Utiliza el archivo **biblioteca.xml** para realizar las siguientes consultas FLOWR. Obtener el mismo formato de resultado.

1. Obtener los autores.

|  |  |
| --- | --- |
| for $x in doc("biblioteca.xml")//author  return $x | ……. |

1. Ahora obtendremos únicamente el elemento <last>

|  |  |
| --- | --- |
| for $x in doc("biblioteca.xml")//last  return $x |  |

1. Devolvemos el mismo resultado pero con formato de “lista con orden” de html:

|  |  |
| --- | --- |
| <ul>{  for $x in doc("biblioteca.xml")//last  return <li>{data($x)}</li>  }</ul> |  |

1. Eliminamos los elementos repetidos.

|  |  |
| --- | --- |
| <ul>{  for $x in distinct-values(doc("biblioteca.xml")//last)  return <li>{$x}</li>  }</ul> |  |

1. Sin elementos repetidos, ordenados alfabéticamente.

|  |  |
| --- | --- |
| <ul>{  for $x in distinct-values(doc("biblioteca.xml")//last)  order by $x  return <li>{$x}</li>  }</ul> |  |

1. En la ventana del editor de XQuery incluye etiquetas de html sin atributos que te permitan obtener un archivo 6.html que puedas visualizar en el explorador de la siguiente manera:

|  |  |
| --- | --- |
| <html>  <body>  <h1>Ejemplo de html</h1>  <hr />  <ol>{  for $x in distinct-values(doc("biblioteca.xml")//last)  order by $x  return <li>{$x}</li>  }</ol>  </body>  </html> |  |

1. Títulos de los libros ordenados alfabéticamente, editados antes del año 2000:

|  |  |
| --- | --- |
| for $x in doc("biblioteca.xml")//book  where $x/@year < 2000  order by $x//title  return data($x//title) |  |

1. Lista ordenada de los apellidos tanto de autores como de editores:

|  |  |
| --- | --- |
| <ol>{  for $x in distinct-values(doc("biblioteca.xml")//(author | editor))  let $c := substring-before(normalize-space($x), " ")  order by $c  return <li>{$c}</li>  }</ol> |  |

Elimina los valores repetidos.

1. Titulo de los libros cuyo precio esté por debajo de los 50€:

|  |  |
| --- | --- |
| for $x in doc("biblioteca.xml")//book  where $x//price < 50  return <libro>{data($x//title)}</libro> |  |

1. Títulos de los libros ordenados de forma descendente escritos en años que empiecen por 1.

|  |  |
| --- | --- |
| for $x in doc("biblioteca.xml")//book  where starts-with($x/@year, "1")  order by $x//title descending  return <libro>{data($x//title)}</libro> |  |