Subject: Trouble with Socket
Posted by jonarmani on Mon, 28 May 2012 20:15:08 GMT
View Forum Message <> Reply to Message

I just started throwing some programs together with U++ (just downloaded the latest stable release today), and am running into the following compiler errors from TheIDE:

C:\MyApps\TestServer\main.cpp:13:2: error: 'Socket' does not name a type
C:\MyApps\TestServer\main.cpp:14:13: error: ISO C++ forbids initialization of member 'port' [-fpermissive]
C:\MyApps\TestServer\main.cpp:14:13: error: making 'port' static [-fpermissive]
C:\MyApps\TestServer\main.cpp:14:13: error: ISO C++ forbids in-class initialization of non-const static member 'port'
C:\MyApps\TestServer\main.cpp:16:2: error: expected unqualified-id before 'if'
C:\MyApps\TestServer\main.cpp:20:2: error: expected unqualified-id before 'if'
C:\MyApps\TestServer\main.cpp:72:1: error: expected '}' at end of input
C:\MyApps\TestServer\main.cpp:72:1: error: expected unqualified-id at end of input
TestServer: 1 file(s) built in (0:01.32), 1326 msecs / file, duration = 1326 msecs, parallelization 0%

There were errors. (0:01.38)

This is even after just a copy/paste of the server code from the Connection-Oriented Socket Tutorial.

Here is my code. Should I put the Socket code somewhere else beside the Top Window? Am I missing a #include?

```
#include <CtrlLib/CtrlLib.h>

using namespace Upp;

#define IMAGECLASS TestImg
#define IMAGEFILE <Test/images.iml>
#include <Draw/iml.h>

struct TheWindow : TopWindow
{
 MenuBar menu;
 Socket accept_socket, data_socket;
 int port = 2000;
 // Listen for connections using _accept_socket;
 if( !ServerSocket(accept_socket, port) ){
    throw Exc("Couldn't bind socket on the local port.");
 }
```

```
// You can do this in a loop to accept many connections:
 if( accept_socket.IsOpen() ){
     dword ip_addr;
     // Hand off successful connection to _data_socket
     if( !accept_socket.IsError() && accept_socket.Accept(data_socket, &ip_addr) )
     {
         Cout() << "Connection from " << FormatIP(m_ipaddr) << "\n";
         // Read from the socket until it is closed, has an error, or you see an end-of-file marker
         // (EOF optional and application-specific)
         while(data_socket.IsOpen() && !data_socket.IsEof() && !data_socket.IsError())
         {
             Cout() << data_socket.Read();
         }
     }
     Cout() << "\n";
 }

 void Exit()
 {
  if(PromptOKCancel("You really wish to exit?")) Break();
 }

 void SubMenu(Bar& bar)
 {
  bar.Add("Exit", TestImg::exit, THISBACK(Exit));
 }

 void MainMenu(Bar& bar)
 {
  bar.Add("Menu", THISBACK(SubMenu));
 }

 virtual void Paint(Draw& w)
 {
  w.DrawRect(GetSize(), Black);
  w.DrawText(20,20,"Server is running", Arial(30), Green);
 }

 typedef TheWindow CLASSNAME;

 TheWindow()
 {
  Title("Test Server");
  AddFrame(menu);
  menu.Set(THISBACK(MainMenu));
 }
};
```

```
GUI_APP_MAIN
{
 TheWindow app;
 app.SetRect(0,0,400,100);
 app.Run();
}
```