Command Reference: If()

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

Version 10.29.00, 2014-05-18

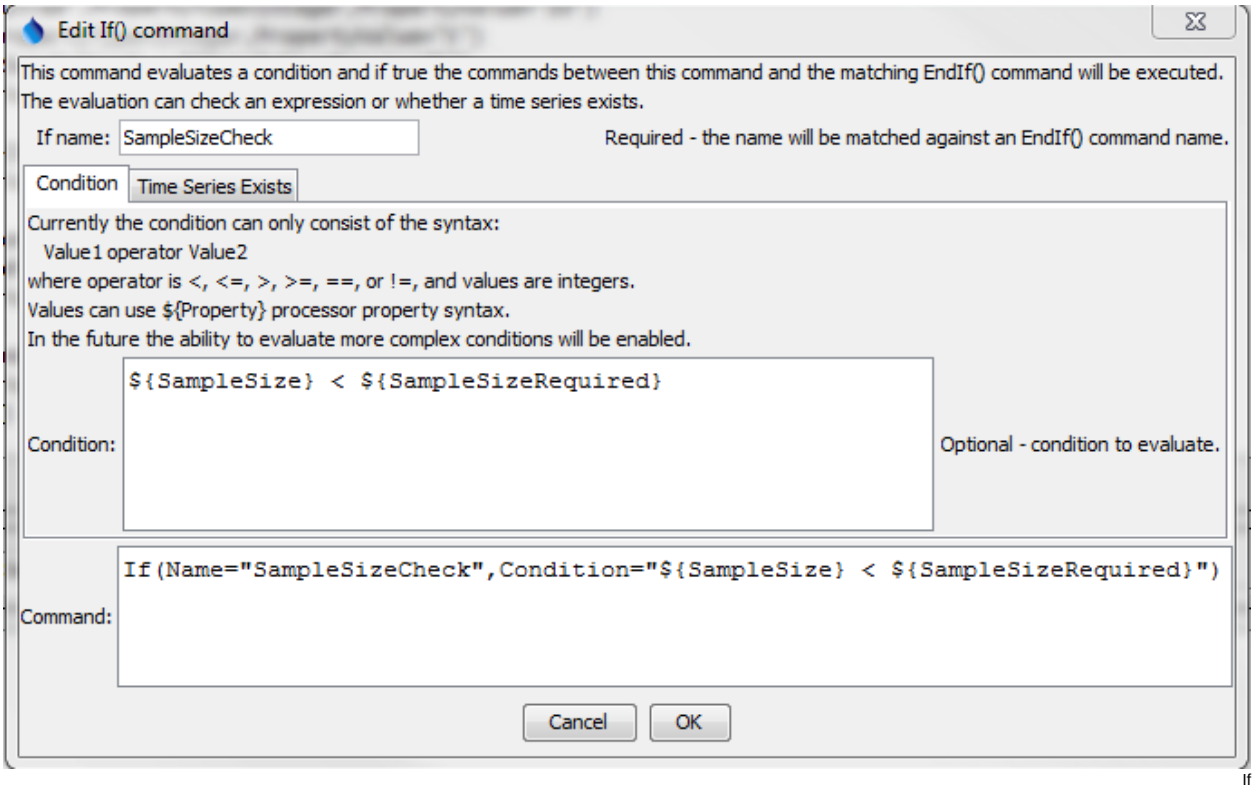
The `If()` command evaluates a conditional statement and if true will result in the commands between `If()` and matching `EndIf()` being executed (the `Name` parameter must match). Currently, there is no "else if" or "else" syntax. The syntax for the conditional statement is restricted to:

Value1 operator Value2

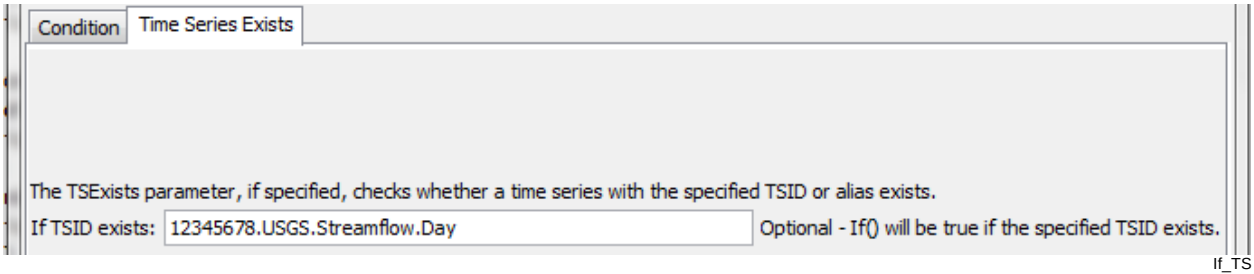
Where the values can be integers or integer processor properties and the operator is one of the following (more functionality will be added in the future):

- <
- <=
- >
- >=
- == (use this to test equality – do not use a single equal sign)
- !=

`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()`, `CopyTable()`, 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.



If() Command Editor Showing Condition to Test



If_TS

If() Command Editor Showing Check for Time Series Existence

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). In these examples processor properties are used to provide condition values.

```
# 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")
```