Subject: Re: use of loops in IDL for hashtable
Posted by penteado on Tue, 08 Sep 2009 02:36:25 GMT
View Forum Message <> Reply to Message

On Sep 7, 11:16 pm, "b.a" <u4565...@anu.edu.au> wrote:
> On Sep 7, 6:08 pm, Wox <s...@nomail.com> wrote:
>
>
>
>> On Sun, 6 Sep 2009 19:01:06 -0700 (PDT), "b.a" <u4565...@anu.edu.au>
>> wrote:
>
>>> Hi,
>
>>> I have 2 hashtables in my IDL program. one is called fileList which
>>> has (id , data) as its (key, value) and second one is colorList
>>> (id , color). I want to define a "foreach" loop so that for each id
>>> of colorList which exists in fileList, it plots the data with the
>>> "color". I mean for each string "id" get the "data" from fileList and
>>> get the "color" from colorList and plot the data in that colour.
>
>>> Can anybody help me with this please.
>
>>> Thank you
>
>> Are you using Craig Markwardt's hash table object? This object has a
>> method "KEYS" which returns all existing keys in a hash table, so you
>> can loop over all keys in colorList (or fileList), use "ISCONTAINED"
>> to check whether the key is in fileList (or colorList) and do your
>> plotting when it is (employing the Get method to get data and color).
>
> Thanks for your reply.
> Yes, I am using Craig Markwardt's hash table object. The problem with
> using KEY() is that the keys are not like {1, 2, 3, 4, ...} that I can
> loop over them. they are like {76, 65, 90, 2, 503, ...}. and I don't
> know the largest key. I have to say for i=0 till some number, go i+1
> and if i+1 is contained in the keys, return its value and rest of the
> things. but I can't specify that "some number". So I think I have to
> use "for each" loop. But I don't know the syntax for it in my case.

KEYS() returns an array of key names. So you loop over the elements of this array. Something like:

```
colorkeys=colorList->keys()
for i=0,n_elements(colorkeys)-1 do if fileList->iscontained(colorkeys
[i]) then *do stuff*
```