

 eBook Gratuit

APPRENEZ

xslt

eBook gratuit non affilié créé à partir des
contributeurs de Stack Overflow.

#xslt

Table des matières

À propos	1
Chapitre 1: Démarrer avec xslt	2
Remarques.....	2
Versions.....	2
Exemples.....	2
Installation ou configuration.....	2
Exemple simple de XSLT.....	3
Chapitre 2: Variables dans xslt	7
Exemples.....	7
Variables Xslt.....	7
Crédits	8

À propos

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [xslt](#)

It is an unofficial and free xslt ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official xslt.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Chapitre 1: Démarrer avec xslt

Remarques

XSLT (Transformations XSL, Transformations de langage de feuilles de style extensibles) est un langage de programmation basé sur XML, dont le but est de manipuler et de transformer le XML. C'est un langage de programmation fonctionnel et un standard W3C. En utilisant XSLT, vous pouvez transformer un ou plusieurs documents XML en un résultat pouvant contenir divers formats de sortie (XML, HTML, texte).

Pour comprendre XSLT, il est préférable d'avoir une bonne connaissance pratique de:

1. XML
2. XPath

Sans ces deux concepts sous-jacents, vous auriez du mal à apprendre le XSLT. Tout d'abord, car un document XSLT est un document XML. Et deuxièmement, les documents XSLT utilisent XPath pour interroger le document XML en cours de transformation.

Lecture recommandée: <https://en.wikipedia.org/wiki/XSLT>

Versions

Version	spécification	Statut	Date de sortie
XSLT 1.0	https://www.w3.org/TR/xslt	Recommandation	1998-08-18
XSLT 2.0	https://www.w3.org/TR/xslt20	Recommandation	2007-01-23
XSLT 3.0	https://www.w3.org/TR/xslt-30	Recommandation du candidat	2015-11-19

Exemples

Installation ou configuration

XSLT est un langage de programmation spécial; Il est largement utilisé pour transformer des documents XML soit dans un format XML différent, en HTML, soit dans des formats textuels.

Il existe deux versions principales de XSLT utilisées: XSLT 1.0 et XSLT 2.0. XSLT 1.0 est plus largement implémenté mais présente de nombreuses restrictions et limitations par rapport à XSLT 2.0; vous devrez décider quelle version utiliser. Si un processeur XSLT 2.0 est disponible pour l'environnement choisi, c'est presque toujours le meilleur choix.

(XSLT 1.0 est sorti en novembre 1999, et de nombreuses implémentations sont apparues en un an ou deux, à la fois par des fournisseurs traditionnels tels que Microsoft, IBM, Sun et Oracle, et

par des passionnés travaillant pendant leur temps libre. Le processeur XSLT 2.0 le plus répandu est celui de Saxon, mais il n'a pas le champ à lui-même: les autres produits autonomes incluent RaptorXML (Altova.), XmlPrime (de CBCL) et Exselt, ainsi que les processeurs 2.0 sont également intégrés dans WebLogic d'IBM, dans le serveur de base de données XML de MarkLogic et dans l'accélérateur XML d'Intel.)

La spécification de XSLT 3.0 est techniquement complète (proposition de recommandation en avril 2017), mais il est préférable de la considérer pour le moment: ne l'envisagez que si vous avez besoin de nouvelles fonctionnalités, telles que la diffusion en continu, les packages, le support JSON. , ou essayer / attraper. Il existe trois implémentations connues: Saxon, Exselt et RaptorXML.

Pour commencer à utiliser XSLT, vous avez plusieurs options:

- Utilisez un outil XSLT en ligne. Il y en a plusieurs disponibles (recherche pour "outil XSLT en ligne"), un IDE en ligne très fiable est <http://xslttransform.net/> . C'est un bon moyen d'obtenir une idée initiale du langage, mais vous ne voudrez plus continuer sur cette voie une fois que vous réaliserez un réel développement.
- Utilisez le moteur XSLT intégré à chaque navigateur. Comme avec les outils en ligne, ceux-ci ont l'avantage de ne rien avoir à installer; mais les navigateurs ne prennent en charge que XSLT 1.0, ils ne prennent en charge que la conversion XML vers HTML, et ils ont un support de débogage très faible. Saxon-JS est un développement très récent, mais toujours expérimental, qui permet d'exécuter XSLT 2.0 (et des parties de XSLT 3.0) dans le navigateur.
- Installez un processeur XSLT (tel que Saxon ou xsltproc). La plupart de ces produits peuvent être appelés à l'aide de la ligne de commande du système d'exploitation ou d'une API pour des langages de programmation spécifiques tels que Java, C, C # ou Python.
- Installez un environnement de développement XML tel que Altova XML Spy, oXygen de SyncroSoft ou Stylus Studio. Bien que cette option soit plus coûteuse, elle offre une capacité de développement et de débogage beaucoup plus riche.

Quelle que soit l'option choisie, vous devez d'abord décider si vous souhaitez utiliser XSLT 1.0 ou XSLT 2.0.

Exemple simple de XSLT

Voici un exemple simple qui utilise XSLT pour convertir des données dans un fichier XML en une table dans un fichier HTML. Vous pouvez l'utiliser pour expérimenter de simples transformations XSLT.

Condition préalable: installez un environnement d'exécution Java et ajoutez l'emplacement de l'environnement JRE à votre variable PATH. (Sous Windows, la plupart des installateurs ajouteront Java à votre chemin.) Si cela fonctionne, vous devriez pouvoir ouvrir une fenêtre de ligne de commande et exécuter la commande `java -version` et obtenir des informations sur votre JRE.

1. Téléchargez le processeur Saxon-HE XSLT pour Java ici: saxon.sourceforge.net et décompressez-le n'importe où sur votre ordinateur.
2. Dans un éditeur de texte, créez un fichier nommé `pets.xml` avec le code suivant:

```
<pets>
  <petType name="Dogs">
    <pet id="123" name="Sparky" vaccineStatus="vaccinated" healthStatus="healthy"/>
    <pet id="234" name="Sadie" vaccineStatus="unvaccinated" healthStatus="sick"/>
    <pet id="345" name="Herman" vaccineStatus="unvaccinated" healthStatus="unknown"/>
  </petType>
  <petType name="Cats">
    <pet id="456" name="Cleo" vaccineStatus="vaccinated" healthStatus="healthy"/>
    <pet id="567" name="Janet" vaccineStatus="unvaccinated" healthStatus="healthy"/>
    <pet id="678" name="Furry" vaccineStatus="vaccinated" healthStatus="sick"/>
  </petType>
</pets>
```

3. Dans un éditeur de texte, créez un fichier nommé `petTransform.xsl` avec le code suivant:

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  version="2.0">

  <!-- handle the root XML element -->
  <xsl:template match="/">
  <html><head>
    <title>Pets that are available for adoption</title>
  </head>
  <body>
    <xsl:apply-templates/>
  </body>
</html>
</xsl:template>

<xsl:template match="pets">
<xsl:apply-templates/>
</xsl:template>

<xsl:template match="petType">
<h2><xsl:value-of select="@name"/></h2>
<table id="{@name}">
  <tr>
    <th colname="id">ID</th>
    <th colname="name">Name</th>
    <th colname="vaccinated">Vaccine status</th>
    <th colname="health">Health status</th>
  </tr>
  <tbody>
    <!-- add a row for each pet in this category -->
    <xsl:for-each select="pet">
      <tr>
        <td colname="id"><xsl:value-of select="@id"/></td>
        <td colname="name"><xsl:value-of select="@name"/></td>
        <td colname="vaccinated"><xsl:value-of select="@vaccineStatus"/></td>
        <td colname="health"><xsl:value-of select="@healthStatus"/></td>
      </tr>
    </xsl:for-each>
  </tbody>
</table>
</xsl:template>
```

```
<!-- ignore the content of other tags because we processed them elsewhere -->
<xsl:template match="*">
<!-- do nothing -->
</xsl:template>

</xsl:stylesheet>
```

4. Ouvrez une fenêtre de ligne de commande et accédez au dossier contenant les fichiers XML et XSLT.
5. Exécutez la commande suivante, où `path_to_saxon.jar` est le chemin d'accès complet au fichier `saxon9he.jar` :

```
java -jar "path_to_saxon.jar" -o
petOutput.html -s:pets.xml -xsl:pettransform.xslt
```

Par exemple:

```
java -jar "C:\Program Files\SaxonHE9-7-0-7J\saxon9he.jar" -o
petOutput.html -s:pets.xml -xsl:pettransform.xslt
```

Assurez-vous de lancer cette commande sur une seule ligne.

6. Ouvrez le fichier de sortie `petOutput.html` dans un éditeur de texte. Ça devrait ressembler à ça:

```
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Pets that are available for adoption</title>
  </head>
  <body>

    <h2>Dogs</h2>
    <table id="Dogs">
      <tr>
        <th colname="id">ID</th>
        <th colname="name">Name</th>
        <th colname="vaccinated">Vaccine status</th>
        <th colname="health">Health status</th>
      </tr>
      <tbody>
        <tr>
          <td colname="id">123</td>
          <td colname="name">Sparky</td>
          <td colname="vaccinated">vaccinated</td>
          <td colname="health">healthy</td>
        </tr>
        <tr>
          <td colname="id">234</td>
          <td colname="name">Sadie</td>
          <td colname="vaccinated">unvaccinated</td>
          <td colname="health">sick</td>
        </tr>
        <tr>
          <td colname="id">345</td>
```

```

        <td colname="name">Herman</td>
        <td colname="vaccinated">unvaccinated</td>
        <td colname="health">unknown</td>
    </tr>
</tbody>
</table>

<h2>Cats</h2>
<table id="Cats">
    <tr>
        <th colname="id">ID</th>
        <th colname="name">Name</th>
        <th colname="vaccinated">Vaccine status</th>
        <th colname="health">Health status</th>
    </tr>
    <tbody>
        <tr>
            <td colname="id">456</td>
            <td colname="name">Cleo</td>
            <td colname="vaccinated">vaccinated</td>
            <td colname="health">healthy</td>
        </tr>
        <tr>
            <td colname="id">567</td>
            <td colname="name">Janet</td>
            <td colname="vaccinated">unvaccinated</td>
            <td colname="health">healthy</td>
        </tr>
        <tr>
            <td colname="id">678</td>
            <td colname="name">Furry</td>
            <td colname="vaccinated">vaccinated</td>
            <td colname="health">sick</td>
        </tr>
    </tbody>
</table>

</body>
</html>

```

7. Ouvrez le fichier de sortie `petOutput.html` dans un navigateur Web. Il devrait afficher les données dans un tableau simple.

Lire Démarrer avec xslt en ligne: <https://riptutorial.com/fr/xslt/topic/1129/demarrer-avec-xslt>

Chapitre 2: Variables dans xslt

Exemples

Variables Xslt

Variable globale: cette variable est disponible partout dans la feuille de style xsl. Cette variable ne doit être que l'enfant de l'élément `<xsl:stylesheet>`.

Variable locale: Cette variable est uniquement disponible là où elle est déclarée.

Voir ci-dessous le code:

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  exclude-result-prefixes="xs"
  version="2.0">
  <xsl:output omit-xml-declaration="yes"/>

  <xsl:variable name="a" select="5"/>      <!-- Global Variable -->

  <xsl:template match="/">
    <xsl:variable name="b" select="2"/> <!--Local Variable -->
    <xsl:value-of select="$a+$b"/>      <!--Addition of 'a' and 'b' -->
  </xsl:template>
</xsl:stylesheet>
```

La sortie du code ci-dessus serait: **7**

Il existe deux manières de définir une valeur pour la variable, par exemple:

Par expression xpath dans l' **attribut @select de l' élément <xsl: variable>** like:

```
<xsl:variable name="apple" select="'RED'"/>
```

OU

Par le **contenu de l'élément <xsl: variable>** comme:

```
<xsl:variable name="apple">RED</xsl:variable>
```

Pour appeler une variable déclarée, utilisez **\$** Sign avec un nom de variable, comme dans le code ci-dessus '\$ a'

Lire Variables dans xslt en ligne: <https://riptutorial.com/fr/xslt/topic/10060/variables-dans-xslt>

Crédits

S. No	Chapitres	Contributeurs
1	Démarrer avec xslt	Community , Daniel Haley , Dimitre Novatchev , Eero Helenius , JLRishe , Lukasz , Mads Hansen , Mathias Müller , Michael Kay , Nate , Tim McMackin
2	Variables dans xslt	palo , Tim C