

---

# Command Reference: AppendFile()

## Append 1+ files to another file

Version 10.12.00, 2012-07-22

The `AppendFile()` command appends one or more files to another file. All or only matching lines from input files can be transferred. This command is useful for appending multiple data files into a single file that can be read by TSTool.

The following dialog is used to edit the command and illustrates the syntax for the command.

**Edit AppendFile() command**

Append the contents of one or more files to another file.  
The input file can be a single file, all files in a folder (\*), or all files matching an extension (\*.csv).  
Use the IncludeText parameter to append only lines that match a pattern using the Java regular expressions.  
It is recommended that the file name be relative to the working directory, which is:  
C:\Develop\TSTool\_SourceBuild\TSTool\test\regression\commands\general\AppendFile

Input file(s):

Output file(s):

Include text:  Optional - include lines matching regular expression (default=include all)

If not found?:  Optional - action if input file is not found (default=Warn)

Command: 

```
AppendFile ( InputFile="Data/ *.csv", OutputFile="Results/Test_AppendFile_Extension_IncludeText_out.csv", IncludeText=".*\Q-\E.*" )
```

AppendFile

**AppendFile() Command Editor**

The command syntax is as follows:

```
AppendFile (Parameter=Value, ...)
```

#### Command Parameters

Parameter	Description	Default
InputFile	The name of one or more files to delete, using the following conventions: <ul style="list-style-type: none"> <li>No * in name – match one file.</li> <li>Filename of * – match all files in input directory (working directory by default).</li> <li>Filename of *.ext – match all files with extension</li> </ul> More options may be supported in the future.	None – must be specified.
OutputFile	The output file that will be appended to. The file is created if it does not exist. Use the RemoveFile() command to remove the old file.	None – must be specified.
IncludeText	A regular expression pattern to include text. This uses the Java regular expressions syntax (see <a href="http://en.wikipedia.org/wiki/Regular_expression">http://en.wikipedia.org/wiki/Regular_expression</a> ).	Transfer all lines.
IfNotFound	Indicate action if the file is not found, one of: <ul style="list-style-type: none"> <li>Ignore – ignore the missing file (do not warn).</li> <li>Warn – generate a warning (use this if the file truly is expected and a missing file is a cause for concern).</li> <li>Fail – generate a failure (use this if the file truly is expected and a missing file is a cause for concern).</li> </ul>	Warn

The following table lists regular expression examples:

InputText Regular Expression	Description
.*\Q-\E.*	Match lines that start with any character, end with any character, and contain a dash. The \Q and \E characters are special characters to start and end a quoted character, and are necessary because the dash has special meaning in a regular expression.