Abstract TCP Server – part 2

Mission

To build a framework for ease the implementation of TCP servers, this include concurrent server with a soft-closedown function.

Background

Previous exercise Abstract TCP Server – part 1

Comments

- Comments of code -- Corey's Tutorial: How to Comment & Document Your Code (video)
- Insert XML Comments https://docs.microsoft.com/en-us/dotnet/csharp/codedoc
- How to use doxygen : http://www.doxygen.nl/manual/starting.html
- Download doxygen https://sourceforge.net/projects/doxygen/

Assignment 1 – Add comments to your Simple Framework

You must add (xml comments or /// comments) to your public methods in the 'AbstractTCPServer'.

Make the comments be a good documentation for others to use i.e. describe the methods, its parameters as well as return values.

Use Doxygen to generate the html-help pages.

Assignment 2 – Soft Shutdown of the Simple Framework

Instead of having, an infinite loop, then support a soft shutdown.

To support soft shutdown in the AbstractTCPServer do following steps.

- Introduce a bool variable running which is initial true.
- Use this variable in the while-loop.
- Implement a method to set the running variable to false
- Implement another method, which listen to the port number of the server plus one e.g. the server 7007, while the 'stop server' have 7008. When a client connect to this stop server – you could make some check for validity – then call the method to change state of the running variable.
- BEFORE the while-loop, make a new Task where to start this 'stop server'.
- Do not call AcceptTCPClient direct. Make an if-statement if (listener.Pending()) -> then call AcceptTcpClient else wait XX sec (using Thread.Sleep(2*1000) = 2 sec)