

Re: GetStdHandle .NET

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.vb/2005-06/msg02101.html>

- *From:* "Jay B. Harlow [MVP - Outlook]" <Jay_Harlow_MVP@xxxxxxxx>
 - *Date:* Wed, 15 Jun 2005 21:43:28 -0500
-

Dragon,

Not sure what your error is about, as strand__sure suggests, can you show more of your code you are attempting to use.

Rather than use Win32 APIs from .NET, have you considered using the System.Console class instead? It encapsulates most of the Win32 Console API functions. With .NET 2.0 (aka VS 2005, aka Whidbey, due out later in 2005). See <http://lab.msdn.microsoft.com/vs2005/> for details) providing even more support.

You can use Console.OpenStandardOutput in VS 2002 & 2003 to get the standard output stream. You can use Console.SetOut to set tell Console.Write what TextWriter (Stream) to use.

Hope this helps
Jay

"Dragon" <no@xxxxxxxxxxxx> wrote in message [news:d8pre1\\$csch\\$1@xxxxxxxxxxxxxxxx](mailto:news:d8pre1$csch$1@xxxxxxxxxxxxxxxx)
| Hello all,
| I ran into trouble with GetStdHandle API. When I'm trying to get standard
| output with GetStdHandle(-11), it returns strange values such as 1548,
| 1876. If I try to WriteConsole() into these handles, it does not work, and
| GetLastError returns ERROR_INVALID_HANDLE. But when I CloseHandle() them,
| it
| succeeds! I tested this in VC++ and VBA, and GetStdHandle(-11) always
| returns 7, which is valid. There is the same problem with standard error,
| but no problems with standard input.
| I declare GetStdHandle as follows:
| Declare Function GetStdHandle Lib "kernel32.dll" (ByVal nStdHandle As _
| Int32) As IntPtr
|
| Any suggestions? Thank you in advance.
|
|
|
|

- **References:**

- ◆ **[GetStdHandle .NET](#)**
 - ◇ From: Dragon

- Prev by Date: **[Re: Long2IP Function?](#)**
- Next by Date: **[How can I print and printpreview a datagrid?](#)**
- Previous by thread: **[Re: GetStdHandle .NET](#)**
- Next by thread: **[timer.start and thread](#)**
- Index(es):
 - ◆ **[Date](#)**
 - ◆ **[Thread](#)**