

---

Subject: How to create a varying variable?

Posted by [atmospheric physics](#) on Mon, 06 Oct 2014 14:13:41 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Hello,

I want to create a varying variable, 'varJJJ' as below:

```
varJJJ = FLTARR(nc,nr)
```

Here JJJ = STRING(indgen(101),FORMAT='(I03)')

Is this possible? Can anyone have a better suggestion?

Thanks in advance,  
Madhavan

---

---

Subject: Re: How to create a varying variable?

Posted by [Matthew Argall](#) on Mon, 06 Oct 2014 14:25:41 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

This discussion ultimately ends with the Execute command, which most agree is a very bad option for naming variables. Instead, try using a list, structure, or concatenated array:

;List

```
myList = list()
```

```
myList(1) -> Add, fltarr(nc, nr)
```

```
myList(2) -> Add, fltarr(nc, nr)
```

;Array

```
myArr = [[[fltarr(nc, nr)]], [[fltarr(nc, nr)]]]
```

;Structure

```
myStruct = create_struct('Tag1', fltarr(nc, nr), 'Tag2', fltarr(nc, nr))
```

---

---

Subject: Re: How to create a varying variable?

Posted by [Matthew Argall](#) on Mon, 06 Oct 2014 14:33:40 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

> ;List

> myList = list()

> myList(1) -> Add, fltarr(nc, nr)

> myList(2) -> Add, fltarr(nc, nr)

Typo. Should be

```
;List  
myList = list()  
myList -> Add, fltarr(nc, nr)  
myList -> Add, fltarr(nc, nr)
```

I suppose you could also create a hash

```
;Hash  
myHash = hash()  
myHash('Key1') = fltarr(nc, nr)  
myHash('Key2') = fltarr(nc, nr)
```

---

---

Subject: Re: How to create a varying variable?  
Posted by [Paul Van Delst\[1\]](#) on Mon, 06 Oct 2014 15:26:43 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

On 10/06/14 10:13, Madhavan Bomidi wrote:

```
> Hello,  
>  
> I want to create a varying variable, 'varJJJ' as below:  
>  
> varJJJ = FLTARR(nc,nr)  
>  
> Here JJJ = STRING(indgen(101),FORMAT='(I03)')  
>  
> Is this possible? Can anyone have a better suggestion?  
>  
> Thanks in advance,  
> Madhavan
```

First thing that occurs to me is: why?

I'm not trying to be obstreperous, just asking for some context since this sort of request is usually more of a design issue than a language one.

cheers,

paulv

p.s. And, FWIW, Matthew's suggestion using a hash is what I would recommend since you can use the dynamic "JJJ" to access the particular value that you want. E.g.

```
IDL> h=hash()
```

Use string of number:

```
IDL> jjj='101'  
IDL> h[jjj] = findgen(20,10)  
IDL> help, h['101']  
<Expression>  FLOAT   = Array[20, 10]
```

Or use actual number (why convert to string?):

```
IDL> nnn=101  
IDL> h[nnn] = dindgen(14)  
IDL> help, h[101]  
<Expression>  DOUBLE  = Array[14]
```

where, as you can see, h['101'] and h[101] reference different values in the hash.

If you have an older version of IDL (i.e. no hashes) then you could put together something similar using pointer arrays.

All of the plumbing (for either methodology) could then be wrapped up into "Init", "Set" and "Get" functions so your code is protected from the implementation details.

---