

Completion

With Code Completion being one of the key features of IntelliJ IDEA, the plugin provides several possibilities to complete keywords, predefined functions, variables and parameters used in XPath expressions, template names and names of parameters that can be passed to a template invocation.

Completion in XPath expressions

It's possible to complete all parameters/variables in scope inside an XPath expression in a normal expression attribute or inside an [attribute value template](#)

```
<xsl:message>
  <xsl:value-of select="concat('Param: ', $my-)" />
</xsl:message>
```

P	\$my-param1
P	\$my-param2
P	\$my-unused-param

Also, all predefined functions and keywords are available for completion, including function signatures.

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

m	starts-with(string, string)	boolean
m	string(any?)	string
m	string-length(string?)	number

The [Quick Documentation Lookup](#) also works in completion lookup lists.

Completion for template names in xsl:call-template

The template's name that is to be called can be completed from a list of all named templates in the current document and included stylesheets

```
<xsl:call-template name="my-template">
  <xsl:with-param name="my-template" />
  <xsl:with-param name="my-template2" />
</xsl:call-template>
```

Completion for template parameters

There's a special completion for parameters that should be passed to a template in a `xsl:call-template` invocation. The completion lists all parameters that are declared by the template and are not yet present in the *argument list* of the invocation, i.e. there's no `xsl:with-param` yet.

```
<xsl:call-template name="my-template">
  <xsl:with-param name="my-param1" select="123" />
  <xsl:with-param name="my-" />
  <xsl:with-param name="my-param2" select="" />
</xsl:call-template>
```

That's especially useful when also using completion to create the `xsl:with-param` tag, because this will automatically trigger the completion for the parameter's name which is required according to the schema; e.g.

```
<xsl:with-p <ctrl-space>
=>
<xsl:with-param name="[lookup list]"
```