

## Re: Setting based 1 Arrays

---

*Source:*

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.vb/2009-05/msg00059.html>

---

- *From:* Mike <[unknown@xxxxxxxxxx](mailto:unknown@xxxxxxxxxx)>
  - *Date:* Sat, 02 May 2009 02:45:46 -0400
- 

Michael C wrote:

Ok, let me see mike. I start a copy of vb6 and type in this code

```
Private Sub Form_Load()  
Dim x(10)  
x(0) = 5  
End Sub
```

hmmmm, seems to me that that array is zero based. If the documentation says otherwise then it is wrong. Generally I don't have documentation from last century on my PC so no I have no read it. However, some other areas of vb6 are most definately 1 based. Which areas are 1 based and which are zero based appears to be random. You're not being very clear mike, perhaps you can cut and paste this mysterious piece of documentation you refer to. Does it say VB6 should be 1 based or zero based?

Michael, I can't confirm this under VB6 because my development machine desktop was corrupted early this year and I now notice I need to restore some lost VB6 dependencies. But in any case, I believe the issue is related to String Indexes more than arrays.

For example:

```
dim x as string  
x = "123456789"  
dim i as integer  
for i = 0 to Len(x)  
print " x(";i;) = "; x(i)  
next i
```

Expected result:

```
x(0) = 1  
x(1) = 1  
x(2) = 2  
x(3) = 3  
x(4) = 4
```

## Re: Setting based 1 Arrays

x(5) = 5  
x(6) = 6  
x(7) = 7  
x(8) = 8  
x(9) = 9

As a side note, I was never aware of Tom's input with

### OPTION BASE 1

I probably would not have used it in our WCVB.BAS header interface file because it has the explicit lower and upper range defined for the arrays field we have.

Type Tuser

```
...  
security(1...NUM_USER_SECURITY) * String * SIZE_SECURITY  
...  
End Type
```

But overall your point is proven – VB.NET is not VB6. The old rules do no apply anymore. :-)

---

.