



XPATH

query language for xml

tutorialspoint
SIMPLY EASY LEARNING

www.tutorialspoint.com



<https://www.facebook.com/tutorialspointindia>



<https://twitter.com/tutorialspoint>

About the Tutorial

XPath is a query language that is used for traversing through an XML document. It is used commonly to search particular elements or attributes with matching patterns.

This tutorial explains the basics of XPath. It contains chapters discussing all the basic components of XPath with suitable examples.

Audience

This tutorial has been designed for beginners to help them understand the basic concepts related to XPath. This tutorial will give you enough understanding on XPath from where you can take yourself to higher levels of expertise.

Prerequisites

Before proceeding with this tutorial, you should have basic knowledge of XML, HTML, and JavaScript.

Disclaimer & Copyright

© Copyright 2018 by Tutorials Point (I) Pvt. Ltd.

All the content and graphics published in this e-book are the property of Tutorials Point (I) Pvt. Ltd. The user of this e-book is prohibited to reuse, retain, copy, distribute or republish any contents or a part of contents of this e-book in any manner without written consent of the publisher.

We strive to update the contents of our website and tutorials as timely and as precisely as possible, however, the contents may contain inaccuracies or errors. Tutorials Point (I) Pvt. Ltd. provides no guarantee regarding the accuracy, timeliness or completeness of our website or its contents including this tutorial. If you discover any errors on our website or in this tutorial, please notify us at contact@tutorialspoint.com.

Table of Contents

About the Tutorial.....	i
Audience	i
Prerequisites	i
Disclaimer & Copyright.....	i
Table of Contents	ii
1. XPATH – OVERVIEW	1
Need for XSL.....	1
What is XPath?	1
2. XPATH — EXPRESSION.....	3
Example	4
3. XPATH — NODES.....	7
XPath Root Node.....	7
XPath Element Node	10
XPath Text Node	14
XPath Attribute Node.....	18
XPath Comment Node.....	21
4. XPATH — ABSOLUTE PATH.....	24
5. XPATH — RELATIVE PATH	27
Example	27
6. XPATH — AXES.....	30
Verify the output.....	32
7. XPATH — OPERATORS.....	33
XPath Comparison Operators.....	33

XPath Boolean Operators	36
XPath Number Operators / Functions	39
XPath String Functions	42
XPath Node Functions	45
8. XPATH — WILDCARD	49
Example	49
9. XPATH — PREDICATE.....	53
Example	53

1. XPath – Overview

Before learning XPath, we should first understand XSL which stands for **Extensible Stylesheet Language**. It is similar to XML as CSS is to HTML.

Need for XSL

In case of HTML documents, tags are predefined such as table, div, span, etc. The browser knows how to add style to them and display them using CSS styles. But in case of XML documents, tags are not predefined. In order to understand and style an XML document, **World Wide Web Consortium (W3C)** developed XSL which can act as an XML-based Stylesheet Language. An XSL document specifies how a browser should render an XML document.

Following are the main parts of XSL:

- **XSLT** — used to transform XML documents into various other types of document.
- **XPath** — used to navigate XML documents.
- **XSL-FO** — used to format XML documents.

What is XPath?

XPath is an official recommendation of the World Wide Web Consortium (W3C). It defines a language to find information in an XML file. It is used to traverse elements and attributes of an XML document. XPath provides various types of expressions which can be used to enquire relevant information from the XML document.

- **Structure Definitions** — XPath defines the parts of an XML document like element, attribute, text, namespace, processing-instruction, comment, and document nodes
- **Path Expressions** — XPath provides powerful path expressions select nodes or list of nodes in XML documents.
- **Standard Functions** — XPath provides a rich library of standard functions for manipulation of string values, numeric values, date and time comparison, node and QName manipulation, sequence manipulation, Boolean values etc.
- **Major part of XSLT** — XPath is one of the major elements in XSLT standard and is must have knowledge in order to work with XSLT documents.
- **W3C recommendation** — XPath is an official recommendation of World Wide Web Consortium (W3C).

One should keep the following points in mind, while working with XPath:

- XPath is core component of XSLT standard.
- XSLT cannot work without XPath.
- XPath is basis of XQuery and XPointer.

2. XPath — Expression

An XPath expression generally defines a pattern in order to select a set of nodes. These patterns are used by XSLT to perform transformations or by XPointer for addressing purpose.

XPath specification specifies seven types of nodes which can be the output of execution of the XPath expression.

- Root
- Element
- Text
- Attribute
- Comment
- Processing Instruction
- Namespace

XPath uses a path expression to select node or a list of nodes from an XML document.

Following is the list of useful paths and expression to select any node/ list of nodes from an XML document.

Expression	Description
node-name	Select all nodes with the given name "nodename"
/	Selection starts from the root node
//	Selection starts from the current node that match the selection
.	Selects the current node
..	Selects the parent of the current node
@	Selects attributes
student	Example: Selects all nodes with the name "student"

class/student	Example: Selects all student elements that are children of class
//student	Selects all student elements no matter where they are in the document

Example

In this example, we've created a sample XML document, students.xml and its stylesheet document **students.xsl** which uses the XPath expressions under **select** attribute of various XSL tags to get the values of roll no, firstname, lastname, nickname and marks of each student node.

students.xml

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="students.xsl"?>
<class>
  <student rollno="393">
    <firstname>Dinkar</firstname>
    <lastname>Kad</lastname>
    <nickname>Dinkar</nickname>
    <marks>85</marks>
  </student>
  <student rollno="493">
    <firstname>Vaneet</firstname>
    <lastname>Gupta</lastname>
    <nickname>Vinni</nickname>
    <marks>95</marks>
  </student>
  <student rollno="593">
    <firstname>Jasvir</firstname>
    <lastname>Singh</lastname>
    <nickname>Jazz</nickname>
    <marks>90</marks>
  </student>
</class>
```


students.xsl

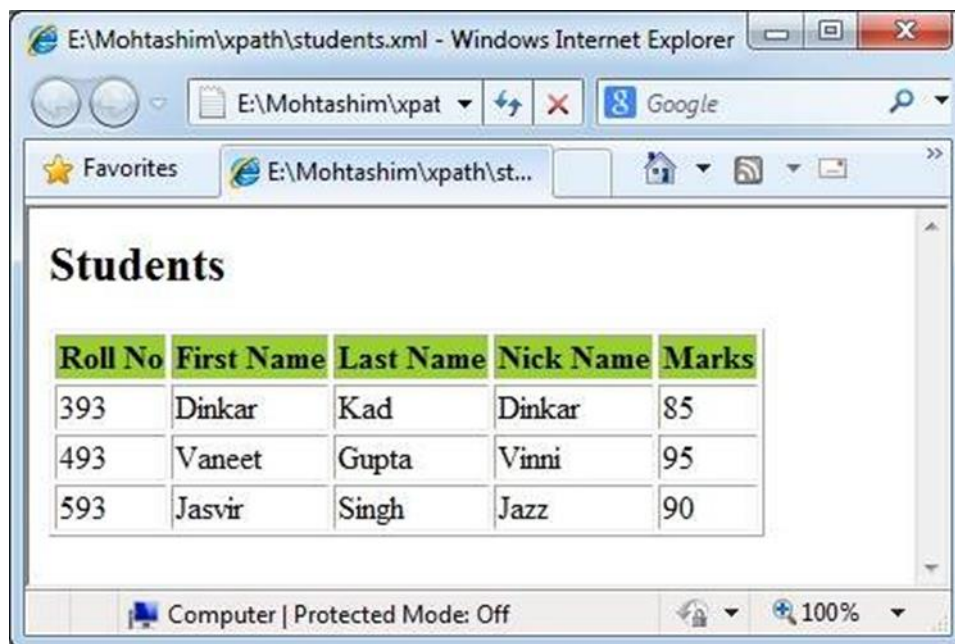
```

<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:template match="/">
  <html>
  <body>
  <h2>Students</h2>
  <table border="1">
    <tr bgcolor="#9acd32">
      <th>Roll No</th>
      <th>First Name</th>
      <th>Last Name</th>
      <th>Nick Name</th>
      <th>Marks</th>
    </tr>
    <xsl:for-each select="class/student">
      <tr>
        <td>
          <xsl:value-of select="@rollno"/>
        </td>
        <td><xsl:value-of select="firstname"/></td>
        <td><xsl:value-of select="lastname"/></td>
        <td><xsl:value-of select="nickname"/></td>
        <td><xsl:value-of select="marks"/></td>
      </tr>
    </xsl:for-each>
  </table>
  </body>
  </html>
</xsl:template>

</xsl:stylesheet>

```

Verify the output

Students

Roll No	First Name	Last Name	Nick Name	Marks
393	Dinkar	Kad	Dinkar	85
493	Vaneet	Gupta	Vinni	95
593	Jasvir	Singh	Jazz	90

3. XPath — Nodes

In this chapter, we'll see the XPath expression in details covering common types of Nodes, XPath defines and handles.

S.N.	Node Type & Description
1	Root Root element node of an XML Document.
2	Element Element node.
3	Text Text of an element node.
4	Attribute Attribute of an element node.
5	Comment Comment

Let us now understand the nodes in detail.

XPath Root Node

Following are the ways to get root element and do the processing afterwards.

Use Wildcard

Use `/*`, wild card expression to select the root node.

```
<p><xsl:value-of select="name(/*)" /></p>
```

Use Name

Use `/class`, to select root node by name.

```
<p><xsl:value-of select="name(/class)" /></p>
```

Use Name with wild card

Use `/class/*`, select all element under root node.

```
<p><xsl:value-of select="name(/class/*)"/></p>
```

Example

In this example, we've created a sample XML document **students.xml** and its stylesheet document **students.xsl** which uses the XPath expressions.

Following is the sample XML used.

students.xml

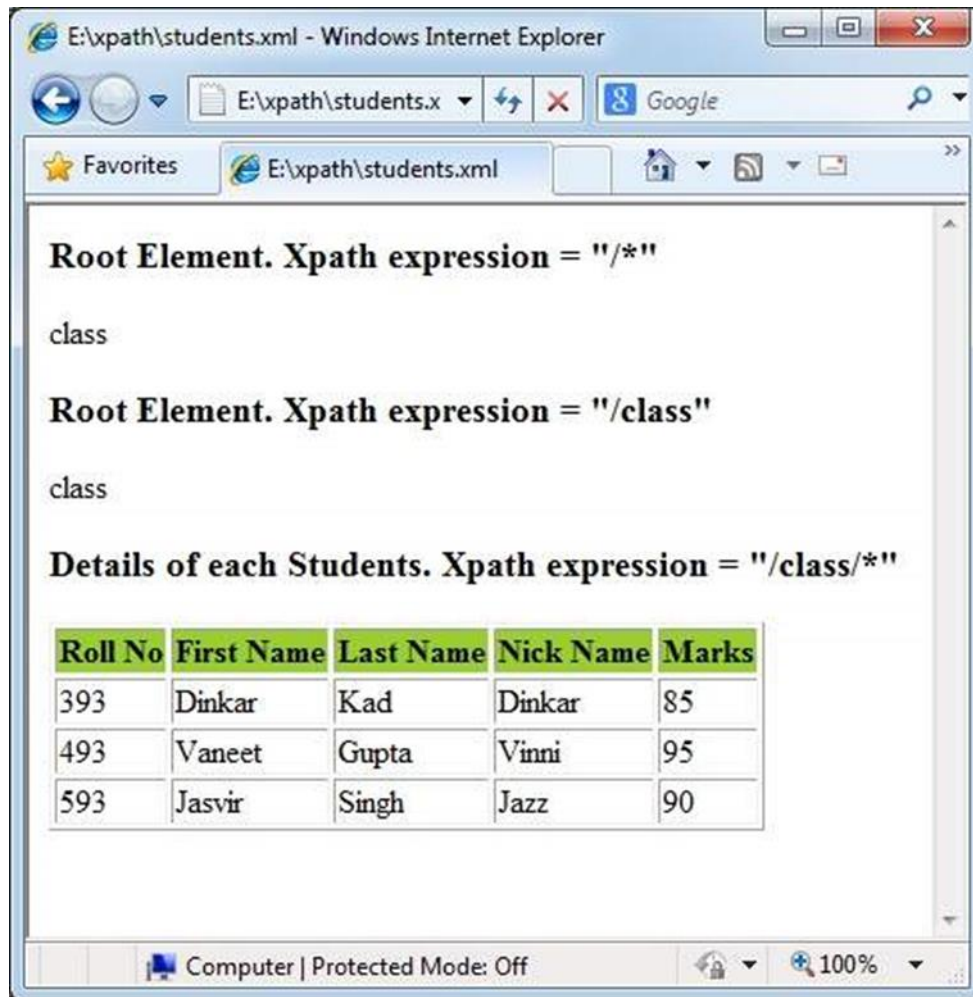
```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="students.xsl"?>
<class>
  <student rollno="393">
    <firstname>Dinkar</firstname>
    <lastname>Kad</lastname>
    <nickname>Dinkar</nickname>
    <marks>85</marks>
  </student>
  <student rollno="493">
    <firstname>Vaneet</firstname>
    <lastname>Gupta</lastname>
    <nickname>Vinni</nickname>
    <marks>95</marks>
  </student>
  <student rollno="593">
    <firstname>Jasvir</firstname>
    <lastname>Singh</lastname>
    <nickname>Jazz</nickname>
    <marks>90</marks>
  </student>
</class>
```

students.xsl

```

<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="/">
<html>
<body>
  <h3>Root Element. Xpath expression = "/*"</h3>
  <p><xsl:value-of select="name(*)"/></p>
  <h3>Root Element. Xpath expression = "/class"</h3>
  <p> <xsl:value-of select="name(/class)"/></p>
  <h3>Details of each Students. Xpath expression = "/class/*"</h3>
  <table border="1">
    <tr bgcolor="#9acd32">
      <th>Roll No</th>
      <th>First Name</th>
      <th>Last Name</th>
      <th>Nick Name</th>
      <th>Marks</th>
    </tr>
    <xsl:for-each select="/class/*">
      <tr>
        <td>
          <xsl:value-of select="@rollno"/>
        </td>
        <td><xsl:value-of select="firstname"/></td>
        <td><xsl:value-of select="lastname"/></td>
        <td><xsl:value-of select="nickname"/></td>
        <td><xsl:value-of select="marks"/></td>
      </tr>
    </xsl:for-each>
  </table>
</body>
</html>
</xsl:template>
</xsl:stylesheet>

```

Verify the output

The screenshot shows a Windows Internet Explorer window titled "E:\xpath\students.xml - Windows Internet Explorer". The address bar shows "E:\xpath\students.x" and the search bar contains "Google". The main content area displays the following text:

Root Element. Xpath expression = "/"

class

Root Element. Xpath expression = "/class"

class

Details of each Students. Xpath expression = "/class/*"

Roll No	First Name	Last Name	Nick Name	Marks
393	Dinkar	Kad	Dinkar	85
493	Vaneet	Gupta	Vinni	95
593	Jasvir	Singh	Jazz	90

The status bar at the bottom shows "Computer | Protected Mode: Off" and a zoom level of "100%".

End of ebook preview

If you liked what you saw...

Buy it from our store @ **<https://store.tutorialspoint.com>**