

NoMainWin

Table of Contents

[NoMainWin](#)

[Closing the Mainwin](#)

The NOMAINWIN command is a compiler directive, not a command as such. If Liberty BASIC "sees" this command any place in the code, there will be no mainwin in the program. It doesn't matter if the code that contains the command is never executed. Liberty BASIC compiles the code at runtime, and if it finds a NOMAINWIN command anywhere in the code, no mainwin is displayed.

In the following example, since the variable "a" is equal to "2", the code within the "if...then" routine is not executed, yet Liberty BASIC "sees" the NOMAINWIN command and suppresses the mainwin.

```
a = 2
if a = 4 then
    nomainwin
end if

notice "There is no mainwin!"
```

In the following example, the code routine containing the NOMAINWIN command is never executed, yet Liberty BASIC "sees" the NOMAINWIN command and suppresses the mainwin.

```
notice "There is no mainwin!"
wait

[branchLabel]
nomainwin
wait
```

Closing the Mainwin

There is no command to close the mainwin. There is also no command to open it. It appears by default unless the program contains a NOMAINWIN command somewhere in the code.