

The background features abstract, overlapping green geometric shapes, primarily triangles and polygons, in various shades of green, creating a modern and dynamic visual effect. The shapes are layered, with some appearing more prominent than others, and they extend from the edges of the frame towards the center.

XSLT

XSL

- ▶ eXtensible Stylesheet Language Transformations
- ▶ transformer des documents XML
- ▶ standard du W3C
- ▶ s'appuie sur XPATH

XSL

- Les fichiers XSL sont des fichiers XML

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

► Output

```
<xsl:output  
  method="html"  
  encoding="UTF-8"  
  indent="yes" />
```

XSL

- Référencer un document XSL dans un document XML

```
<?xml-stylesheet type="text/xsl" href= "fichier.xsl" ?>
```

Template

```
<xsl:template />
```

- ▶ Attribut **match** : expression XPATH selectionnant des éléments
- ▶ Attribut **name** : permet d'identifier un template pour l'utiliser dans d'autres templates
- ▶ On définit l'un ou l'autre

foreach

```
<xsl:for-each select="XPATH">  
  <xsl:value-of select="@attr" />  
</xsl:for-each>
```

Value-of

```
<xsl:value-of select="element1" ></xsl:value-of>
```

- Sélectionne la première valeur s'il y en a plusieurs

If

```
<xsl:if test="@sexe = 'masculin'">
  <p>
    <xsl:value-of select="element1" ></xsl:value-of>
    <xsl:value-of select="element2" ></xsl:value-of>
  </p>
</xsl:if>
```

a = b
not(a = b)
a = b
a < b
a <= b
a > b
a >= b

Choose

```
<xsl:choose>  
  <xsl:when test="@a = 'test'">  
  
    </xsl:when>  
    <xsl:when test="@a = 'test'">  
  
      </xsl:when>  
      <xsl:otherwise>  
  
        </xsl:otherwise>  
      </xsl:choose>
```

Apply Templates

```
<xsl:apply-templates select="XPATH" />
```

variables

```
<xsl:variable name="variable1" select="'test'" />
```

```
<xsl:variable name="variable2" select="test/test/test[test='test']" />
```

```
<xsl:value-of select="$variable1"/>
```