

Current Object Value

D2000 VBApi - Work with the current object value

The interface [D2000 VBApi](#) allows you to get the current object value and modify it. Before getting the value or its modification, it is necessary to open the particular object ([VBApiOpenObject](#)).

Getting the value of an object is performed by calling the procedure **VBApiGetObjectValue**.

Declaration

```
Declare Sub VBApiGetObjectValue Lib "vbapi.dll" Alias  
"_GetObjectValue@12" (ByRef name As String, ByRef unival As VBApi_TUniVal,  
ByRef ok As Long)
```

Parameters

name	Name of open object.
unival	Structure that represents the current object value.
ok	Return value - call success.

Return value

Value of the parameter *ok*.

Return value	Description
0	Successful call. The parameter <i>unival</i> contains the current object value.
1	Incorrect object name (also in case, that the object is not opened).
2	No logged on user.
3	Unexpected internal error.
4	Incorrect parameters.

Setting the value of an object is performed alike by calling the procedure **VBApiChangeObjValue**. It is possible to set a value of the object of **User variable** type.

Declaration

```
Declare Sub VBApiChangeObjValue Lib "vbapi.dll" Alias "_ChangeObjValue@12"  
(ByRef name As String, ByRef unival As VBApi_TUniVal, ByRef ok As Long)
```

Parameters

name	Name of open object.
unival	Structure representing the current object value.
ok	Return value - call success.

Return value

Value of the parameter *ok*.

Return value	Description
0	Calling successful. The parameter <i>unival</i> contains the current object value.
1	Incorrect object name (also in case, that the object is not opened)
2	No logged on user.
3	Unexpected internal error.
4	Incorrect parameters.

Description

When setting the value, the value type is checked (*unival*) which must be equal with the current value type of the particular object ([user variable](#)).

Note

It is suitable to implement a modification of the current value, in regard to a lot of items in the structure [VB Api_TUniVal](#), in such a way that you should detect the current value at first (to fill the structure), carry out the required modifications and then call the procedure **VBApiChangeObjValue**.

Access to an item of an object of Value array type

To get the items' values of [Value array](#) type object use the function **VBApiGetArrObjectValue**. Compared to the function [VBApiGetObjectValue](#), there are two extra parameters - index of required value and the indicator of exceeded array size (index).

Declaration

```
Declare Sub VBApiGetArrObjectValue Lib "vbapi.dll" Alias
    "_GetArrObjectValue@20" (ByRef name As String, ByRef unival As
    VBApi_TUniVal, ByVal index As Long, ByRef bOutOfRange As Long, ByRef ok As
    Long)
```

Parameters

name	Name of open object.
unival	Structure representing the current object value.
index	Index of required array item.
bOutOfRange	Out-of-limit attribute.
ok	Return value - call success.

Return value

Value of the parameter *ok*.

Return value	Description
0	Calling successful. The parameter <i>unival</i> contains the current object value.
1	Incorrect object name (also in case, that the object is not opened)
2	No logged on user.
3	Unexpected internal error.
4	Incorrect parameters.

Note

If the parameter *name* is the name of an object, that is not Array type, and the parameter *index* = 1, then calling the function is performed successfully and returns the object value. The function **VBApiGetArrObjectValue** therefore can substitute the function [VBApiGetObjectValue](#).

Access to an item of an object of Structured variable type

To get the items' values of [Structured variable](#) object use the function **VBApiGetRecItemObjectValue**. Compared to the function [VBApiGetArrObjectValue](#), there is one extra parameter to enter the name of an item of the structure *itemName*.

Declaration

```
Declare Sub VBApiGetRecItemObjectValue Lib "vbapi.dll" Alias
    "_GetRecItemObjectValue@24" (ByRef name As String, ByRef unival As
    VBApi_TUniVal, ByVal index As Long, ByRef itemName As String, ByRef
    bOutOfRange As Long, ByRef ok As Long)
```

Parameters

name	Name of open object.
unival	Structure representing the current object value.
index	Index of required array item.
itemName	Name of structure item.
bOutOfRange	Out-of-limit attribute.
ok	Return value - call success.

Return value

Value of the parameter *ok*.

Return value	Description
0	Calling successful. The parameter <i>unival</i> contains the current object value.
1	Incorrect object name (also in case, that the object is not opened)
2	No logged on user.
3	Unexpected internal error.
4	Incorrect parameters.



Related pages:

[D2000 VBApi](#)