



XPath

<notes>

<note>

<to id="1"> Knysliukas </to>

<to id="2"> Drakonas </to>

<from> Pukuotukas </from>

</note>

</notes>

Selecting

Expression	Selects...
<i>nodename</i>	all child nodes of the named node
/	from the root node
//	nodes in the document from the current node that match the selection no matter where they are
.	the current node
..	the parent of the current node
@	attribute

Examples

Path Expression	Result
note	Selects all the child nodes of the 'note' element
/notes	Selects the root element 'notes' Note: slash (/) represents an absolute path
note/to	Selects all 'to' elements that are children of 'note'
//note	Selects all 'note' elements no matter where they are in the document
note//to	Selects all 'to' elements that are descendant of the 'notes' element, no matter where they are under the 'notes' element
//@lang	Selects all attributes that are named lang

<notes>

<note>

<to id="1"> Knysliukas </to>

<to id="2"> Drakonas </to>

<from> Pukuotukas </from>

</note>

</notes>

- `note/to[1]`
- `note/to[last()-1]`
- `note/to[position()<5]`
- `//message[@lang]`
- `//message[@lang='eng']`
- `note/message[weight>35.00]/title`

Predicates

- **note/to[1]**
- **<notes>**
- **<note>**
- **<to id="1"> Knysliukas </to>**
- **<to id="2"> Drakonas </to>**
- **<from> Pukuotukas </from>**
- **</note>**
- **</notes>**

- `note/to[last()-1]`
- `<notes>`
- `<note>`
- `<to id="1"> Knysliukas </to>`
- `<to id="2"> Drakonas </to>`
- `<from> Pukuotukas </from>`
- `</note>`
- `</notes>`

note/to[position()<5]

- **<notes>**
- **<note>**
- **<to id="1"> Knysliukas </to>**
- **<to id="2"> Drakonas </to>**
- **<from> Pukuotukas </from>**
- **</note>**
- **</notes>**

Predicates

- `//message[@id]`
- `//@id`
- `<notes>`
- `<note>`
- `<to id="1"> Knysliukas </to>`
- `<to id="2"> Drakonas </to>`
- `<from> Pukuotukas </from>`
- `</note>`
- `</notes>`

Predicates

- `//message[@lang='eng']`
- `//to[@id='2']`
- `<notes>`
- `<note>`
- `<to id="1"> Knysliukas </to>`
- `<to id="2"> Drakonas </to>`
- `<from> Pukuotukas </from>`
- `</note>`
- `</notes>`

Predicates

- `note/to[1]`
- `<notes>`
- `<note>`
- `<to id="1"> Knysliukas </to>`
- `<to id="2"> Drakonas </to>`
- `<from> Pukuotukas </from>`
- `</note>`
- `</notes>`

Signs

	Description
*	any element node
@*	any attribute
node()	any node of any kind
	combines two sets

<notes>

<note>

<to> Knysliukas </to>

<to> Drakonas </to>

<from> Pukuotukas </from>

</note>

</notes>

Relationship of Nodes

- Parent
- Children
- Siblings
- Ancestors
- Descendants

- ancestor
- attribute
- Child
- child::book
- child::*
- ancestor::book
- child::* / child::price