

# File tool

\$file allows generating an output report file in a template file.

## \$file.silentCreate(templateFilename)

A shortcut to create a file, the output filename of which is the template name, not an import context object. For example:

```
$file.silentCreate('overview.html')
```

	Name	Type	Description
<b>Parameter(s)</b>	templateFilename	java.lang.String	<ul style="list-style-type: none"><li>The <b>templateFilename</b> parameter can be either an absolute or relative path to a file in the file system.</li><li>If the file is not found in the file system, the filename would be searched as an Attached File Element in a MagicDraw project in the File property.</li></ul>
<b>Return</b>	-	void	

## \$file.silentCreate(attachedFile)

A shortcut to create a file, the output filename of which is the template name, not an import context object. For example:

```
$file.silentCreate($attachedFile)
```

	Name	Type	Description
<b>Parameter(s)</b>	attachedFile	com.nomagic.uml2.ext.magicdraw.classes.mdkernel.Element	An Attached File Element.
<b>Return</b>	-	void	

## \$file.silentCreate(templateFilename, importObject)

A shortcut to create a file, the output filename of which is the template name. For example:

```
$file.silentCreate('overview.html', '')
```

	Name	Type	Description
<b>Parameter(s)</b>	templateFilename	java.lang.String	<ul style="list-style-type: none"><li>The <b>templateFilename</b> parameter can be either an absolute or relative path to a file in the file system.</li><li>If the file is not found in the file system, the filename would be searched as an Attached File Element in a MagicDraw project in the File property.</li></ul>
	importObject	java.lang.Object	An object reference, which will be \$importer in the template file. You can use the \$importer variable in the template file to get data.
<b>Return</b>	-	void	-

## \$file.silentCreate(attachedFile, importObject)

A shortcut to create a file, the output filename of which is the template name. For example:

```
$file.silentCreate($attachedFile, '')
```

	Name	Type	Description
--	------	------	-------------

<b>Parameter(s)</b>	attachedFile	com.nomagic.uml2.ext.magicdraw.classes.mdkernel.Element	An Attached File Element.
	importObject	java.lang.Object	An object reference, which will be \$importer in the template file. You can use the \$importer variable in the template file to get data.
<b>Return</b>	-	void	-

### **\$file.silentCreate(templateFilename, outputFileName, importObject)**

Generate a report output from a given template name. For example:

```
$file.silentCreate('overview.html','overview.html','')
```

	Name	Type	Description
<b>Parameter(s)</b>	templateFilename	java.lang.String	<ul style="list-style-type: none"> <li>The <b>templateFilename</b> parameter can be either an absolute or relative path to a file in the file system.</li> <li>If the file is not found in the file system, the filename would be searched as an Attached File Element in a MagicDraw project in the File property.</li> </ul>
	outputFileName	java.lang.String	The output filename.
	importObject	java.lang.Object	An object reference, which will be \$importer in the template file. You can use the \$importer variable in the template file to get data.
<b>Return</b>	-	void	-

### **\$file.silentCreate(attachedFile, outputFileName, importObject)**

Generate a report output from a given template name. For example:

```
$file.silentCreate($attachedFile,'overview.html','')
```

	Name	Type	Description
<b>Parameter(s)</b>	attachedFile	com.nomagic.uml2.ext.magicdraw.classes.mdkernel.Element	An Attached File Element.
	outputFileName	java.lang.String	The output filename.
	importObject	java.lang.Object	An object reference, which will be \$importer in the template file. You can use the \$importer variable in the template file to get data.
<b>Return</b>	-	void	-

### **\$file.silentCreate(templateType, templateFilename, outputname, importObject)**

Generate a report output from a given template name. For example:

```
$file.silentCreate('html','overview','overview','')
```

	Name	Type	Description
<b>Parameter(s)</b>	templateType	java.lang.String	A template type such as rtf, html, or htm.

	template Filename	java. lang. String	<ul style="list-style-type: none"> <li>The <b>templateFilename</b> parameter can be either an absolute or relative path to a file in the file system.</li> <li>If the file is not found in the file system, the filename would be searched as an Attached File Element in a MagicDraw project in the File property.</li> </ul>
	outputFile Name	java. lang. String	An output filename without a filename extension.
	importOb ject	java. lang. Object	An object reference, which will be \$importer in the template file. You can use the \$importer variable in the template file to get data.
<b>Return</b>	-	void	-

## \$file.silentCreate(templateType, attachedFile, outputname, importObject)

Generate a report output from a given template name. For example:

```
$file.silentCreate('html', $attachedFile, 'overview', '')
```

	Name	Type	Description
<b>Parameter(s)</b>	template Type	java.lang.String	A template type such as rtf, html, or htm.
	attached File	com.nomagic.uml2.ext. magicdraw.classes. mdkernel.Element	An Attached File Element.
	outputFile Name	java.lang.String	An output filename without a filename extension.
	importOb ject	java.lang.Object	An object reference, which will be \$importer in the template file. You can use the \$importer variable in the template file to get data.
<b>Return</b>	-	void	-

## \$file.create(templateFilename)

A shortcut to create a file, the output filename of which is the template name, not an import context object. For example:

```
$file.create('overview.html')
```

	Name	Type	Description
<b>Parameter(s)</b>	template Filename	java. lang. String	<ul style="list-style-type: none"> <li>The <b>templateFilename</b> parameter can be either an absolute or relative path to a file in the file system.</li> <li>If the file is not found in the file system, the filename would be searched as an Attached File Element in a MagicDraw project in the File property.</li> </ul>
<b>Return</b>	-	java. lang. String	An output pathname. It will return an empty string if there is an error.

## \$file.create(attachedFile)

A shortcut to create a file, the output filename of which is the template name, not an import context object. For example:

```
$file.create($attachedFile)
```

	Name	Type	Description
--	------	------	-------------

<b>Parameter(s)</b>	attached File	com.nomagic.uml2.ext.magicdraw.classes.mdkernel.Element	An Attached File Element.
<b>Return</b>	-	java.lang.String	An output pathname. It will return an empty string if there is an error.

### **\$file.create(templateFilename, importObject)**

A shortcut to create a file, the output filename of which is the template name. For example:

```
$file.create('overview.html', '')
```

	Name	Type	Description
<b>Parameter(s)</b>	templateFilename	java.lang.String	<ul style="list-style-type: none"> <li>The <b>templateFilename</b> parameter can be either an absolute or relative path to a file in the file system.</li> <li>If the file is not found in the file system, the filename would be searched as an Attached File Element in a MagicDraw project in the File property.</li> </ul>
	importObject	java.lang.Object	An object reference, which will be importer in the template file. You can use the \$importer variable in the template file to get data.
<b>Return</b>	-	java.lang.String	An output pathname. It will return an empty string if there is an error.

### **\$file.create(attachedFile, importObject)**

A shortcut to create a file, the output filename of which is the template name. For example:

```
$file.create($attachedFile, '')
```

	Name	Type	Description
<b>Parameter(s)</b>	attached File	com.nomagic.uml2.ext.magicdraw.classes.mdkernel.Element	An Attached File Element.
	importObject	java.lang.Object	An object reference, which will be importer in the template file. You can use the \$importer variable in the template file to get data.
<b>Return</b>	-	java.lang.String	An output pathname. It will return an empty string if there is an error.

### **\$file.create(templateFilename, outputFileName, importObject)**

Generate a report output from a given template name. For example:

```
$file.create('overview.html', 'overview.html', '')
```

	Name	Type	Description
<b>Parameter(s)</b>	templateFilename	java.lang.String	<ul style="list-style-type: none"> <li>The <b>templateFilename</b> parameter can be either an absolute or relative path to a file in the file system.</li> <li>If the file is not found in the file system, the filename would be searched as an Attached File Element in a MagicDraw project in the File property.</li> </ul>
	outputFileName	java.lang.String	An output filename.
	importObject	java.lang.Object	An object reference, which will be \$importer in the template file. You can use the \$importer variable in the template file to get data.

<b>Return</b>	-	java.lang.String	An output pathname. It will return an empty string if there is an error.
---------------	---	------------------	--

### **\$file.create(attachedFile, outputFileName, importObject)**

Generate a report output from a given template name. For example:

```
$file.create($attachedFile, 'overview.html', '')
```

	Name	Type	Description
<b>Parameter(s)</b>	attachedFile	com.nomagic.uml2.ext.magicdraw.classes.mdkernel.Element	An Attached File Element.
	outputFileName	java.lang.String	An output filename.
	importObject	java.lang.Object	An object reference, which will be \$importer in the template file. You can use the \$importer variable in the template file to get data.
<b>Return</b>	-	java.lang.String	An output pathname. It will return an empty string if there is an error.

### **\$file.create(templateType, templateFilename, outputname, importObject)**

Generate a report output from a given template name. For example:

```
$file.create('html', 'overview', 'overview', '')
```

	Name	Type	Description
<b>Parameter(s)</b>	templateType	java.lang.String	A template type such rtf, html, and htm.
	templateFilename	java.lang.String	<ul style="list-style-type: none"> <li>The <b>templateFilename</b> parameter can be either an absolute or relative path to a file in the file system.</li> <li>If the file is not found in the file system, the filename would be searched as an Attached File Element in a MagicDraw project in the File property.</li> </ul>
	outputname	java.lang.String	An output filename without a filename extension.
	importObject	java.lang.Object	An object reference, which will be \$importer in the template file. You can use the \$importer variable in the template file to get data.
<b>Return</b>	-	java.lang.String	An output pathname. It will return an empty string if there is an error.

### **\$file.create(templateType, attachedFile, outputname, importObject)**

Generate a report output from a given template name. For example:

```
$file.create('html', $attachedFile, 'overview', '')
```

	Name	Type	Description
<b>Parameter(s)</b>	templateType	java.lang.String	A template type such rtf, html, and htm.
	attachedFile	com.nomagic.uml2.ext.magicdraw.classes.mdkernel.Element	An Attached File Element.

	outputname	java.lang.String	An output filename without a filename extension.
	importObject	java.lang.Object	An object reference, which will be \$importer in the template file. You can use the \$importer variable in the template file to get data.
<b>Return</b>	-	java.lang.String	An output pathname. It will return an empty string if there is an error.

### \$file.createAndWait(templateFilename)

Open a new template engine to generate a report and return a path to generate the file. The generated report will be given the template filename. For example:

```
$file.createAndWait('overview.html')
```

	Name	Type	Description
<b>Parameter(s)</b>	templateFilename	java.lang.String	<ul style="list-style-type: none"> <li>The <b>templateFilename</b> parameter can be either an absolute or relative path to a file in the file system.</li> <li>If the file is not found in the file system, the filename would be searched as an Attached File Element in a MagicDraw project in the File property.</li> </ul>
<b>Return</b>	-	java.lang.String	An absolute path to generate a file.

### \$file.createAndWait(attachedFile)

Open a new template engine to generate a report and return a path to generate the file. The generated report will be given the template filename. For example:

```
$file.createAndWait($attachedFile)
```

	Name	Type	Description
<b>Parameter(s)</b>	attachedFile	com.nomagic.uml2.ext.magicdraw.classes.mdkernel.Element	An Attached File Element.
<b>Return</b>	-	java.lang.String	An absolute path to generate a file.

### \$file.createAndWait(templateFilename, contextValue)

Open a new template engine to generate a report and return a path to generate the file. The generated report will be named after the template filename. For example:

```
$file.createAndWait('overview.html', $var)
```

	Name	Type	Description
<b>Parameter(s)</b>	templateFilename	java.lang.String	<ul style="list-style-type: none"> <li>The <b>templateFilename</b> parameter can be either an absolute or relative path to a file in the file system.</li> <li>If the file is not found in the file system, the filename would be searched as an Attached File Element in a MagicDraw project in the File property.</li> </ul>
	contextValue	java.lang.Object	An additional object added to a new template context with a default \$importer name.
<b>Return</b>		java.lang.String	

### \$file.createAndWait(attachedFile, contextValue)

Open a new template engine to generate a report and return a path to generate the file. The generated report will be named after the template filename. For example:

```
$file.createAndWait($attachedFile,$var)
```

	Name	Type	Description
<b>Parameter(s)</b>	attached File	com.nomagic.uml2.ext.magicdraw.classes.mdkernel.Element	An Attached File Element.
	contextValue	java.lang.Object	An additional object added to a new template context with a default \$importer name.
<b>Return</b>		java.lang.String	

### **\$file.createAndWait(templateFileName, outputFileName, contextValue)**

Open a new template engine to generate a report and return a path to generate the file. "importer" will be used as the name of a context value. For example:

```
$file.createAndWait('overview.html','overview.html',$var)
```

	Name	Type	Description
<b>Parameter(s)</b>	templateFileName	java.lang.String	<ul style="list-style-type: none"> <li>The <b>templateFileName</b> parameter can be either an absolute or relative path to a file in the file system.</li> <li>If the file is not found in the file system, the filename would be searched as an Attached File Element in a MagicDraw project in the File property.</li> </ul>
	outputFileName	java.lang.String	An output report filename.
	contextValue	java.lang.Object	An additional object added to a new template context with a default \$importer name.
<b>Return</b>		java.lang.String	An absolute path to generate a file.

### **\$file.createAndWait(attachedFile, outputFileName, contextValue)**

Open a new template engine to generate a report and return a path to generate the file. "importer" will be used as the name of a context value. For example:

```
$file.createAndWait($attachedFile,'overview.html',$var)
```

	Name	Type	Description
<b>Parameter(s)</b>	attached File	com.nomagic.uml2.ext.magicdraw.classes.mdkernel.Element	An Attached File Element.
	outputFileName	java.lang.String	An output report filename.
	contextValue	java.lang.Object	An additional object added to a new template context with a default \$importer name.
<b>Return</b>		java.lang.String	An absolute path to generate a file.

### **\$file.createAndWait(templateFileName, outputFileName, ContextName, contextValue)**

Open a new template engine to generate a report and return a path to generate the file. For example:

```
$file.createAndWait('overview.html','overview.html','var',$var)
```

	Name	Type	Description
<b>Parameter(s)</b>	templateFilename	java.lang.String	<ul style="list-style-type: none"> <li>The <b>templateFilename</b> parameter can be either an absolute or relative path to a file in the file system.</li> <li>If the file is not found in the file system, the filename would be searched as an Attached File Element in a MagicDraw project in the File property.</li> </ul>
	outputFilename	java.lang.String	An output report filename.
	contextName	java.lang.String	Add an object context's name.
	contextValue	java.lang.Object	An additional object added to a new template context with a default \$importer name.
<b>Return</b>	-	java.lang.String	An absolute path to generate a file.

### **\$file.createAndWait(attachedFile, outputFileName, ContextName, contextValue)**

Open a new template engine to generate a report and return a path to generate the file. For example:

```
$file.createAndWait($attachedFile, 'overview.html', 'var', $var)
```

	Name	Type	Description
<b>Parameter(s)</b>	attachedFile	com.nomagic.uml2.ext.magicdraw.classes.mdkernel.Element	An Attached File Element.
	outputFilename	java.lang.String	An output report filename.
	contextName	java.lang.String	Add an object context's name.
	contextValue	java.lang.Object	An additional object added to a new template context with a default \$importer name.
<b>Return</b>	-	java.lang.String	An absolute path to generate a file.

### **\$file.createAndWait(templateFileName, outputFileName, context)**

Open a new template engine to generate a report and return a path to generate the file. For example:

```
$file.createAndWait('overview.html', 'overview.html', $map)
```

	Name	Type	Description
<b>Parameter(s)</b>	templateFilename	java.lang.String	<ul style="list-style-type: none"> <li>The <b>templateFilename</b> parameter can be either an absolute or relative path to a file in the file system.</li> <li>If the file is not found in the file system, the filename would be searched as an Attached File Element in a MagicDraw project in the File property.</li> </ul>
	outputFilename	java.lang.String	An output report filename.
	context	java.util.Map<String, Object>	An additional context added to a new template.
<b>Return</b>	-	java.lang.String	An absolute path to generate a file.

### **\$file.createAndWait(attachedFile, outputFileName, context)**



Open a new template engine to generate a report and return a path to generate the file. For example:

```
$file.createAndWait($attachedFile, 'overview.html', $map)
```

	Name	Type	Description
<b>Parameter(s)</b>	attachedFile	com.nomagic.uml2.ext.magicdraw.classes.mdkernel.Element	An Attached File Element.
	outputFilename	java.lang.String	An output report filename.
	context	java.util.Map<String, Object>	An additional context added to a new template.
<b>Return</b>	-	java.lang.String	An absolute path to generate a file.

### \$file.copy(inputFilename)

Copy an input file to an output file using the same name in the binary format. For example:

```
$file.copy('icon.gif')
```

	Name	Type	Description
<b>Parameter(s)</b>	inputFilename	java.lang.String	An input filename
<b>Return</b>	-	java.lang.String	An output pathname. It will return an empty string if there is an error.

### \$file.copy(inputFilename, outputFilename)

Copy an input file to an output file in the binary format. For example:

```
$file.copy('icon.gif', 'icon.gif')
```

	Name	Type	Description
<b>Parameter(s)</b>	inputFilename	java.lang.String	An input filename.
	outputFilename	java.lang.String	An output filename.
<b>Return</b>	-	java.lang.String	An output pathname. It will return an empty string if there is an error.

### \$file.exists(pathname)

Test if a file denoted by a specific pathname exists. By default, the current directory refers to a template location. For example:

```
$file.exists("$template.resourceLocation/myimage.png")
$file.exists("C:/myfolder/myimage.png")
$file.exists("mytemplate.txt")
```

	Name	Type	Description
<b>Parameter(s)</b>	pathname	java.lang.String	A pathname string.

<b>Return</b>	-	java.lang.String	True if and only if a file or directory denoted by a specific pathname exists; otherwise, false.
---------------	---	------------------	--

### **\$file.computeName(directory, name)**

Create a pathname string from a given directory and name.

	<b>Name</b>	<b>Type</b>	<b>Description</b>
Parameter(s)	directory	java.lang.String	A parent directory of a file.
	name	java.lang.String	A filename (excludes the filename extension).
Return	-	java.lang.String	A pathname from the given directory and name.

For example:

```
$file.computeName('actors', $ac.ID, 'html')
output: 'actor/_123456789.html'
```

### **\$file.computeName(directory, name, fileType)**

Create a pathname string from a given directory, name, and file type.

	<b>Name</b>	<b>Type</b>	<b>Description</b>
Parameter(s)	directory	java.lang.String	A parent directory of a file.
	name	java.lang.String	A filename (excluding the extension).
	fileType	java.lang.String	A file type, such as rtf, txt, and html.
Return	-	java.lang.String	A pathname from the given directory, name, and file type.