**XSLT**

* XSLT stands for XSL Transformations
* XSL (eXtensible Stylesheet Language) is a language for transforming XML documents.
* XSLT is the most important part of XSL
* XSLT transforms an XML document into another XML document
* XSLT uses XPath to navigate in XML documents

**Templates Function (<xsl: template>)**

An XSL style sheet consists of one or more set of rules that are called templates.

**Value-of Function (<xsl: value-of>)**

Value-of element is used to extract the value of a selected node.

**XSLT Code for Value-of Function**

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

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

<Employees>

<id>

<xsl:value-of select=*"//id"*/>

</id>

<name>

<xsl:value-of select=*"//name"*/>

</name>

<dept>

<xsl:value-of select=*"//dept"*/>

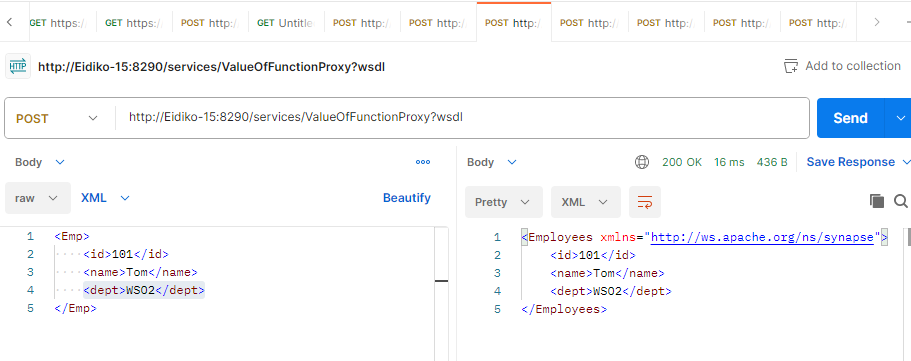
</dept>

</Employees>

</xsl:template>

</xsl:stylesheet>

**Test in Postman**

****

**For-each Function**

For-each element allows you to do looping in XSLT. (<xsl: for-each>)

**XSLT Code for For-each Function**

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

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

<Employees>

<xsl:for-each select=*"Employees/Emp"*>

<Employee>

<id>

<xsl:value-of select=*"id"*/>

</id>

<name>

<xsl:value-of select=*"name"*/>

</name>

<status>success</status>

</Employee>

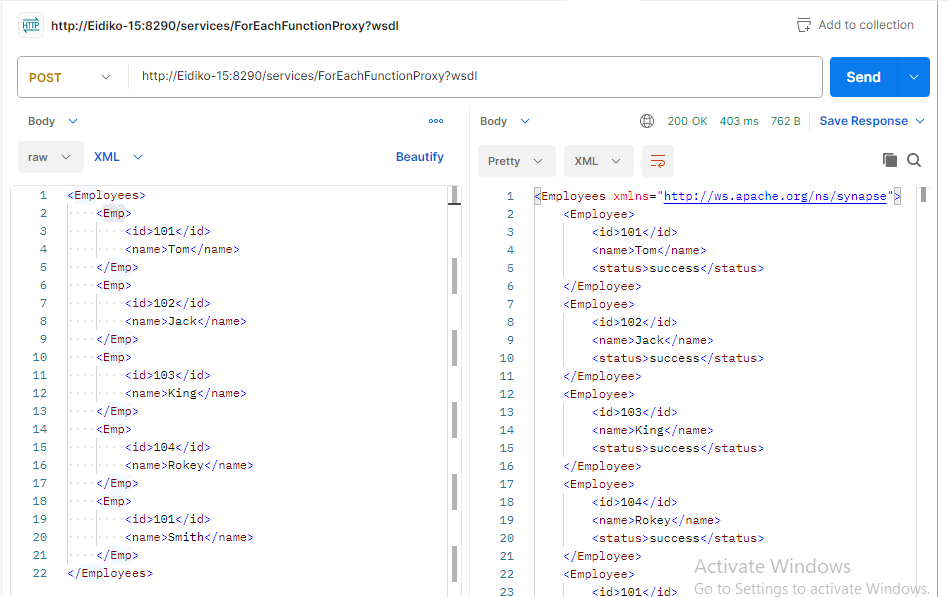
</xsl:for-each>

</Employees>

</xsl:template>

</xsl:stylesheet>

**Test in Postman**



**If function (<xsl: if>)**

If element is used to put a conditional test against the content of the XML file.

**XSLT Code for if Function**

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

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

<Products>

<xsl:for-each select=*"products/product"*>

<xsl:if test=*"price > 25000"*>

<product>

<name>

<xsl:value-of select=*"name"*/>

</name>

<price>

<xsl:value-of select=*"price"*/>

</price>

<status>It is more than 25000</status>

</product>

</xsl:if>

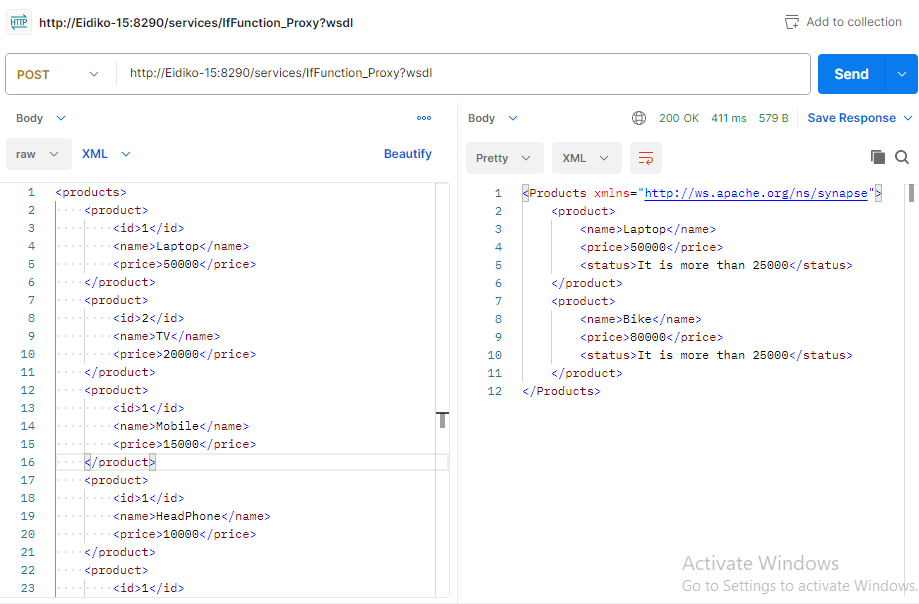
</xsl:for-each>

</Products>

</xsl:template>

</xsl:stylesheet>

**Test in Postman**



**Choose Function (<xsl: choose>)**

The element is used in conjunction with **<xsl: when>** and <**xsl: otherwise>** to express multiple conditional tests.

**XSLT Code for choose Function**

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

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

<Products>

<xsl:for-each select=*"products/product"*>

<xsl:choose>

<xsl:when test=*"price > 25000"*>

<product>

<name>

<xsl:value-of select=*"name"*/>

</name>

<price>

<xsl:value-of select=*"price"*/>

</price>

<status>It is more than 25000</status>

</product>

</xsl:when>

<xsl:otherwise>

<product>

<name>

<xsl:value-of select=*"name"*/>

</name>

<price>

<xsl:value-of select=*"price"*/>

</price>

<status>It is less than 25000</status>

</product>

</xsl:otherwise>

</xsl:choose>

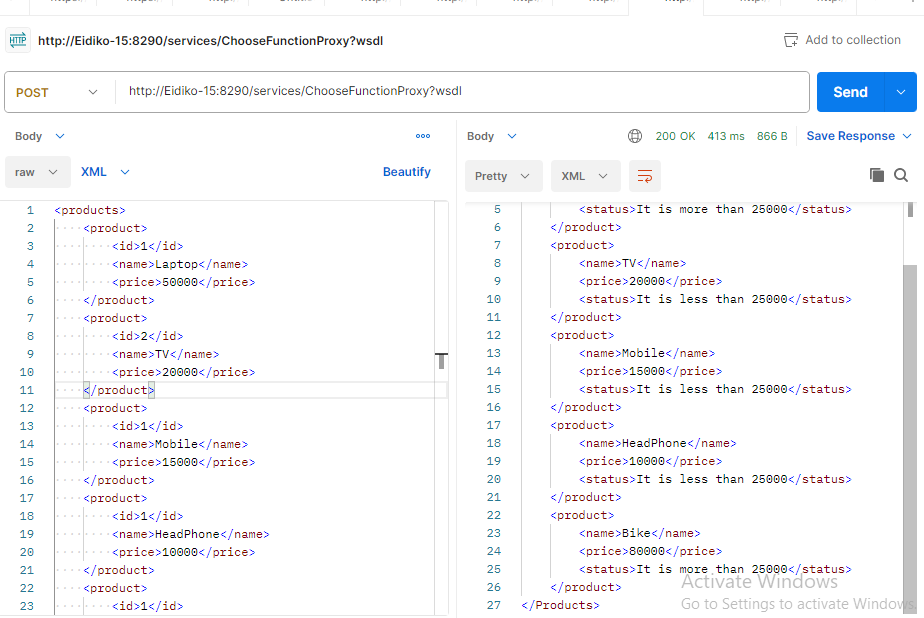
</xsl:for-each>

</Products>

</xsl:template>

</xsl:stylesheet>

**Test in Postman**



**Sort Function**

Sort element is used to sort the output.

**XSLT Code for sort Function**

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

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

<Products>

<xsl:for-each select=*"products/product"*>

<xsl:sort select=*"price"*/>

<product>

<id>

<xsl:value-of select=*"id"*/>

</id>

<name>

<xsl:value-of select=*"name"*/>

</name>

<price>

<xsl:value-of select=*"price"*/>

</price>

</product>

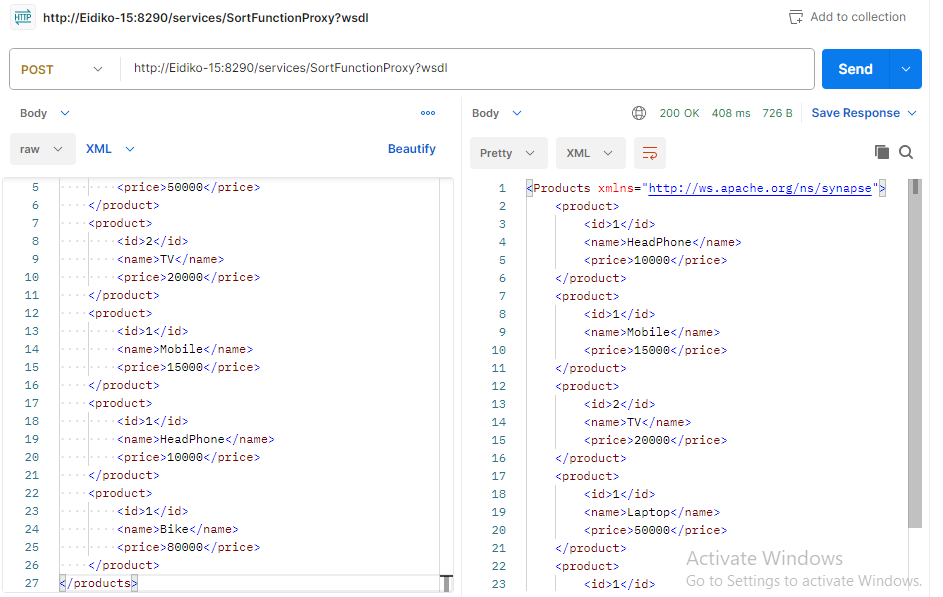
</xsl:for-each>

</Products>

</xsl:template>

</xsl:stylesheet>

**Test in Postman**



Note : Store XSLT code in Registry or Local entry call code using XSLT mediator