

```

    endfor
End

```

Range-Based For Loop

A range-based for loop iterates over all elements of a wave. The basic syntax is:

```

for(<type> varName : <wave expression>)
    <loop body>
endfor

```

The range-based for loop feature was added in Igor Pro 9.00.

During each iteration of the loop, the specified loop variable contains the value of the corresponding wave element. For example:

```

Function Example1()
    Make/O/T/N=(2,2) tw = "p=" + num2str(p) + ",q=" + num2str(q)
    for (String s : tw)
        Print s
    endfor
End

```

Executing Example1 prints this:

```

p=0,q=0
p=1,q=0
p=0,q=1
p=1,q=1

```

You can omit the loop variable type if the loop variable was defined earlier in the function. You can also omit it if the wave expression is a wave reference defined earlier in the function either explicitly or automatically as in Example1. The for statement in Example1 could be written like this:

```

for (s : tw)      // Omit "String" because the tw is a known wave reference

```

Here is an example in which the wave expression is not a wave reference so the loop variable type is required:

```

Function Example2()
    for (String s : ListToTextWave(IndependentModuleList(";" , ";" )))
        Print s
    endfor
End

```

Example2 could be rewritten like this:

```

Function Example3()
    WAVE/T tw = ListToTextWave(IndependentModuleList(";" , ";" ))
    for (s : tw)
        Print s
    endfor
End

```

A range-based for loop iterates over all elements of the wave, regardless of its dimensionality. For a multi-dimensional wave, it iterates in column-major order. For a 2D wave, this means that it iterates over all rows of column 0, then all rows of column 1, and so on for each column.

If the wave is numeric, the type of the loop variable does not need to be an exact match for the wave type. For example, the loop variable can be double even if the wave is an integer wave.

If the wave is a text wave, the loop variable must be a string. If the wave is a data folder reference wave, the loop variable must be a DFREF. If the wave is a wave reference wave, the loop variable must be a WAVE.