

[Home](#) x [Database Functions](#) x [Definitions](#) x [Error handling](#) x [Events handling](#) x [Methods](#) x  
[Find](#) x [GetChunk](#) x [Objects](#) x [Module](#) x [Properties](#) x [On \(control event\)](#) x  
[On \(database event\)](#) x [On \(dialog event\)](#) x [On \(form event\)](#) x [ProcBodyLine](#) x [ProcCountLines](#) x  
[ProcOfLine](#) x [ProcStartLine](#) x [Shortcuts](#) x [Snippets of Code](#) x [Module example](#) x

## Module example

Query the properties of a Basic module

tags:  
[Snippets of Code](#)

```
Const cstModule = "myModule"
Const cstProc = "mySub"
Const vbext_pk_Proc = 0 ' A Sub or Function procedure
Const cstStringToFind = "some string"

Dim oModule As Object, sProc As String, iProcType As Integer
Dim vStartLine As Variant, vStartColumn As Variant, vEndLine As Variant, vEndColumn As Variant

Set oModule = Application.AllModules(cstModule)
With oModule
    DebugPrint "Name = " & .Name
    DebugPrint "# of lines = " & .CountOfLines
    DebugPrint "# of declaration lines = " & .CountOfdeclarations
    DebugPrint "Lines 26 to 31 = " & .Lines(26, 6)
    DebugPrint "# of lines in proc " & cstProc & " = " & .ProcOfLines(cstProc, iProcType)
    DebugPrint "Start line in proc " & cstProc & " = " & .ProcStartLine(cstProc, iProcType)
    DebugPrint "Start body line in proc " & cstProc & " = " & .ProcBodyLine(cstProc, iProcType)
    ' Line 35 is located within procedure sProc (of type iProcType)
    sProc = .ProcOfLine(35, iProcType)
    ' Arguments are left uninitialized to consider the whole procedure
    If .Find(cstStringToFind, vStartLine, vStartColumn, vEndLine, vEndColumn)
        DebugPrint "Find: " & vStartLine & " " & vStartColumn & " " & vEndLine & " " & vEndColumn
    End If
End With
TraceConsole()
```

Bookmark this page » » [Module example](#)