

## Re: Parameters in command line

**Source:** <http://coding.derkeiler.com/Archive/Perl/comp.lang.perl.misc/2005-02/2038.html>

---

**From:** A. Sinan Unur (*Iusa\_at\_llenroc.ude.invalid*)

**Date:** 02/20/05

Date: 20 Feb 2005 21:47:56 GMT

John Bokma <postmaster@castleamber.com> wrote in  
news:Xns96039C7D6183Fcastleamber@130.133.1.4:

> *Brian McCauley wrote:*  
>  
>> *yusufdestina wrote:*  
>>  
>>> *Can someone point me to a direction for this?*  
>>> *I have an application that opens another perl executable file. [PAR]*  
>>> *How can I pass vars to the perl appl.?*  
>>> *ex: testfile.exe name=value ...*  
>>  
>> *It is confusing to use an .exe suffix on a Perl script.*  
>  
> *Read a bit more about PAR and pp and then you understand.*

I am not so sure the answer to the OP's question has anything whatsoever to do with PAR.

There are two possibilities:

1. testfile.exe's source is under the OP's control. In this case, brian's advice applies.
2. testfile.exe's source is not under the OP's control. In this case, the original language testfile.exe was written in and the fact that it was compiled with PAR becomes irrelevant. If testfile.exe does expect and handle parameters specified on the command line, then the OP can use the system function to do so.

If testfile.exe is not designed and written to expect command line parameters, then it is not possible to pass command line parameters to testfile.exe.

To sum up,

- If (1) applies, refactor code as per Brian's suggestion.  
If (2) applies read testfile.exe's documentation and

comp.lang.perl.misc: Re: Parameters in command line

perldoc -f system

This answer does not depend on the precise way in which testfile.exe came to existence.

Sinan