

Apply-Macro Input

Input is supplied to [apply-macro](#) via an input generator.

filenametokenizer

The filenametokenizer input generator allows one to recursively search a directory looking for files that match a specified regular expression. Match groups in the regular expression can be symbolically named as tokens, these tokens being passed into the apply-macro.

The filenametokenizer takes two nested elements: a [FileSet](#) and a `regex` that nests one or more token elements.

```
<filenametokenizer>
  <!-- the files to scan -->
  <fileset dir="${framework.modules.dir}" includes="**/*"/>
  <regex
    pattern="(.*)-(.*).(xml)">
    <token name="filename" matchgroup="0"/>
    <token name="filep1" matchgroup="1"/>
    <token name="filep2" matchgroup="2"/>
    <token name="extension" matchgroup="3"/>
  </regex>
</filenametokenizer>
```

stringlist

The stringlist element allows one to define a delimited string.

```
<stringlist param="item" delimiter="," value="one,two,three"/>
```

join

The join element can take one input generator and a keyedparamvalues set and map the two together.

```
<join>
  <keyedparamvalues>
    <paramvaluepair param="dName" value="dave"/>
    <paramvaluepair param="dType" value="Deployment"/>
  </keyedparamvalues>
  <stringlist param="item" delimiter="," value="one,two,three"/>
</join>
```

mapperinput

See [mapperinput](#) for details.

propertiesquery

See [propertiesquery](#) for details.

queryresults

See [queryresults](#) for details.