Command Reference: If()

Start a block of commands as part of a conditional "if"

Version 10.27.00, 2014-03-20

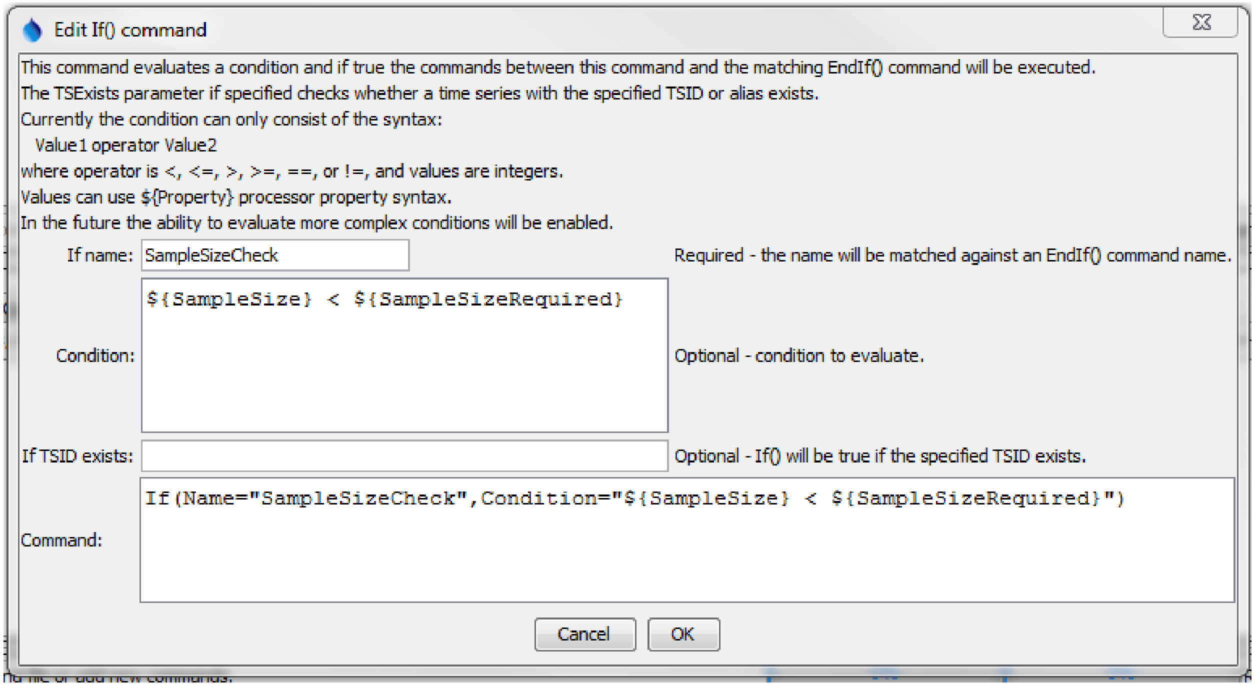
The `If()` command evaluates a conditional statement and if true will result in the commands between `If()` and matching `EndIf()` being executed. The `If()` and `EndIf()` commands must have the same value for the `Name` parameter to allow TSTool to determine the start and end of the block. Currently, there is no "else if" or "else" syntax. The syntax for the conditional statement is restricted to:

Value1 operator Value2

Where the values can be integers and the operator is one of the following (more functionality will be added in the future):

- <
- <=
- >
- >=
- ==
- !=

`If()` commands can be nested. All nested `If()` commands must evaluate to true to execute the commands within the deepest level of nesting. Some commands, including `SelectTimeSeries()` and `ReadTimeSeriesList()` now set a property that can be used for checks. The following dialog is used to edit this command and illustrates the command syntax.



Edit If() command

This command evaluates a condition and if true the commands between this command and the matching EndIf() command will be executed.
The TSExists parameter if specified checks whether a time series with the specified TSID or alias exists.
Currently the condition can only consist of the syntax:
  Value1 operator Value2
where operator is <, <=, >, >=, ==, or !=, and values are integers.
Values can use ${Property} processor property syntax.
In the future the ability to evaluate more complex conditions will be enabled.

If name: SampleSizeCheck — Required - the name will be matched against an EndIf() command name.

Condition: `${SampleSize} < ${SampleSizeRequired}` — Optional - condition to evaluate.

If TSID exists: — Optional - If() will be true if the specified TSID exists.

Command:

```
If(Name="SampleSizeCheck",Condition="${SampleSize} < ${SampleSizeRequired}")
```

If() Command Editor

The command syntax is as follows:

```
If(Parameter=Value,…)
```

Command Parameters

Parameter	Description	Default
`Name`	The name of the "if" command, which will be matched with the name of an `EndIf()` command to indicate the block of commands in the "if" command.	None – must be specified.
`Condition`	The conditional statement to evaluate.	`Condition` and/or `TSExists` must be specified.
`TSExists`	Specify a TSID or alias to match.	`Condition` and/or `TSExists` must be specified.

The following example illustrates combinations of `If()` and `Message()` commands (indentation indicates line continuation for messages).

```
# Test evaluating an integer condition where integer is supplied by property
#@expectedStatus Warning
StartLog(LogFile="Results/Test_If_IntegerProperty_LT_IntegerProperty.TSTool.log")
SetProperty(PropertyName="SampleSizeRequired",PropertyType=Integer,
  PropertyValue="10")
SetProperty(PropertyName="SampleSize",PropertyType=Integer,PropertyValue="5")
If(Name="SampleSizeCheck",Condition="${SampleSize} < ${SampleSizeRequired}")
Message(Message="Sample size (${SampleSize}) is less than required
  ${SampleSizeRequired}",CommandStatus=WARNING)
EndIf(Name="SampleSizeCheck")
If(Name="SampleSizeCheck2",Condition="${SampleSize} > ${SampleSizeRequired}")
Message(Message="Sample size (${SampleSize}) is >= than required
  ${SampleSizeRequired}",CommandStatus=WARNING)
EndIf(Name="SampleSizeCheck2")
If(Name="SampleSizeCheck3Outer",Condition="${SampleSize} < ${SampleSizeRequired}")
If(Name="SampleSizeCheck3InnerTrue",Condition="${SampleSize} == 5")
Message(Message="Sample size (${SampleSize}) is == 5",CommandStatus=WARNING)
EndIf(Name="SampleSizeCheck3InnerTrue")
If(Name="SampleSizeCheck3InnerFalse",Condition="${SampleSize} != 6")
Message(Message="Sample size (${SampleSize}) is not == 6",CommandStatus=WARNING)
EndIf(Name="SampleSizeCheck3InnerFalse")
EndIf(Name="SampleSizeCheck3Outer")
```