Subject: Is it possible to show a console window in GUI_APP_MAIN?
Posted by sevenjay on Thu, 16 Oct 2008 08:36:01 GMT
View Forum Message <> Reply to Message

Well.....I know maybe it's a stupid question.
Is it possible to show a console window in GUI_APP_MAIN?

In my application, I have a command need users input their password and they can see the log of the command.
To have users' trust, I don't want to send pw from GUI then catch log back from the command message.

On Linux the code example is like this:

```
function called in GUI_APP_MAIN
int function()
{
 printf("this is a test program\n");

 const char * send_argv[]={"sudo","ls","-al",NULL};
 pid_t pid = fork();
 switch(pid) {
  case -1: //parent fork failed
   return P2M_FORK_FAIL;
   break;
  case  0: //child process
   if(-1==execvp(send_argv[0], send_argv)); //execvp("sudo",{"sudo","ls","-al",NULL})
    return P2M_EXEC_FAIL;
   break;
  default: //parent process
   wait(NULL);
   return P2M_OK;
   break;
 }
}
```

But even printf() can not show a message...
If there is any idea, please tell me.
Thanks a lot!

Subject: Re: Is it possible to show a console window in GUI_APP_MAIN?
Posted by pepe11 on Fri, 17 Oct 2008 05:52:50 GMT
View Forum Message <> Reply to Message

Hi,

Do you mean GUI + Console in one aplication ?In WIN is it possible. I use it for debuging in real

time application(very old aplication). But i do not whether that is implemeted in U++.

```
#include <wincon.h>
void InitDebugPrintf(char *Format)
{
 COORD Size;

 InitializeCriticalSection( &Criticalsection );
 ConsoleCreated = TRUE;
 ConsoleCreated = AllocConsole();
 ConsoleScreenBuffer = CreateConsoleScreenBuffer( GENERIC_WRITE, 0, NULL,
CONSOLE_TEXTMODE_BUFFER, NULL );
 Size.X = 120;
 Size.Y = 1024;
 SetConsoleScreenBufferSize( ConsoleScreenBuffer, Size );
 if( ConsoleScreenBuffer != NULL ) SetConsoleActiveScreenBuffer( ConsoleScreenBuffer );
 SetConsoleTitle(Format);

}
void CancelDebugPrintf()
{
 DeleteCriticalSection( &Criticalsection );
 fclose(stream);
 fclose(stream1);
 CloseHandle( ConsoleScreenBuffer );
// FreeConsole();
}
void DebugPrintf( unsigned char Type, char *Format, ... )
{
 char LocalBuffer[128];
 int SizeWritten;
 DWORD SizeToWriteToConsole;
 va_list ap;

 va_start( ap, Format );
 if(( SizeWritten  = _vsnprintf( LocalBuffer , sizeof( LocalBuffer ), Format, ap )) != -1 )
 {
  LocalBuffer[sizeof(LocalBuffer)-1] = '\0';
  if( ConsoleCreated == FALSE ) InitDebugPrintf("PEPE I");
  EnterCriticalSection( &Criticalsection );
    if( ConsoleScreenBuffer != NULL )
    {
     SetConsoleTextAttribute(ConsoleScreenBuffer,FOREGROUND_RED |
FOREGROUND_GREEN | FOREGROUND_BLUE);
     WriteConsole( ConsoleScreenBuffer, TEXT(LocalBuffer), SizeToWriteToConsole =
SizeWritten, &SizeToWriteToConsole, NULL );
    }
  LeaveCriticalSection( &Criticalsection );
```

```
 }
 va_end( ap );
}
```

Pepe

Subject: Re: Is it possible to show a console window in GUI_APP_MAIN?
Posted by sevenjay on Sat, 18 Oct 2008 11:33:56 GMT
View Forum Message <> Reply to Message

thank pepe11, you let me have a direction.
I think.....
Why can We use a console under the Windows NT? Maybe it is a Window. A DOS emulators?
May it is also be like in X-window.
So something I need is to create a console window.

Thanks a lot.

Subject: Re: Is it possible to show a console window in GUI_APP_MAIN?
Posted by mr_ped on Sat, 18 Oct 2008 19:36:03 GMT
View Forum Message <> Reply to Message

Maybe you can also check TheIDE internal console used to output compilation/search/run input/output when used together with "Debug/Run options../Standard output: Console".

This one works in the same way both under windows and linux and it's faster than ordinary shell (bash or cmd.exe).
I'm just not sure about input possibilities, and where to look for this one in the TheIDE sources.

Subject: Re: Is it possible to show a console window in GUI_APP_MAIN?
Posted by sevenjay on Wed, 05 Nov 2008 05:35:18 GMT
View Forum Message <> Reply to Message

mr_ped wrote on Sat, 18 October 2008 21:36Maybe you can also check TheIDE internal console used to output compilation/search/run input/output when used together with "Debug/Run options../Standard output: Console".

This one works in the same way both under windows and linux and it's faster than ordinary shell (bash or cmd.exe).
I'm just not sure about input possibilities, and where to look for this one in the TheIDE sources.
Thanks for your idea.

The console will be in the TheIDE and can not input anything to itself.
Well.... I did it by using xterm in Linux. You can try "cmd /K dir" command if you want to do it In windows.
Here is the sample code:

```
int exec_xterm()
{
 const char * temp_argv[100]={"xterm", "-sb", "-rightbar", "-hold", "-e", "sudo", "ls", NULL};
 char * const * send_argv=(char * const * )temp_argv;//force convert type

 pid_t pid = fork();//-----------------------------------------------------

 switch(pid) {
  case -1: //parent fork failed
   perror("parent fork failed");
   return FORK_FAIL;
   break;
  case  0: //child process
   if(-1==execvp(send_argv[0],send_argv));
   {
    perror("exec failed");
    return EXEC_FAIL;
   }
   break;
  default: //parent process
   wait(NULL);
   return OK;
   break;
 }
 //-----------------------------------------------------------
}
```

Hope it may be helpful for someone...