Subject: Fatal error appear every time when use SetQTF
Posted by akebee on Thu, 21 Jun 2012 12:42:33 GMT

```
for(;;)
{
// GuiLock __;
 if(Thread::IsShutdownThreads())
  break;

 String last = view.GetQTF();
 String now = last + "&[8 [@4 show the qtf string ] ]";
 view.SetQTF(now);
 view.ScrollEnd();

 Sleep(20);

}
```

I use above thread to set the richtextview,but the fatal error
appear every time when i lauch it! Please tell me whoat`s wrong.

File Attachments

Subject: Re: Fatal error appear every time when use SetQTF
Posted by mirek on Thu, 21 Jun 2012 15:06:51 GMT

Well, first of all, commenting out // GuiLock is wrong.

I guess you might have done this because it seemed to be frozen - it would be no wonder as
GuiLock, as written, would cover Sleep call, leaving virtually no chance for main thread to run...

First of all, try:

```
for(;;)
{
        {
 GuiLock __;
 if(Thread::IsShutdownThreads())
  break;

 String last = view.GetQTF();
 String now = last + "&[8 [@4 show the qtf string ] ]";
 view.SetQTF(now);
```

```
  view.ScrollEnd();
 }
 Sleep(20);

}
```

Subject: Re: Fatal error appear every time when use SetQTF
Posted by akebee on Thu, 21 Jun 2012 15:18:04 GMT

er...but if i uncomment the GuiLock__, the program will be very slowly!!!!

Subject: Re: Fatal error appear every time when use SetQTF
Posted by mirek on Thu, 21 Jun 2012 18:56:36 GMT

akebee wrote on Thu, 21 June 2012 11:18er...but if i uncomment the GuiLock__, the program will be very slowly!!!!

Have you tried with putting that Sleep out of block?

Do you understand an explanation why it was slow?

Mirek

Subject: Re: Fatal error appear every time when use SetQTF
Posted by akebee on Thu, 21 Jun 2012 21:36:43 GMT

mirek,i finally aware of you add the brackets before the Sleep,
```
     {
   GuiLock __;
   if(Thread::IsShutdownThreads())
    break;

   String last = view.GetQTF();
   String now = last + "&[8 [@4 show the qtf string ] ]";
   view.SetQTF(now);
   view.ScrollEnd();
  }
  Sleep(20);
```

so the ~GuiLock() have changce to run......and it works well this time.