

1° Esempio di Web Server: Interroga un server HTTP (da cui l'uso della porta 80), il cui indirizzo IP viene letto da console e scrive su console quello che il server ha risposto (in pratica invia tramite lo stream il sorgente della pagina). Se il server fosse www.google.com a schermo si leggerà qualcosa del genere:

```
<!doctype html><html itemscope="" itemtype="http://schema.org/WebPage"><head> <meta content="/images/google_favicon_128.png" itemprop="image"><title>Google</title> ...
```

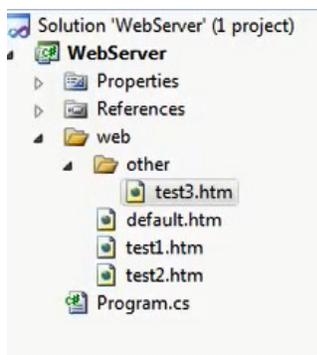
Se apriamo Firefox per aprire www.google.com e con un click del mouse destro scegliamo "Visualizza sorgente pagina" vedremo lo stesso risultato.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.IO;
using System.Net.Sockets;

namespace PMWB
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Please enter the name of a server");
            string server = Console.ReadLine();
            TcpClient client = new TcpClient(server, 80); // 80 is the web port (HTTP)
            StreamReader sr = new StreamReader(client.GetStream());
            StreamWriter sw = new StreamWriter(client.GetStream());

            try
            {
                sw.WriteLine("GET / HTTP/1.0\n\n");
                sw.Flush();

                string data = sr.ReadLine();
                while (data != null)
                {
                    Console.WriteLine(data);
                    data = sr.ReadLine();
                }
                client.Close();
            }
            catch (Exception e)
            {
            }
        }
    }
}
```



2° Esempio di Web Server: Questo server interagirà con un client Firefox: in pratica il server rimane disponibile sulla porta 1302 della macchina locale (localhost). Quando l'utente tramite Firefox (o altro browser) scriverà la richiesta <http://localhost:1302> il server risponderà inviando la pagina richiesta. Per fare alcuni test occorre creare dei file html da inserire nella cartella del progetto (vedi immagine a sinistra). Occorre notare che un WebServer che riceve una richiesta deve rispondere con dei codici di stato e dei relativi messaggi che sono stati definiti per il protocollo HTTP.

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.IO;
using System.Net.Sockets;

namespace WebServer
{
    class Program
    {
        static void Main(string[] args)
        {
            TcpListener listener = new TcpListener(1302);
            listener.Start();

            while (true)
            {
                Console.WriteLine("Waiting for a connection");
                TcpClient client = listener.AcceptTcpClient();
                StreamReader sr = new StreamReader(client.GetStream());
                StreamWriter sw = new StreamWriter(client.GetStream());
                try
                {
                    // client request

                    string request = sr.ReadLine();
                    Console.WriteLine(request);
                    string[] tokens = request.Split(' ');
                    string page = tokens[1];
                    if (page == "/")
                    {
                        page = "default.htm";
                    }

                    // Find the file
                    StreamReader file = new StreamReader("../web" + page);
                    sw.WriteLine("HTTP/1.0 200 OK\n");

                    // Send the file
                    string data = file.ReadLine();
                    while (data != null)
                    {
                        sw.WriteLine(data);
                        sw.Flush();
                        data = file.ReadLine();
                    }
                }
                catch (Exception e)
                {
                    //error
                    sw.WriteLine("HTTP/1.0 404 OK\n");
                    sw.WriteLine("<H1>SORRY! We couldnt' find your file!</H1>");
                    sw.Flush();
                }
                client.Close();
            }
        }
    }
}

```