Killing a global variable reduces clutter and saves a bit of memory. You can not kill a system variable or local variable.

To kill all global variables in the current data folder, use KillVariables/A/Z.

String Variables

You create user string variables by calling the `String` operation from the command line or in a procedure. The syntax is:

```
String [/G] strName [=strExpr] [,strName [=strExpr]... ]
```

The optional /G flag specifies that the string is to be global, and it overwrites any existing string variable.

The string variable is initialized when it is created if you supply the initial value with a string expression using =*strExpr*. If you create a string variable and specify no initializer it is initialized to the empty string ("").

When you call String from the command line or from a macro, the string variable is initialized to the specified initial value or to the empty string ("") if you provide no initial value.

When you declare a local string variable in a user-defined function, it is null (has no value) until you assign a value to it, via either the initial value or a subsequent assignment statement. Igor generates an error if you use a null local string variable in a user-defined function.

When you call String in a procedure, the new string is local to that procedure unless you include the /G flag. When you call String from the command line, the new string is always global.

You can create more than one string variable at a time by separating the names and optional initializers for multiple string variables with a comma.

Here is an example of variable creation with initialization:

```
String str1 = "This is string 1", str2 = "This is string 2"
```

Since /G was not used, these strings would be global if you invoked String directly from the command line or local if you invoked it in a procedure.

`String/G strName` can be invoked whether or not a variable of the given name already exists. If it does exist as a string, its contents are not altered by the operation unless the operation includes an initial value for the string.

You can kill (delete) a global string using the Data Browser or the `KillStrings` operation. The syntax is:

```
KillStrings [flags] [stringName [,stringName ]...]
```

There are two optional *flags*:

A	Kills all global strings in the current data folder. If you use /A, omit stringName.
/Z	Doesn't generate an error if a global string to be killed does not exist.

For example, to kill global string `myGlobalString` without worrying about whether it was previously defined, use the command:

```
KillStrings/Z myGlobalString
```

Killing a string reduces clutter and saves a bit of memory. You can not kill a local string.

To kill all global strings in the current data folder, use KillStrings/A/Z.

Local and Parameter Variables in Procedures

You can create variables in procedures as parameters or local variables. These variables exist only while the procedure is running. They can not be accessed from outside the procedure and do not retain their values