sindicats -> que es, ventajas y desventajas y crear un fichero

RSS-> quin elements son obligatoris

Quins el ordre

-> Mostra el resultat de un XST. saber interpretarlo

que es el XPath, perque s’utiliza

XML

* XML is primarily designed to **store and exchange inform ation.**

# 1. USOS

1. **Develop a program:** Because it is very simple to work with XML files, you could develop a program that takes the XML data and generates the optimal output.
2. **Use CSS**: In many cases a simple solution will be to use CSS to render information more user-friendly using a browser. *Es pot utilitzar css per fer el xml més llegible o més bonic. Més semblant al html*
3. **Transform the document:** an alternative solution is to transform the document into another format that is designed to be viewed such as: PDF, HTML, XHTML, etc. *Transformarlo en un XHTML o altres formats. Agafa la informació i canviar el format perquè sigui més llegible.*

## A. Desenvolupament

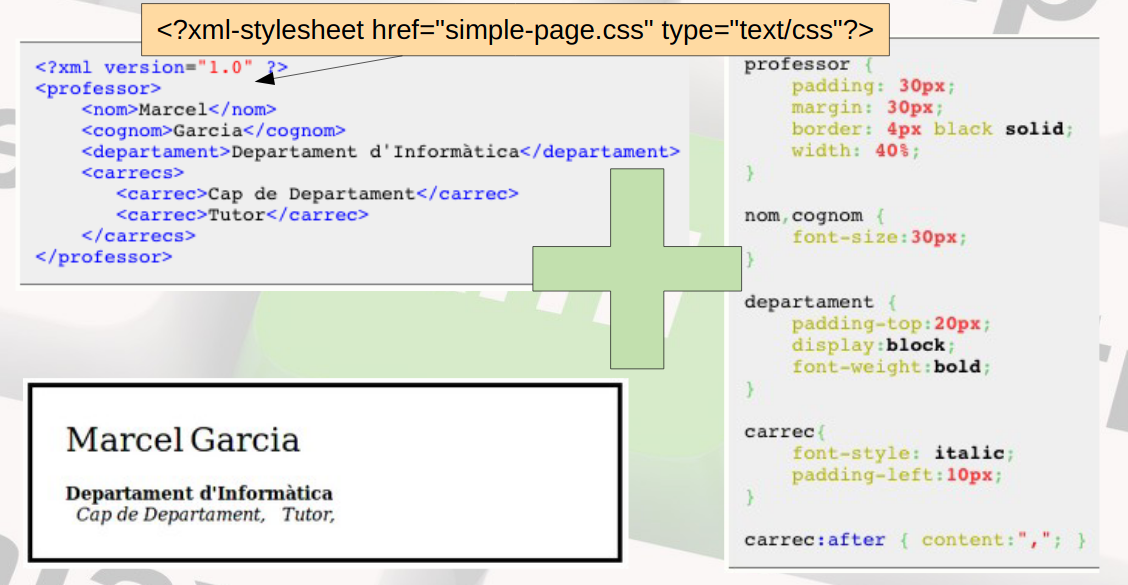
1. [Java with XML](https://www.tutorialspoint.com/java_xml/index.htm)
2. Python with XML
3. C++ with XML
4. C# with XML
5. .NET with XML.

## B. Usos CCS

In many cases, if the goal is to make the document more enjoyable, the simplest solution may be to **view it with CSS.** *Per fer-ho més simple o visible.*

CSS is especially interesting when you want to make **simple adaptations.** If you have an XML document like the following, and want to represent it as a view card. A CSS document containing these lines can be used:

*Declarar al nostre xml el css: es podria veure de la següent manera:*



## C. Transformar

To try and get everything CSS could not do, a new template language was created:

* **XSL** **(extensible stylesheet language).**

Currently XSL is a languagues family used to define **transformations** and **presentations** of XML documents.

*Transformar el XML en altres tipos de documents.*

*El XML té subllenguatges.*

The XSL family is made up of three languages:

* **XSL-FO (**XSL formatting objects): A language for defining the **format** to be applied to a **document**.
* **XSLT** (XSL transformations): a language for **transforming XML documents.**
* **XPath:** A language for **accessing parts of XML documents.**

*XSLT is a language for transforming XML documents.*

*XPath is a language for navigating in XML documents.*

*XQuery is a language for querying XML documents.*

### *El CSS es el full d’estil del HTML. El XSL es el gull d’estil del XML. No comparteixen el mateix tipos de format però si la idea. És més que un llenguatge de fulles d’estil, té més de llenguatges.*

# 2. XSLT

* An XSLT stylesheet is an XML document. It must be well formed.
* The XSLT file will **always** be saved in separate files with the extension .xsl. *Es guarden en un fitxer apart amb extensió xsl*
* They must start with an XML declaration:*Per dintre tindre codi xml*

| <? xml version = "1.0"?>. |
| --- |

* The root element of the XSLT is **stylesheet**. This element **will contain all the others,** and must be preceded by the **xsl alias** corresponding to the namespace for XSLT stylesheets.  *L’arrel será una fulla d’estils: stylesheet. Contindrá tota la resta d’elements. <xsl>*

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

All your doubts about how XSLT works may be solved in this slides presentation:

<http://roble.pntic.mec.es/jtal0007/xmlfp/xslt/UnidadDidactica4XSLT-curso2010-11.pdf>

* **XSL (eXtensible Stylesheet Language) is a styling language for XML.**
* **XSLT stands for XSL Transformations.**

### 

### 

### 

### 

### Exemple:

| <?xml version="1.0"?>  <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>Title</th>  <th>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> |
| --- |

### 

# ESTRUCTURA

* *Crear un fitxer nou xml.*
* *L’àrrel sempre será:*

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

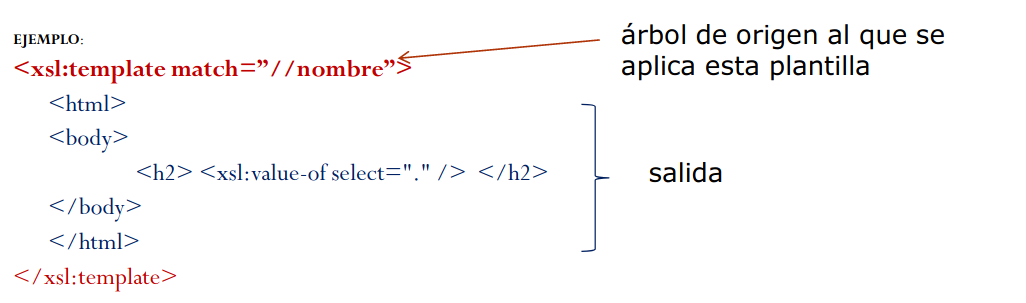
* *Primer: crear etiqueta* ***template***

*L’etiqueta match: es pot posar una plantilla*

***/*** *: coincidirá o posicionará el nostre fitxer xml en el primer node.*

| *<xsl:template match="/">* |
| --- |

* *Etiqueta <html><body>* ***(fixes)***

******

* *Per a cada cd, que estiguin dintre de catàleg, escriurà una nova fila, dintre d’una taula. y per a cada valor, (value-of) posará el text.*

| *<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>* |
| --- |

* *Per tancar*

| *</xsl:template> </xsl:stylesheet>* |
| --- |