Subject: Re: code snippet
Posted by m.hadfield on Wed, 30 May 2001 00:38:48 GMT
View Forum Message <> Reply to Message

From: "Craig Markwardt" <craigmnet@cow.physics.wisc.edu>

```
>       choice = (['first','second','third','fourth'])[key]
>       choice = (['first','second','third','fourth'])(key)
>       choice = (['first','second','third','fourth'])([key])
>       choice = (['first','second','third','fourth'])[(key)]
>  One of these four expressions has a slightly different meaning than
>  the other.  As a trivia test to the newer users of IDL on the
>  newsgroup (or the older ones!), can you say which one is different?
```

I think I qualify as one of the older ones.

IMHO you could (and should) eliminate some of the confusion by declaring "compile_opt STRICTARR" or "compile_opt IDL2" in your startup file so that numbers 2 and 3 are illegal.

Now, this

```
    choice = (['first','second','third','fourth'])[key]
```

is the same as this

```
    choice = (['first','second','third','fourth'])[(key)]
```

and indeed they are both the same as this

```
   choice = (['first','second','third','fourth'])[((((((key))))))]
```

because all those parentheses around "key" have no effect.

But this

```
    choice = (['first','second','third','fourth'])[[key]]
```

is different from all the others because the index is now a 1-D array, so the result is also a 1-D array.

But what about these

```
    choice = (['first','second','third','fourth'])[[[[[[key]]]]]]
    choice = (['first','second','third','fourth'])[([([(key)])])]
    choice = ([['first','second'],['third','fourth']])[[key]]
    choice = ([['first','second','third'],'fourth'])[[key]]
```

I'll leave them as an exercise (I think they're all valid).

Mark Hadfield
m.hadfield@niwa.cri.nz http://katipo.niwa.cri.nz/~hadfield
National Institute for Water and Atmospheric Research

--
Posted from clam.niwa.cri.nz [202.36.29.1]
via Mailgate.ORG Server - http://www.Mailgate.ORG