

Resumen de XPath y XQuery para examen

XPath: Navegación y selección en XML

1. Ejes de navegación:

- `child::`: Selecciona nodos hijos.
- `descendant::`: Selecciona todos los nodos descendientes.
- `self::`: Selecciona el nodo actual.
- `parent::`: Selecciona el nodo padre.
- `ancestor::`: Selecciona todos los ancestros.
- `following-sibling::` y `preceding-sibling::`: Selecciona nodos hermanos siguientes o anteriores.

2. Expresiones de ruta (path expressions):

- `/`: Nodo raíz o separa niveles.
- `//`: Selecciona nodos independientemente de su nivel.
- `elemento`: Selecciona nodos específicos desde la posición actual.
- `*`: Comodín para cualquier elemento.
- `@atributo`: Selecciona un atributo.

3. Funciones principales:

- **Funciones de cadenas:**
 - `substring(cadena, inicio, longitud)`: Devuelve una subcadena.
 - `string-length(cadena)`: Longitud de una cadena.
 - `upper-case(cadena)` y `lower-case(cadena)`: Convierte cadenas a mayúsculas o minúsculas.
 - `contains(cadena1, cadena2)`: Verifica si `cadena1` contiene `cadena2`.
- **Funciones contextuales:**
 - `position()`: Proporciona la posición del nodo dentro de un conjunto.
 - `last()`: Último nodo del conjunto.
- **Funciones agregadas:**
 - `count()`, `sum()`, `avg()`, `max()`, `min()`.

4. Operadores comunes:

- Aritméticos: `+`, `-`, `div`, `mod`.
- Comparativos: `<`, `<=`, `>`, `>=`, `=`.
- Lógicos: `and`, `or`, `not()`.

Ejemplo: Seleccionar libros con precio mayor a 25: `//libro[precio > 25]`.

XQuery: Consultas avanzadas en XML

1. Características principales:

- Extiende XPath: cualquier expresión XPath es válida en XQuery.
- Sintaxis basada en las cláusulas FLWOR:
 - **for**: Itera sobre un conjunto de nodos.
 - **let**: Asigna valores a variables.
 - **where**: Aplica filtros.
 - **order by**: Ordena resultados.
 - **return**: Especifica el valor de retorno.

Ejemplo básico:

```
for $libro in doc("catalogo.xml")//libro
where $libro/precio > 25
return $libro/titulo
```

2. Funciones adicionales:

- **Transformaciones**: Permite generar otros formatos como HTML o JSON.

- Ejemplo: Generar una lista HTML.

```
<ul>
{
  for $libro in doc("catalogo.xml")//libro
  return <li>{data($libro/titulo)}</li>
}
</ul>
```

- **Condicionales**: Utiliza **if-then-else**.

- Ejemplo:

```
for $libro in doc("catalogo.xml")//libro
return if ($libro/precio > 30) then "Caro" else "Barato"
```

Consejos prácticos:

- Practica con ejemplos de XPath y XQuery en BaseX.
- Familiarízate con las expresiones más comunes, como rutas, ejes y funciones de cadenas.
- Comprende bien las cláusulas FLWOR para XQuery, ya que son clave en las consultas avanzadas.
- Usa herramientas como BaseX para probar y depurar tus consultas.