```
	SVAR SJC_YUNITS
	Printf "Y Units: %s\r", SJC_YUNITS

	return 0
End
```

The code above assumes that the header contains the ##NPOINTS label from which the variables VJC_NPOINTS and SJC_YUNITS are created. If you can't guarantee that the file contains such a label, then you must use **NVAR**/Z and **NVAR_Exists** to test for the existence of the variable before using it.

If you need to determine which variables were created at runtime, use the **GetIndexedObjName** function and test each name for the SJC_ or VJC_ prefix.

Another problem with header variables in functions is that they leave a lot of clutter around. You can clean up like this:

```
KillVariables/Z VJC_NPOINTS
KillStrings/Z SJC_YUNITS
```

Loading Sound Files

The **SoundLoadWave** operation, which was added in Igor Pro 7, loads data from various sound file formats.

See **Sound** on page IV-245 for general information on Igor's sound-related features.

Loading Waves Using Igor Procedures

One of Igor's strong points is that it you can write procedures to automatically load, process and graph data. This is useful if you have accumulated a large number of data files with identical or similar structures or if your work generates such files on a regular basis.

The input to the procedures is one or more data files. The output might be a printout of a graph or page layout or a text file of computed results.

Each person will need procedures customized to his or her situation. In this section, we present some examples that might serve as a starting point.

Variables Set by File Loaders

The LoadWave operation creates the numeric variable V_flag and the string variables S_fileName, S_path, and S_waveNames to provide information that is useful for procedures that automatically load waves. When used in a function, the LoadWave operation creates these as local variables.

Most other file loaders create the same or similar output variables.

LoadWave sets the string variable S_fileName to the name of the file being loaded. This is useful for annotating graphs or page layouts.

LoadWave sets the string variable S_path to the full path to the folder containing the file that was loaded. This is useful if you need to load a second file from the same folder as the first.

LoadWave sets the variable V_flag to the number of waves loaded. This allows a procedure to process the waves without knowing in advance how many waves are in a file.

LoadWave also sets the string variable S_waveNames to a semicolon-separated list of the names of the loaded waves. From a procedure, you can use the names in this list for subsequent processing.