
Subject: Re: using LOCATIONS keyword of IDLgrText?

Posted by [Laurens](#) on Fri, 02 Feb 2007 16:20:45 GMT

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

I found a way to make it work, though I don't know if its efficient or nice to do;

had to give a reform command to the positions keyword:

```
, POSITIONS=reform(posArr,3,i)
```

I'm open to suggestions for nices solutions though this works fine for me...

Laurens

Laurens wrote:

> Hi, once again,

>

> I'm starting to feel sorry for myself for not being able to find what I
> want to know; I kinda knī½w that the solution should be somewhere in the
> groups and somehow I think it could be constructed by all the
> J.D.-Smith-bracket-fun on David's site, but I can't make anything out of
> it (call me stupid lol).

>

> Problem: I'm having an IDLgrText-object. I need multiple texts displayed
> on a curve, so I already have an array of strings for that (keyword
> STRINGS). I'm also creating an array for the positions for each string:

>

```
> newPos = [x,y,z]
```

```
> IF (n_elements(posArray) EQ 0) THEN $
```

```
>   posArray = [newPos] $
```

```
> ELSE $
```

```
>   posArray = [posArray,[newPos]]
```

>

```
> IDLgrText->SetProperty, POSITIONS=posArray
```

>

> Somehow though, that doesn't work; the first text is displayed correctly
> but the next ones go to 0,0,0 which tells me that IDL doesn't like the
> positions somehow.

>

> Is my expansion of the Array faulty or does the POSITIONS keyword just
> not work :-) (one can always hope its not his own fault, right :P)

>

> I'm thankful once again :-)

>

>

> Cheers, Laurens

Subject: Re: using LOCATIONS keyword of IDLgrText?
Posted by [badjelly.witch](#) on Sat, 03 Feb 2007 19:07:12 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Feb 3, 5:20 am, Laurens <mr....@inter.nl.net> wrote:
> I found a way to make it work, though I don't know if its
> efficient or nice to do;
>
> had to give a reform command to the positions keyword:
>
> , POSITIONS=reform(posArr,3,i)
>
> I'm open to suggestions for nicer solutions though this works
> fine for me...

That looks nice enough to me, but if to avoid flattening the array when you extend it you can add a couple more brackets:

```
posArray = [[posArray],[newPos]]
```

By the way, extending an array repeatedly like this gets *very* inefficient when the array is large, because you're actually creating a whole new array every time. But for your application this shouldn't be a problem.
