
Subject: Re: file_basename(): bug or feature?

Posted by [Paul Van Delst\[1\]](#) on Thu, 24 Feb 2005 21:40:53 GMT

[View Forum Message](#) <> [Reply to Message](#)

David Fanning wrote:

> Benjamin Hornberger writes:

>

>

>> On Unix machines, IDL's file_basename() function doesn't split a path in
>> Windows format (with backslashes). On Windows machines, it splits both
>> (yes, I know, on Windows you can also use both to specify a path). Is
>> this a bug or a feature?

>>

>> Linux:

>>

>> IDL> filename='d:\path\to\file.ex'

>> IDL> print,file_basename(filename)

>> d:\path\to\file.ex

>> IDL> filename='d:/path/to/file.ex'

>> IDL> print,file_basename(filename)

>> file.ex

>>

>> Windows:

>>

>> IDL> filename='d:\path\to\file.ex'

>> IDL> print,file_basename(filename)

>> file.ex

>> IDL> filename='d:/path/to/file.ex'

>> IDL> print,file_basename(filename)

>> file.ex

>>

>> I have a situation where a Linux machine has to handle Windows paths,
>> where this is quite annoying ...

>>

>> Any comments?

>

>

> I'm not sure it's a bug OR a feature. More likely the
> sensible way to write the program, given the original
> purpose. :-)

>

> My FSC_BASE_FILENAME program has the same problem,
> simply because I use PATH_SEP to obtain the path
> separator. I think it would be difficult to write
> a generic routine to handle all types of path separators
> properly,

Couldn't you just loop over the types of path separators? (e.g. ['/', '\', ':', ..etc] for

the various systems? Are any of the path separators on different systems allowable filename characters on others?

paulv

--

Paul van Delst
CIMSS @ NOAA/NCEP/EMC
