

CreationDate

For an example using a base name, see *Generating a Series of Names from a Base Name* next.

Generating a Series of Names from a Base Name

If you are creating a series of waves, such as when loading a data file containing multiple columns, it is sometimes convenient to generate names of the form <baseName><number>, for example, wave0, wave1, wave2. To do this, you pass the base name as the nameInStr parameter and set bit 2 (input name is base name) of the options parameter.

If you disallow creation of names that are in use for existing objects (bit 1 of options cleared), CreateDataObjectName appends a number to the base name such that the output name is not in use for an object in the namespace of the type of object specified by objectType. In this case, suffixNum is not used and you should pass 0 for it.

If you allow creation of names that are in use for existing objects (bit 1 of options set), CreateDataObjectName appends the digits representing suffixNum to the base name to create the output name. In this case, you must pass the desired suffix number in each call to CreateDataObjectName.

Here is an example demonstrating this technique:

```
Function/S CreateSeriesOfWavesWithBaseName(dfr, basePath, beLiberal, allowOverwrite,
    numWavesToCreate)
    DFREF dfr                                // : for current data folder
    String basePath
    int beLiberal
    int allowOverwrite
    int numWavesToCreate

    int options = 4                          // inNameIsBaseName
    if (beLiberal)
        options += 1
    endif
    if (allowOverwrite)
        options += 2
    endif

    int suffixNum = 0
    String list = ""

    int i
    for(i=0; i<numWavesToCreate; i+=1)
        String outName = CreateDataObjectName(dfr, basePath, 1, suffixNum, options)
        Make/O/N=(100) dfr:$outName
        list += outName + ";"
        if (allowOverwrite)
            suffixNum += 1                  // suffixNum for next call
        endif
    endfor

    return list
End
```

With overwrite disallowed, you must create the object each time through the loop because that is the only way that CreateDataObjectName can determine that the wave created in a previous iteration exists.

With overwrite allowed, you do not need to create the object each time through the loop although that is normally what you want.

See Also

[Object Names](#) on page III-501, [Programming with Liberal Names](#) on page IV-168, [CheckName](#), [CleanupName](#), [UniqueName](#)

CreationDate

CreationDate (waveName)

Returns creation date of wave as an Igor date/time value, which is the number of seconds from 1/1/1904.

The returned value is valid for waves created with Igor Pro 3.0 or later. For waves created in earlier versions, it returns 0.

See Also

[ModDate](#).