



CICLO: [DAM]
**MÓDULO DE [LENGUAJE DE MARCAS
Y SISTEMAS DE GESTIÓN DE
INFORMACIÓN]**

[Tarea N° 06]

Alumno:
[Juan Carlos Filter Martín]
[15456141A]

Contenido

1. Documentos que se adjuntan a este informe.....	4
2. RA06_i) Se han utilizado herramientas para gestionar la información almacenada en bases de datos nativas.....	4
Instalación base de datos XML eXist.....	4
Descargar, instalar JDK y agregar a las variables de entornos.....	4
Descargar XML eXist BD.....	6
Instalar XML eXist BD.....	7
3. RA06_f) Se han utilizado sistemas gestores de bases de datos relacionales en el almacenamiento de información en formatos de intercambio de datos.....	9
Acceder a la aplicación, comprobar puertos y arrancar el servicio.....	9
4. RA06_c) Se han establecido tecnologías eficientes de almacenamiento de información en función de sus características.....	13
Documento XML.....	13
5. RA06_g) Se han utilizado técnicas específicas para crear documentos de intercambio de datos a partir de información almacenada en bases de datos relacionales.....	15
Crear una colección.....	15
Subimos el documento XML a la colección.....	16
6. RA06_h) Se han identificado las características de los sistemas.....	17
1-Obtener la información de los miembros del personal cuya permanencia se cuente en días..	17
2-Obtener la información de los miembros del personal cuya permanencia se cuente en meses	18
3-Obtener la información de los miembros del personal cuyo salario sea mayor de 10000.....	19
4-Obtener la información de los miembros del personal cuyo salario sea mayor de 10000 y su categoría distinta de "Analista de Desarrollo".....	20
5-Obtener la información de los miembros del personal cuya categoría sea distinta de "Diseñador" o "Programador"	21
7. RA06_d) Se han identificado lenguajes y herramientas para el tratamiento y almacenamiento de información y su inclusión en documentos de intercambio de datos.....	22
Consulta 1 utilizado el lenguaje XQuery.....	22
Consulta 2 utilizado el lenguaje XQuery.....	22
Consulta 3 utilizado el lenguaje XQuery.....	22
Consulta 4 utilizado el lenguaje XQuery.....	22
Consulta 5 utilizado el lenguaje XQuery.....	22

1. Documentos que se adjuntan a este informe.

A continuación se detallan los documentos que componen la presente entrega de la tarea:

1. Informe de elaboración de la tarea.
2. Fichero XML baseDatos_xml.xml

2. RA06_i) Se han utilizado herramientas para gestionar la información almacenada en bases de datos nativas.

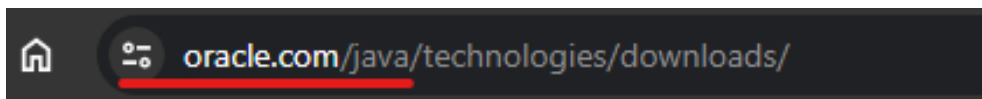
Instalación base de datos XML eXist

Descargar, instalar JDK y agregar a las variables de entornos

- ✖ Lo primero es tener el JDK instalado para ello podemos comprobarlo en la terminal: **java -versión**

```
C:\Users\Juan Carlos>java -version
java version "22" 2024-03-19
Java(TM) SE Runtime Environment (build 22+36-2370)
Java HotSpot(TM) 64-Bit Server VM (build 22+36-2370, mixed mode, sharing)
```

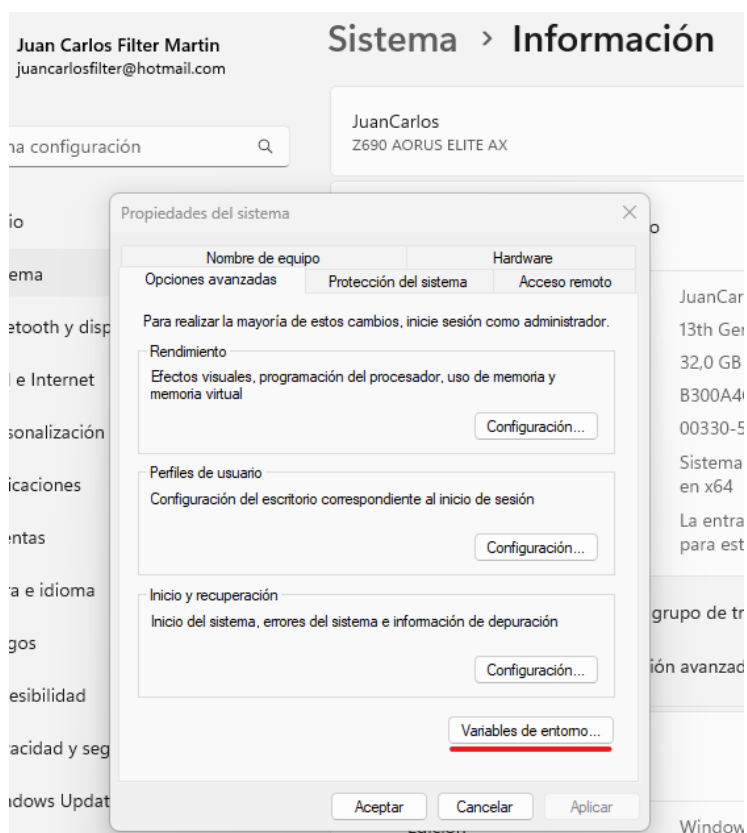
- ✖ Si no estuviera instalado tendríamos que ir a la web de oracle > java downloads



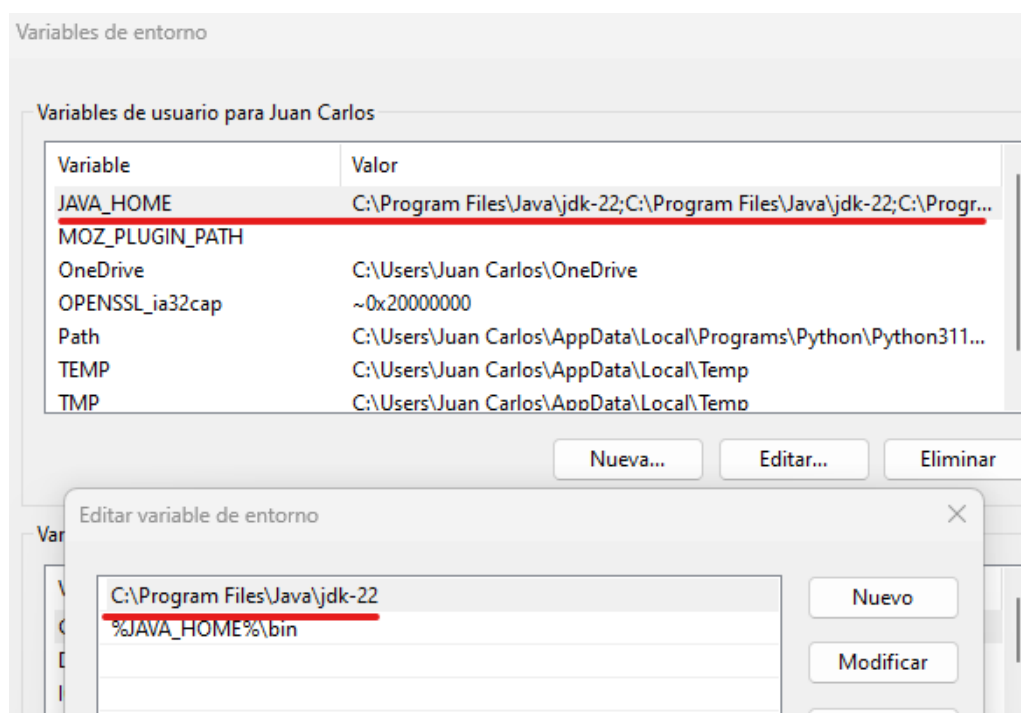
- ✖ Y descargar la ultima versión del JDK

Java downloads			Tools and resources	Java archive	
JDK 22	JDK 21	JDK 17	GraalVM for JDK 22	GraalVM for JDK 21	GraalVM for JDK 17
JDK Development Kit 22 downloads					
JDK 22 binaries are free to use in production and free to redistribute, at no cost, under the Oracle No-Fee Terms and Conditions (NFTC).					
JDK 22 will receive updates under these terms, until September 2024, when it will be superseded by JDK 23.					
Linux	macOS	Windows			
Product/file description		File size	Download		
x64 Compressed Archive		185.52 MB	https://download.oracle.com/java/22/latest/jdk-22_windows-x64_bin.zip (sha256)		
x64 Installer		163.91 MB	https://download.oracle.com/java/22/latest/jdk-22_windows-x64_bin.exe (sha256)		
x64 MSI Installer		162.07MB	https://download.oracle.com/java/22/latest/jdk-22_windows-x64_bin.msi (sha256)		

- ✖ Posteriormente se va a añadir a las variables de entorno desde las propiedades del equipo. **(esto no es necesario)**

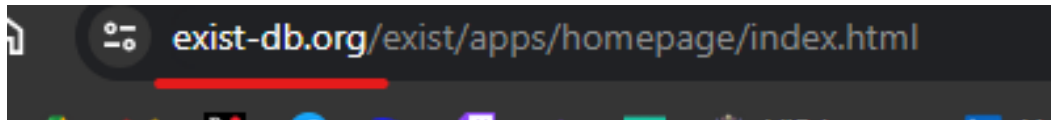


- ✖ En la variable JAVA_HOME > editar... > y agregamos la ruta donde se encuentra el JDK

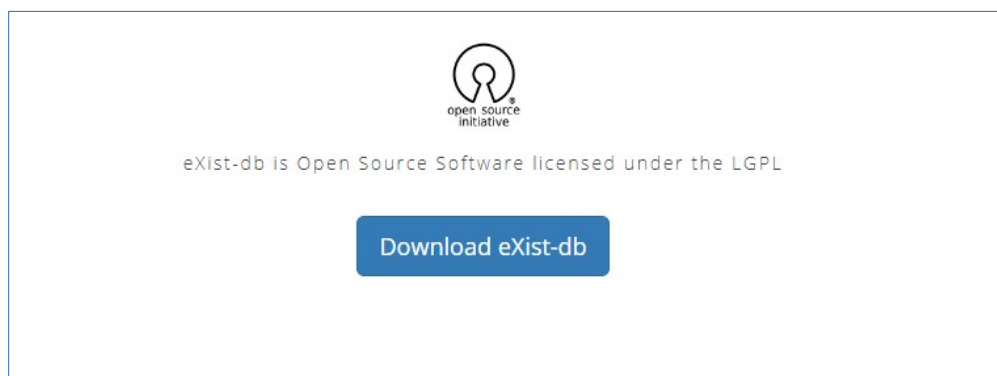


Descargar XML eXist BD

- ✖ Ahora podremos instalar la base de datos XML eXist, para ello vamos a la siguiente web



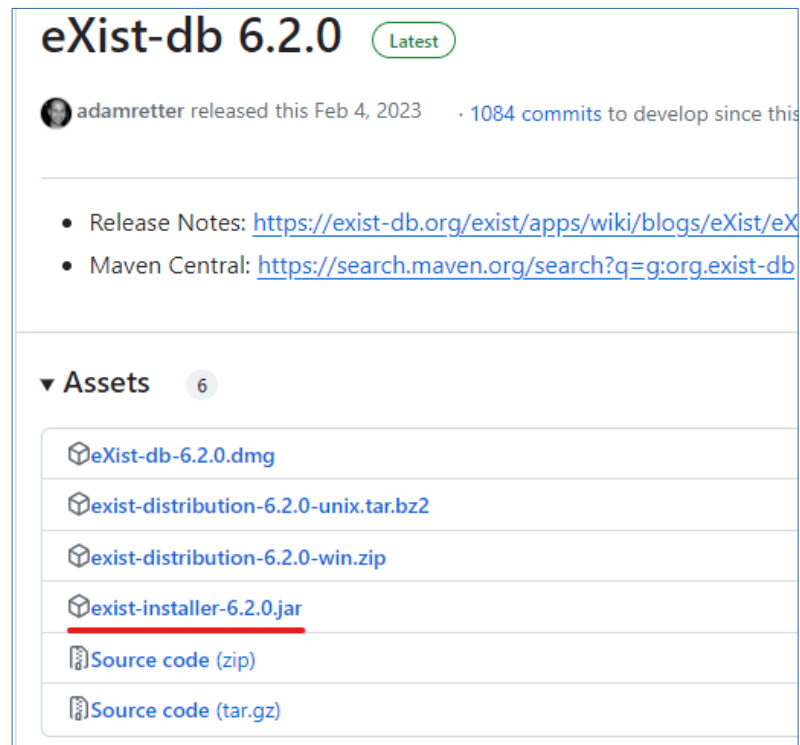
- ✖ Download eXist-db



- ✖ Entramos en última versión estable



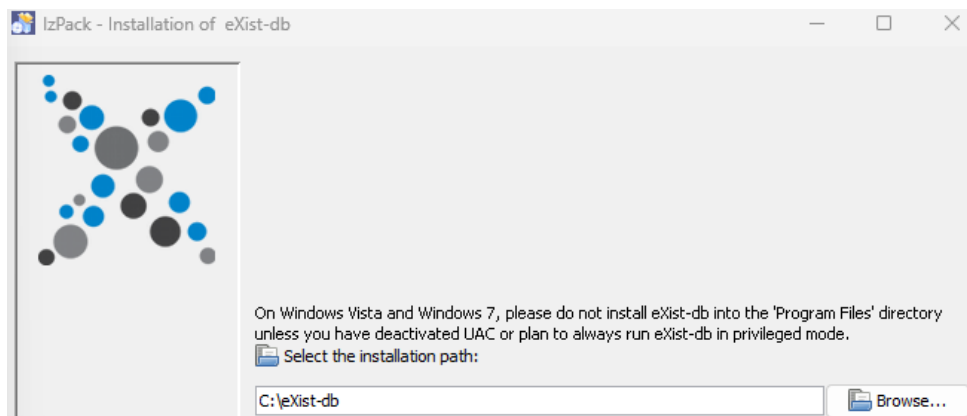
- ✖ Lo que nos llevará a github y descargaremos el .jar



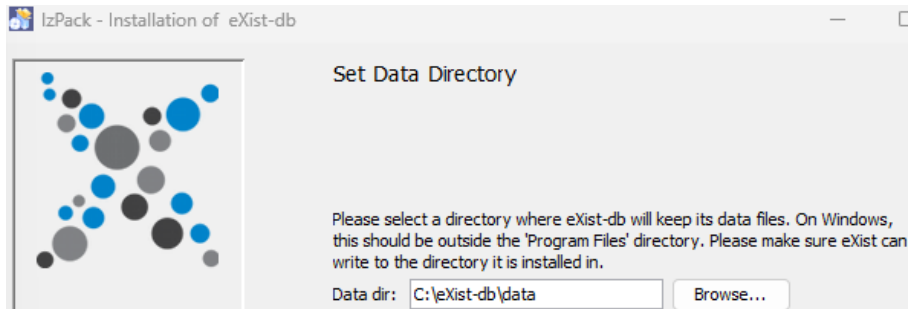
Instalar XML eXist BD

- ✖ Ejecutamos el .jar y procedemos a la instalación, simplemente seguir los pasos del asistente. Algunos de estas ventanas son:

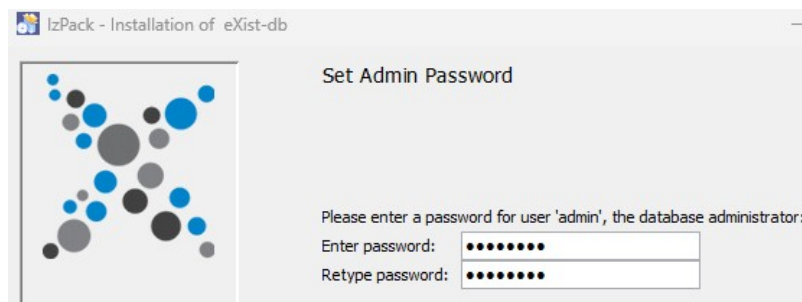
→ Elegir directorio de instalación



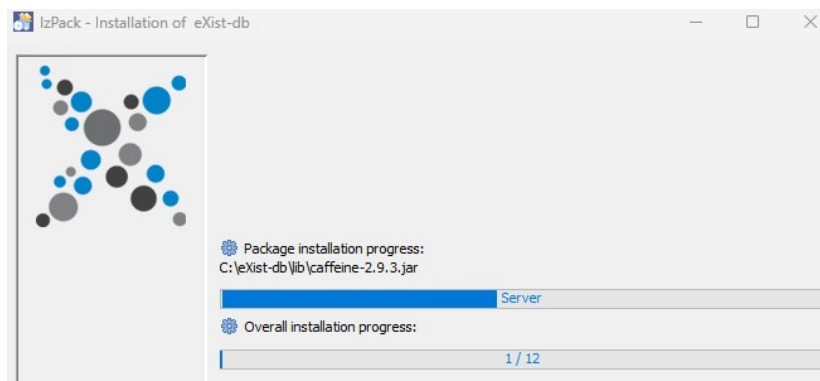
→ Directorio donde se van a guardar los archivos de datos



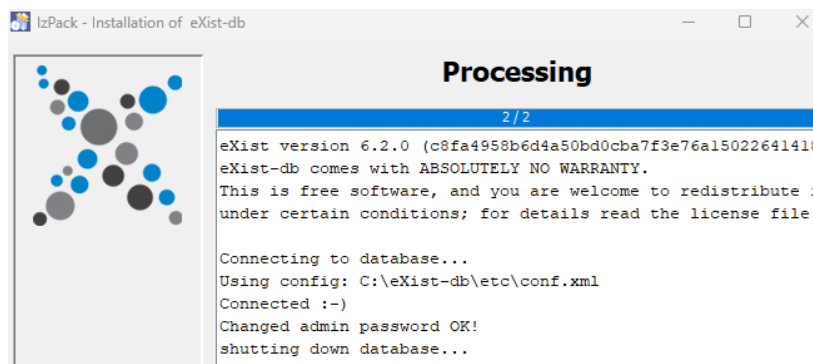
→ Contraseña para el administrador: **20081992**



→ Instalación en curso:



→ Se configura: crea la base de datos, se cambia la contraseña de admin y apaga BD

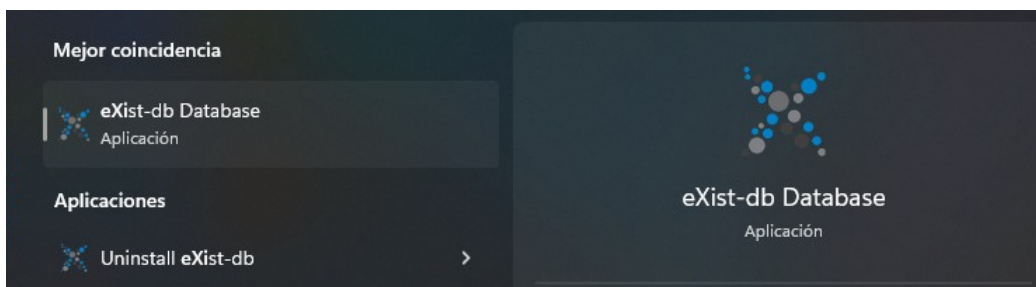


× Con esto ya tendríamos la base de datos XML eXist instalada

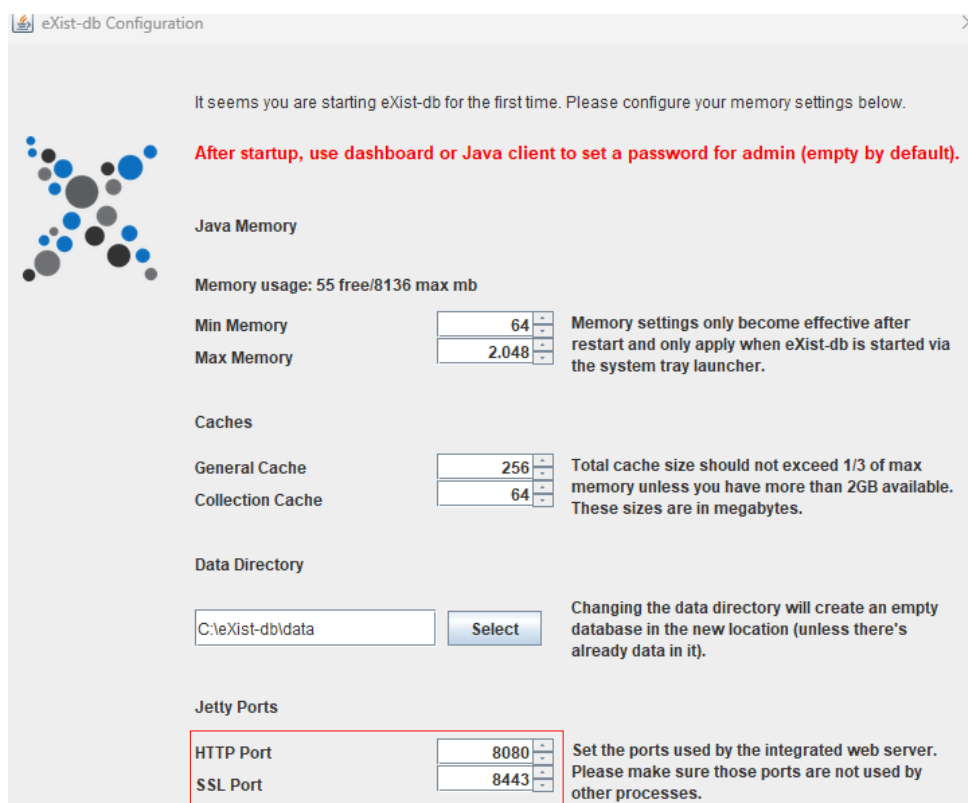
3. RA06_f) Se han utilizado sistemas gestores de bases de datos relacionales en el almacenamiento de información en formatos de intercambio de datos.

Acceder a la aplicación, comprobar puertos y arrancar el servicio

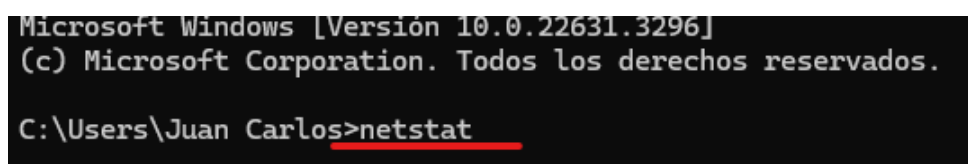
- ✗ Iniciamos la aplicación.



- ✗ Y vemos el puerto que usa eXist y comprobamos desde la terminal los puertos usados.



- ✗ Se puede comprobar que el puerto que usa por defecto eXist no está en uso. Mediante el comando **netstat** podemos ver los puertos con conexiones activas

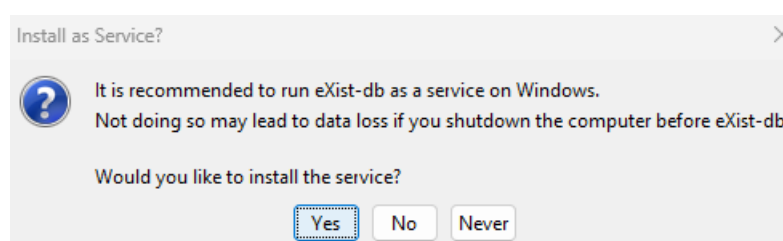



```
C:\Users\Juan Carlos>netstat

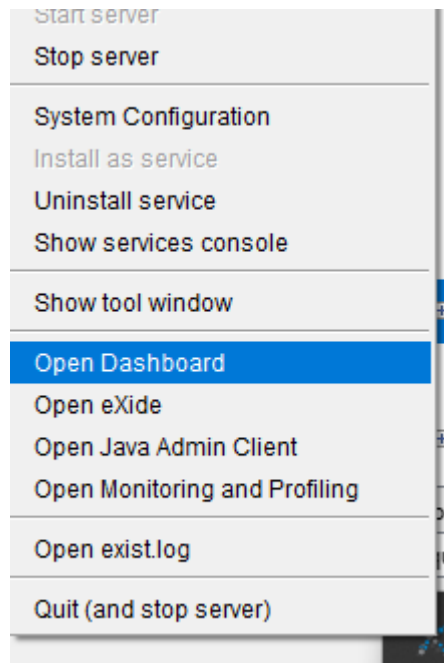
Conexiones activas

Proto  Dirección local      Dirección remota      Estado
TCP    127.0.0.1:1844        lm:56842              ESTABLISHED
TCP    127.0.0.1:6463        lm:56808              ESTABLISHED
TCP    127.0.0.1:9010        lm:56771              ESTABLISHED
TCP    127.0.0.1:9100        lm:56803              ESTABLISHED
TCP    127.0.0.1:28196       lm:56661              ESTABLISHED
TCP    127.0.0.1:28196       lm:56667              ESTABLISHED
TCP    127.0.0.1:28196       lm:56669              ESTABLISHED
TCP    127.0.0.1:28196       lm:56676              ESTABLISHED
TCP    127.0.0.1:28196       lm:56695              ESTABLISHED
TCP    127.0.0.1:49670       lm:56614              ESTABLISHED
TCP    127.0.0.1:56502       lm:65001              ESTABLISHED
TCP    127.0.0.1:56614       lm:49670              ESTABLISHED
TCP    127.0.0.1:56661       lm:28196              ESTABLISHED
TCP    127.0.0.1:56667       lm:28196              ESTABLISHED
TCP    127.0.0.1:56669       lm:28196              ESTABLISHED
TCP    127.0.0.1:56676       lm:28196              ESTABLISHED
TCP    127.0.0.1:56695       lm:28196              ESTABLISHED
TCP    127.0.0.1:56758       lm:56773              ESTABLISHED
TCP    127.0.0.1:56760       lm:56772              ESTABLISHED
TCP    127.0.0.1:56771       lm:9010               ESTABLISHED
TCP    127.0.0.1:56772       lm:56760              ESTABLISHED
TCP    127.0.0.1:56773       lm:56758              ESTABLISHED
TCP    127.0.0.1:56774       lm:56775              ESTABLISHED
TCP    127.0.0.1:56775       lm:56774              ESTABLISHED
TCP    127.0.0.1:56776       lm:56777              ESTABLISHED
TCP    127.0.0.1:56777       lm:56776              ESTABLISHED
```

- × Guardamos la configuración y procederá a hacer las instalaciones necesarias y el servicio.









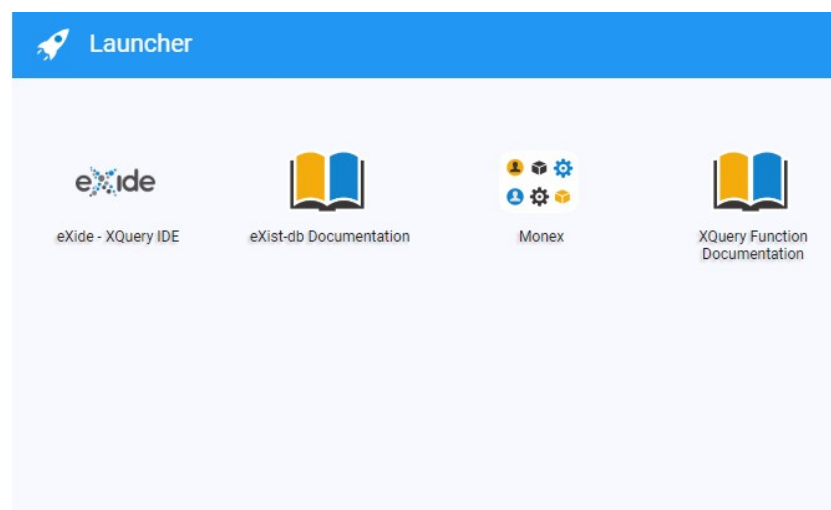
- ✖ Para acceder a eXistdb podemos comprobar que al iniciarlo se queda ejecutado en segundo plano.
- ✖ Entonces para ello pulsamos en el icono > **Open Dashboard**. (Da acceso a un panel para acceder a las utilidades de manejo de eXist)



- ✖ Dentro de este menú si hacemos login de admin tendremos un panel con más opciones como puede ser backup entre otras.

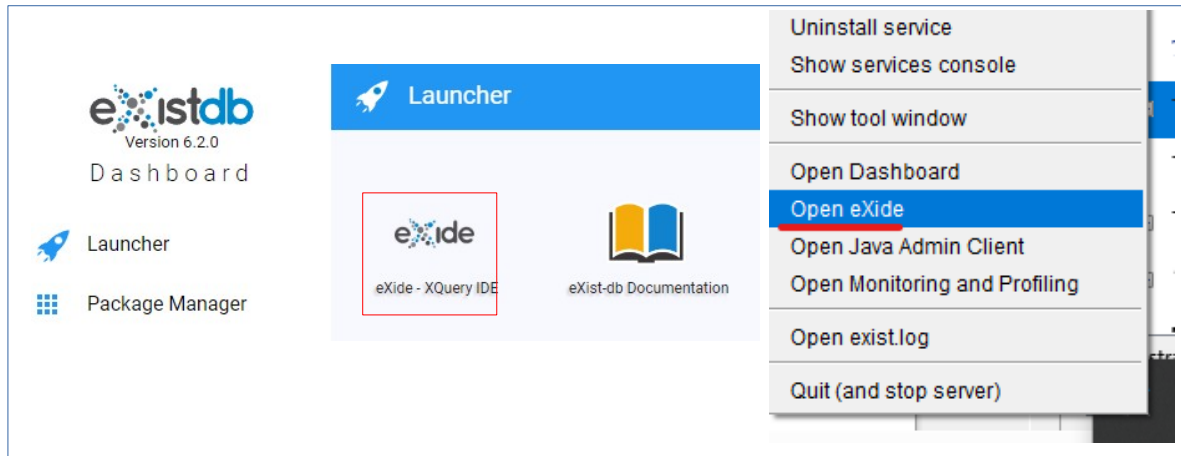


-  Launcher
-  Package Manager
-  User Manager
-  Backup
-  Settings
-  Logout admin



✖ Para iniciar la BD eXist podemos hacerlos de dos formas:

- desde el propio panel **eXide-XQuery IDE**
- desde el icono en segundo plano y pulsar en **Open eXide**



✖ Y nos mostrará un panel como el siguiente:



4. RA06_c) Se han establecido tecnologías eficientes de almacenamiento de información en función de sus características.

Documento XML

```
<?xml version="1.0" encoding="UTF-8"?>
<personal_havok_summers>
  <empleado idEmpresa="1AAAB09">
    <categoria>Programador Junior</categoria>
    <salario>12.000</salario>
    <permanencia tiempo="dias">1000</permanencia>
  </empleado>
  <empleado idEmpresa="3AXF01">
    <categoria>Programador Junior</categoria>
    <salario>14.500</salario>
    <permanencia tiempo="años">2</permanencia>
  </empleado>
  <empleado idEmpresa="9XX756A">
    <categoria>Programador Senior</categoria>
    <salario>17.000</salario>
    <permanencia tiempo="meses">8</permanencia>
  </empleado>
  <empleado idEmpresa="9$DD90">
    <categoria>Programador Senior</categoria>
    <salario>16.500</salario>
    <permanencia tiempo="dias">300</permanencia>
  </empleado>
  <empleado idEmpresa="AZ0091">
    <categoria>Jefe de Proyecto</categoria>
    <salario>25.000</salario>
    <permanencia tiempo="años">7</permanencia>
  </empleado>
  <empleado idEmpresa="1">
    <categoria>Gerente</categoria>
    <salario>50.000</salario>
    <permanencia tiempo="años">12</permanencia>
  </empleado>
  <empleado idEmpresa="AS155AA1">
    <categoria>Analista de Desarrollo</categoria>
    <salario>21.300</salario>
    <permanencia tiempo="años">3</permanencia>
  </empleado>
  <empleado idEmpresa="APQ131SL1">
    <categoria>Analista de Desarrollo</categoria>
    <salario>23.250</salario>
    <permanencia tiempo="años">2</permanencia>
  </empleado>
  <empleado idEmpresa="XA8923FH">
    <categoria>Diseñador Web</categoria>
    <salario>17.755</salario>
    <permanencia tiempo="meses">9</permanencia>
  </empleado>
</personal_havok_summers>
```



- × Parte del documento XML

```
baseDatos_xml.xml X
DES > baseDatos_xml.xml > XSLT/XPath for Visual Studio Code > personal_havok_summers
<?xml version="1.0" encoding="UTF-8"?>

<personal_havok_summers>

  <empleado idEmpresa="1AAB09">

    <categoria>Programador Junior</categoria>

    <salario>12.000</salario>

    <permanencia tiempo="dias">1000</permanencia>

  </empleado>

  <empleado idEmpresa="3AXF01">

    <categoria>Programador Junior</categoria>

    <salario>14.500</salario>

    <permanencia tiempo="años">2</permanencia>

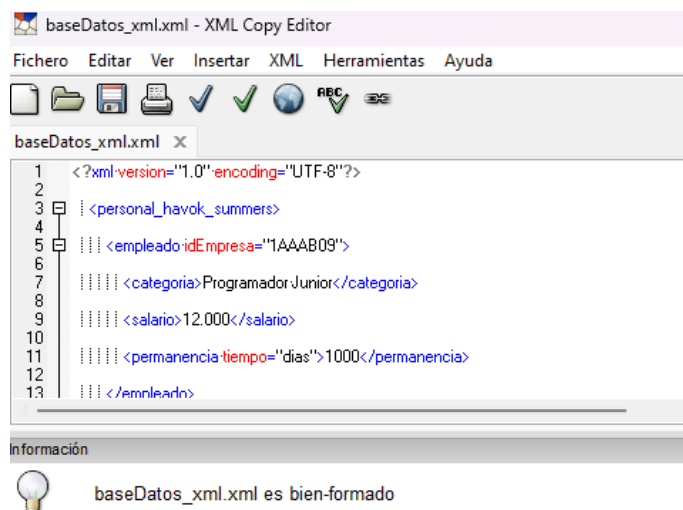
  </empleado>

  <empleado idEmpresa="9XX756A">

    <categoria>Programador Senior</categoria>

    <salario>17.000</salario>
```

- × Está bien formado

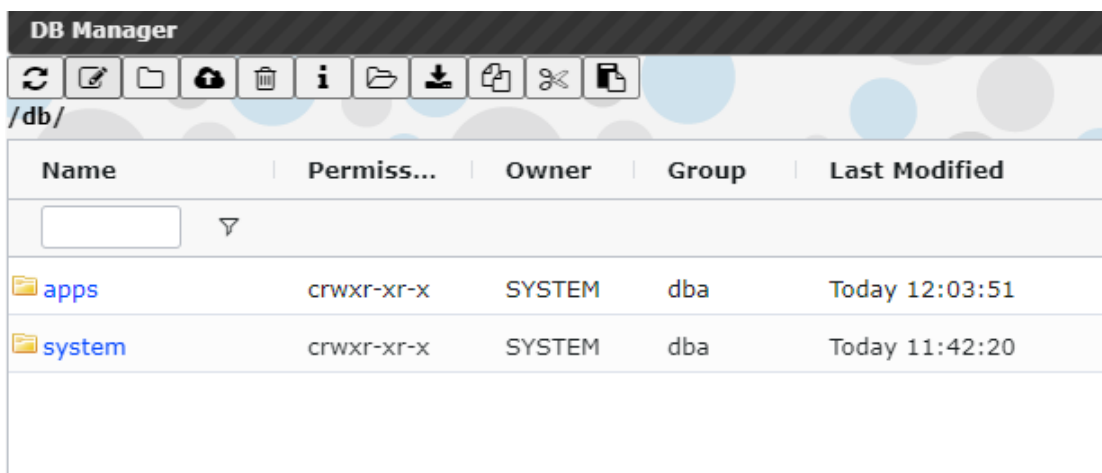


5. RA06_g) Se han utilizado técnicas específicas para crear documentos de intercambio de datos a partir de información almacenada en bases de datos relacionales.

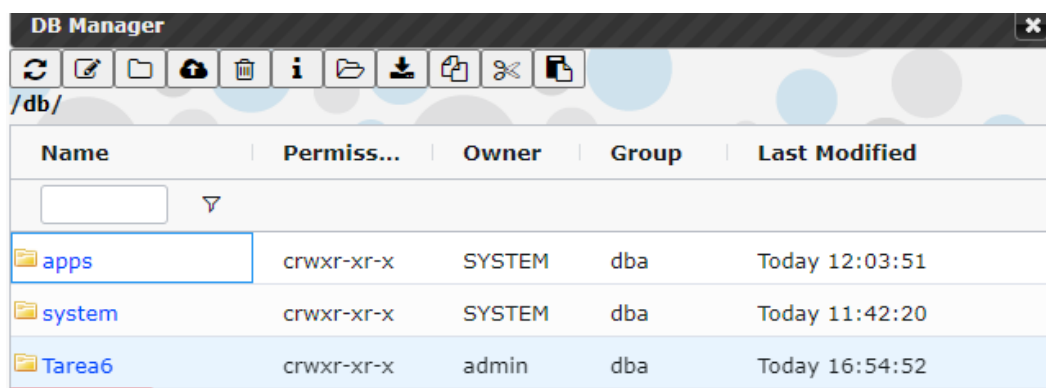
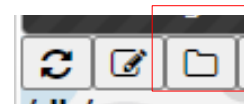
Crear una colección

× Los pasos a seguir para crear una colección son los siguientes:

- Logueado como admin.
- File> Manager

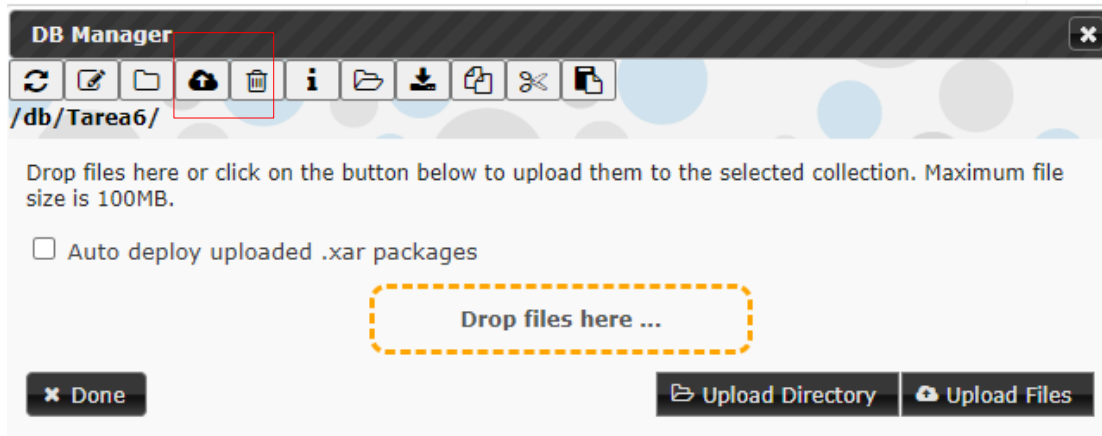


- Pulsamos en el siguiente icono y creamos la colección

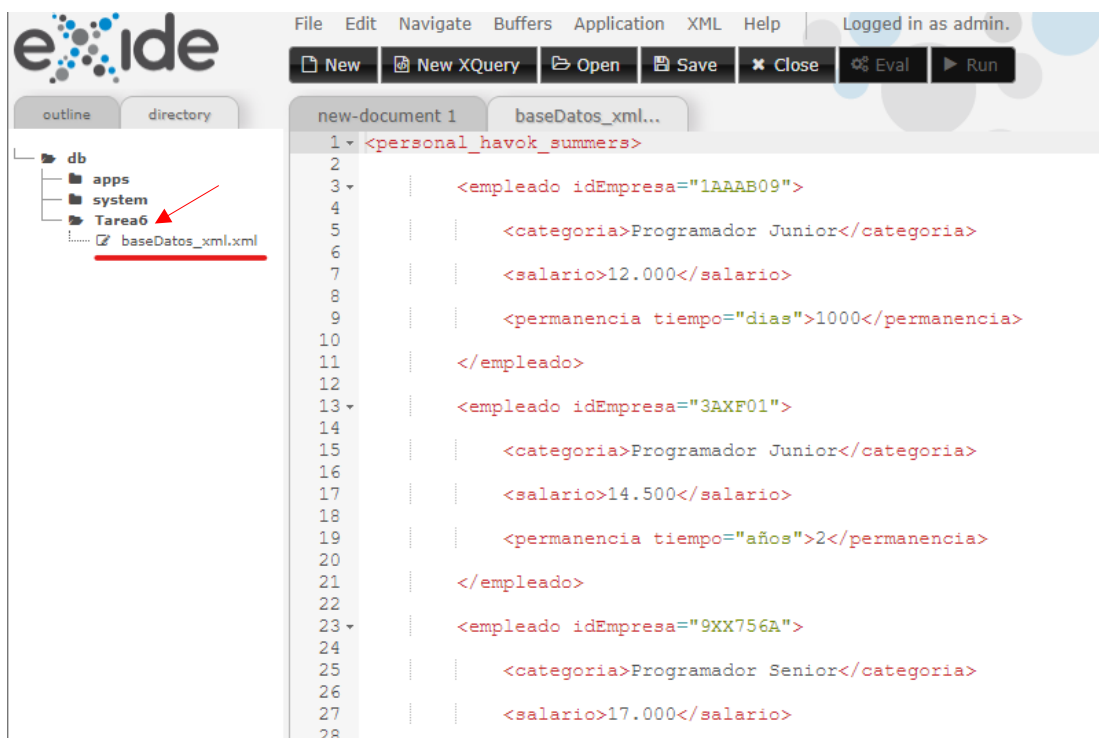
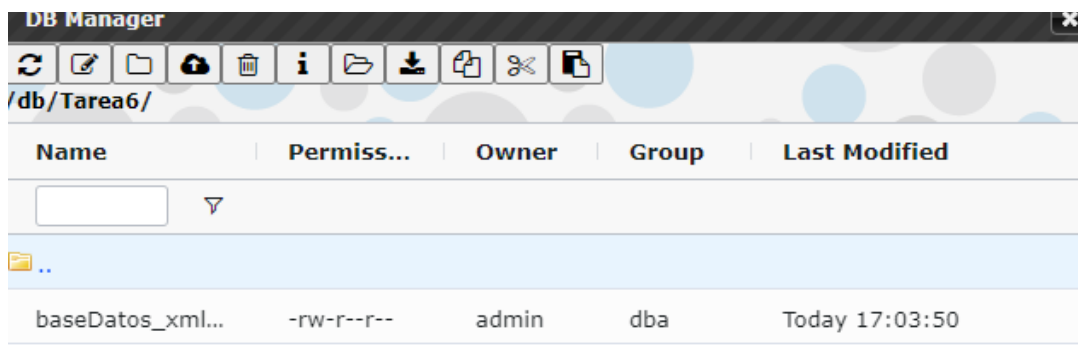


Subimos el documento XML a la colección

- Desde la colección creada pulsamos en el siguiente icono para importar el documento.



- Subimos el fichero y si comprobamos de nuevo la colección, se puede ver como aparece el documento xml



6. RA06_h) Se han identificado las características de los sistemas

1-Obtener la información de los miembros del personal cuya permanencia se cuente en días.

✱ Esta consulta se lee de la siguiente manera:

Almacena la palabra reservada \$empleado lo siguiente: dentro del doc "baseDatos_xml.xml" todo lo que contenga las etiquetas de la siguiente jerarquía /personal_havok_summers/empleado.

Usando esa palabra reservada \$empleado y con la condición where + /permanencia/@tiempo buscamos que la permanencia tiempo sea "**días**".

→ Devuelve \$empleado

Esto es una palabra reservada donde se va a almacenar cada resultado de la expresión XPath

```
baseDatos_xml...  new-document 1*
1 xquery version "3.1";
2 for $empleado in doc("Tarea6/baseDatos_xml.xml")/personal_havok_summers/empleado
3 where $empleado/permanencia/@tiempo = "días"
4 return $empleado
```

__new__1

Adaptive Output Number of results: 10 Indent Live Preview Highlight Index Match

```
1 <empleado idEmpresa="1AAB09">
    <categoria>Programador Junior</categoria>
    <salario>12.000</salario>
    <permanencia tiempo="días">1000</permanencia>
</empleado>
2 <empleado idEmpresa="9$DD90">
    <categoria>Programador Senior</categoria>
    <salario>16.500</salario>
    <permanencia tiempo="días">300</permanencia>
</empleado>
```

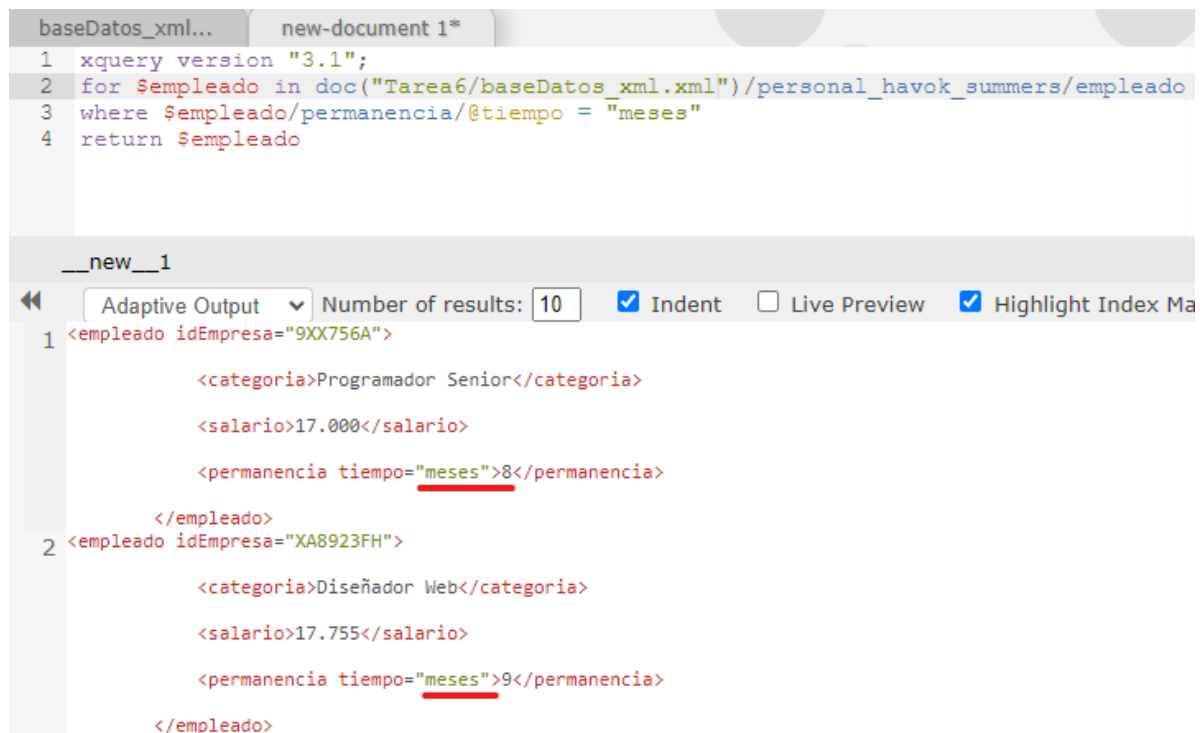

2-Obtener la información de los miembros del personal cuya permanencia se cuente en meses

- × Esta consulta se lee de la siguiente manera (exactamente igual a la anterior pero cambiando días por meses):

Almacena la palabra reservada \$empleado lo siguiente: dentro del doc "baseDatos_xml.xml" todo lo que contenga las etiquetas de la siguiente jerarquía /personal_havok_summers/empleado.

Usando esa palabra reservada \$empleado y con la **condición where** + /permanencia/@tiempo buscamos que la permanencia tiempo sea **"meses"**.

→ Devuelve \$empleado



The screenshot shows an XQuery editor with two tabs: 'baseDatos_xml...' and 'new-document 1*'. The query in the editor is:

```
1 xquery version "3.1";
2 for $empleado in doc("Tarea6/baseDatos_xml.xml")/personal_havok_summers/empleado
3 where $empleado/permanencia/@tiempo = "meses"
4 return $empleado
```

Below the query, the results are displayed in a table with two columns: 'Adaptive Output' and 'Number of results: 10'. The results are shown in a tree view with the following structure:

```
1 <empleado idEmpresa="9XX756A">
  <categoria>Programador Senior</categoria>
  <salario>17.000</salario>
  <permanencia tiempo="meses">8</permanencia>
</empleado>
2 <empleado idEmpresa="XA8923FH">
  <categoria>Diseñador Web</categoria>
  <salario>17.755</salario>
  <permanencia tiempo="meses">9</permanencia>
</empleado>
```

3-Obtener la información de los miembros del personal cuyo salario sea mayor de 10000.

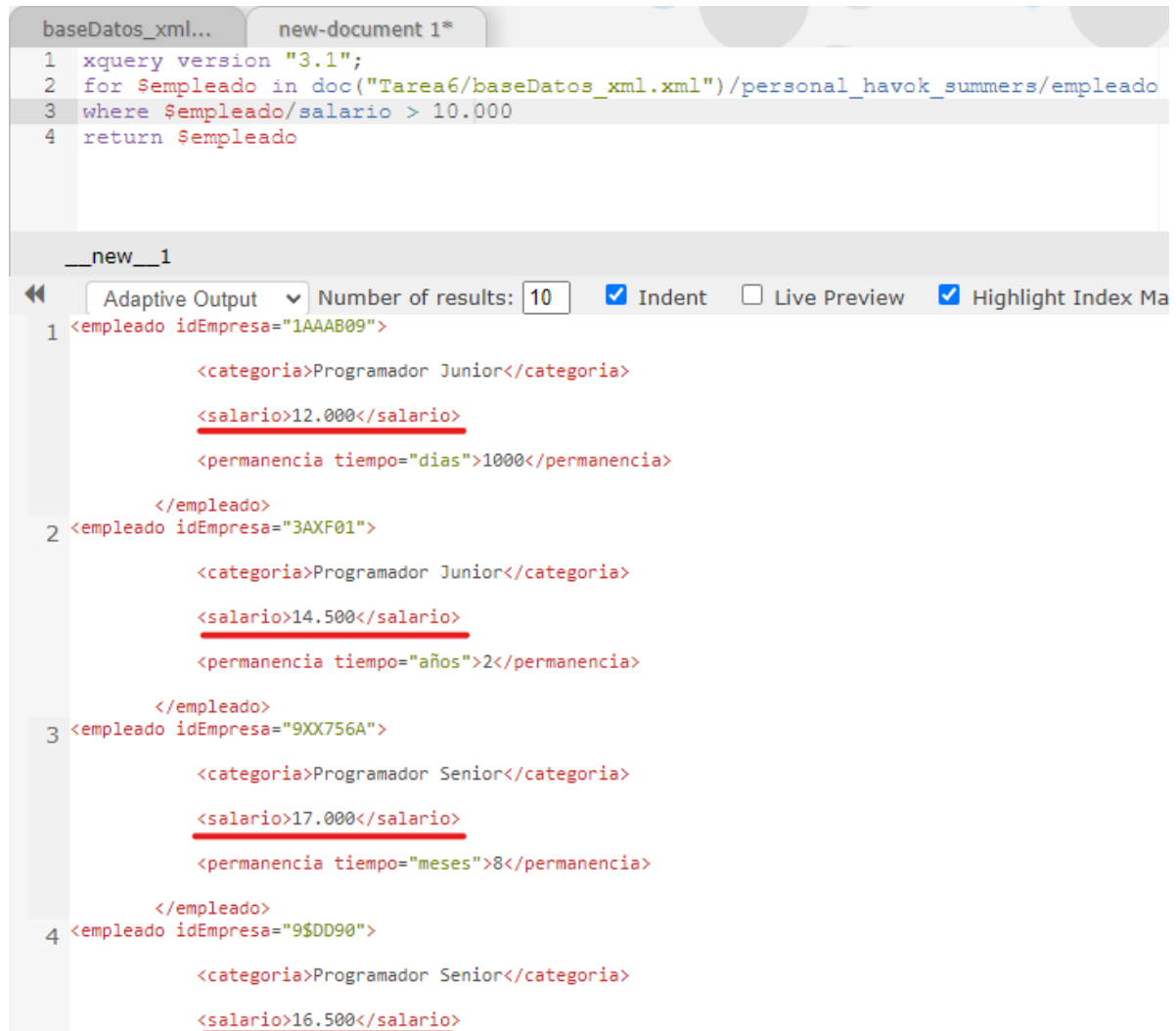
- ✖ Esta consulta se lee de la siguiente manera:

Almacena la palabra reservada \$empleado lo siguiente: dentro del doc "baseDatos_xml.xml" todo lo que contenga las etiquetas de la siguiente jerarquía /personal_havok_summers/empleado.

Usando esa palabra reservada \$empleado y con la **condición where** + /salario indicamos que solo recoja los **mayores a 10.000**

→ Devuelve \$empleado

- ✖ Se muestra algunos de los varios empleados que obtenemos como resultado de que su sueldo sea mayor a 10.000



The screenshot shows a software interface with two tabs: 'baseDatos_xml...' and 'new-document 1*'. The 'new-document 1*' tab is active and contains an XQuery. Below the query, there is a control bar with 'Adaptive Output', 'Number of results: 10', and checkboxes for 'Indent', 'Live Preview', and 'Highlight Index Ma'. The results are displayed as a list of XML elements, with the first four shown. Each result is an <empleado> element containing <idEmpresa>, <categoria>, <salario>, and <permanencia tiempo=> attributes. The salary values are 12.000, 14.500, 17.000, and 16.500, all of which are underlined in red.

```
1 xquery version "3.1";
2 for $empleado in doc("Tarea6/baseDatos_xml.xml")/personal_havok_summers/empleado
3 where $empleado/salario > 10.000
4 return $empleado
```

__new__1

Adaptive Output Number of results: 10 Indent Live Preview Highlight Index Ma

```
1 <empleado idEmpresa="1AAB09">
    <categoria>Programador Junior</categoria>
    <salario>12.000</salario>
    <permanencia tiempo="días">1000</permanencia>
</empleado>
2 <empleado idEmpresa="3AXF01">
    <categoria>Programador Junior</categoria>
    <salario>14.500</salario>
    <permanencia tiempo="años">2</permanencia>
</empleado>
3 <empleado idEmpresa="9XX756A">
    <categoria>Programador Senior</categoria>
    <salario>17.000</salario>
    <permanencia tiempo="meses">8</permanencia>
</empleado>
4 <empleado idEmpresa="9$DD90">
    <categoria>Programador Senior</categoria>
    <salario>16.500</salario>
```

4-Obtener la información de los miembros del personal cuyo salario sea mayor de 10000 y su categoría distinta de "Analista de Desarrollo".

- ✖ Esta consulta se lee de la siguiente manera:

Almacena la palabra reservada \$empleado lo siguiente: dentro del doc "baseDatos_xml.xml" todo lo que contenga las etiquetas de la siguiente jerarquía /personal_havok_summers/empleado.

Usando esa palabra reservada \$empleado y con la **condición where** + /salario indicamos que solo recoja los **mayores a 10.000** y con **not** indicamos que la categoría no sea "Analista de Desarrollo"

→ Devuelve \$empleado

- ✖ Estos son algunos de los empleados que obtenemos al usar la siguiente consulta xQuery

```
baseDatos_xml...  new-document 1*
1  xquery version "3.1";
2  for $empleado in doc("Tarea6/baseDatos_xml.xml")/personal_havok_summers/empleado
3  where $empleado/salario > 10.000 and not($empleado/categoria = "Analista de Desarrollo")
4  return $empleado
```

__new__1

Adaptive Output Number of results: 10 Indent Live Preview Highlight Index Matches

```
1 <empleado idEmpresa="1AAAB09">
    <categoria>Programador Junior</categoria>
    <salario>12.000</salario>
    <permanencia tiempo="días">1000</permanencia>
</empleado>
2 <empleado idEmpresa="3AXF01">
    <categoria>Programador Junior</categoria>
    <salario>14.500</salario>
    <permanencia tiempo="años">2</permanencia>
</empleado>
3 <empleado idEmpresa="9XX756A">
    <categoria>Programador Senior</categoria>
    <salario>17.000</salario>
    <permanencia tiempo="meses">8</permanencia>
</empleado>
4 <empleado idEmpresa="9$DD90">
    <categoria>Programador Senior</categoria>
    <salario>16.500</salario>
    <permanencia tiempo="días">300</permanencia>
</empleado>
5 <empleado idEmpresa="AZ0091">
    <categoria>Jefe de Proyecto</categoria>
    <salario>25.000</salario>
    <permanencia tiempo="años">7</permanencia>
```

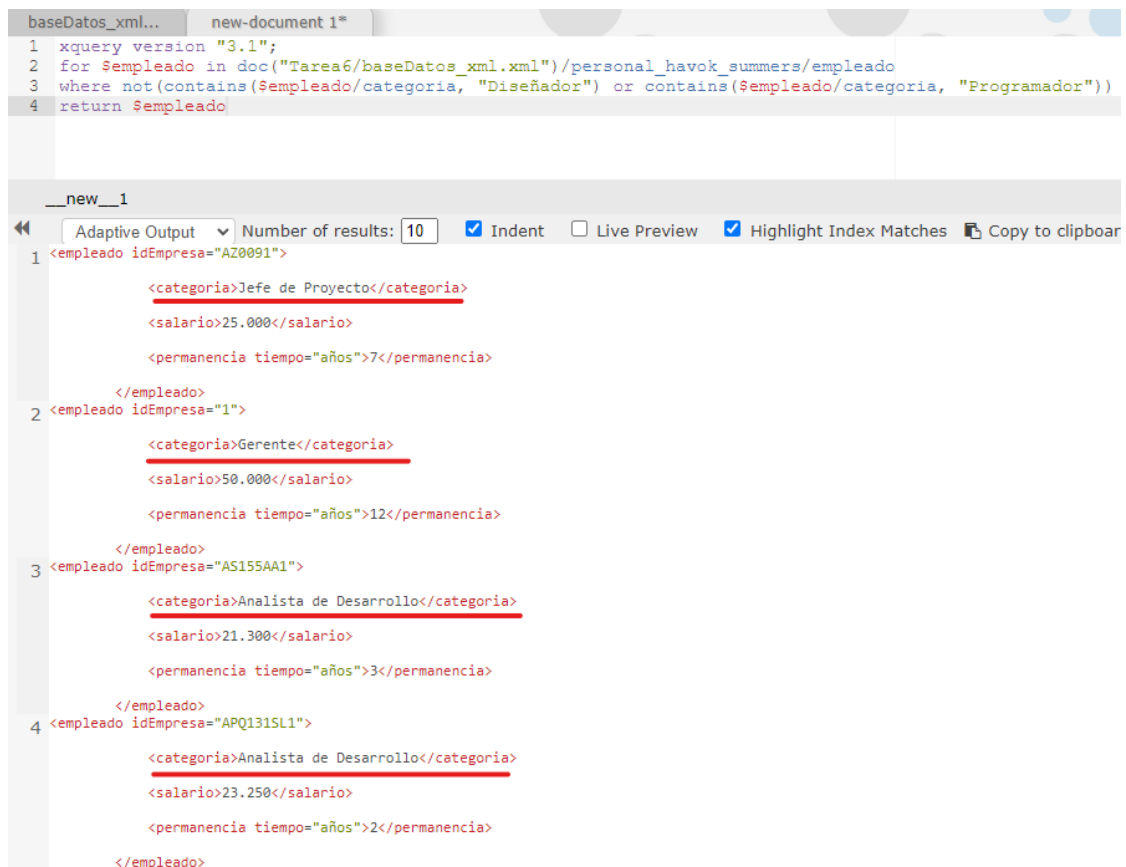
5-Obtener la información de los miembros del personal cuya categoría sea distinta de "Diseñador" o "Programador"

✖ Esta consulta se lee de la siguiente manera:

Almacena la palabra reservada \$empleado lo siguiente: dentro del doc "baseDatos_xml.xml" todo lo que contenga las etiquetas de la siguiente jerarquía /personal_havok_summers/empleado.

Usando esa palabra reservada \$empleado y con la **condición where** le indicamos con **not** que **no muestre** lo que contenga "**Diseñador**" dentro de categoría o que contenga "**Programador**"

→ Devuelve \$empleado



The screenshot shows an XQuery editor with a query in the top pane and its results in the bottom pane. The query is as follows:

```
1 xquery version "3.1";
2 for $empleado in doc("Tarea6/baseDatos_xml.xml")/personal_havok_summers/empleado
3 where not(contains($empleado/categoria, "Diseñador") or contains($empleado/categoria, "Programador"))
4 return $empleado
```

The results pane shows four XML documents, each representing an employee. The first three are filtered results, and the fourth is a document that was not filtered (likely because it was not found or did not match the criteria). The filtered results are:

```
1 <empleado idEmpresa="AZ0091">
  <categoria>Jefe de Proyecto</categoria>
  <salario>25.000</salario>
  <permanencia tiempo="años">7</permanencia>
</empleado>
2 <empleado idEmpresa="1">
  <categoria>Gerente</categoria>
  <salario>50.000</salario>
  <permanencia tiempo="años">12</permanencia>
</empleado>
3 <empleado idEmpresa="AS155AA1">
  <categoria>Analista de Desarrollo</categoria>
  <salario>21.300</salario>
  <permanencia tiempo="años">3</permanencia>
</empleado>
4 <empleado idEmpresa="APQ131SL1">
  <categoria>Analista de Desarrollo</categoria>
  <salario>23.250</salario>
  <permanencia tiempo="años">2</permanencia>
</empleado>
```

7. RA06_d) Se han identificado lenguajes y herramientas para el tratamiento y almacenamiento de información y su inclusión en documentos de intercambio de datos.

Consulta 1 utilizado el lenguaje XQuery

```
baseDatos_xml...  new-document 1*
1  xquery version "3.1";
2  for $empleado in doc("Tarea6/baseDatos_xml.xml")/personal_havok_summers/empleado
3  where $empleado/permanencia/@tiempo = "dias"
4  return $empleado
```

Consulta 2 utilizado el lenguaje XQuery

```
baseDatos_xml...  new-document 1*
1  xquery version "3.1";
2  for $empleado in doc("Tarea6/baseDatos_xml.xml")/personal_havok_summers/empleado
3  where $empleado/permanencia/@tiempo = "meses"
4  return $empleado
```

Consulta 3 utilizado el lenguaje XQuery

```
baseDatos_xml...  new-document 1*
1  xquery version "3.1";
2  for $empleado in doc("Tarea6/baseDatos_xml.xml")/personal_havok_summers/empleado
3  where $empleado/salario > 10.000
4  return $empleado
```

Consulta 4 utilizado el lenguaje XQuery

```
baseDatos_xml...  new-document 1*
1  xquery version "3.1";
2  for $empleado in doc("Tarea6/baseDatos_xml.xml")/personal_havok_summers/empleado
3  where $empleado/salario > 10.000 and not($empleado/categoria = "Analista de Desarrollo")
4  return $empleado
```

Consulta 5 utilizado el lenguaje XQuery

```
baseDatos_xml...  new-document 1*
1  xquery version "3.1";
2  for $empleado in doc("Tarea6/baseDatos_xml.xml")/personal_havok_summers/empleado
3  where not(contains($empleado/categoria, "Diseñador") or contains($empleado/categoria, "Programador"))
4  return $empleado
```