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();
}
```

Subject: Re: Trouble with Socket
Posted by mirek on Tue, 29 May 2012 06:17:50 GMT
View Forum Message <> Reply to Message

jonarmani wrote on Mon, 28 May 2012 16:15Am I missing a #include?

Indeed. You need to add Web package and #include "Web/Web.h".

Alternatively (and better), socket encapsulation was recently refactored into new TcpSocket class, which is part of Core (so you have it alread included).

See http://www.ultimatepp.org/src$Core$TcpSocket$en-us.html

Mirek

Subject: Re: Trouble with Socket
Posted by jonarmani on Fri, 01 Jun 2012 02:47:56 GMT
View Forum Message <> Reply to Message

Thanks for your response. I'm now trying to figure out how to use the TcpSocket class correctly. Looks like it's what I need, I just need to be a better programmer.

Subject: Re: Trouble with Socket
Posted by jonarmani on Sun, 03 Jun 2012 17:04:14 GMT
View Forum Message <> Reply to Message

Ahhh! After messing with this for so long, I figured out I needed to download the latest nightly release for Socket or TcpSocket stuff to work! I was using version 4193 previously, which I believe was the latest "stable" release.

Is it standard practice for everyone to be using the latest nightly release? The one I grabbed was version 5021.

Subject: Re: Trouble with Socket
Posted by BioBytes on Sun, 03 Jun 2012 18:19:24 GMT
View Forum Message <> Reply to Message

Hi Jonarmani,

A lot of improvments are generally implemented in the nightly builds. My way is to rename the installation directory of my current stable U++ version and install the lastest build (ex. I rename Upp directory Upp-5010 for example and install the latest version ex. U++5021 in Upp).

So I can try the new build with minimal risk and can return easily to the previous stable version by renaming back the Upp-XXXX to Upp.

Regards

Biobytes

Subject: Re: Trouble with Socket
Posted by jonarmani on Mon, 11 Jun 2012 14:49:11 GMT
View Forum Message <> Reply to Message

Ok, I really like that suggestion. Thank you.

Subject: Re: Trouble with Socket
Posted by jash08 on Wed, 29 Aug 2012 09:34:33 GMT
View Forum Message <> Reply to Message

Quote:Sets timeout for all operations. Zero means that all operations return immediatelly (in that case it is usually a good idea to perform some sort of external blocking on socket or socket group using e.g. SocketWaitEvent). Null means operations are blocking (but they still can invoke WhenProgress periodically if defined). Other values specify a number of milliseconds. Note: It is possible to adjust timeout before any single TcpSocket operation. Returns *this.

Thanks for the information. i really appreciate it a lot.

click here