

# XSL

Mait Kommusaar

# XSL perekond

- XSL Transformatsioonid
- XSL Formatting Objects
- XPath

# Ajalugu

- Ajendiks DSSL funktsionaalsus XML-i jaoks
- DSSL - Document Style Semantics and Specification Language
- November 1999 - XPath ja XSLT W3C Recommendation
- Oktoober 2001 - XSL-FO W3C Recommendation

# XSL-FO

- **XSL Formatting Objects**
- XEP, XSLFormatter, XINC,
- FOP

# XPath

- Info otsimine
- Arvutuste tegemine
- XML-i puu kujuline struktuur

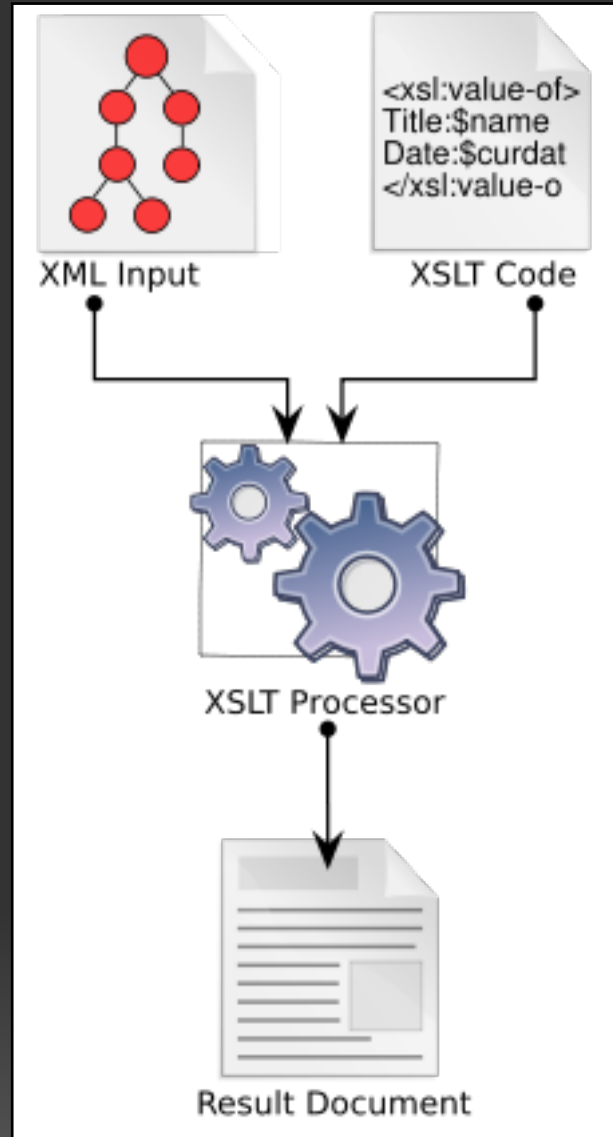
# XSLT

- XSLT tähendab XSL Transformations
- XSLT on XSL-i kõige tähtsam osa
- XSLT teisendab XML dokumendi teiseks XML dokumendiks
- XSLT kasutab XPath navigeerimiseks XML dokumendis

# XSLT Implementatsioonid

- **Kliendi poolsed**
  - MSXML
  - TransforMiix
- **Serveri poolsed**
  - Saxon
  - Xalan

# XSLT tööpõhimõte

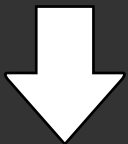




# XSLT Näide

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<?xml-stylesheet type="text/xsl" href="example1.xsl"?>
<catalog>
  <cd>
    <title>Hide your heart</title>
    <artist>Bonnie Tyler</artist>
    <country>UK</country>
    <company>CBS Records</company>
    <price>9.90</price>
    <year>1988</year>
  </cd>
  <cd>
    <title>Greatest Hits</title>
    <artist>Dolly Parton</artist>
    <country>USA</country>
    <company>RCA</company>
    <price>9.90</price>
    <year>1982</year>
  </cd>
</catalog>
```

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="/">
  <html>
    <body>
      <h2>My CD Collection</h2>
      <table border="1">
        <tr bgcolor="#9acd32">
          <th align="left">Title</th>
          <th align="left">Artist</th>
        </tr>
        <xsl:for-each select="catalog/cd">
          <tr>
            <td><xsl:value-of select="title"/></td>
            <td><xsl:value-of select="artist"/></td>
          </tr>
        </xsl:for-each>
      </table>
    </body>
  </html>
</xsl:template> </xsl:stylesheet>
```



## My CD Collection

Title	Artist
Hide your heart	Bonnie Tyler
Greatest Hits	Dolly Parton



# Tähtsamad XSLT elemendid

- `xsl:apply-templates`
- `xsl:choose`
- `xsl:for-each`
- `xsl:if`
- `xsl:stylesheet`
- `xsl:template`
- `xsl:variable`
- `xsl:value-of`
- `xsl:call-template`

# <xsl:apply-templates> element

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:template match="/">
  <html>
    <body>
      <table border="1">
        <xsl:for-each select="catalog/cd">
          <tr>
            <td>
              <xsl:apply-templates select="artist" />
            </td>
          </tr>
        </xsl:for-each>
      </table>
    </body>
  </html>
</xsl:template>

<xsl:template match="artist">
  <xsl:value-of select="."/>
</xsl:template>

<xsl:template match="artist[.='Dolly Parton']">
  <b><xsl:value-of select="."/></b>
</xsl:template>

</xsl:stylesheet>
```

# <xsl:choose> element

```
<xsl:choose>
  <xsl:when test="1=2">
    yks
  </xsl:when>
  <xsl:when test="1=1">
    kaks
  </xsl:when>
  <xsl:otherwise>
    kymme
  </xsl:otherwise>
</xsl:choose>
```

# <xsl:for-each> element

```
<xsl:for-each select="catalog/cd">  
    <xsl:value-of select="title"/>  
</xsl:for-each>
```

# <xsl:for-each> filtreerimine

## Operaatorid:

- = (equal)
- != (not equal)
- &lt; less than
- &gt; greater than

## Näide:

```
<xsl:for-each select="catalog/cd[artist='Dolly Parton']">
    <xsl:value-of select="artist"/>
</xsl:for-each>
```

# <xsl:if> element

```
<xsl:if test="1=1">  
  true  
</xsl:if>
```

# <xsl:stylesheet>

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

  Stylesheet content.

</xsl:stylesheet>
```



# <xsl:template> element

```
<xsl:template match="/">
  <html>
    <body>
      <h2>Root element matched</h2>
    </body>
  </html>
</xsl:template>
```

# <xsl:variable> element

```
<xsl:variable name="x" select="artist"/>  
<xsl:value-of select="$x"/>
```

# <xsl:value-of> element

```
<xsl:for-each select="catalog/cd">
  <xsl:sort select="artist"/>
  <tr>
    <td><xsl:value-of select="title"/></td>
    <td><xsl:value-of select="artist"/></td>
  </tr>
</xsl:for-each>
```

# <xsl:sort> element

```
<xsl:sort  
  select="expression"  
  lang="language-code"  
  data-type="text|number|qname"  
  order="ascending|descending"  
  case-order="upper-first|lower-first"/>
```

## Näide:

```
<xsl:for-each select="catalog/cd">  
  <xsl:sort select="artist"/>  
  <tr>  
    <td><xsl:value-of select="title"/></td>  
    <td><xsl:value-of select="artist"/></td>  
  </tr>  
</xsl:for-each>
```

# <xsl:call-template>

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:template match="/">
  <html>
    <xsl:call-template name="templ">
      <xsl:with-param name="param1" select="123"/>
    </xsl:call-template>
  </html>
</xsl:template>

<xsl:template name="templ">
  <xsl:param name="param1"/>
  <xsl:value-of select="$param1"/>
</xsl:template>

</xsl:stylesheet>
```

# Funktsioonid

- ceiling
- concat
- count
- floor
- normalize-space
- position
- round
- string
- string-length
- substring
- substring-after
- substring-before
- sum
- translate