

---

Subject: Re: Looking for IDL code documentation standards  
Posted by [Michael Wallace](#) on Tue, 17 May 2005 18:21:18 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

>> Reserved Words - lowercase (e.g. begin, end, do, for, if, while)  
>>  
>> Variables: lowercase, words separated by "\_" (e.g. my\_var)  
>> Routines: lowercase, words separated by "\_" (e.g. my\_func)  
>> Keywords: UPPERCASE, words separated by "\_" (e.g. MY\_KEYWORD)  
>>  
>> Classes: MixedCase, no separation character (e.g. MyWonderfulClass)  
>> Methods: MixedCase, no separation character (e.g. GetProperty)  
>  
>  
> I find it useful to distinguish, in my code, names of routines that are  
> part of the IDL distribution from names of contributed routines. I do this  
> in the following way:  
>  
> PRINT, SIN(MyFunc(x))  
>  
> and apologize to all those who detest the all-uppercase way. What I  
> advocate is just only to make clear in \*some\* way that "sin" is part of  
> the IDL distribution, whereas "myfunc" isn't.  
>  
> This helps avoid confusion and makes it easier to track down problems  
> caused by collisions in that small world called namespace.

Makes perfect sense. One of the rules that I didn't elaborate on was that I prepend a short "namespace" to the front of my functions and procedures. If I have a project called "Mike's Cool Project," the typical routine name would look like something like mcp\_function or mcp\_procedure. Any objects related to the project have the same convention (e.g. MCPgrCoolTextThingy).

-Mike

---