
Subject: Re: Passing variables

Posted by [jeffwax1](#) on Thu, 23 Aug 2007 15:33:52 GMT

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

> The easiest way is typically to use a common block. At the start of
> everyone program that want to access some variables, put a line like
> this:
>
> common, block_name, variable1, variable2,
>
> A couple rules about common blocks:
>
> 1) The names of the variables don't have to be the same in every
> program.
> 2) The number of variables matter. All common blocks should have the
> same number of variables.
> 3) The positions of the variables matter. Since names don't matter,
> the first variable in one program will always be the same as the first
> variable in another program, even if they have much different names
> 4) Common blocks last until you quit IDL. If you declare it once, it
> stays declared.
> 5) Common blocks can't be extended. This means that if you create a
> common block and later realize you need to add more variables to it,
> you'll have to quit IDL (or type .reset) in order to "destroy" the
> common block.

In the help files the only examples I see of common blocks are inside
procedures. Can I put them inside a function, or even my main
program? I tried putting:

```
common, share, B
```

in both my main program and the program with the function I'm using,
and both came back as syntax errors.

On second thought, after trying to answer my own program, I made a
test program along the lines of:

```
pro test  
common, share, A
```

```
...
```

```
end
```

IDL is insisting that the "common,share,A" is a syntax error.
