Note that parentheses must be used in this type of statement. That is a result of the precedence of $ relative to :.

When used at the beginning of a path, the $ operator works in a special way and must be used on the entire path:

```
String path1 = "root:folder1:subfolder1:wave3"
Display $path1
```

When liberal names are used within a path, they must be in single quotes. For example:

```
Display root:folder1:'subfolder 1':'wave 3'
String path1 = "root:folder1:'subfolder 1':'wave 3'"
Display $path1
```

However, when a simple name is passed in a string, single quotes must not be used:

```
Make 'wave 1'
String name
name = "'wave 1'"        // Wrong.
name = "wave 1"          // Correct.
Display $name
```

Data Folder Operations and Functions

Most people will use the Data Browser to create, view and manipulate data folders. The following operations will be mainly used by programmers, who should read **Programming with Data Folders** on page IV-169.

```
NewDataFolder path
SetDataFolder path
KillDataFolder path
DuplicateDataFolder srcPath, destPath
MoveDataFolder srcPath, destPath
MoveString srcPath, destPath
MoveVariable srcPath, destPath
MoveWave wave, destPath [newname]
RenameDataFolder path, newName
Dir
```

The following are functions that are used with data folders.

```
GetDataFolder(mode [, dfr ])
CountObjects(pathStr,type)
GetIndexedObjName(pathStr,type,index)
GetWavesDataFolder(wave,mode)
DataFolderExists(pathStr)
DataFolderDir(mode)
```

Data Folders Reference Functions

Programmers can utilize data folder references in place of paths. Data folder references are lightweight objects that refer directly to a data folder whereas a path, consisting of a sequence of names, has to be looked up in order to find the actual target folder.

Here are functions that work with data folder references:

```
GetDataFolder(mode [, dfr ])
GetDataFolderDFR()
GetIndexedObjNameDFR(dfr, type, index)
GetWavesDataFolderDFR(wave)
CountObjectsDFR(dfr, type)
DataFolderRefChanges(dfr, changeType)
DataFolderRefStatus(dfr)
```