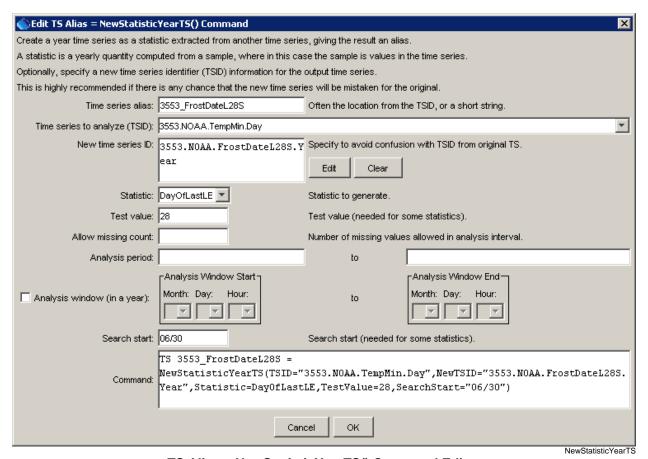
# Command Reference: TS Alias = NewStatisticYearTS()

Create a new yearly time series containing a statistic determined from a time series

Version 08.15.00, 2008-05-04

The TS Alias = NewStatisticYearTS() command processes a time series and creates a new yearly time series, assigning the result an alias so that it can be more easily manipulated. Each yearly value in the resulting time series contains a statistic determined from the sample of points from the corresponding year in the original time series. For example, if the original time series has a daily time step, then the sample that is analyzed will be 365 or 366 values (depending on leap year). This command is useful because it operates on the raw time series data. Other commands (e.g., ChangeInterval() can produce a similar result, for example converting a daily time series to monthly average). The NewStatisticYearTS() command is being phased in, offering statistics that have been requested to meet a need. It is envisioned the list of statistics will increase over time and that additional optional parameters may be added (e.g., to indicate how to handle missing data in the sample).

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



TS Alias = NewStatisticYearTS() Command Editor

The command syntax is as follows:

TS Alias = NewStatisticYearTS(Parameter=value,...)

### **Command Parameters**

Parameter	Parameter Description			
Alias	The alias of the new time series, which can be used	None – must be		
	instead of the TSID in other commands.	specified.		
TSID	The time series identifier (or alias) of the time series to	None – must be		
	analyze.	specified.		
NewTSID	The time series identifier to be assigned to the new time	Use the same		
	series, which is useful to avoid confusion with the original	identifier as the		
	time series.	original time series,		
		with an interval of		
		Year and a data type		
		matching the statistic.		
Statistic	See the <b>Available Statistics</b> table below.	None – must be		
		specified.		
TestValue	A test value used when analyzing the statistic.	This parameter is		
		required for some		
		statistics and not used		
		for others.		
Allow	The number of missing values allowed in the source	-1 – allow any		
Missing	interval(s) in order to produce a result. This capability	number of missing		
Count	should be used with care because it may result in data that	values		
	are not representative of actual conditions.			
TestStart	Specify the start in a year to apply the test, for example to	This parameter is		
	limit the test to a season.	planned for		
		implementation in the		
		future.		
TestEnd	Specify the end in a year to apply the test, for example to	This parameter is		
	limit the test to a season.	planned for		
		implementation in the		
		future.		
AnalysisStart	The date/time for the analysis start, using a precision that	Analyze the full		
	matches the original time series, and 9999 for year, for	period.		
	example 9999-01-01 00.			
AnalysisEnd	The date/time for the analysis start, using a precision that	Analyze the full		
	matches the original time series, and 9999 for year, for	period.		
g 1.5:	example 9999-10-31 00.			
SearchStart	The date/time to begin a data search when processing the	This parameter is		
	statistic. For example, for the DayOfLastLE statistic	optional for the		
	applied to daily data, specify the MM/DD or MM-DD of the	DayOf statistics. By		
	day to begin searching backwards in the year.	default, searches start		
		on Jan 1 for forward		
		searches and Dec 31		
		for backward		
		searches.		

## **Available Statistics**

Statistic	Description	Limitations
CountOfGE	Count of values in a year >= TestValue.	Analysis is limited to daily data.
CountOfGT	Count of values in a year > TestValue.	Analysis is limited to daily data.
CountOfLE	Count of values in a year <= TestValue.	Analysis is limited to daily data.
CountOfLT	Count of values in a year < TestValue.	Analysis is limited to daily data.
DayOfFirstGE	Julian day of the year (1-366) for the first	Analysis is limited to daily data.
	data value >= TestValue. Searches start	
	at the start of the year and move forward	
	(SearchStart will reset the search start).	
DayOfFirstGT	Similar to DayOfFirstGE, for values >	Analysis is limited to daily data.
	TestValue.	
DayOfFirstLE	Similar to DayOfFirstGE, for values <=	Analysis is limited to daily data.
	TestValue.	
DayOfFirstLT	Similar to DayOfFirstGE, for values <	Analysis is limited to daily data.
	TestValue.	
DayOfLastGE	Julian day of the year (1-366) for the last	Analysis is limited to daily data.
	data value >= TestValue. Searches start	
	at the end of the year and move backward	
	(SearchStart will reset the search start).	
DayOfLastGT	Similar to DayOfLastGE, for values >	Analysis is limited to daily data.
	TestValue.	
DayOfLastLE	Similar to DayOfLastGE, for values <=	Analysis is limited to daily data.
	TestValue.	
DayOfLastLT	Similar to DayOfLastGE, for values <	Analysis is limited to daily data.
	TestValue.	
DayOfMax	Julian day of the year (1-366) for the	Analysis is limited to daily data.
	maximum value in the time series.	
DayOfMin	Julian day of the year (1-366) for the	Analysis is limited to daily data.
	minimum value in the time series.	
Max	Maximum value in a year.	Analysis is limited to daily data.
Min	Minimum value in a year.	Analysis is limited to daily data.

# Handling of Missing Data at the End of the Period for DayOfXXXXxx Statistics with SearchStart

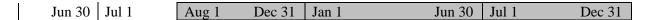
Day of year statistics that start the search on a given day in the year handle missing data at the start or end of the period as discussed in the following examples. For discussion purposes, SearchStart is June 30 for backward searches (to find the last occurrence in a year) and July 1 for forward searches (to find the first occurrence in a year).

#### Searching Forward with Gap at Start

Jun 30	Jul 1	Aug 1	Dec 31	Jan 1	Jun 30	Jul 1	Dec 31

In the above case, the period July 1 to July 31 would be treated as missing and the statistic would only be computed if AllowMissingCount is greater than the number of missing values in this initial period.

#### Searching Backward with Gap at Start



In the above case, the period on and before June 30 would be treated as missing and the statistic would not be computed in any case.

#### Searching Forward with Gap at End



In the above, case the period on and after July 1 would be treated as missing and the statistic would not be computed in any case.

#### Searching Backward with Gap at End



In the above case, the period between March 15 and June 30 would be treated as missing and the statistic would only be computed if AllowMissingCount is greater than the number of missing values in this initial period.

# **Examples**

The following example commands file computes the last spring frost date for 28 degrees and 32 degrees, searching backwards from June 30 each year, and the first fall frost date for 32 and 28 degrees, searching forwards from July 1 each year:

```
StartLog(LogFile="FrostDates HydroBase.log")
SetOutputPeriod(OutputStart="1950-01",OutputEnd="2004-12")
# 3553 - GREELEY UNC
3553.NOAA.TempMin.Day~HydroBase
TS 3553_FrostDateL28S = NewStatisticYearTS(TSID="3553.NOAA.TempMin.Day",
  NewTSID="3553.NOAA.FrostDateL28S.Year",
  Statistic=DayOfLastLE, TestValue=28,
  SearchStart="06/30")
TS 3553_FrostDateL32S = NewStatisticYearTS(TSID="3553.NOAA.TempMin.Day",
 NewTSID="3553.NOAA.FrostDateL32S.Year",
  Statistic=DayOfLastLE, TestValue=32,
  SearchStart="06/30")
TS 3553_FrostDateF32F = NewStatisticYearTS(TSID="3553.NOAA.TempMin.Day",
 NewTSID="3553.NOAA.FrostDateF32F.Year",
  Statistic=DayOfFirstLE, TestValue=32,
  SearchStart="07/01")
TS 3553_FrostDateF28F = NewStatisticYearTS(TSID="3553.NOAA.TempMin.Day",
  NewTSID="3553.NOAA.FrostDateF28F.Year",
  Statistic=DayOfFirstLE, TestValue=28,
  SearchStart="07/01")
Free(TSID="*.*.TempMin.*")
writeStateCU("Results/Test.FrostDates")
```