

---

Subject: Re: Wrong documentation for BitGet()?

Posted by [wlandsman](#) on Tue, 03 May 2016 18:05:54 GMT

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

---

Yes, it looks like the documentation might be been written before the IDL code was written ;-)

The situation is even worse for the .bitset method. Again, the documentation says that the bit position is 1-based whereas it is actually 0-based. But the documentation is also incorrect about how a vector of positions is applied.

The example is

```
IDL> num = [0, 0, 0, 0]
IDL> PRINT, num.BitSet([1, 2, 3, 4])
1 2 4 8
```

whereas what IDL actually displays is  
30 30 30 30

The position vector applies to *every* element of the input array, setting them all to  $2^1 + 2^2 + 2^3 + 2^4 = 30$

--Wayne

On Tuesday, May 3, 2016 at 1:34:02 PM UTC-4, Paulo Penteado wrote:

```
> Hello,
>
> I was reading the help on the BitGet method,
>
> https://www.harrisgeospatial.com/docs/idl_integer.html#BitGe t
>
> And it seems there is a typo. It says
>
> Print the 3rd bit value of each element of an array:
> num = INDGEN(8)
> PRINT, num.BitGet( 3 )
> IDL prints:
> 0 0 0 0 1 1 1 1
>
> My experimenting indicates the argument of BitGet is the bit number, starting from zero, not
> from one as the help indicates. To get the 3rd bit, and obtain the result shown in the example, it
> seems it should be num.BitGet(2).
>
> Is this correct or am I missing something?
>
> Paulo
```

---