

#1 S2 **'Abs Function Example**

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'ABS() - absolute value  
    dim a as single  
    dim b as integer  
  
    a = 2.34  
    b = abs(a)  
    msgbox str$(b)  
'end sub
```

¹ XCMD_Abs

² Abs Example

#3 \$4 **'AddIni Function Example {bmc no_dos.bmp}**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of AddIni

    'add the contents of myini.ini to win.ini
    a% = AddIni("myini.ini", "win.ini")
    if a% then
        msgbox "INI settings added to Win.INI"
    else
        msgbox "AddIni failed."
    end if
'end sub
```

³ XCMD_AddIni

⁴ AddIni Example

#5 S6 'And Operator Exanmple

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}
'sub main()
    'AND statement
    dim a as integer
    dim b as integer

    a = 5
    b = 9
    if (a < 6) AND (b > 8) then
        msgbox "Both conditions were true."
    else
        msgbox "Both conditions were not true."
    end if

    if (a < 6) AND (b > 9) then
        msgbox "Both conditions were true."
    else
        msgbox "Both conditions were not true."
    end if
'end sub
```

⁵ XCMD_And

⁶ And Example

#7 S8 'AnswerBox Function Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'AnswerBox example  
  
    'This is a default answer box without  
    'specifying the buttons  
    r% = AnswerBox("Example prompt?")  
    msgbox str$(r%)    'display the result  
    r% = AnswerBox("Example prompt 2?","&Maybe","&Ok","Maybe &Not")  
    msgbox str$(r%)    'display the result  
'end sub
```

⁷ XCMD_AnswerBox

⁸ AnswerBox Example

^{#9} ^{S10} **'AppActivate Statement Example** **{bmc no_dos.bmp}**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'AppActivate example  
  
    'activate Applications Manager  
    AppActivate "Applications Manager"  
'end sub
```

⁹ XCMD_AppActivate

¹⁰ AppActivate Example

#11 S12 **'AppClose Statement Example** {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'AppClose example  
  
    'search for a copy of Notepad that is running  
    appname$ = AppFind$("Notepad")  
  
    'close the copy of Notepad that was found  
    AppClose appname$  
'end sub
```

¹¹ XCMD_AppClose

¹² AppClose Example

#13 #14 **'AppFileName\$ Function Example** {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of AppFileName$

    'find a copy of Notepad that is running
    appname$ = AppFind$("Notepad")

    'get the name of the program it was executed from
    appfile$ = AppFileName$(appname$)

    'display the file name
    MsgBox appfile$
'end sub
```

¹³ XCMD_AppFileName_DOLLAR

¹⁴ AppFileName\$ Example

#15 S16 **'AppFind Function Example** {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of AppFind$  
  
    appname$ = AppFind$("Notepad")  
    MsgBox appname$  
'end sub
```

¹⁵ XCMD_AppFind

¹⁶ AppFind Example

#17 S18 'AppGetActive\$ Function Example {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of AppGetActive$  
  
    'activate a copy of Notepad  
    appname$ = AppFind$("Notepad")  
    AppActivate appname$  
  
    'now see if it is active  
    appname$ = AppGetActive$()  
    MsgBox appname$  
'end sub
```

¹⁷ XCMD_AppGetActive_DOLLAR

¹⁸ AppGetActive\$ Example

#19 S20 'AppGetPosition Statement Example {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of AppGetPosition
    dim x, y, w, h as integer

    'Get the position, width and height of Notepad
    appname$ = AppFind$("Notepad")
    AppGetPosition x, y, w, h, appname$

    'display them
    MsgBox appname$+ " is at "+str$(x)+ ", "+str$(y)+ " with width of "+str$(w)+ " and height of "+str$(h)
'end sub
```

¹⁹ XCMD_AppGetPosition

²⁰ AppGetPosition Example

#21 S22 'AppGetState Function Example {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of AppGetState

    appname$ = AppFind$("Notepad")
    appstate% = AppGetState(appname$)
    if appstate% = WS_MAXIMIZED then
        MsgBox appname$ + " is Maximized - " +str$(appstate%)
    else
        if appstate% = WS_MINIMIZED then
            MsgBox appname$ + " is Minimized - " + str$(appstate%)
        else
            if appstate% = WS_RESTORED then
                MsgBox appname$ + " is Restored - " + str$(appstate%)
            else
                MsgBox appname$ + " is in an unknown state - " + str$(appstate%)
            end if
        end if
    end if
end sub
```

²¹ XCMD_AppGetState

²² AppGetState Example

#23 S24 'AppHide Statement Example {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of AppHide and AppShow  
  
    appname$ = AppFind$("Notepad")  
    AppHide appname$  
    MsgBox appname$+ " is now hidden."  
    AppShow appname$  
    MsgBox appname$+ " is no longer hidden."  
'end sub
```

²³ XCMD_AppHide

²⁴ AppHide Example

#25 S26 **'AppList Statement Example** **{bmc no_dos.bmp}**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'example of AppList  
    dim appnames(1) as string  
  
    AppList appnames  
    for i% = lbound(appnames) to ubound(appnames)  
        MsgBox appnames(i%)  
    next i%  
'end sub
```

²⁵ XCMD_AppList

²⁶ AppList Example

#27 S28 **'AppMaximize Statement Example** {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'example of AppMaximize, AppRestore, and AppMinimize

    appname$ = AppFind$("Notepad")
    AppMaximize appname$
    msgbox appname$+ " is maximized."
    AppRestore appname$
    msgbox appname$+ " is restored."
    AppMinimize appname$
    msgbox appname$+ " is minimized."
'end sub
```

²⁷ XCMD_AppMaximize

²⁸ AppMaximize Example

#29 S30 **'AppMinimize Statement Example** {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'example of AppMaximize, AppRestore, and AppMinimize  
  
    appname$ = AppFind$("Notepad")  
    AppMaximize appname$  
    msgbox appname$+ " is maximized."  
    AppRestore appname$  
    msgbox appname$+ " is restored."  
    AppMinimize appname$  
    msgbox appname$+ " is minimized."  
'end sub
```

²⁹ XCMD_AppMinimize

³⁰ AppMinimize Example

#31 S32 **'AppMove Statement Example** **{bmc no_dos.bmp}**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'example of AppMove
    dim ax, ay as integer

    ' get current position of Notepad to save
    appname$ = AppFind$("Notepad")
    AppGetPosition ax, ay, 0, 0, appname$

    ' move Notepad around a little
    for i% = 0 to 200
        AppMove i%, i%, appname$
    next i%

    AppMove ax, ay, appname$
'end sub
```

³¹ XCMD_AppMove

³² AppMove Example

#33 #34 **'AppRestore Statement Example** {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'example of AppMaximize, AppRestore, and AppMinimize  
  
    appname$ = AppFind$("Notepad")  
    AppMaximize appname$  
    msgbox appname$+ " is maximized."  
    AppRestore appname$  
    msgbox appname$+ " is restored."  
    AppMinimize appname$  
    msgbox appname$+ " is minimized."  
'end sub
```

³³ XCMD_AppRestore

³⁴ AppRestore Example

#35 #36 **'AppSetState Statement Example** {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'example of AppSetState  
  
    appname$ = AppFind$("Notepad")  
    AppSetState WS_MAXIMIZED, appname$  
    msgbox "Notepad is maximized"  
    AppSetState WS_RESTORED, appname$  
    msgbox "Notepad is restored"  
    AppSetState WS_MINIMIZED, appname$  
    msgbox "Notepad is minimized"  
'end sub
```

³⁵ XCMD_AppSetState

³⁶ AppSetState Example

#37 S38 'AppShow Statement Example {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'example of AppHide and AppShow  
  
    appname$ = AppFind$("Notepad")  
    AppHide appname$  
    MsgBox appname$+ " is now hidden."  
    AppShow appname$  
    MsgBox appname$+ " is no longer hidden."  
'end sub
```

³⁷ XCMD_AppShow

³⁸ AppShow Example

#39 \$40 'AppSize Statement Example {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'example of AppSize
    dim ax, ay, aw, ah as integer

    appname$ = AppFind$("Notepad")
    AppSetState WS_RESTORED, appname$
    AppGetPosition ax, ay, aw, ah, appname$
    for i% = 10 to 200
        AppSize i%, i%, appname$
    next i%
    AppSize aw, ah, appname$
    AppMove ax, ay, appname$
    AppSetState WS_MINIMIZED, appname$
'end sub
```

³⁹ XCMD_AppSize

⁴⁰ AppSize Example

#41 S42 'AppType Function Example {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'example of AppType

    msgbox "After you press OK, you have 10 seconds      to activate an app to check."
    sleep 10000      'wait for 10 seconds
    appname$ = AppGetActive$()
    at% = AppType(appname$)
    select case at%
        case TYPE_DOS
            msgbox "DOS Application"
        case TYPE_WINDOWS
            msgbox "WINDOWS Application"
    end select
'end sub
```

⁴¹ XCMD_AppType

⁴² AppType Example

#43 S44 **'ArrayDims Function Example**

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'example of ArrayDims  
    dim apps(1) as string  
    dim oapps(1,1) as string  
  
    'ArrayDims shows how many dimensions an array  
    '   was declared with  
    msgbox "apps(1) has "+str$(arraydims(apps))+ " dimension(s)."  
    msgbox "oapps(1,1) has "+str$(arraydims(oapps))+ "dimension(s)."  
'end sub
```

⁴³ XCMD_ArrayDims

⁴⁴ ArrayDims Example

#45 #46 **'ArraySort Statement Example**

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'example of ArraySort  
    dim apps(1) as string  
  
    'get list of running applications  
    AppList apps  
    'sort it  
    ArraySort apps  
    'display them  
    for i% = lbound(apps) to ubound(apps)  
        msgbox apps(i%)  
    next i%  
'end sub
```

⁴⁵ XCMD_ArraySort

⁴⁶ ArraySort Example

#47 S48 'Asc Function Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'example of ASC()  
    dim acode as integer  
    dim astr as string  
  
    astr = "This is a string."  
    'the ASC function returns the ASCII code for  
    'the first character of the given string  
    acode = asc(astr)  
    msgbox "The first character of the string (" + astr + ") has an ASCII code of " + str$(acode)  
'end sub
```


#49 \$50 **'AskBox\$ Function Example**

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'example of AskBox$  
  
    userInput$ = AskBox$("Key in something below:", "This is the default value.")  
    msgbox "You entered -> "+userInput$  
'end sub
```

⁴⁹ XCMD_AskBox_DOLLAR

⁵⁰ AskBox\$ Example

^{#51} ^{\$52} **'AskPassword\$ Function Example**

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'example of AskPassword$  
  
    mypassword$ = AskPassword$("Type in a fake password:")  
    msgbox "The password you typed was -> "+mypassword$  
'end sub
```

⁵¹ XCMD_AskPassword_DOLLAR

⁵² AskPassword\$ Example

#53 #54 **'Atn Function Example**

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'example of ATN (arctangent)  
    dim mypi as double  
  
    mypi = 4 * atn(1)  
    msgbox str$(mypi)  
'end sub
```

⁵³ XCMD_Atn

⁵⁴ Atn Example

#55 #56 **'Beep Statement Example**

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'example of BEEP  
  
    beep  
'end sub
```

#57 #58 **'ButtonEnabled Function Example** {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of ButtonEnabled

    appn$ = AppFind$("Notepad")
    if appn$ <> "" then
        AppActivate appn$
        Menu "File.Page Setup"
        if ButtonEnabled("OK") then
            msgbox "OK button is enabled."
        else
            msgbox "OK button is not enabled."
        end if
    end if
end sub
```

⁵⁷ XCMD_ButtonEnabled

⁵⁸ ButtonEnabled Example

#59 \$60 'ButtonExists Function Example {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of ButtonExists

    appn$ = AppFind$("Notepad")
    if appn$ <> "" then
        AppActivate appn$
        Menu "File.Page Setup"
        if ButtonExists("Cancel") then
            msgbox "Cancel button exists."
        else
            msgbox "Cancel button does not exist."
        end if
    end if
end sub
```

⁵⁹ XCMD_ButtonExists

⁶⁰ ButtonExists Example

#61 S62 'Call Statement Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}
sub doubleit (a as integer)
    msgbox str$(a * 2)
end sub

sub main()
    'Example of the Call statement

    'the following statements do the same thing
    Call doubleit (3)
    doubleit(3)
end sub
```

#63 S64 'CDBl Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'example of CDBL
    dim adouble as double
    dim asingle as single
    dim ainteger as integer
    dim along as long

    asingle = pi
    adouble = pi
    ainteger = 30000
    along = 88000
    adouble = cdbl(ainteger)
    msgbox str$(adouble)
    adouble = cdbl(along)
    msgbox str$(adouble)
    msgbox str$(asingle)
    adouble = pi
    msgbox str$(adouble)
    'Now convert the single to double. You'll see a
    'change in the value due to conversion of a
    'smaller precision to double precision.
    adouble = cdbl(asingle)
    msgbox str$(adouble)
'end sub
```


#65 S66 'ChDir Statement Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'example of ChDir  
  
    msgbox CurDir$("C")  
    ChDir "C:\"  
    msgbox CurDir$("C")  
'end sub
```

#67 S68 'ChDrive Statement Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'example of ChDrive  
  
    msgbox CurDir$()  
    chdrive "D"  
    msgbox CurDir$()  
'end sub
```

⁶⁷ XCMD_ChDrive

⁶⁸ ChDrive Example

#69 S70 'Chr\$ Function Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'example of Chr$()  
  
    'List the letters of the alphabet  
    startcode = asc("A")  
    alphstr$ = ""  
    msgbox "The alphabet starts at code "+str$(startcode)  
    for i% = startcode to startcode +25  
        alphstr$ = alphstr$ +chr$(i%)  
    next i%  
    msgbox alphstr$  
'end sub
```

⁶⁹ XCMD_Chr_DOLLAR

⁷⁰ Chr\$ Example

#71 S72 'CInt Function Example

```
{bmc xc_copy2.bmp}  {bmc xc_copy.bmp}  {bmc xc_print.bmp}  {bmc xc_close.bmp}  
'sub main()  
    'Example of CINT  
    dim a as integer  
  
    a = cint(pi)    'convert PI to an integer  
    msgbox str$(a)  
'end sub
```

⁷¹ XCMD_CInt

⁷² CInt Example

#73 S74 'Clipboard\$ Statement and Function Example

```
{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}
'sub main()
    'Example of Clipboard$(), Clipboard$, and
    'ClipboardClear
    'WARNING: after you run this, your clipboard
    'contents will be gone

    msgbox Clipboard$()
    ClipboardClear
    msgbox Clipboard$()
    Clipboard$ "I was here!"
    msgbox Clipboard$()
    ClipboardClear
'end sub
```

⁷³ XCMD_Clipboard_DOLLAR

⁷⁴ Clipboard\$ Example

#75 #76 **'ClipboardClear Statement Example**

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of Clipboard$(), Clipboard$, and  
    'ClipboardClear  
    'WARNING: after you run this, your clipboard  
    'contents will be gone  
  
    msgbox Clipboard$()  
    ClipboardClear  
    msgbox Clipboard$()  
    Clipboard$ "I was here!"  
    msgbox Clipboard$()  
    ClipboardClear  
'end sub
```

⁷⁵ XCMD_ClipboardClear

⁷⁶ ClipboardClear Example

#77 S78 'CLng Function Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of CLNG  
    dim a as long  
  
    a = clng(pi)    'convert PI to a long integer  
    msgbox str$(a)  
'end sub
```

⁷⁷ XCMD_CLng

⁷⁸ CLng Example

#79 S80 'CSng Function Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of CSNG()  
  
    dim adouble as double  
    dim asingle as single  
  
    adouble =pi  
    msgbox str$(adouble)  
    asingle = csng(adouble)  
    msgbox str$(asingle)  
'end sub
```


#81 S82 'CStr Function Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of CSTR  
  
    dim adouble as double  
    dim astring as string  
  
    adouble = pi  
    astring = cstr(adouble)  
    msgbox astring  
'end sub
```

#83 S84 'CurDir\$ Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of CurDir$()  
    dim dstr as string  
  
    dstr = CurDir$()  
    msgbox dstr  
    dstr = CurDir$("F")    'for a particular drive  
    msgbox dstr  
'end sub
```

⁸³ XCMD_CurDir_DOLLAR

⁸⁴ CurDir\$ Example

#85 S86 'Date\$ Statement Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of the Date$ command (not function)  
    dim currentdate as string  
    dim newdate as string  
  
    currentdate = Date$  
    newdate = "1-1-1980"      'This is the earliest  
                              'date that can be  
                              'assigned  
  
    Date$ = newdate  
    msgbox Date$  
    Date$ = currentdate  
    msgbox Date$  
'end sub
```

^{#87} ^{S88} **'Date\$ Function Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of Date$()  
    dim dstr as string  
  
    dstr = Date$  
    msgbox dstr  
    dstr = Date$()  
    msgbox dstr  
'end sub
```

⁸⁷ XCMD_Date_DOLLAR_Function

⁸⁸ Date\$ Function Example

#89 S90 **'DateSerial Function Example**

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'example of DateSerial  
    dim dserial as double  
  
    dserial = DateSerial(1899,12,30) 'Earliest date should be 0  
    msgbox str$(dserial)  
    dserial = DateSerial(1999,12,30) '100 years later (note leap years included)  
    msgbox str$(dserial)  
'end sub
```

#91 S92 'DateValue Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of DateValue  
    dim dval as double  
  
    dval = DateValue(Date$)   'value of current date  
    msgbox str$(dval)  
'end sub
```

⁹¹ XCMD_DateValue

⁹² DateValue Example

#93 S94 'Day Function Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of Day() function (day of month)  
    'will show the current date's day of month.  
    msgbox str$(Day(DateValue(Date$)))  
'end sub
```

⁹³ XCMD_Day

⁹⁴ Day Example

#95 \$96 **'DCLHomeDir\$ Function Example**

```
{bmc xc_copy2.bmp}    {bmc xc_copy.bmp}    {bmc xc_print.bmp}    {bmc xc_close.bmp}  
'sub main()  
    'DCLHomeDir$ example  
  
    a$ = DCLHomeDir$  
    msgbox a$  
'end sub
```

⁹⁵ XCMD_DCLHomeDir_DOLLAR

⁹⁶ DCLHomeDir\$ Example

#97 S98 'DCLOS\$ Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of DCLOS() function

    a% = DCLOS()
    select case a%
        case 0
            msgbox "Windows"
        case 1
            msgbox "DOS"
    end select
'end sub
```

⁹⁷ XCMD_DCLOS_DOLLAR

⁹⁸ DCLOS\$ Example

#99 \$100 'DCLVersion\$ Function Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of DCLVersion$()  
  
    a$ = DCLVersion$()  
    msgbox a$  
'end sub
```

#101 S102 **'Declare Statement Example** {bmc no_dos.bmp}

```
{bmc xc_copy2.bmp}  {bmc xc_copy.bmp}  {bmc xc_print.bmp}  {bmc xc_close.bmp}
declare sub MessageBeep lib "user" (byval n as integer)
declare function GetDebugState lib "user" alias "GetSystemDebugState" () as long

'sub main()
    'Example of Declare statement
    'All declared external functions and procedures
    'exist outside of any DCL function or
    'subroutines.
    dim debugstate as long

    MessageBeep(-1)      'standard system beep
    debugstate = GetDebugState()
    msgbox str$(debugstate)
'end sub
```

#103 S104 'DEFtype Statement Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

DEFInt i - k, x - z

DEFStr s - v

DEFDb1 a - c

'sub main()

 'example of DEFtype statement

 'This statement is used to define certain

 'letters of the alphabetic character set to

 'be used as a certain type of variable.

 i = 1

 msgbox str\$(i)

 j = 2.5

 msgbox str\$(j)

 s = "test"

 msgbox s

 a = 3

 msgbox str\$(a)

 b = 5.6

 msgbox str\$(b)

'end sub

#105 \$106 **'DesktopCascade Statement Example {bmc no_dos.bmp}**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

'sub main()

 'Example of DesktopCascade

 DesktopCascade

'end sub

#107 S108 **'DesktopSetColors Statement Example** **{bmc no_dos.bmp}**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'example of DesktopSetColors  
  
    msgbox "Click to change to WingTips"  
    DesktopSetColors "WingTips"  
    msgbox "Click to change to Arizona"  
    DesktopSetColors "Arizona"  
'end sub
```

'DesktopSetWallpaper Statement Example**{bmc no_dos.bmp}**{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of DesktopSetWallpaper

    msgbox "Click to set wallpaper to CASTLE tiled."
    DesktopSetWallpaper "castle.bmp",true
    msgbox "Click to set wallpaper to CASTLE centered."
    DesktopSetWallpaper "castle.bmp",false
    msgbox "Click to clear wallpaper."
    DesktopSetWallpaper "",true
'end sub
```

#111 \$112 **'DesktopTile Statement Example {bmc no_dos.bmp}**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

'sub main()

 'Example of DesktopTile

 DesktopTile

'end sub

¹¹¹ XCMD_DesktopTile

¹¹² DesktopTile Example

#113 \$114 **'Dialog Statement and Function Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()

    'Example of using the DIALOG function and command

    'Here is our dialog box specification.
    Begin Dialog UserDialog 16,32,148,72, "Sample"
        OKButton 54,11,41,14
        CancelButton 55,40,41,14
    End Dialog

    'Create a dialog box type variable
    dim mydialog as UserDialog

    'First the function method
    retval% = Dialog(mydialog)
    select case retval%
        case -1
            msgbox "OK pressed."
        case 0
            msgbox "Cancel pressed."
    end select

    'Now the command method
    'First create an error handler in case "Cancel"
    'is pressed.
    on error goto cancelpressed
    Dialog mydialog
    msgbox "OK pressed."
    goto endprog
cancelpressed:
    msgbox "Cancel pressed."
endprog:
    on error goto 0
'end sub
```

#115 \$116 'Dim Statement Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Examples of the DIM statement

    'These two statements declare integers
    dim a as integer
    dim b%

    a = 1
    b% = 1

    'These two statements declare strings
    dim s as string
    dim t$

    s = "test"
    t$ = "test"

    'Array declarations
    'When an array is declared, its elements are
    'numbered 0 through N-1 where N is the size of
    'the array. Using the OPTION BASE statement,
    'you can begin numbering at 1.

    dim j(5) as integer      'array of 5 integer values
    for x% = 0 to 4
        j(x%) = x%
        msgbox str$(j(x%))
    next x%

    dim k(5) as string       'array of 5 strings
    for x% = 0 to 4
        k(x%) = chr$(65+x%)
        msgbox k(x%)
    next x%

    dim l(3,3) as string     'two-dimensional array
                                'containing a total of
                                '9 strings, 3 by 3

    for x% = 0 to 2
        for y% = 0 to 2
            l(x%,y%) = str$(x%)+str$(y%)
```

¹¹⁵ XCMD_Dim

¹¹⁶ Dim Example

```
        msgbox 1(x%,y%)
    next y%
next x%

'Using explicit array subscripts
dim p(6 to 10) as integer
for x% = 6 to 10
    p(x%) = x%
    msgbox str$(p(x%))
next x%
'end sub
```

#117 \$!18 **'DirExists Function Example** {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of DirExists  
  
    a$ = "C:\DOS"  
    if DirExists(a$) then  
        msgbox a$+ " exists"  
    else  
        msgbox a$+ " does not exist"  
    end if  
'end sub
```

#119 \$120 'Dir\$ Function Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of Dir$() function  
  
    a$ = dir$("c:\*.bat")    'initialize file search  
                             '(filespec parameter required)  
  
    do  
        msgbox a$           'display file name  
        a$ = dir$()         'next file name  
                             '(no parameter specified)  
  
    loop while a$ <> ""  
'end sub
```

#121 S122 **'DiskDrives Statement Example**

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of DiskDrives command  
    dim ddrives() as string  
  
    DiskDrives ddrives  
    for i% = lbound(ddrives) to ubound(ddrives)  
        alldrives$ = alldrives$ +ddrives(i%)  
    next i%  
    msgbox alldrives$  
'end sub
```

¹²¹ XCMD_DiskDrives

¹²² DiskDrives Example

#123 #124 **DiskFree Function Example**

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of DiskFree function  
    dim ddrives() as string  
    dim freespace as long  
  
    DiskDrives ddrives  
    for i% = lbound(ddrives) to ubound(ddrives)  
        if ddrives(i%) <> "A" and ddrives(i%) <> "B" then  
            freespace = DiskFree(ddrives(i%))  
            msgbox "Free space on drive "+ddrives(i%)+ ": is "+str$(freespace)  
        end if  
    next i%  
'end sub
```

¹²³ XCMD_DiskFree

¹²⁴ DiskFree Example

'Do...Loop Statement Example

```

{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}
'sub main()
    'Examples of DO-LOOP statements
    'All of the following loops perform the same task.
    'The difference is primarily in WHEN the loop exit
    'condition is checked.

    a% = 1
    do while a% = 1
        a% = msgbox("DO-WHILE-LOOP Click cancel to go to next loop.",1)
    loop

    a% = 1
    do until a% <> 1
        a% = msgbox("DO-UNTIL-LOOP Click cancel to go to next loop.",1)
    loop

    a%=1
    do
        a% = msgbox("DO-LOOP-WHILE Click cancel to go to next loop.",1)
    loop while a% = 1

    a% = 1
    do
        a% = msgbox("DO-LOOP-UNTIL Click cancel to go to next loop.",1)
    loop until a% <> 1

    a% = 1
    do
        a% = msgbox("DO-LOOP Click cancel to go to next loop.",1)
        if a% <> 1 then
            exit do
        end if
    loop
'end sub

```


#127 \$128

'DoEvents Statement Example

{bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Examples of DoEvents and DoEvents()  
  
    DoEvents  
    x% = DoEvents()  
'end sub
```

¹²⁷ XCMD_DoEvents

¹²⁸ DoEvents Example

'DoKeys Statement Example**{bmc no_dos.bmp}**{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of DoKeys  
    dim alttab as string  
  
    alttab = "%{TAB}"  
    msgbox "Press OK to do first Alt-Tab"  
    DoKeys alttab  
    msgbox "Press OK to Alt-Tab back to original application"  
    DoKeys alttab  
'end sub
```

#131 S132 **'EnableStopScript Statement Example** {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of EnableStopScript

    EnableStopScript ESS_ENABLE
        'control-break enabled
    EnableStopScript ESS_DISABLE
        'control-break disabled
    EnableStopScript ESS_INTERACTIVE
        'control-break only in script editor
'end sub
```

#133 \$134 **'End Statement Example**

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of END  
  
    msgbox "You'll see this one."  
    end  
    msgbox "You won't see this one."  
'end sub
```

#135 \$136

'Environ\$ Function Example

{bmc no_win.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of environ$() function  
  
    a$ = environ$("path")  
    msgbox a$  
'end sub
```

¹³⁵ XCMD_Environ_DOLLAR

¹³⁶ Environ\$ Example

#137 \$138 'Err Statement and Function Example

```
{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}
'sub main()
    'Example of Err(), Error, and Error$()
    dim i as integer

    i=1
nexterror:
    on error goto errortrap
    select case i
        case 1
            error 100
        case 2
            error 101
        case 3
            error 102
    end select
end

errortrap:
    msgbox "Error #"+str$(err())+ "; "+Error$(err())
    i = i +1
    goto nexterror
'end sub
```

#139 \$140 **'Error Statement and Function Example**

```
{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}
'sub main()
    'Example of Err(), Error, and Error$()
    dim i as integer

    i=1
nexterror:
    on error goto errortrap
    select case i
        case 1
            error 100
        case 2
            error 101
        case 3
            error 102
    end select
end

errortrap:
    msgbox "Error #"+str$(err())+ "; "+Error$(err())
    i = i +1
    goto nexterror
'end sub
```

#141 S142 **'Exclusive Statement Example {bmc no_dos.bmp}**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Examples of the Exclusive statement  
  
    Exclusive TRUE      'no multitasking for now  
    Exclusive FALSE    'allow other programs to run  
                        'again  
'end sub
```


#143 S144 **'Exp Function Example**

```
{bmc xc_copy2.bmp}    {bmc xc_copy.bmp}    {bmc xc_print.bmp}    {bmc xc_close.bmp}  
'sub main()  
    'Example of EXP function  
  
    dim result as double  
  
    result = exp(1)  
    msgbox str$(result)  
'end sub
```

#145 \$146

'FileCopy Function Example

{bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of FileCopy

    a% = FileCopy("c:\*.bat","c:\batch")
    if a% then
        msgbox "File copy successful."
    else
        msgbox "File copy failed."
    end if
'end sub
```

¹⁴⁵ XCMD_FileCopy

¹⁴⁶ FileCopy Example

#147 S148 'FileDateTime Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of FileDateTime  
    dim fdt as double  
    dim sfdt as string  
  
    fdt = FileDateTime("C:\AUTOEXEC.BAT")  
    sfdt = str$(month(fdt)) + "/" + str$(day(fdt)) + "/" + str$(year(fdt)) + " " + str$(hour(fdt)) +  
    ":" + str$(minute(fdt)) + ":" + str$(second(fdt))  
    msgbox sfdt  
'end sub
```

#149 \$150 **'FileDirs Statement Example**

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of FileDirs statement  
    dim fdirs() as string  
  
    'list all directories on drive C:  
    FileDirs fdirs,"C:\*.*"  
  
    for i% = lbound(fdirs) to ubound(fdirs)  
        fdirlist$ = fdirlist +fdirs(i%) + "; "  
    next i%  
    msgbox fdirlist$  
'end sub
```

#151 #152 **'FileExists Function Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of FileExists() function  
  
    if FileExists("C:\AUTOEXEC.BAT") then  
        msgbox "Autoexec Exists."  
    else  
        msgbox "Autoexec Does Not Exists."  
    end if  
    if FileExists("D:\NADA.FIL") then  
        msgbox "NADA.FIL Exists."  
    else  
        msgbox "NADA.FIL Does Not Exist."  
    end if  
'end sub
```

'FileLen Function Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of the FileLen function  
    dim flen as long  
  
    flen = FileLen("C:\AUTOEXEC.BAT")  
    msgbox "Autoexec.Bat is"+str$(flen)+ " bytes long."  
'end sub
```

#155 \$156 **'FileList Statement Example**

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of FileList statement  
    dim ffiles() as string  
    dim atrib as integer  
  
    atrib = ATTR_NORMAL  
    filelist ffiles,"C:\*.*",atrib  
    for i% = lbound(ffiles) to ubound(ffiles)  
        flist$ = flist$ +ffiles(i%) + "; "  
    next i%  
    msgbox flist$  
'end sub
```

FileParse Statement Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Examples of FileParse$()  
    dim filespec as string  
  
    filespec = "C:\DOS\COMMAND.COM"  
  
    'full file specification  
    a$ = FileParse$(filespec,0)  
    msgbox "Full Filespec = " + a$  
  
    'drive  
    a$ = FileParse$(filespec,1)  
    msgbox "Drive = " + a$  
  
    'path  
    a$ = FileParse$(filespec,2)  
    msgbox "Path = " + a$  
  
    'name  
    a$ = FileParse$(filespec,3)  
    msgbox "Filename = " + a$  
  
    'root  
    a$ = FileParse$(filespec,4)  
    msgbox "File Rootname = " + a$  
  
    'extension  
    a$ = FileParse$(filespec,5)  
    msgbox "File Extension = " + a$  
'end sub
```


#159 \$!60 'FileType Function Example

```
{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}
sub DisplayFileType (ftype as integer)
    select case ftype
        case TYPE_DOS
            msgbox "DOS"
        case TYPE_WINDOWS
            msgbox "WINDOWS"
        case else
            msgbox "Unknown Filetype"
    end select
end sub

sub main()
    'example of FileType function

    ftype% = FileType("C:\DOS\COMMAND.COM")
    DisplayFileType(ftype)
    ftype% = FileType("C:\DOS\DOSSHELL.EXE")
    DisplayFileType(ftype)
    ftype% = FileType("D:\WINDOWS\notepad.exe")
    DisplayFileType(ftype)
end sub
```

#161 \$162

'FindFile\$ Function Example

{bmc no_dos.bmp}

{bmc xc_copy2.bmp}

{bmc xc_copy.bmp}

{bmc xc_print.bmp}

{bmc xc_close.bmp}

```
'sub main()
    'Example of FindFile$

    fp$ = FindFile$("notepad.exe")
    if fp$ <> "" then
        msgbox "File found as "+fp$
    else
        msgbox "File not found."
    end if
'end sub
```

¹⁶¹ XCMD_FindFile_DOLLAR

¹⁶² FindFile\$ Example

#163 S164 **'Fix Function Example**

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of the Fix() function  
    dim adouble as double  
    dim aint as integer  
  
    adouble = pi  
    msgbox str$(adouble)  
    aint = fix(adouble)  
    msgbox str$(aint)  
'end sub
```

#165 \$166 **'For...Next Statement Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()

    'Examples of FOR-NEXT loops
    dim x as integer      'used as iteration variable
                           'for FOR-NEXT

    dim xstart as integer
    dim xend as integer

    'simple form
    for x = 1 to 5
        msgbox str$(x)
    next x

    'step form
    for x = 1 to 10 step 2
        msgbox str$(x)
    next x

    'backward form
    for x = 10 to 1 step -2
        msgbox str$(x)
    next x

    'change of indexes
    for x = 69 to 74
        msgbox chr$(x)
    next x

    'variable range
    xstart = 69
    xend = 74
    for x = xstart to xend
        msgbox "VAR - "+chr$(x)
    next x

    'premature exit
    for x = 1 to 10
        if x = 6 then
            exit for
        end if
        msgbox "Exit - "+str$(x)
    next x
```

¹⁶⁵ XCMD_For_Next

¹⁶⁶ For...Next Example

'end sub

#167 \$168

'FreeFile Function Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of FreeFile%()  
    dim nextfile as integer  
  
    nextfile = FreeFile()  
    msgbox str$(nextfile)  
'end sub
```

¹⁶⁷ XCMD_FreeFile

¹⁶⁸ FreeFile Example

#169 S170 **'GetAttr Function Example**

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of GetAttr  
  
    i% = GetAttr("C:\IO.SYS")  
    msgbox str$(i%)  
'end sub
```

#171 S172 **'GetEnv Function Example {bmc no_dos.bmp}**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of SetEnv and GetEnv$()
    tmp$ = GetEnv$("TMP",ENV_WINDOWS)
    tv$ = AskBox$("New Value For TMP Environment Variable:")
    a% = SetEnv("TMP",tv$,ENV_WINDOWS)
    msgbox "New value for TMP is "+ GetEnv$("TMP",ENV_WINDOWS)
    'restore old value
    a% = SetEnv("TMP",tmp$,ENV_WINDOWS)
'end sub
```


#173 S174 **'GoSub Statement Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Examples of GOSUB and RETURN  
  
    for x% = 1 to 5  
        gosub mylabel  
    next x%  
end  
  
mylabel:  
    msgbox "Here we are!"  
    return  
'end sub
```

#175 \$176 **'Goto Statement Example**

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of GOTO  
  
    for x% = 1 to 5  
        if x% = 3 then goto enditall  
    next x%  
    msgbox "Error"  
    end  
  
enditall:  
    msgbox "Ended properly"  
    end  
'end sub
```

#177 \$178 'Hex\$ Function Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of Hex$() function  
  
    i% = 31  
    h$ = hex$(i%)  
    msgbox h$  
'end sub
```

#179 S180 **'HLine Statement Example {bmc no_dos.bmp}**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Examples of HLINE  
  
    ViewPortOpen  
    ViewPortClear  
    Print "Here is some test data."  
    HLine 50  
    sleep 2000  
    HLine -50  
    ViewPortClose  
'end sub
```

#181 #182 'Hour Function Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of hour() function  
    dim dt as double  
  
    dt = Now  
    msgbox str$(hour(dt)) 'current hour  
'end sub
```

#183 S184 **'HPage Statement Example {bmc no_dos.bmp}**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Examples of HPage  
  
    ViewPortOpen  
    ViewPortClear  
    Print "This is some test data"  
    HPage 1  
    sleep 2000  
    HPage -1  
    ViewPortClose  
'end sub
```

#185 \$186 **'HScroll Statement Example {bmc no_dos.bmp}**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of HSCROLL

    ViewPortOpen
    ViewPortClear
    Print "This is some test data for the viewport scroll test."
    sleep 2000
    HScroll 50 '50 percent scroll
    sleep 2000
    HScroll 0      'no scroll
    sleep 2000
    ViewPortClose
'end sub
```

#187 \$188 **'If...Then...Else Statement Example**

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example IF-THEN-ELSE-END IF statement  
    dim a%  
  
    if a% = 1 then  
        'do this stuff if a is 1  
    elseif a% = 2 then  
        'otherwise if a is 2 then do this stuff  
    else  
        'if a is neither 1 nor 2 then do this stuff  
    end if  
'end sub
```


#189 \$190 **'InputBox\$ Function Example**

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of InputBox$() function  
  
    a$ = InputBox$("Enter your description:",    "Description","",100,200)  
    msgbox a$  
'end sub
```

#191 \$192 **InStr Function Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of InStr() function
    dim teststring as string

    teststring = "The quick brown fox jumps over the lazy dog."
    'search starting at position 1
    i% = InStr(1,teststring,"quick")
    if i% = 0 then
        msgbox "'quick' was not found"
    else
        msgbox "'quick' was found at position"+str$(i%)
    end if
    'search starting at position 10
    i% = InStr(10,teststring,"quick")
    if i% = 0 then
        msgbox "'quick' was not found"
    else
        msgbox "'quick' was found at position"+str$(i%)
    end if
'end sub
```

#193 \$194

'Int Function Example

```
{bmc xc_copy2.bmp}    {bmc xc_copy.bmp}    {bmc xc_print.bmp}    {bmc xc_close.bmp}  
'sub main()  
    'example of INT function  
    dim adouble as double  
  
    adouble = pi  
    msgbox str$(adouble)  
    msgbox str$(int(adouble))  
'end sub
```

¹⁹³ XCMD_Int

¹⁹⁴ Int Example

#195 \$196

'Item\$ and ItemCount Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of Item$() and ItemCount  
    dim pathstr as string  
  
    pathstr = environ$("PATH") 'get the path  
    msgbox "There are"+str$(itemcount(pathstr,";"))+" items in the path."  
    msgbox pathstr  
    msgbox item$(pathstr,3,4,";")  
'end sub
```

195 XCMD_Item_DOLLAR

196 Item\$ Example

#197 \$198

'Kill Statement Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of KILL command  
  
    kill "c:\junk.txt"  
'end sub
```

¹⁹⁷ XCMD_Kill

¹⁹⁸ Kill Example

#199 \$200 **'LBound Function Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}
'sub main()
 'Examples of LBOUND
 dim ia1(8) as integer
 dim ia2(65 to 70) as integer

 msgbox str\$(lbound(ia1))
 msgbox str\$(lbound(ia2))
'end sub

#201 \$202 **'LCase\$ Function Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of LCase$  
    dim teststr as string  
  
    teststr = "THIS IS A TEST"  
    msgbox teststr  
    teststr = lcase$(teststr)  
    msgbox teststr  
'end sub
```

'Left\$ Function Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of left$()  
    dim teststr as string  
  
    teststr = "This is a test."  
    msgbox teststr  
    teststr = left$(teststr,7)  
    msgbox teststr  
'end sub
```


'Len Function Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of Len()  
    dim teststr as string  
  
    teststr = "This is a test."  
    msgbox "'" + teststr + "' is " + str$(len(teststr)) + " characters long."  
'end sub
```

'Let Statement Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
```

```
    'Examples of Let statement
```

```
        let a% = 1
```

```
        let s$ = "test"
```

```
'end sub
```

#209 \$210 **'Line\$ Function Example**

```
{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}
'sub main()
    'Example of Line$()
    dim crlf as string
    dim testlines as string

    crlf = chr$(13) +chr$(10)
    'carriage return followed by a linefeed
    testlines = "line 1" +crlf
    testlines = testlines + "line 2" +crlf
    testlines = testlines + "line 3" +crlf
    testlines = testlines + "line 4"
    msgbox testlines
    msgbox line$(testlines,2,3)
'end sub
```

#211 \$212 'LineCount Function Example

```
{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}
'sub main()
    'Example of LineCount
    dim crlf as string
    dim testlines as string

    crlf = chr$(13) + chr$(10)
    'carriage return followed by a linefeed
    testlines = "line 1" + crlf
    testlines = testlines + "line 2" +crlf
    testlines = testlines + "line 3" +crlf
    testlines = testlines + "line 4"
    msgbox testlines
    msgbox str$(linecount(testlines))+ " lines total."
'end sub
```

#213 \$214 **'Log Function Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of log() function  
  
    msgbox "Log of 1 is "+str$(log(1))  
    msgbox "Log of 2 is "+str$(log(2))  
'end sub
```

#215 \$216 **'LTrim\$ Function Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}
'sub main()
 'Example of LTrim\$()
 dim teststring as string

 teststring = " testing "
 msgbox "*" +teststring + "*"
 msgbox "*" +ltrim\$(teststring) + "*"
'end sub

#217 S218 **'MCI Function Example {bmc no_dos.bmp}**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example MCI command  
    'NOTE: This program may not run on your machine.  
  
    dim resultstr as string  
    dim errorstr as string  
  
    MCI("play cdrom from 1 to 20",resultstr,errorstr)  
'end sub
```

#219 \$220 **'Menu Statement Example {bmc no_dos.bmp}**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of Menu command  
  
    aname$ = AppFind$("Notepad")  
    AppActivate aname$  
    Menu "File.Page Setup"  
'end sub
```


#221 \$222 **'Mid\$ Function Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of mid$() function and command
    dim teststr as string

    teststr = "This is a test."
    msgbox teststr
    msgbox mid$(teststr,3,5)      'starting at 3 and
                                'retrieving 5 characters

    mid$(teststr,3,5) = "      "
    msgbox teststr
    msgbox mid$(teststr,3,5)      'starting at 3 and
                                'retrieving 5 characters
'end sub
```

'Minute Function Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of minute() function  
    dim dt as double  
  
    dt = Now  
    msgbox str$(minute(dt))    'current minute  
'end sub
```

#225 \$226 **'MkDir Statement Example**

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of mkdir  
  
    MkDir "C:\testdir"  
'end sub
```

'Mod Operator Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of MOD  
  
    msgbox str$(5 mod 3)        'should display 2--the  
                               'remainder after division  
  
'end sub
```

#229 \$230 **'Month Function Example**

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of Month  
  
    msgbox str$(month(Now))    'display current month  
'end sub
```

'MsgBox Statement and Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Examples of MsgBox function and statement

    for i% = 0 to 5
        MsgBox "message", i%, "title"
    next i%
    for i% = 0 to 5
        result = MsgBox("message", i%, "title")
        select case result
            case 1
                msgbox "OK was pressed"
            case 2
                msgbox "Cancel was pressed"
            case 3
                msgbox "Abort was pressed"
            case 4
                msgbox "Retry was pressed"
            case 5
                msgbox "Ignore was pressed"
            case 6
                msgbox "Yes was pressed"
            case 7
                msgbox "No was pressed"
        end select
    next i%

    'In this dialog box the second button is the default.
    result=msgbox("Hello World",3+256,"title")

    'In this dialog box the Stop symbol is displayed.
    result=msgbox("Hello World",3+256+16,"title")

'end sub
```

#233 \$234 **'Name Statement Example**

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of Name statement  
  
    Name "C:\JUNK.TXT" as "C:\NEWJUNK.TXT"  
        'renames JUNK.TXT to NEWJUNK.TXT  
'end sub
```

#235 \$236

'NetAttach Function Example

{bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of NetAttach

    s$ = AskBox$("Server to attach to:")
    u$ = AskBox$("Username:")
    p$ = AskPassword$("Password:")
    if NetAttach(s$,u$,p$) then
        msgbox("Attach Successful.")
    else
        msgbox("Attach Failed.")
    end if
'end sub
```

235 XCMD_NetAttach

236 NetAttach Example

'NetConnectDrive Function Example**{bmc no_dos.bmp}**{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of NetConnectDrive

    lp$ = AskBox$("Local drive to connect to:")
    np$ = AskBox$("Network path to connect:")
    if NetConnectDrive(lp$,np$) then
        msgbox "Drive connected."
    else
        msgbox "Drive connection failed."
    end if
'end sub
```

'NetDetach Function Example**{bmc no_dos.bmp}**{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of NetDetach

    s$ = AskBox$("Server To Detach:")
    if NetDetach(s$) then
        msgbox "Server Detached."
    else
        msgbox "Error Detaching Server."
    end if
end sub
```

'NetDirectoryRights Function Example**{bmc no_dos.bmp}**{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of NetDirectoryRights

    p$ = AskBox$("Path to get rights for:")
    r$ = AskBox$("Rights to search for:")
    if NetDirectoryRights(p$,r$) then
        msgbox "You have "+r$+ " rights for "+p$
    else
        msgbox "You do not have "+r$+ " rights for "+p$
    end if
'end sub
```

#243 S244 'NetDisconnectDrive Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of NetDisconnectDrive() function
    dim drive as string
    dim netpath as string

    drive = "k"
    netpath = "sweng_1/sys2:"

    if NetConnectDrive(drive,netpath) then
        msgbox "Drive "+drive+ " has been connected."
        if NetDisconnectDrive(drive) then
            msgbox "Drive "+drive+ " has been disconnected."
        else
            msgbox "Disconnect Failed."
        end if
    else
        msgbox "Connection Failed."
    end if
'end sub
```

'NetGetDirectoryRights Function Example**{bmc no_dos.bmp}**{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of NetGetDirectoryRights()  
  
    p$ = AskBox$("Path to get rights list for:")  
    r$ = NetGetDirectoryRights$(p$)  
    msgbox "Your rights for "+p$+ " are "+r$  
'end sub
```

#247 S248 **'NetMemberOf Function Example** {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of NetMemberOf

    g$ = AskBox$("Group To Search For:")
    s$ = AskBox$("Server To Search:")
    if NetMemberOf(g$,s$) then
        msgbox "You are a member of that group."
    else
        msgbox "You are not a member of that group."
    end if
'end sub
```

#249 \$250 **'NetStationID Function Example** {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of NetStationID$()  
  
    nsi$ = NetStationID$()  
    msgbox "Your ethernet address is: "+nsi$  
'end sub
```

#251 \$252 **'NetUserName Function Example {bmc no_dos.bmp}**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of NetUserName  
  
    s$ = AskBox$("Server:")  
    nul$ = NetUserName$()  
    nu2$ = NetUserName$(s$)  
    msgbox "You are "+nul$+ " on your primary server, and "+nu2$+ " on "+s$+ "."  
'end sub
```


'NetworkStatus Function Example**{bmc no_dos.bmp}**{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of NetworkStatus function

    ns% = NetworkStatus()
    msgbox str$(ns%)
    if ns% AND NS_ACTIVE then
        msgbox "Network is active."
    else
        msgbox "Network is not active."
    end if
    if ns% AND NS_LOGGEDON then
        msgbox "Logged On."
    else
        msgbox "Not Logged On."
    end if
'end sub
```

'Not Operator Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of NOT operator  
    dim testvar as integer  
  
    testvar = TRUE  
    if testvar then  
        msgbox "TestVar is TRUE"  
    end if  
  
    testvar = FALSE  
    if NOT testvar then  
        msgbox "TestVar is NOT TRUE"  
    end if  
'end sub
```

#257 \$258

'Now Function Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of the Now function  
    dim cdt as double  
  
    cdt = Now  
    msgbox "Current date is "+str$(month(cdt))+ "/" +str$(day(cdt))+ "/" +str$(year(cdt))  
'end sub
```

#259 \$260 **'Null Function Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}
'sub main()
 'Example of the Null function
 dim teststr as string

 teststr = "testing"
 msgbox "*" + teststr + "*"
 teststr = null
 msgbox "*" + teststr + "*"
'end sub

#261 \$262

'Oct\$ Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
```

```
    'Example of Oct$() function
```

```
        msgbox Oct$(9)
```

```
'end sub
```

261 XCMD_Oct_DOLLAR

262 Oct\$ Example

#263 S264 'On Error Statement Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of ON-ERROR statement  
  
    on error goto elabel      'enable error trap  
    error 101                 'simulate an error  
    end  
  
elabel:                      'error trap label  
    on error goto 0           'disable error trapping  
    end  
'end sub
```

#265 \$266 **'OpenFileName\$ Function Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}
'sub main()
 'Example of OpenFileName\$

 selfile\$ = OpenFileName\$("Open File", "All Files:*.bmp, *.wmf;Bitmaps:*.bmp;Metafiles:*.wmf")
 msgbox "Selected File = "+selfile\$
'end sub

#267 \$268 **'Option Base Statement Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}
Option Base 1

```
'sub main()  
    'Example of Option Base statement  
    dim a(5) as integer  
  
    msgbox str$(lbound(a))  
'end sub
```


#269 \$270 'Or Operator Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'OR statement  
    dim a as integer  
    dim b as integer  
  
    a = 5  
    b = 9  
    if (a < 6) OR (b > 8) then  
        msgbox "One of the conditions was true."  
    else  
        msgbox "Neither condition was true."  
    end if  
  
    if (a < 6) OR (b > 9) then  
        msgbox "One of the conditions was true."  
    else  
        msgbox "Neither condition was true."  
    end if  
'end sub
```

#271 \$272

'PI Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of the PI function  
    dim adouble as double  
  
    adouble = pi  
    msgbox str$(adouble)  
'end sub
```

²⁷¹ XCMD_PI

²⁷² PI Example

'PopupMenu Function Example**{bmc no_dos.bmp}**{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of PopupMenu
    dim apps$() as string
    dim result as integer

    AppList apps$
    result = PopupMenu(apps$)
    if result >= lbound(apps$) then
        msgbox "You chose "+apps$(result)
    else
        msgbox "You chose nothing -"+str$(result)
    end if
end sub
```

#275 \$276 **'Print Statement Example**

```
{bmc xc_copy2.bmp}  {bmc xc_copy.bmp}  {bmc xc_print.bmp}  {bmc xc_close.bmp}  
'sub main()  
    'Example of Print  
  
    ViewPortOpen  
    ViewPortClear  
    print "this is some data"  
    sleep 5000  
    ViewPortClose  
'end sub
```

'Print # Statement Example{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
  open "test.dat" for output as #1  
  print #1,10,34,"Hello World";  
  a = 10  
  s$ = "this is a test"  
  print #1,a;s$,  
  print #1,67  
  close #1  
'end sub
```

'PrinterGetOrientation Function Example**{bmc no_dos.bmp}**{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of PrinterGetOrientation and
    'PrinterSetOrientation
    dim oldorient as integer

    oldorient = PrinterGetOrientation()
    select case oldorient
        case PO_PORTRAIT
            msgbox "Printer is set up for portrait print."
        case PO_LANDSCAPE
            msgbox "Printer is set up for landscape print."
    end select
    PrinterSetOrientation oldorient
'end sub
```

'PrinterSetOrientation Statement Example**{bmc no_dos.bmp}**{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of PrinterGetOrientation and
    'PrinterSetOrientation
    dim oldorient as integer

    oldorient = PrinterGetOrientation()
    select case oldorient
        case PO_PORTRAIT
            msgbox "Printer is set up for portrait print."
        case PO_LANDSCAPE
            msgbox "Printer is set up for landscape print."
    end select
    PrinterSetOrientation oldorient
'end sub
```

'PrintFile Function Example{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of PrintFile  
  
    taskid = printfile("c:\testfile.txt")  
    msgbox "Your file is printing as task #"+str$(taskid)  
'end sub
```


#285 \$286

'Random Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of Random function  
  
    ViewPortOpen  
    ViewPortClear  
    for j% = 1 to 10  
        i% = Random(1,100)  
        print i%  
    next j%  
    sleep 5000  
    ViewPortClose  
'end sub
```

#287 \$288

'Randomize Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of Randomize statement  
  
    Randomize 65  
    Randomize  
'end sub
```

#289 \$290 **'ReadINI\$ Function Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of ReadIni$  
  
    winshell$ = ReadIni$("boot","shell","system.ini")  
    msgbox winshell$  
'end sub
```

#291 \$292 **'ReadINISection Statement Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of ReadIniSection  
    dim iniitems() as string  
  
    ReadIniSection "boot",iniitems,"system.ini"  
    ViewPortOpen  
    print "[Boot] section items from SYSTEM.INI"  
    print " "  
    for i% = lbound(iniitems) to ubound(iniitems)  
        print iniitems(i%)  
    next i%  
    sleep 5000  
    ViewPortClose  
'end sub
```

'ReDim Statement Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of ReDim  
    dim stuff(5) as integer  
  
    msgbox str$(ubound(stuff))  
    redim stuff(10)  
    msgbox str$(ubound(stuff))  
'end sub
```

#295 \$296

'REM Statement Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

'sub main()

 'Example of REM

REM this is also a comment

'end sub

295 XCMD_REM

296 REM Example

#297 \$298

'RefreshIni Statement Example {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
```

```
    'Example of RefreshIni
```

```
        RefreshIni
```

```
'end sub
```

²⁹⁷ XCMD_RefreshIni

²⁹⁸ RefreshIni Example

#299 S300 'Reset Statement Example

```
{bmc xc_copy2.bmp}  {bmc xc_copy.bmp}  {bmc xc_print.bmp}  {bmc xc_close.bmp}  
'sub main()  
    'Example of RESET statement  
  
    Reset  
        'close all open files (writes out i/o buffers)  
'end sub
```


#301 S302

'RestoreEnv Function Example

{bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of SaveEnv and RestoreEnv  
  
    a% = SaveEnv(ENV_WINDOWS)  
    a% = RestoreEnv(ENV_WINDOWS)  
'end sub
```

³⁰¹ XCMD_RestoreEnv

³⁰² RestoreEnv Example

'Resume Statement Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of Resume statement  
  
    on error goto testerror  
    error 101  
    msgbox "resumed as anticipated"  
    on error goto 0  
    end  
  
testerror:  
    resume next  
'end sub
```

#305 #306

'Return Statement Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Examples of GOSUB and RETURN  
  
    for x% = 1 to 5  
        gosub mylabel  
    next x%  
end  
  
mylabel:  
    msgbox "Here we are!"  
    return  
'end sub
```

#307 \$308

'Right\$ Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of right$()  
    dim teststr as string  
  
    teststr = "This is a test."  
    msgbox teststr  
    teststr = right$(teststr,7)  
    msgbox teststr  
'end sub
```

³⁰⁷ XCMD_Right_DOLLAR

³⁰⁸ Right\$ Example

#309 S310 **'Rmdir Statement Example**

```
{bmc xc_copy2.bmp}    {bmc xc_copy.bmp}    {bmc xc_print.bmp}    {bmc xc_close.bmp}  
'sub main()  
    'Example of Rmdir  
  
    Rmdir "C:\testdir"  
'end sub
```

#311 \$312 **'Rnd Function Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}
'sub main()
 'Example of Rnd function

 ViewPortOpen
 Randomize
 for i% = 1 to 10
 print Rnd(1)
 next i%
 sleep 5000
 ViewPortClose
'end sub

#313 S314 **'RTrim\$ Function Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}
'sub main()
 'Example of RTrim\$()
 dim teststring as string

 teststring = " testing "
 msgbox "*" +teststring + "*"
 msgbox "*" +rtrim\$(teststring) + "*"
'end sub

#315 \$316

'SaveEnv Function Example

{bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

'sub main()

 'Example of SaveEnv and RestoreEnv

 a% = SaveEnv(ENV_WINDOWS)

 a% = RestoreEnv(ENV_WINDOWS)

'end sub

³¹⁵ XCMD_SaveEnv

³¹⁶ SaveEnv Example

#317 #318 **'SaveFileName\$ Function Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of SaveFileName$  
  
    selfile$ = SaveFileName$("Open File","All Files:*.bmp, *.wmf;Bitmaps:*.bmp;Metafiles:*.wmf")  
    msgbox "Selected File = "+selfile$  
'end sub
```

'Second Function Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of second() function  
    dim dt as double  
  
    dt = Now  
    msgbox str$(second(dt))....'current second  
'end sub
```

#321 #322 **'Select...Case Statement Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of SELECT-CASE statement  
  
    i% = 1  
    select case i%  
        case 1  
            msgbox "i% is 1"  
        case 2  
            msgbox "i% is 2"  
    end select  
'end sub
```

'SelectBox Function Example {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'example of SelectBox function  
    dim alist() as string  
  
    AppList alist  
    result% = SelectBox("Application List", "Pick One", alist)  
    msgbox "You selected "+alist(result%)  
'end sub
```

'SendKeys Statement Example {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of SendKeys  
    dim alttab as string  
  
    alttab = "%{TAB}"  
    msgbox "Press OK to do first Alt-Tab"  
    SendKeys alttab,TRUE  
    msgbox "Press OK to Alt-Tab back to original application"  
    SendKeys alttab,FALSE  
'end sub
```

#327 S328

'SetAttr Statement Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
```

```
    'Example of SetAttr
```

```
    SetAttr "C:\AUTOEXEC.BAT",0
```

```
    'set file attribute to NORMAL
```

```
'end sub
```

³²⁷ XCMD_SetAttr

³²⁸ SetAttr Example

'SetEnv Function Example**{bmc no_dos.bmp}**{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of SetEnv and GetEnv$()
    tmp$ = GetEnv$("TMP",ENV_WINDOWS)
    tv$ = AskBox$("New Value For TMP Environment Variable:")
    a% = SetEnv("TMP",tv$,ENV_WINDOWS)
    msgbox "New value for TMP is "+ GetEnv$("TMP",ENV_WINDOWS)
    'restore old value
    a% = SetEnv("TMP",tmp$,ENV_WINDOWS)
'end sub
```

#331 S332

'Sgn Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of SGN() function  
  
    i% = -2  
    msgbox str$(sgn(i%))  
    i% = 0  
    msgbox str$(sgn(i%))  
    i% = 2  
    msgbox str$(sgn(i%))  
'end sub
```

³³¹ XCMD_Sgn

³³² Sgn Example

#333 S334 **'Shell Function Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}
'sub main()
 'Example of Shell function

 taskid% = Shell("notepad.exe",1)
'end sub

#335 \$336

'Sin Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of Sin() function  
  
    result# = sin(0)  
    msgbox str$(result#)  
    result# = sin(1)  
    msgbox str$(result#)  
'end sub
```

#337 S338

'Sleep Statement Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

'sub main()

 'Example of Sleep command

 msgbox "Press OK to sleep 5 seconds"

 sleep 5000

'end sub

³³⁷ XCMD_Sleep

³³⁸ Sleep Example

'SleepUntil Function Example**{bmc no_dos.bmp}**{bmc xc_copy2.bmp}{bmc xc_copy.bmp}{bmc xc_print.bmp}{bmc xc_close.bmp}

```

'sub main()
    'Example of SleepUntil

    t$ = time$()      'get current time
    t$ = left$(t$,5)   'get hours and
                        'minutes only
    m$ = right$(t$,2) 'get minutes value
    h$ = left$(t$,2)   'get hour value
    x = val(m$)        'get the value of the minutes
    y = val(h$)        'get value of hour string
    x = x + 2          'increment minutes
                        'to wait 2 minutes

    if x > 59 then
        y = y + 1
        x = x - 59
    end if
    h$ = str$(y)       'convert hours
                        'back to string
    h$ = right$(h$,len(h$)-1)
        'get rid of leading blank from conversion
    if len(h$) = 1 then h$ = "0" + h$
    m$ = str$(x)       'convert minutes
                        'back to string likewise
    m$ = right$(m$,len(m$)-1)
    if len(m$) = 1 then m$ = "0" + m$
    t$ = h$+ ":"+m$    'reconstruct time
                        'value

    retval% = SleepUntil(t$,TRUE, "Press CANCEL to quit.(","Sleeping Until "+t$,TRUE,TRUE)
'end sub

```

#341 S342

'Snapshot Statement Example {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of SnapShot statement

    SnapShot 4 'active window
    taskid% = Shell("pbrush.exe")
    DoKeys "+{INSERT}" 'paste clipboard
                                'into paintbrush
'end sub
```

³⁴¹ XCMD_Snapshot

³⁴² Snapshot Example

#343 #344 **'Space\$ Function Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}
'sub main()
 'Example of Space\$() function

 stuff\$ = null
 msgbox "*" + stuff\$ + "
 stuff\$ = space\$(10)
 msgbox "*" + stuff\$ + "
'end sub

#345 #346 **'Sqr Function Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}
'sub main()
 'Example of SQR function

 msgbox str\$(sqr(100))
'end sub

#347 S348

'Stop Statement Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of Stop statement  
  
    Stop  
'end sub
```


'StrComp Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of StrComp function

    'Case-sensitive comparison
    teststr$ = "This is a test string"
    result% = StrComp(teststr$, "THIS IS A TEST STRING", 0)

    select case result%
        case -1
            msgbox "teststr$ is less than the compare string"
        case 0
            msgbox "teststr$ is equal to the compare string"
        case 1
            msgbox "teststr$ is greater than compare string"
    end select

    'Case-insensitive comparison
    result% = StrComp(teststr$, "THIS IS A TEST STRING", 1)
    select case result%
        case -1
            msgbox "teststr$ is less than the compare string"
        case 0
            msgbox "teststr$ is equal to the compare string"
        case 1
            msgbox "teststr$ is greater than compare string"
    end select
'end sub
```

#351 3352

'Str\$ Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of str$() function  
  
    i% = 3  
    s$ = str$(i%)  
    msgbox s$  
'end sub
```

³⁵¹ XCMD_Str_DOLLAR

³⁵² Str\$ Example

#353 S354 'String\$ Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of String$() function  
  
    s$ = String$(10,65)  
    msgbox s$  
    s$ = String$(10,"B")  
    msgbox s$  
'end sub
```

#355 #356

'SystemFreeMemory Function Example

```
{bmc xc_copy2.bmp}    {bmc xc_copy.bmp}    {bmc xc_print.bmp}    {bmc xc_close.bmp}  
'sub main()  
    'Example of SystemFreeMemory  
  
    msgbox "Free Memory ="&str$(SystemFreeMemory)  
'end sub
```

#357 S358

'SystemFreeResources Function Example

{bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of SystemFreeResources  
  
    msgbox "Free Resources =" +str$(SystemFreeResources)+ "%"  
'end sub
```

³⁵⁷ XCMD_SystemFreeResources

³⁵⁸ SystemFreeResources Example

#359 #360 **'SystemMouseTrails Statement Example {bmc no_dos.bmp}**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of SystemMouseTrails  
  
    SystemMouseTrails TRUE  
    msgbox "Move the Mouse Around Now"  
    SystemMouseTrails FALSE  
'end sub
```

#361 S362

'SystemRestart Statement Example

{bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
```

```
    'Example of SystemRestart
```

```
    'WARNING:  this will restart Windows if you run it
```

```
        SystemRestart
```

```
'end sub
```

³⁶¹ XCMD_SystemRestart

³⁶² SystemRestart Example

#363 S364

'SystemTotalMemory Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'example of SystemTotalMemory  
  
    msgbox "Total System Memory ="&str$(SystemTotalMemory)  
'end sub
```

³⁶³ XCMD_SystemTotalMemory

³⁶⁴ SystemTotalMemory Example

#365 \$366 **'SystemWindowsDirectory\$ Function Example**

```
{bmc xc_copy2.bmp}    {bmc xc_copy.bmp}    {bmc xc_print.bmp}    {bmc xc_close.bmp}  
'sub main()  
    'Example of SystemWindowsDirectory$  
  
    msgbox SystemWindowsDirectory$  
'end sub
```

³⁶⁵ XCMD_SystemWindowsDirectory_DOLLAR

³⁶⁶ SystemWindowsDirectory\$ Example

#367 S368

'SystemWindowsVersion\$ Function Example

{bmc

no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

'sub main()

 'Example of SystemWindowsVersion\$

 msgbox SystemWindowsVersion\$

'end sub

³⁶⁷ XCMD_SystemWindowsVersion_DOLLAR

³⁶⁸ SystemWindowsVersion\$ Example

#369 S370

'Tan Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of Tan() function  
  
    i# = Tan(1)  
    msgbox str$(i#)  
'end sub
```

³⁶⁹ XCMD_Tan

³⁷⁰ Tan Example

#371 #372 **'Time\$ Statement Example**

```
{bmc xc_copy2.bmp}    {bmc xc_copy.bmp}    {bmc xc_print.bmp}    {bmc xc_close.bmp}  
'sub main()  
    'Example of Time command  
  
    time$ = "21:30:40"  
'end sub
```

#373 S374

'Time\$ Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of Time$() function  
  
    msgbox Time$()  
'end sub
```

³⁷³ XCMD_Time_Function

³⁷⁴ Time Function

#375 \$376

'Timer Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of Timer function  
    dim tval as long  
  
    tval = Timer  
    msgbox str$(tval)  
'end sub
```

#377 S378

'TimeSerial Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of TimeSerial function  
    dim tser as double  
  
    tser = TimeSerial(21,40,44)  
'end sub
```

³⁷⁷ XCMD_TimeSerial

³⁷⁸ TimeSerial Example

#379 S380

'TimeValue Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of TimeValue function  
    dim tval as double  
  
    tval = TimeValue("21:40:44")  
'end sub
```


#381 S382

'Trim\$ Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of Trim$()  
    dim teststring as string  
  
    teststring = "      testing      "  
    msgbox "*" +teststring + "*"   
    msgbox "*" +trim$(teststring) + "*"   
'end sub
```

381 XCMD_Trim_DOLLAR

382 Trim\$ Example

#383 S384

'UBound Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Examples of UBOUND  
    dim ia1(8) as integer  
    dim ia2(65 to 70) as integer  
  
    msgbox str$(ubound(ia1))  
    msgbox str$(ubound(ia2))  
'end sub
```

#385 #386 **'UCase\$ Function Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of UCase$  
    dim teststr as string  
  
    teststr = "this is a test"  
    msgbox teststr  
    teststr = ucase$(teststr)  
    msgbox teststr  
'end sub
```

#387 S388

'Val Function Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of Val function  
  
    teststr$ = "123.3"  
    tval# = val(teststr$)  
    msgbox str$(tval)  
'end sub
```

#389 \$390

'Viewport Example {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of ViewPort commands

    'Create a veiwport window
    ViewPortOpen "My ViewPort"
    For i% = 1 to 20
        Print i%
    Next i%
    msgbox "Press OK to clear the viewport."
    ViewPortClear
    msgbox "Press OK to close the viewport."
    ViewPortClose
'end sub
```

389 XCMD_ViewportClear

390 Viewport Example

#391 \$392 **VLine Statement Example** {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Examples of VLINE  
  
    ViewPortOpen  
    ViewPortClear  
    for i% = 1 to 50  
        Print "Here is some test data."  
    next i%  
    VLine 50  
    sleep 2000  
    VLine -50  
    ViewPortClose  
'end sub
```

#393 S394

'VPage Statement Example {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
  'Examples of VPage
```

```
  ViewPortOpen  
  ViewPortClear  
  for i% = 1 to 50  
    Print i%  
  next i%  
  VPage 1  
  sleep 2000  
  VPage -1  
  ViewPortClose
```

```
'end sub
```

#395 \$396

'VScroll Statement Example

{bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of VSCROLL

    ViewPortOpen
    ViewPortClear
    for i% = 1 to 50
        Print "Test data for viewport scroll test."
    next i%
    sleep 2000
    VScroll 50 '50 percent scroll
    sleep 2000
    VScroll 1        'no scroll
    sleep 2000
    ViewPortClose
'end sub
```

³⁹⁵ XCMD_VScroll

³⁹⁶ VScroll Example

'WaitForTaskCompletion Function Example**{bmc****no_dos.bmp}**{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of WaitForTaskCompletion  
  
    taskid% = Shell("notepad",1)  
    'The next statement pauses until Notepad  
    'is shut down  
    WaitForTaskCompletion taskid%  
    msgbox "All done."  
'end sub
```

#399 \$400 **'Weekday Function Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of Weekday()

    wday% = weekday(Now)
    select case wday%
        case 1
            msgbox "Today Is Sunday"
        case 2
            msgbox "Today Is Monday"
        case 3
            msgbox "Today Is Tuesday"
        case 4
            msgbox "Today Is Wednesday"
        case 5
            msgbox "Today Is Thursday"
        case 6
            msgbox "Today Is Friday"
        case 7
            msgbox "Today Is Saturday"
    end select
'end sub
```

#401 \$402 **'While...Wend Statement Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of WHILE-WEND  
  
    i% = 4  
    while i% > 1  
        msgbox "not yet"  
        i% = i% -1  
    wend  
    msgbox str$(i%)  
'end sub
```

#403 S404

'WinActivate Statement Example

{bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of WinActivate  
  
    appn$ = AppFind$("Notepad")  
    WinActivate appn$  
'end sub
```

⁴⁰³ XCMD_WinActivate

⁴⁰⁴ WinActivate Example

#405 \$406

'WinClose Statement Example

{bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of WinClose  
  
    appn$ = AppFind$("Notepad")  
    WinClose appn$  
'end sub
```

405 XCMD_WinClose

406 WinClose Example

#407 \$408

'WinFind Function Example

{bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of WinFind  
  
    appn$ = AppFind$("Notepad")  
    hWnd% = WinFind(appn$)  
    msgbox str$(hWnd%)  
'end sub
```

⁴⁰⁷ XCMD_WinFind

⁴⁰⁸ WinFind Example

#409 \$410

'WinList Function Example {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()
    'Example of WinList
    dim hWindows() as integer

    WinList hWindows
    for i% = lbound(hWindows) to ubound(hWindows)
        msgbox str$(hWindows(i%))
    next i%
'end sub
```

⁴⁰⁹ XCMD_WinList

⁴¹⁰ WinList Example

#411 \$412 **'WinMaximize Statement Example** {bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of WinMaximize  
  
    appn$ = AppFind$("Notepad")  
    WinMaximize appn$  
'end sub
```


#413 \$414 **'WinMinimize Statement Example {bmc no_dos.bmp}**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of WinMinimize  
  
    appn$ = AppFind$("Notepad")  
    WinMinimize appn$  
'end sub
```

#415 \$416

'WinMove Statement Example

{bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of WinMove  
  
    appn$ = AppFind$("Notepad")  
    for i% = 0 to 100  
        WinMove i%, i%, appn$  
    next i%  
'end sub
```

⁴¹⁵ XCMD_WinMove

⁴¹⁶ WinMove Example

#417 S418

'WinRestore Statement Example

{bmc no_dos.bmp}

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of WinRestore  
  
    appn$ = AppFind$("Notepad")  
    WinRestore appn$  
'end sub
```

⁴¹⁷ XCMD_WinRestore

⁴¹⁸ WinRestore Example

'WinSize Statement Example**{bmc no_dos.bmp}**{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of WinSize  
  
    appn$ = AppFind$("Notepad")  
    for i% = 1 to 200  
        WinSize i%, i%, appn$  
    next i%  
'end sub
```

#421 \$422 **'Word\$ Function Example**

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of WORD$() and WORDCOUNT functions  
    dim teststr as string  
  
    teststr = "The quick brown fox jumps over the lazy dog."  
    for i% = 1 to wordcount(teststr)  
        msgbox word$(teststr,i%,i%)  
    next i%  
'end sub
```

WordCount Function Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of WORD$( ) and WORDCOUNT functions  
    dim teststr as string  
  
    teststr = "The quick brown fox jumps over the lazy dog."  
    for i% = 1 to wordcount(teststr)  
        msgbox word$(teststr,i%,i%)  
    next i%  
'end sub
```

WriteINI Statement Example

```
{bmc xc_copy2.bmp}   {bmc xc_copy.bmp}   {bmc xc_print.bmp}   {bmc xc_close.bmp}  
'sub main()  
    'Example of WriteIni statement  
  
    WriteIni "anewsection","anewitem","value","win.ini"  
'end sub
```

#427 S428 'Xor Operator Example

```
{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}
'sub main()
    'Example of XOR
    dim a as integer
    dim b as integer

    a = 5
    b = 9
    if (a < 6) XOR (b > 8) then
        msgbox "Both conditions were not the same."
    else
        msgbox "Both conditions were the same--either TRUE or FALSE."
    end if

    if (a < 6) XOR (b > 9) then
        msgbox "Both conditions were not the same."
    else
        msgbox "Both conditions were the same--either TRUE or FALSE."
    end if
'end sub
```


#429 \$430

'Year Function Example

{bmc xc_copy2.bmp} {bmc xc_copy.bmp} {bmc xc_print.bmp} {bmc xc_close.bmp}

```
'sub main()  
    'Example of Year  
  
    msgbox str$(year(Now))   'display current year  
'end sub
```