Subject: Input value at main ()
Posted by darthspawn on Tue, 25 Aug 2009 12:51:24 GMT
View Forum Message <> Reply to Message

with a c++ program I can define some input values

int main(int argc, char *argv[])

how can I do it with u++? I search information on the forum and in the manual section of the site, but I don't found anything. Thank you!
Sam

Subject: Re: Input value at main ()
Posted by masu on Tue, 25 Aug 2009 13:15:52 GMT
View Forum Message <> Reply to Message

Hi,

you may use CommandLine() that return a String Vector.

Matthias

Subject: Re: Input value at main ()
Posted by darthspawn on Tue, 25 Aug 2009 14:26:01 GMT
View Forum Message <> Reply to Message

Thank you! if someone needs the code I used this:

```
String str;
int argc = CommandLine().GetCount();
const Vector<String>& argv = CommandLine();

if(argc < 1 || argc >2)  {
    return;
}
else {
    str = argv[0];
}
```

Subject: Re: Input value at main ()
Posted by sergeynikitin on Tue, 25 Aug 2009 14:49:52 GMT
View Forum Message <> Reply to Message

My piece of code - can someone correct the it, or maybe for someone, and so it will be useful.

```
#include <plugin/pcre/Pcre.h>
 .........
 .........


GUI_APP_MAIN
{

 int argc = CommandLine().GetCount();
 const Vector<String>& argv = CommandLine();

 if((argc >= 1) && ((argv[0]=="/?")||(argv[0]=="/h")||(argv[0]=="-h")||(argv[0]=="--help")))
 {
  Cout()  << "PrintProject-SV ver "<<projectversion<< "\n\n"
    << "Command Line Syntax: PrintProject2 --export-tr|--cfg=parametrs.cfg|--help \n"
    << "   --cfg=filname.cfg - Set alternative config file\n"
    << "      (/cfg=filename and /c=filename - alternative forms of above command)\n"
    << "      (filename default in win32.exe folder or at home in POSIX systems)\n"
    << "\n"
    << "   --export-tr localizationcode - export localization file to edit runtime localization\n"
    << "      (for example --export-tr ruru  - this export russian localization in Unicode encoding\n"
    << "       ruru0 - for windows-1250 encoding\n"
    << "       ruru1 - for windows-1251 encoding\n"
    << "       ruru2 - for windows-1252 encoding\n"
    << "       ruru3 - for windows-1253 encoding\n"
    << "       ruru4 - for windows-1254 encoding\n"
    << "       ruru5 - for windows-1255 encoding\n"
    << "       ruru6 - for windows-1256 encoding\n"
    << "       ruru7 - for windows-1257 encoding\n"
    << "       ruruA - for iso-8859-1 encoding\n"
    << "       ruruB - for iso-8859-2 encoding\n"
    << "       ruruC - for iso-8859-3 encoding\n"
    << "       ruruD - for iso-8859-4 encoding\n"
    << "       ruruE - for iso-8859-5 encoding\n"
    << "       ruruF - for iso-8859-6 encoding\n"
    << "       ruruG - for iso-8859-7 encoding\n"
    << "       ruruH - for iso-8859-8 encoding\n"
    << "       ruruI - for iso-8859-9 encoding\n"
    << "       ruruJ - for iso-8859-10 encoding\n"
    << "\n\n";
  return;
 }
 if((argc >= 1) && ((argv[0]=="/?")||(argv[0]=="/h")||(argv[0]=="-h")||(argv[0]=="--help"))) {
```

```
  Cout() << "Commandline=\""<<argv[0] <<"\"\n";
 }

 if(argc >= 1) {
  RegExp regexp1("^(?:--cfg|/cfg|-c)\\=(.*)$",RegExp::MULTILINE);
  if (regexp1.Match(argv[0])){
   cfg_filename = regexp1[0];
   cfg_set = true;
   Cout() << "CFG_FILE=\""<<cfg_filename <<"\"\n";
  }
 }

 .........
 .........
 Other code
 .........
 .........

}
```