PID 5340 /F

Para verificar que se ha matado correctamente ponemos este comando:

netstat -ano | findstr :8080

Una vez que tengamos esto bien configurado vamos a arrancarlo:

C:\exist\bin

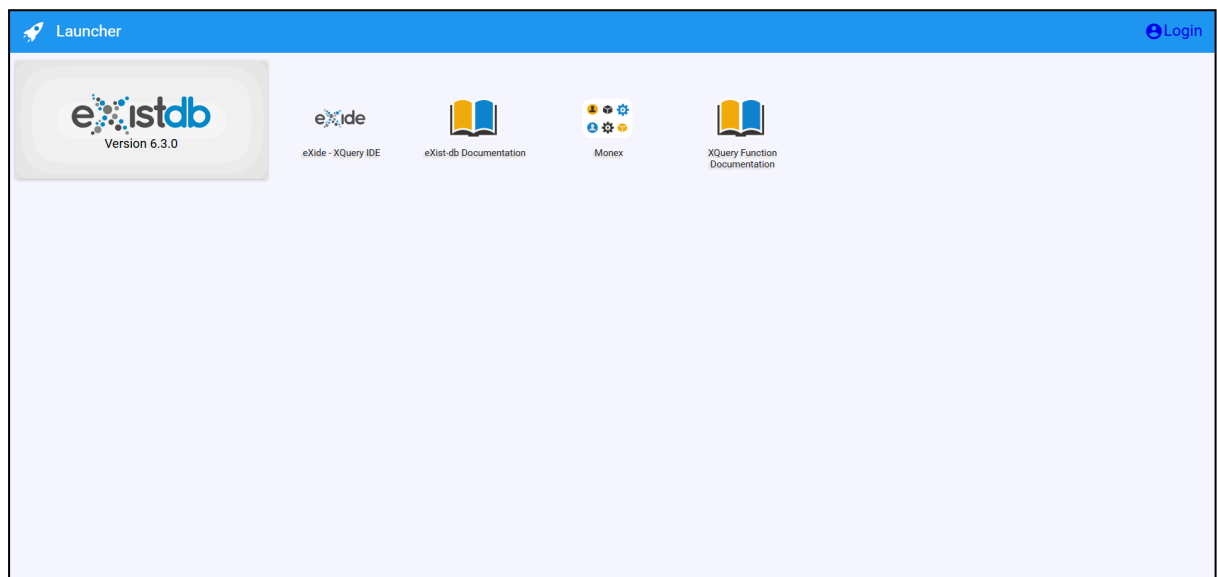
Buscamos el archivo startup.bat y le damos doble click:

Nos tiene que aparecer lo siguiente:

```
04 feb 2025 11:07:16,729 [main] INFO (JettyStart.java [run]:289) - Server has started, listening on:
04 feb 2025 11:07:16,731 [main] INFO (JettyStart.java [run]:291) - http://192.168.20.118:8080/
04 feb 2025 11:07:16,731 [main] INFO (JettyStart.java [run]:291) - https://192.168.20.118:8443/
04 feb 2025 11:07:16,731 [main] INFO (JettyStart.java [run]:294) - Configured contexts:
04 feb 2025 11:07:16,732 [main] INFO (JettyStart.java [run]:300) - /exist (eXist XML Database)
04 feb 2025 11:07:16,734 [main] INFO (JettyStart.java [run]:316) - /exist/iprange (IPrange filter)
04 feb 2025 11:07:16,734 [main] INFO (JettyStart.java [run]:300) - / (eXist-db portal)
04 feb 2025 11:07:16,735 [main] INFO (JettyStart.java [run]:316) - /iprange (IPrange filter)
04 feb 2025 11:07:16,735 [main] INFO (JettyStart.java [run]:324) - -----
```

Metemos en el navegador la url con el puerto 8080.

Nos debe de aparecer esta pantalla:

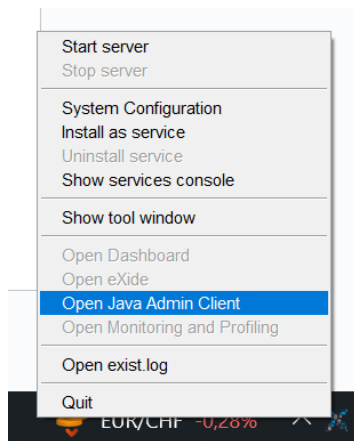


Nos vamos a loguear en la aplicación.

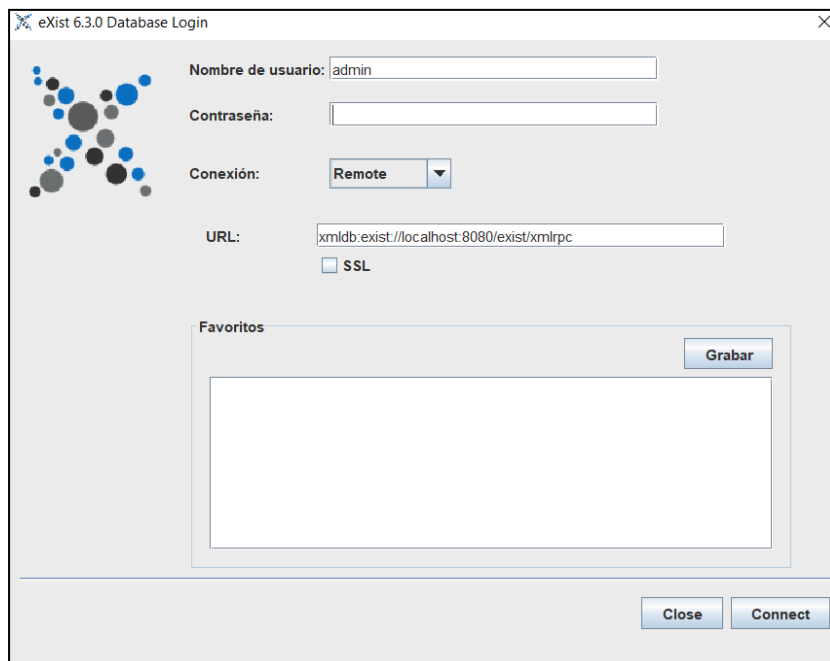
¿Cómo sabemos cuál es el usuario y la contraseña?

Le volvemos a meter en los iconos ocultos:





Le damos a Open Java Admin Client y nos aparecerá la siguiente ventana:



Como muestra en la ventana el usuario es admin y no tiene contraseña. Nos vamos al buscador donde nos tenemos que loguear y nos logueamos:

Login

User

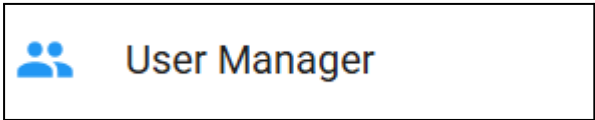
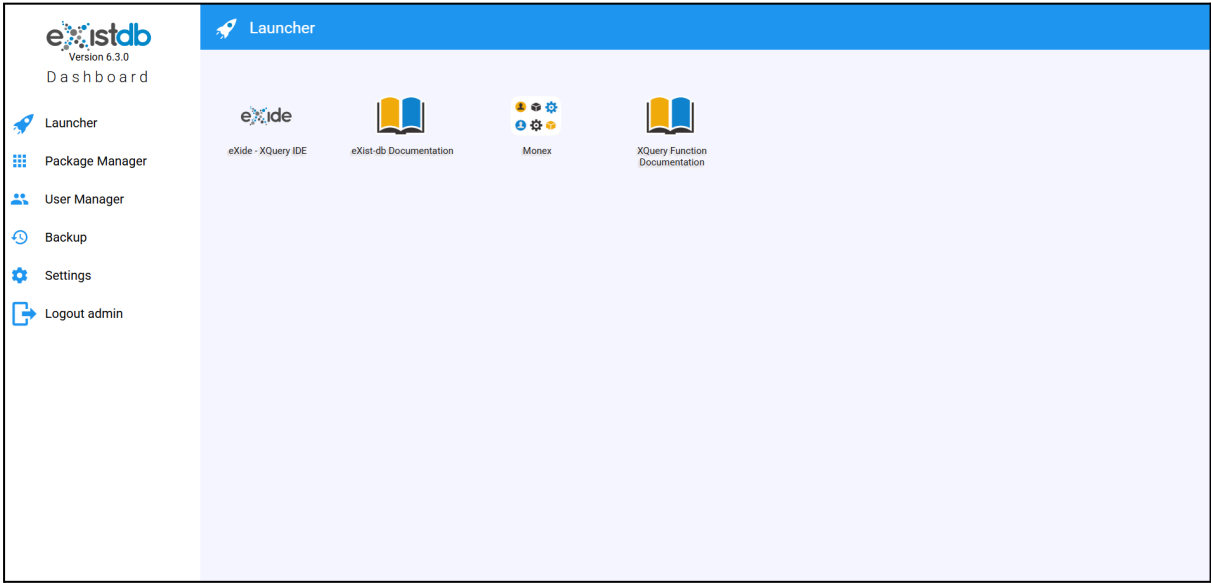
admin

Password

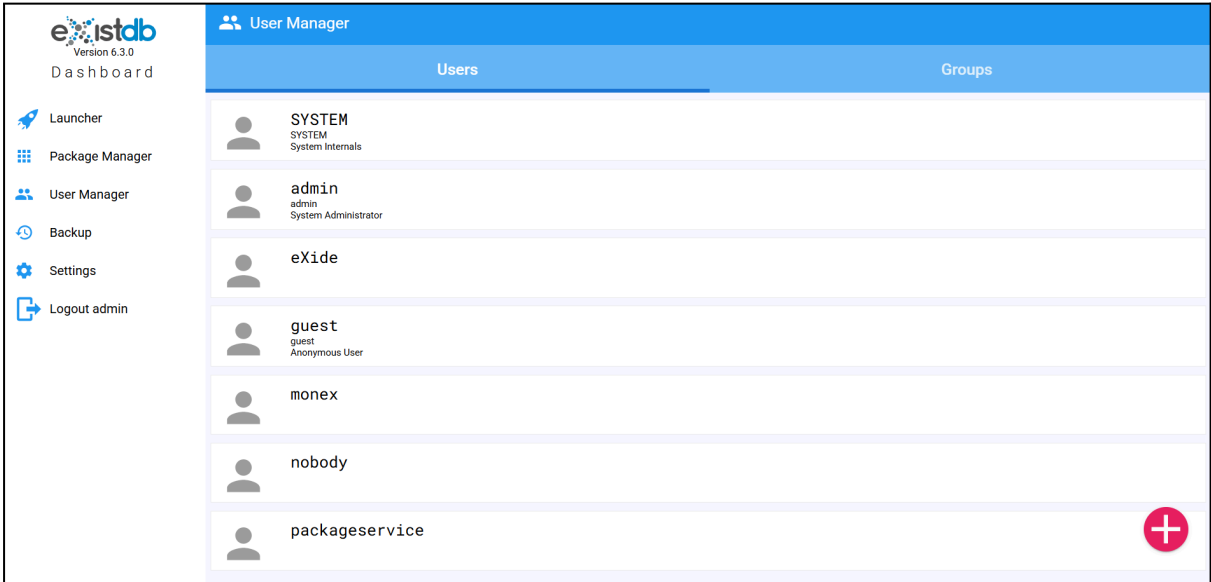
Wrong password or invalid user(must be member of group dba)

LOGIN

Una vez le damos a login se nos abrirá esta otra ventana:



Le damos a user Manager, vamos a crear un usuario.



En esta ventana le damos al “+”.

User Manager

User

User Name*

Laura

Full Name

Laura Juncos

Description

Password

....

Confirm

....

umask

022

Rellenamos los campos, bajamos y marcamos las dos primeras opciones:

Member of groups

dba

eXide

guest

monex

nogroup

packageservice

SAVE

DELETE USER

Una vez lo tengamos así le damos a “Save”.

User Manager

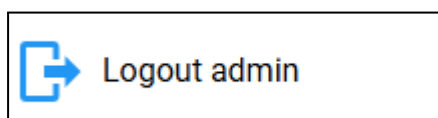
Users

Groups

Laura

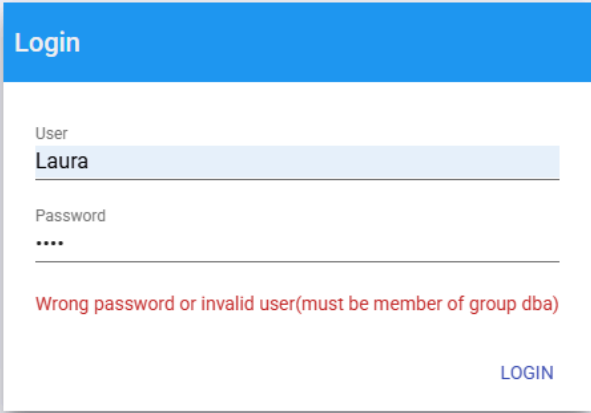
Laura Juncos

Ya aparece nuestro usuario ahí. Perfecto ya lo tenemos creado.



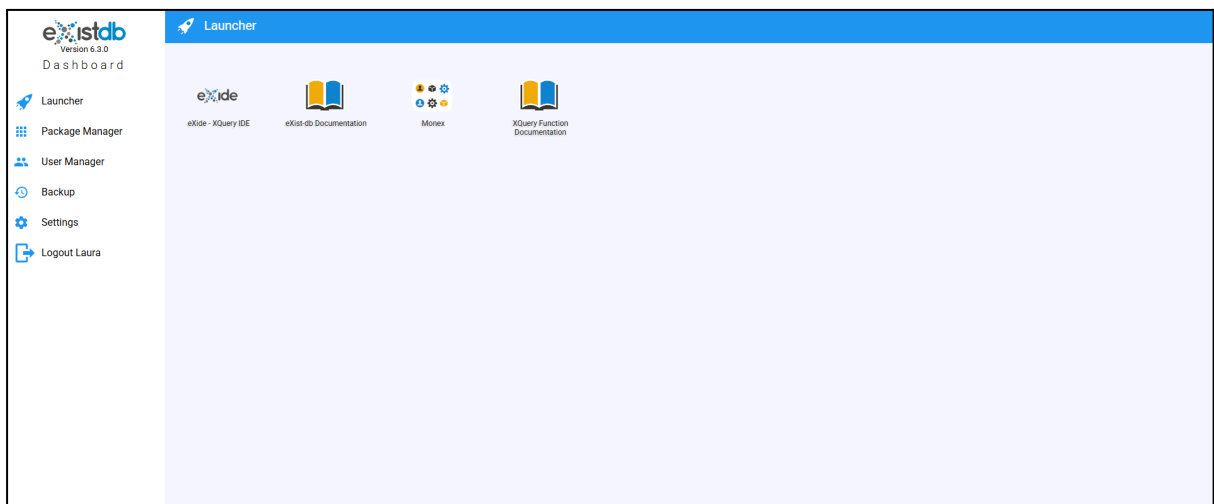
Ahora salimos del administrador.

Le damos otra vez a Login y nos logueamos con nuestro usuario recién creado.



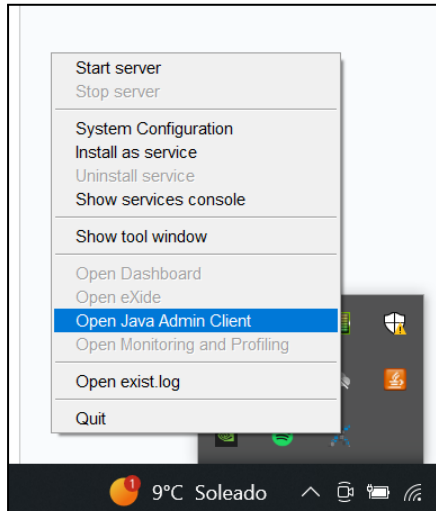
The image shows a login form titled "Login" with a blue header. It contains two input fields: "User" with the text "Laura" and "Password" with four dots. Below the fields is a red error message: "Wrong password or invalid user(must be member of group dba)". At the bottom right is a blue "LOGIN" button.

Una vez que tengamos el usuario y la contraseña le damos a Login.

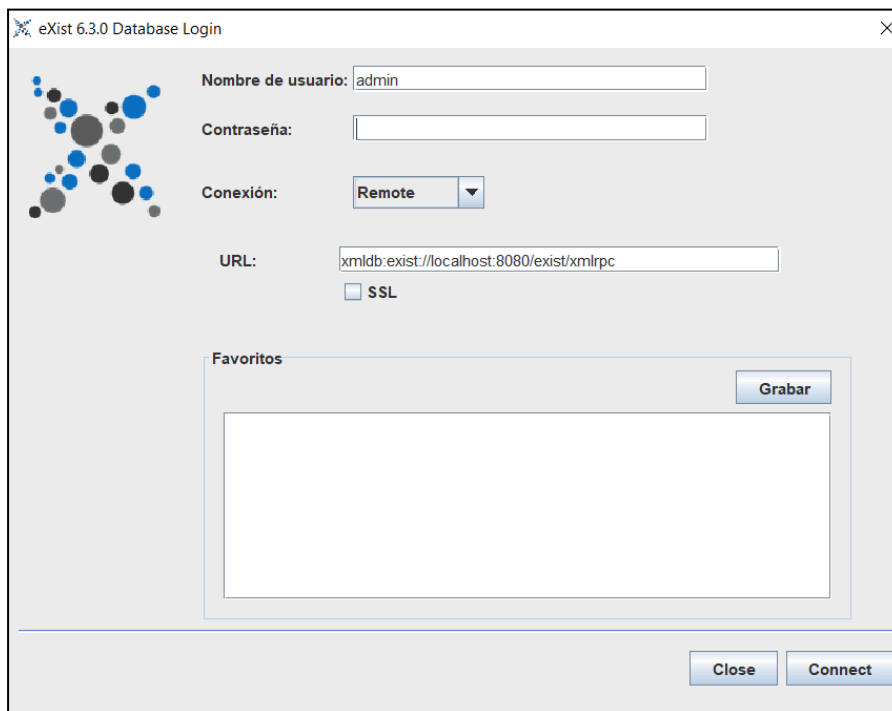


Nos debe aparecer la misma pantalla pero ahora con nuestro usuario.

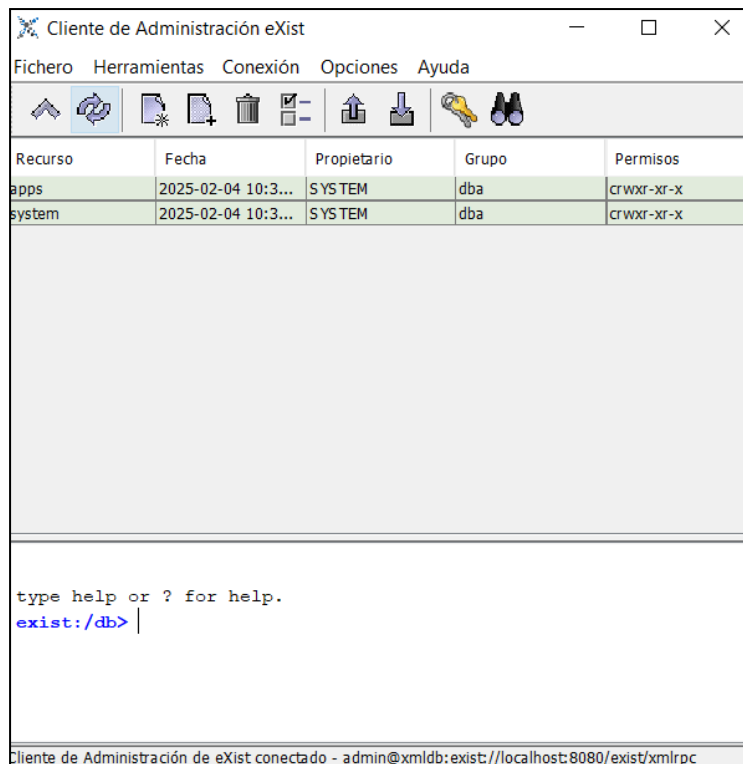
Ahora vamos a crear colecciones. Nos vamos otra vez a los iconos ocultos:



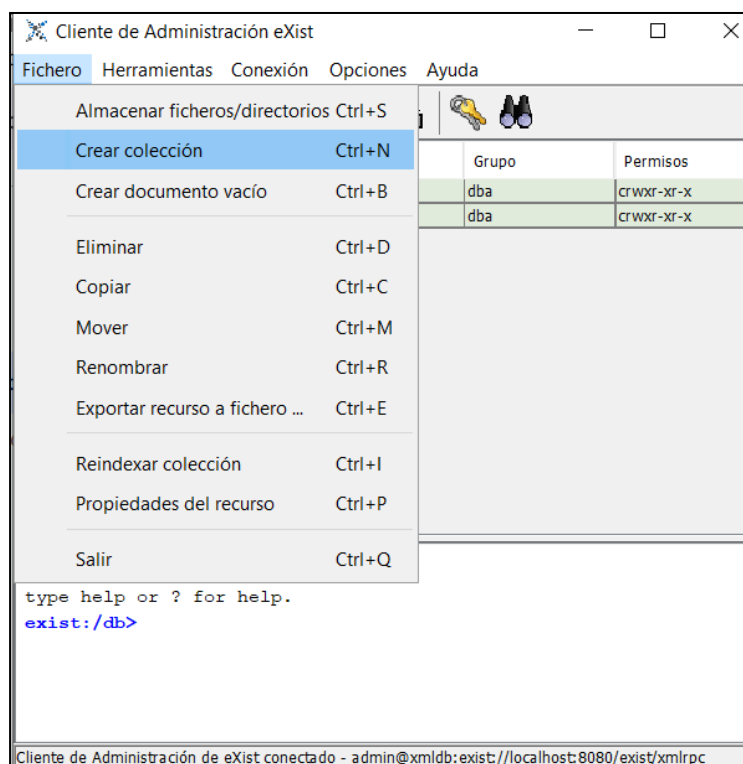
Una vez aquí le damos a Open Java Admin Client:



Una vez entramos en esta pantalla le damos a "Connect".



Nos aparecerá esta ventana. Ahora le damos a “Fichero” y a “Crear colección”:



Entrada

Por favor, introduzca el nombre de la nueva colección

Prueba1

Aceptar Cancelar

Nos aparecerá esta ventana para introducir el nombre de la nueva colección, podemos de nombre Prueba1 y le daremos a “Aceptar”.

Recurso	Fecha	Propietario	Grupo	Permisos
Prueba1	2025-02-04 12:1...	admin	dba	crwxr-xr-x
apps	2025-02-04 10:3...	S Y S T E M	dba	crwxr-xr-x
system	2025-02-04 10:3...	S Y S T E M	dba	crwxr-xr-x

Ahora en las tablas nos aparece la colección que hemos creado, le damos doble click a la colección:

Prueba1	2025-02-04 12:1...	admin	dba	crwxr-xr-x
---------	--------------------	-------	-----	------------

Cliente de Administración eXist

Fichero Herramientas Conexión Opciones Ayuda

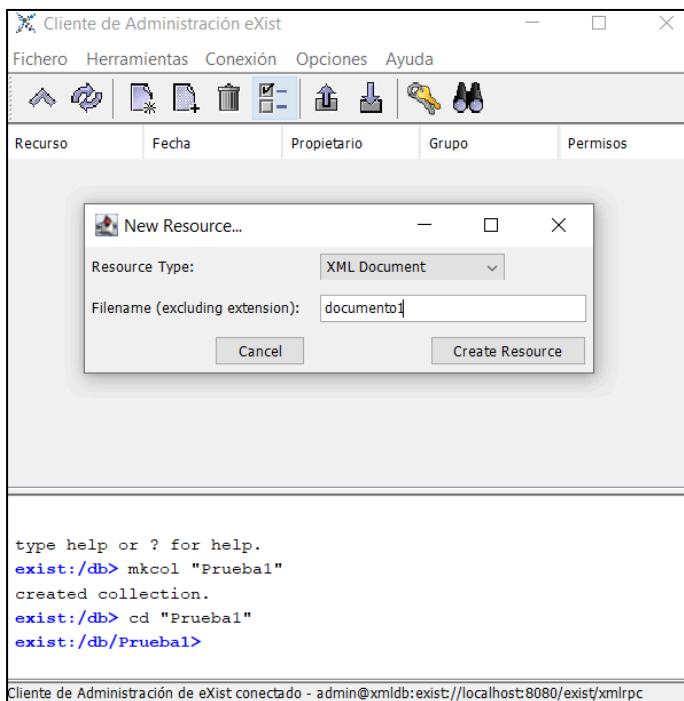
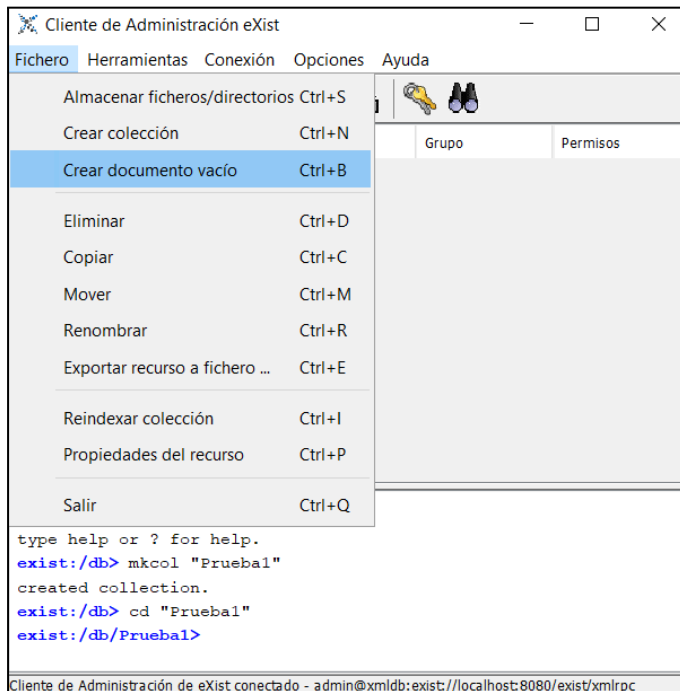
Recurso Fecha Propietario Grupo Permisos

type help or ? for help.
 exist:/db> mkcol "Prueba1"
 created collection.
 exist:/db> cd "Prueba1"
 exist:/db/Prueba1>

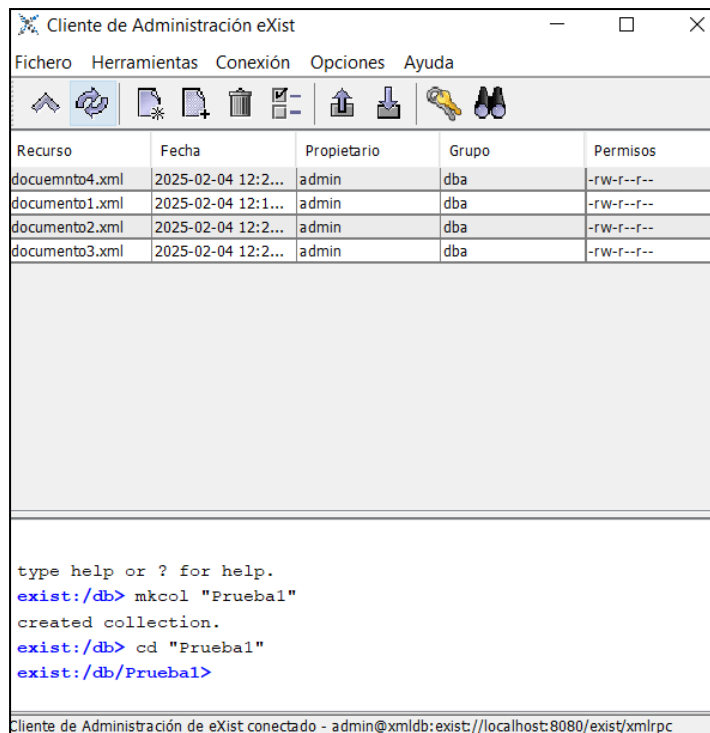
Cliente de Administración de eXist conectado - admin@xmldb:exist://localhost:8080/exist/xmlrpc

Como veis abajo estamos dentro de la colección pero la colección está vacía.

Ahora le damos a fichero y le damos a documento vacío:



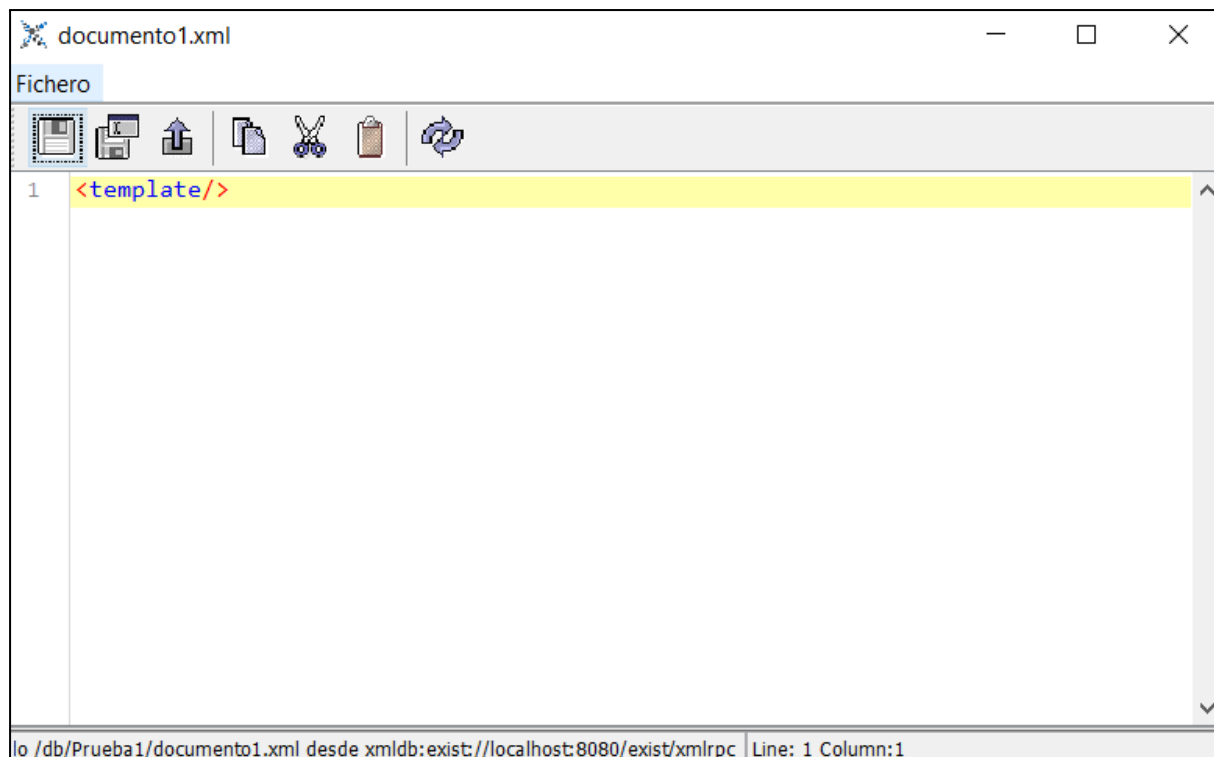
Nos aparecerá esta ventana, le pondremos de nombre "documento1" y le daremos a "Create Resource", vamos a hacer este paso hasta que tengamos 4 documentos y los llamaremos "documento1", "documento2", "documento3", "documento4".



Nos tendría que quedar así.

Ahora le vamos a dar doble click a documento1:

documento1.xml	2025-02-04 12:1...	admin	dba	-rw-r--r--
----------------	--------------------	-------	-----	------------

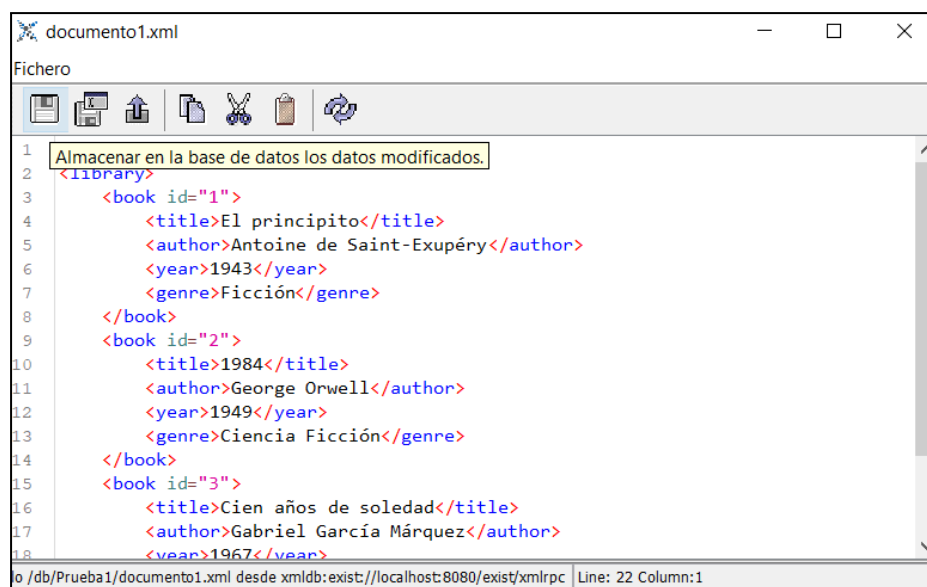


Nos aparecerá esto, esta ventana es para definir el documento y luego hacer consultas de ello.

En mi caso **documento 1** será de libros, **documento 2** será de razas de perros, **documento 3** será de marcas de coches y **documento 4** de equipos de fútbol.

DOCUMENTO 1:

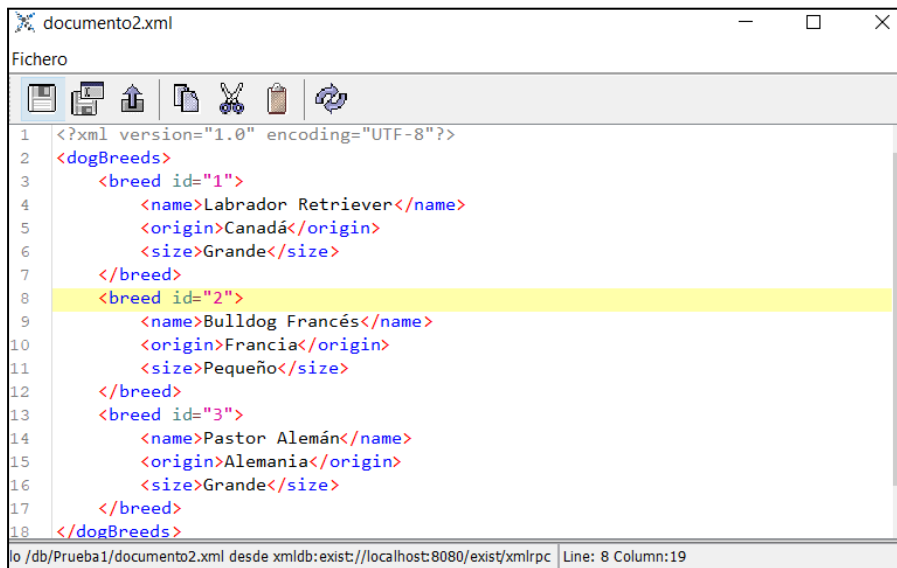
```
<?xml version="1.0" encoding="UTF-8"?>
<library>
  <book id="1">
    <title>El principito</title>
    <author>Antoine de Saint-Exupéry</author>
    <year>1943</year>
    <genre>Ficción</genre>
  </book>
  <book id="2">
    <title>1984</title>
    <author>George Orwell</author>
    <year>1949</year>
    <genre>Ciencia Ficción</genre>
  </book>
  <book id="3">
    <title>Cien años de soledad</title>
    <author>Gabriel García Márquez</author>
    <year>1967</year>
    <genre>Realismo mágico</genre>
  </book>
</library>
```



Lo guardamos y lo cerramos.

DOCUMENTO 2:

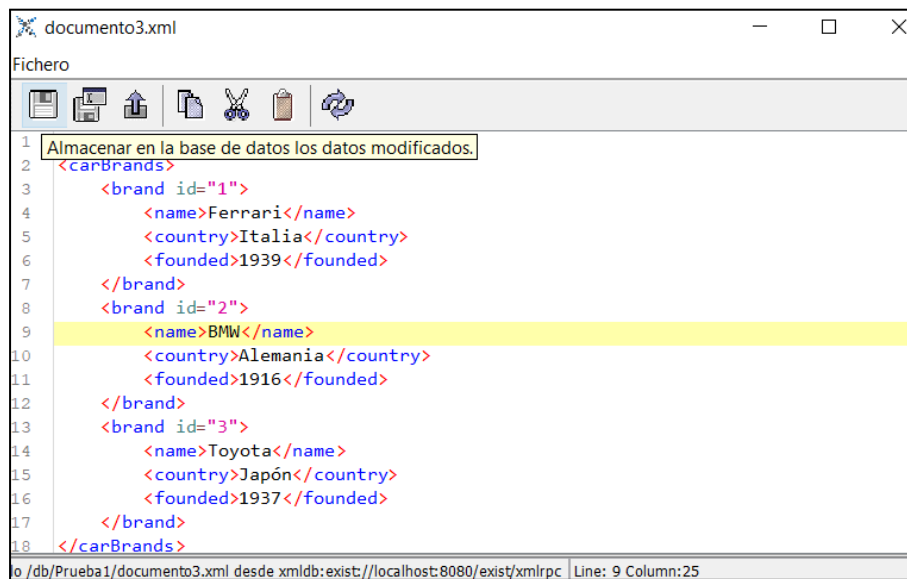
```
<?xml version="1.0" encoding="UTF-8"?>
<dogBreeds>
  <breed id="1">
    <name>Labrador Retriever</name>
    <origin>Canadá</origin>
    <size>Grande</size>
  </breed>
  <breed id="2">
    <name>Bulldog Francés</name>
    <origin>Francia</origin>
    <size>Pequeño</size>
  </breed>
  <breed id="3">
    <name>Pastor Alemán</name>
    <origin>Alemania</origin>
    <size>Grande</size>
  </breed>
</dogBreeds>
```



Lo guardamos y lo cerramos.

DOCUMENTO 3:

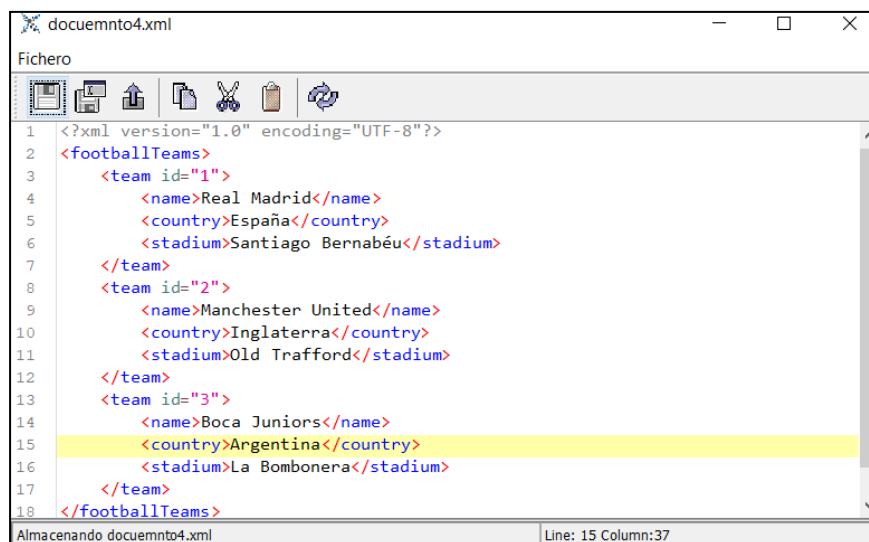
```
<?xml version="1.0" encoding="UTF-8"?>
<carBrands>
  <brand id="1">
    <name>Ferrari</name>
    <country>Italia</country>
    <founded>1939</founded>
  </brand>
  <brand id="2">
    <name>BMW</name>
    <country>Alemania</country>
    <founded>1916</founded>
  </brand>
  <brand id="3">
    <name>Toyota</name>
    <country>Japón</country>
    <founded>1937</founded>
  </brand>
</carBrands>
```



Lo guardamos y lo cerramos.

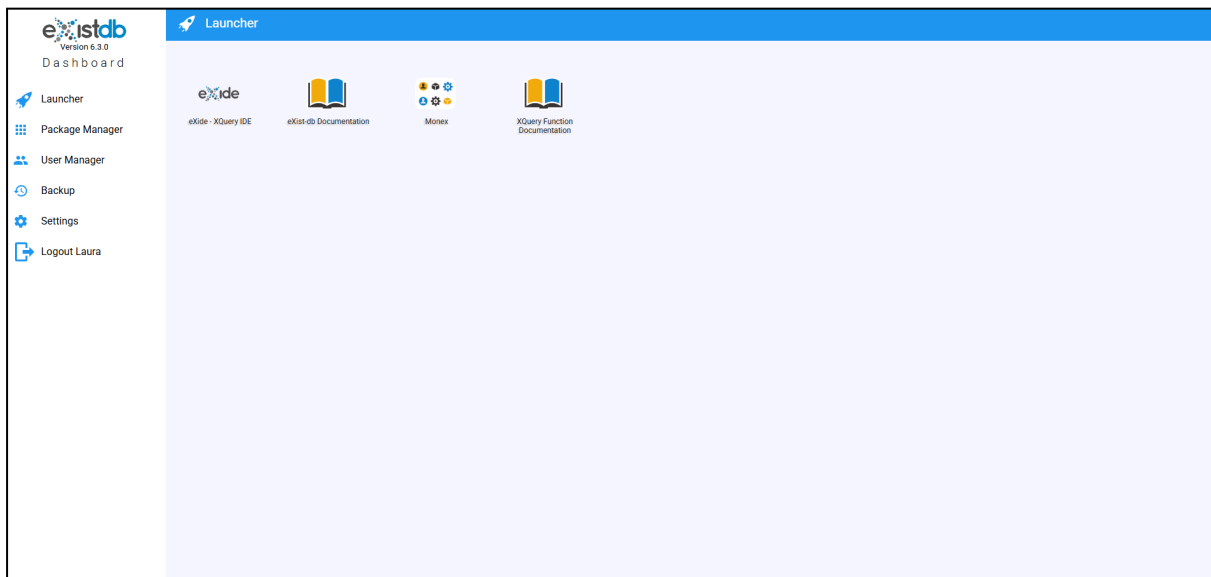
DOCUMENTO 4:

```
<?xml version="1.0" encoding="UTF-8"?>
<footballTeams>
  <team id="1">
    <name>Real Madrid</name>
    <country>España</country>
    <stadium>Santiago Bernabéu</stadium>
  </team>
  <team id="2">
    <name>Manchester United</name>
    <country>Inglaterra</country>
    <stadium>Old Trafford</stadium>
  </team>
  <team id="3">
    <name>Boca Juniors</name>
    <country>Argentina</country>
    <stadium>La Bombonera</stadium>
  </team>
</footballTeams>
```



Lo guardamos y lo cerramos.

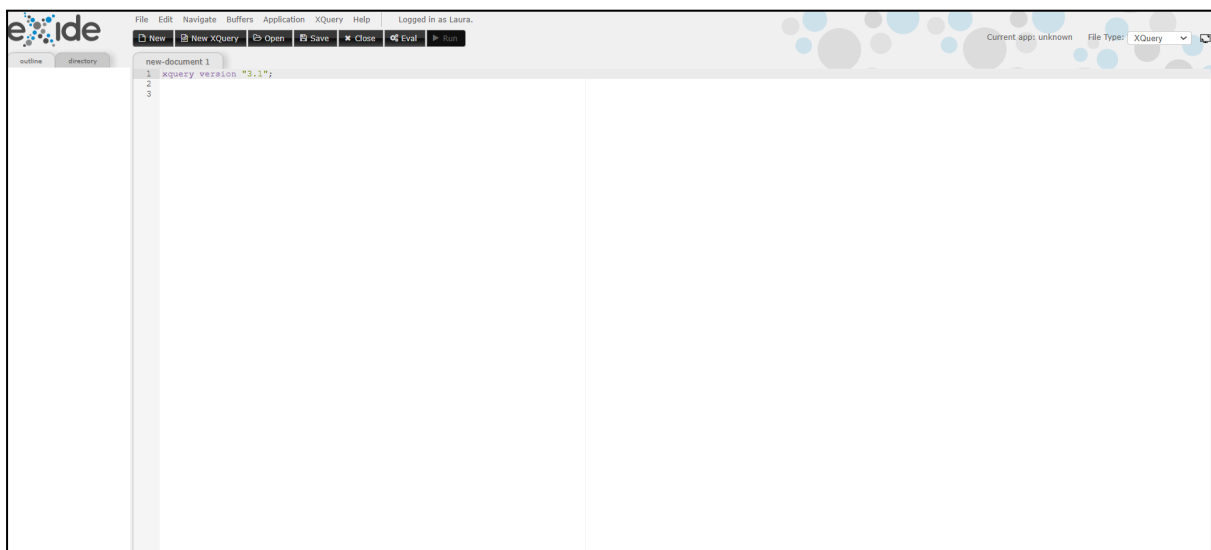
Una vez que tenemos los 4 documentos completos nos vamos otra vez a la web:



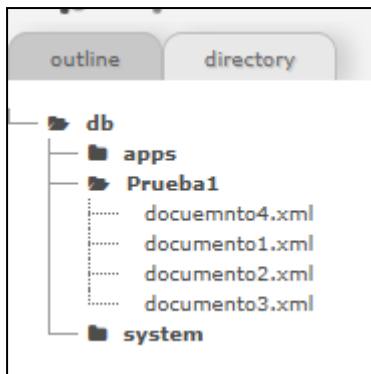
Vamos a hacer las consultas:



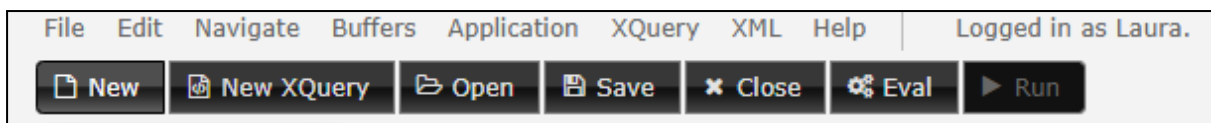
Nos metemos aquí:



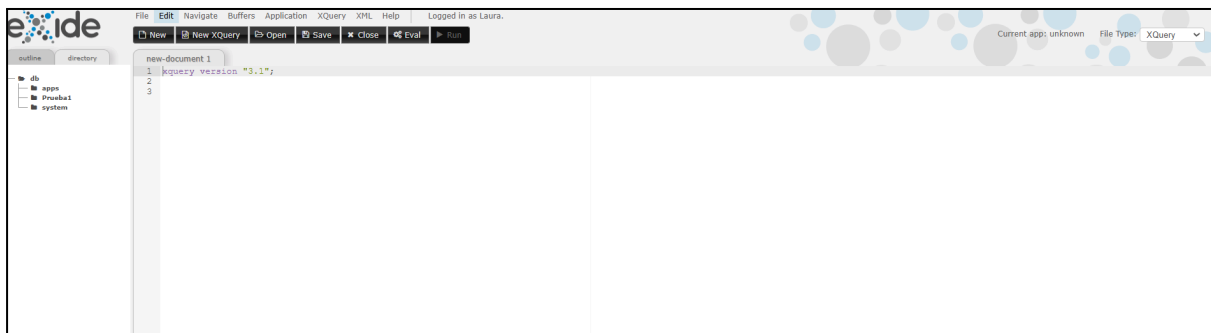
Nos aparece esta ventana de aquí, una vez aquí arriba a la izquierda le damos a directory y buscamos nuestra colección:



Aquí nos aparece todo lo he hemos creado:



En este menú le damos a “New XQuery”:

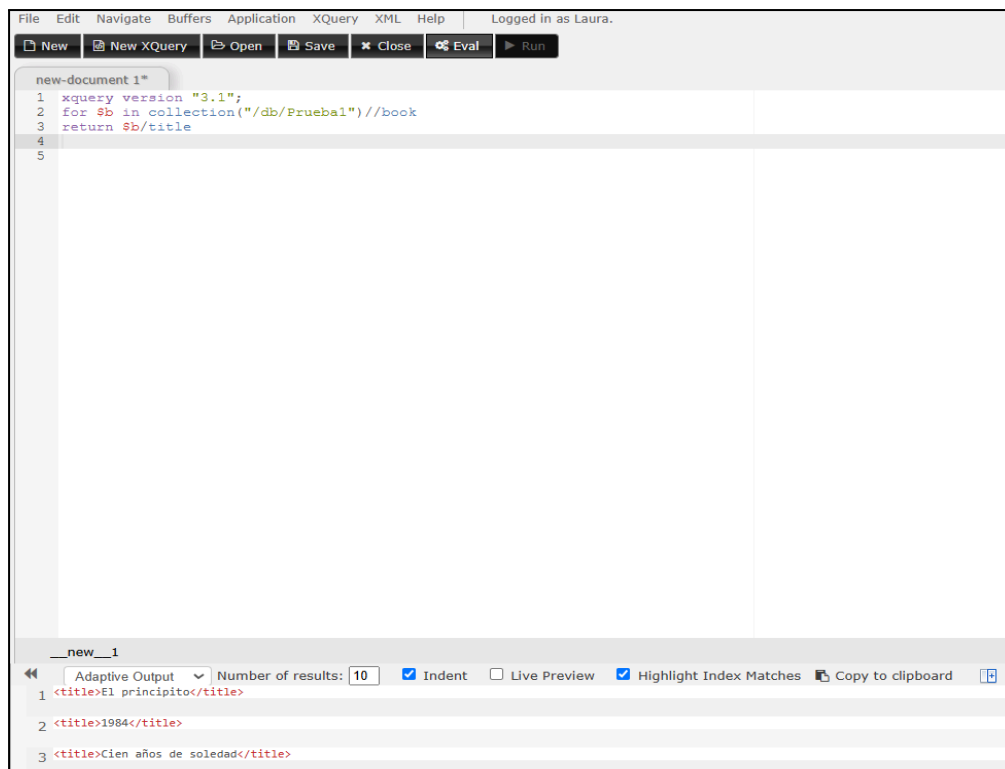


Nos aparecerá esta ventana, vamos a ponerle una consulta al primer documento:

DOCUMENTO 1:

```
for $b in collection("/db/Prueba1")
return $b/title
```

Esta consulta nos devolverá los títulos:



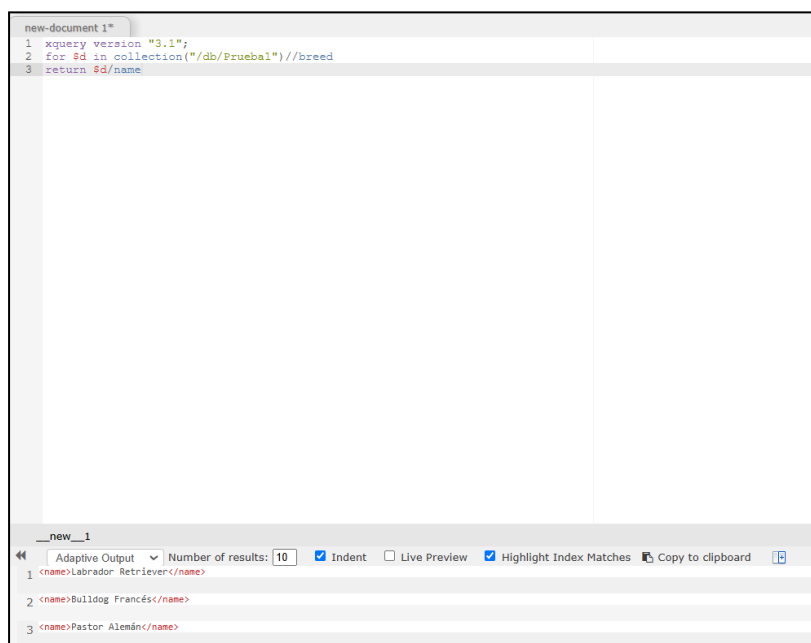
Si no aparece la consulta nada más ponerla le damos a “Eval”.

Vamos a hacer una consulta para cada documento:

DOCUMENTO 2:

```
for $d in collection("/db/Prueba1")//breed
return $d/name
```

Esta consulta nos devolverá los nombres de las razas:



DOCUMENTO 3:

```
for $c in collection("/db/Prueba1")//brand
where $c/country = "Japón"
return $c/name
```

Esta consulta devolverá el nombre del coche que sea de Japón:



The screenshot shows a document editor window titled "new-document 1*". The editor contains the following XQuery code:

```
1 xquery version "3.1";
2 for $c in collection("/db/Prueba1")//brand
3 where $c/country = "Japón"
4 return $c/name
```

Below the editor, there is a toolbar with a left arrow, a dropdown menu set to "Adaptive Output", a "Number of results:" field set to "10", a checked "Indent" checkbox, and an unchecked "Live Preview" checkbox. The output area shows the result of the query:

```
1 <name>Toyota</name>
```

DOCUMENTO 4:

```
for $t in collection("/db/Prueba1")//team
where $t/country = "España"
return $t/name
```

Esta consulta te devolverá el nombre del equipo que sea de España:



The screenshot shows a MongoDB xquery interface. The top pane, titled 'new-document 1*', contains the following xquery code:

```
1 xquery version "3.1";  
2 for $t in collection("/db/Prueba1")//team  
3 where $t/country = "España"  
4 return $t/name  
5
```

The bottom pane, titled '__new__1', shows the result of the query. It includes a control bar with 'Adaptive Output' (a dropdown menu), 'Number of results: 10', and a checked checkbox. Below this, the first result is displayed as:

```
1 <name>Real Madrid</name>
```

Por si estas consultas no son suficientes vamos a crear otra consulta más para cada documento:

DOCUMENTO 1:

```
for $b in collection("/db/Prueba1/libros.xml")//book
where $b/year > 1950
return concat($b/title, " - ", $b/author)
```

Esta consulta saca el título y autor de los libros que su año sea mayor al 1950:



The screenshot shows a document editor with two tabs: 'new-document 1*' and 'documento1.xml'. The 'new-document 1*' tab is active and contains the following XQuery code:

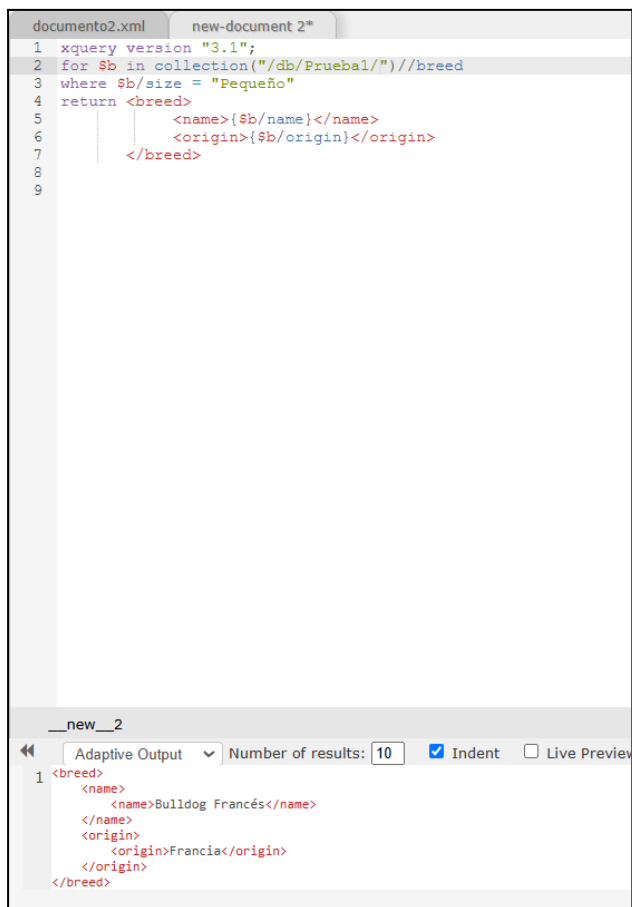
```
1 xquery version "3.1";
2 for $b in collection("/db/Prueba1")//book
3 where $b/year > 1950
4 return concat($b/title, " - ", $b/author)
5
```

At the bottom of the editor, there is a control bar with a left arrow, a dropdown menu set to 'Adaptive Output', a 'Number of results:' field set to '10', and a checked 'Indent' checkbox. Below this, the first result is displayed as a string: '1 "Cien años de soledad - Gabriel García Márquez"'. The text 'Cien años de soledad' is in green and '- Gabriel García Márquez' is in black.

DOCUMENTO 2:

```
for $b in collection("/db/Prueba1/")//breed
where $b/size = "Pequeño"
return <breed>
  <name>{$b/name}</name>
  <origin>{$b/origin}</origin>
</breed>
```

Esta consulta saca las razas de perros pequeños y su origen:



The screenshot shows a code editor with two tabs: 'documento2.xml' and 'new-document 2*'. The 'new-document 2*' tab is active and contains the following XQuery code:

```
1 xquery version "3.1";
2 for $b in collection("/db/Prueba1/")//breed
3 where $b/size = "Pequeño"
4 return <breed>
5   <name>{$b/name}</name>
6   <origin>{$b/origin}</origin>
7   </breed>
8
9
```

Below the code editor, there is a preview window titled '__new__2'. It shows the result of the XQuery, which is an XML document with one <breed> element. The <breed> element has two children: <name> and <origin>. The <name> element has the value 'Bulldog Francés' and the <origin> element has the value 'Francia'.

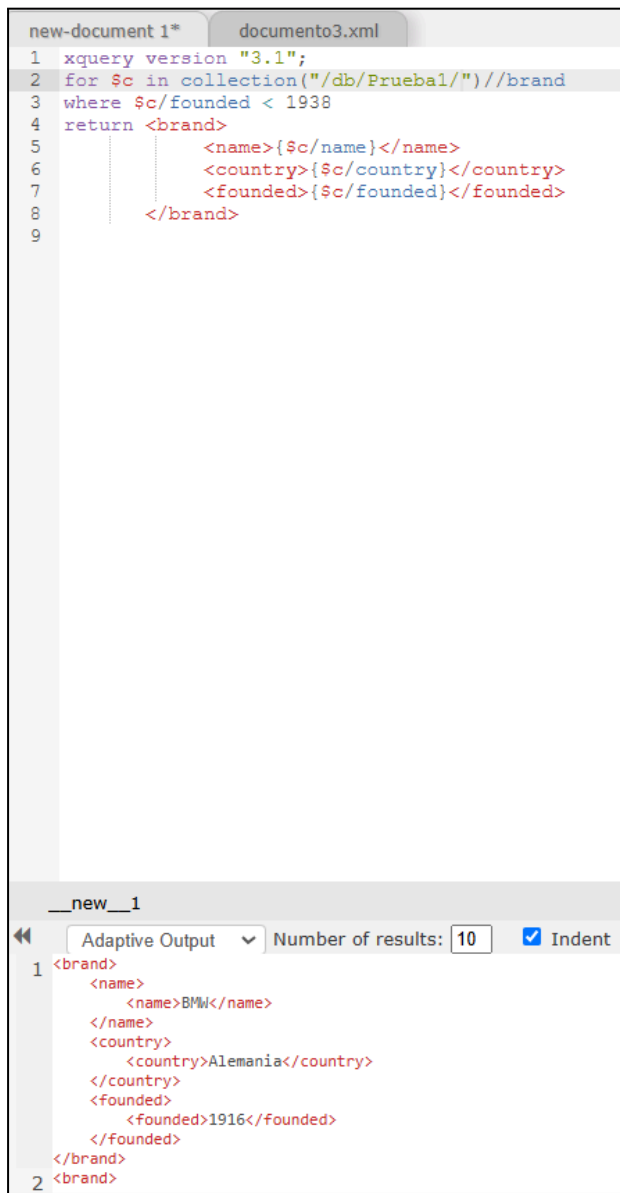
Adaptive Output | Number of results: 10 | ☒ Indent | ☐ Live Preview

```
1 <breed>
  <name>
    <name>Bulldog Francés</name>
  </name>
  <origin>
    <origin>Francia</origin>
  </origin>
</breed>
```


DOCUMENTO 3:

```
for $c in collection("/db/Prueba1/")//brand
where $c/founded < 1938
return <brand>
<name>{$c/name}</name>
<country>{$c/country}</country>
<founded>{$c/founded}</founded>
</brand>
```

Esta consulta es para obtener las marcas fundadas antes de 1938:



The screenshot shows a document editor with two tabs: 'new-document 1*' and 'documento3.xml'. The 'documento3.xml' tab is active and contains the following XQuery code:

```
1 xquery version "3.1";
2 for $c in collection("/db/Prueba1/")//brand
3 where $c/founded < 1938
4 return <brand>
5     <name>{$c/name}</name>
6     <country>{$c/country}</country>
7     <founded>{$c/founded}</founded>
8 </brand>
9
```

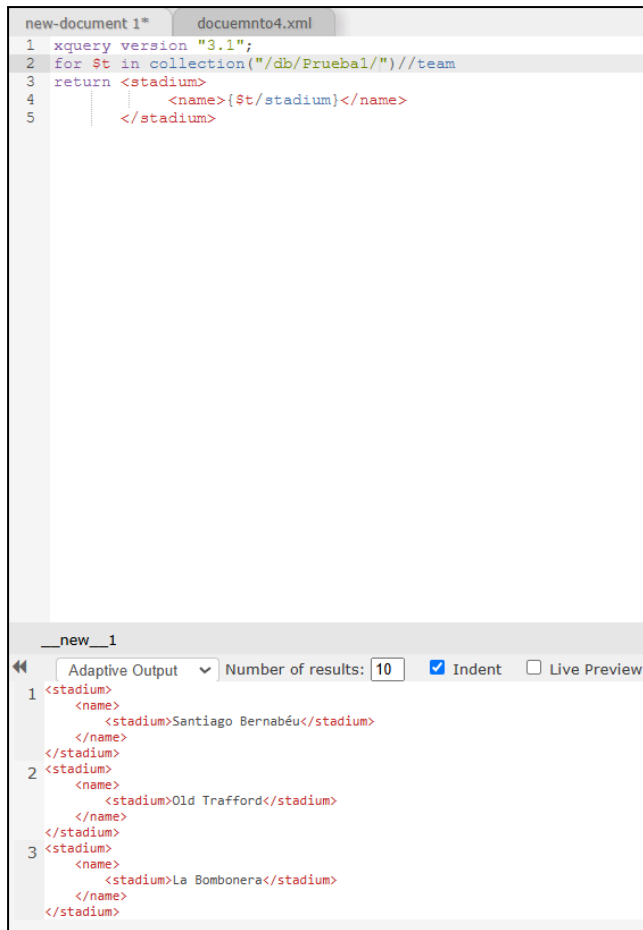
Below the code editor, there is a preview window titled 'new_1'. It shows the XML output of the query. The output is an XML document with a root element 'brand' (line 1) containing three child elements: 'name' (line 2), 'country' (line 3), and 'founded' (line 4). The values are 'BMW' (line 2), 'Alemania' (line 3), and '1916' (line 4). The output is formatted with indentation. The preview window also includes a dropdown menu for 'Adaptive Output', a 'Number of results: 10' field, and a checked 'Indent' checkbox.

```
1 <brand>
2   <name>BMW</name>
3   <country>Alemania</country>
4   <founded>1916</founded>
5 </brand>
6 <brand>
```

DOCUMENTO 4:

```
for $t in collection("/db/Prueba1/")//team
return <stadium>
<name>{$t/stadium}</name>
</stadium>
```

Esta consulta es para obtener todos los estadios:



The screenshot shows a document editor with two tabs: 'new-document 1*' and 'docuemnto4.xml'. The 'docuemnto4.xml' tab is active and contains the following XQuery code:

```
1 xquery version "3.1";
2 for $t in collection("/db/Prueba1/")//team
3 return <stadium>
4       <name>{$t/stadium}</name>
5       </stadium>
```

Below the code editor, there is a preview section titled '___new___1'. It includes a dropdown menu set to 'Adaptive Output', a 'Number of results' field set to '10', and two checkboxes: 'Indent' (checked) and 'Live Preview' (unchecked). The preview displays the XML output of the query, showing three stadium entries:

```
1 <stadium>
  <name>
    <stadium>Santiago Bernabéu</stadium>
  </name>
</stadium>
2 <stadium>
  <name>
    <stadium>Old Trafford</stadium>
  </name>
</stadium>
3 <stadium>
  <name>
    <stadium>La Bombonera</stadium>
  </name>
</stadium>
```