
4. XSLT

PB138

XSLT

- messages.xml + messages.xslt = result.xml
- XSLT is composed of templates.

```
<xsl:stylesheet version="1.0"  
xmlns:xsl="http://www.w3.  
org/1999/XSL/Transform">
```

```
<xsl:template match="/">
```

Running the Transformation

- Try transformation from Netbeans and from Java
 - HELLO WORLD
-

Basic Transformations

- `/` vs `/messages`
 - Evaluation of the algorithm
 - `apply-templates` will again apply the whole template file to each child
 - `text()` matching
 - The implicit templates
 - We can override them!
-

Implicit Templates

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

```
<xsl:template match="text()|@"*>  
  <xsl:value-of select="."/>  
</xsl:template>
```

Functional calling

- create a table "id, text"
 - sms text in red color
 - other text in blue

Modes, foreach and IFs

- Write out SMS first!
 - `<xsl:choose>`
 - `<xsl:if>`
 - `<xsl:for-each>`
-

Other important concepts

- `xsl:sort`
 - `xsl:element`
 - `xsl:call-template` (with parameters)
-

xsl:element, xsl:attribute

```
<xsl:element name="{@element}">
```

```
<xsl:attribute name="id">ID1</xsl:attribute>
```

- Output the names in "id" elements:

```
<el1>
```

```
  Filip Nguyen
```

```
</el1>
```
