

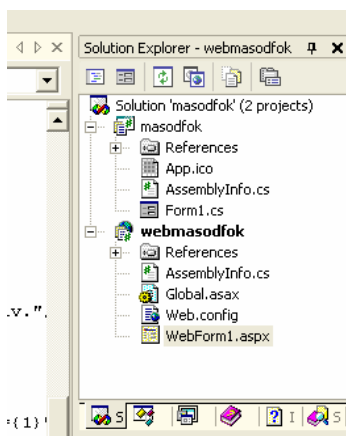
## XVI.2. Webes alkalmazások alapjai

Napjainkban az internetes elérhetőség, rendelkezésre állás már-már szinte alapkövetelmény. Ennek a felhasználói igénynek a kielégítésére a keretrendszer lehetőséget ad Web alapú alkalmazások készítésére.

Az ilyen jellegű alkalmazás készítésének alapfeltétele az, hogy egy web kiszolgáló eléréséhez megfelelő jogosultsággal rendelkezünk. Ez a gyakorlatban az alábbiakat jelenti:

- Azon a gépen ahol fejleszteni szeretnénk, először fel kell installálni az Internet Information Service (IIS) szolgáltatást.
- Fel kell installálni a Visual Studio.NET alkalmazást.
- Ez létrehoz két felhasználói csoportot a számítógépen, *Debugger Users* és *VS Developers* névvel.
- Azokat a fejlesztőket rakjuk bele ezekbe a csoportokba, amelyektől ilyen alkalmazások fejlesztését várjuk. (A *Debugger Users* csoportba minden fejlesztőt bele kell rakni, különben az operációs rendszer a nyomkövetési módban fordított állományt nem engedi a keretrendszerből futtatni!)

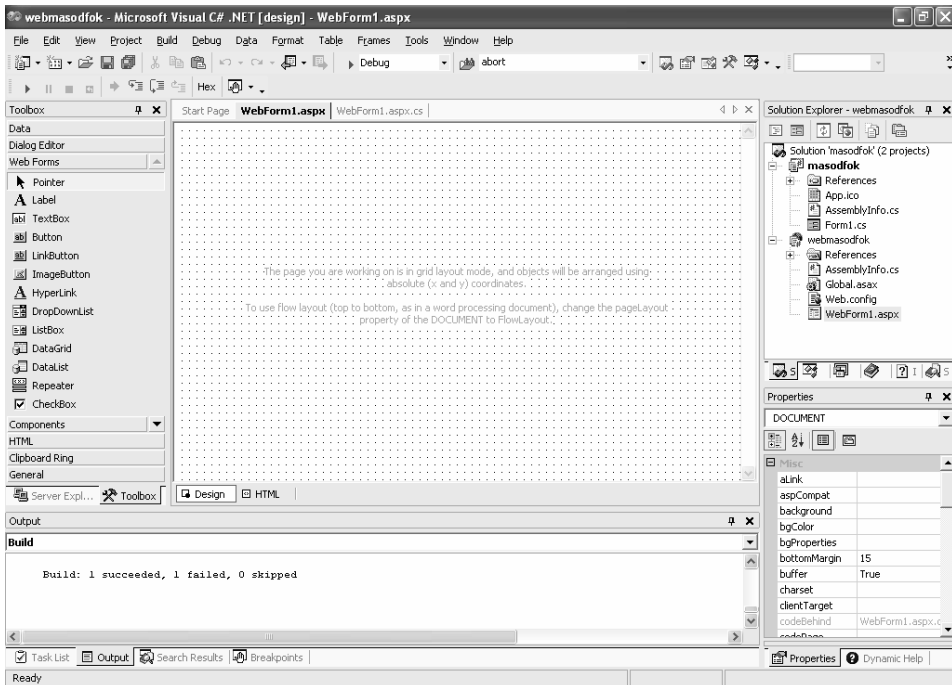
Ha a fenti feltételek megvannak, akkor *ASP.NET Web Application* sablont választva készíthetünk webes alkalmazást. Ezek alapján készítsünk egy *webmasodfok* projektet a fejlesztőkörnyezetben, ahogyan az a következő képen látszik, az előző *masodfok* projekt mellett elhelyezve. (A keretrendszer *Solution* fogalma több projektet enged egy keretbe foglalni, hiszen gyakran előfordul, hogy egy feladatmegoldást nem egy projekttel célszerű megadni. Példánk esetében nincs erről szó.)



28. ábra

Miután megadtuk a nevet, elkészül a következő üres weboldal. A létrehozott *Webform1.aspx* az üres HTML oldal (ezért is van HTML nézete) és a háttérben meghúzódnó programfájl (*Webform1.aspx.cs*). Látható, hogy a *Toolbox* ablakban a korábbi *Windows Forms* felirat helyett *Web Forms* olvasható, mutatva, hogy ezek a vezérlők web alapú alkalmazásokhoz használhatóak. A *Tulajdonság* ablak egyetlen objektuma a *DOCUMENT* objektum lesz, ami természetesen magának a HTML dokumentumnak felel meg. Ezek tulajdonságértékeit módosíthatjuk, például a *Title* mező értékét, megadva ezzel a weboldal címkéjét, vagy a *bgColor* paraméterrel beállítva a kívánt háttérszínt.

A kapott képernyő a következő alakú lesz:



29. ábra

A *Windows Form* megoldáshoz hasonlóan készítsük el a másodfokú egyenlet megoldását ebben a környezetben is. Ehhez első lépésként alakítsuk ki a programunk felületét az előző példához hasonlóan, majd a nyomógomb eseménykezelő függvényét definiáljuk.

A kapott forráskód (*Webform1.aspx.cs*) a következő lesz:

```
using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Web;
using System.Web.SessionState;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.HtmlControls;

namespace webmasodfok
{
    /// <summary>
    /// Summary description for WebForm1.
    /// </summary>
    public class WebForm1 : System.Web.UI.Page
    {
        protected System.Web.UI.WebControls.Label Label1;
        protected System.Web.UI.WebControls.Label Label2;
        protected System.Web.UI.WebControls.Label Label3;
        protected System.Web.UI.WebControls.Label Label4;
        protected System.Web.UI.WebControls.TextBox a;
        protected System.Web.UI.WebControls.TextBox b;
        protected System.Web.UI.WebControls.TextBox c;
        protected System.Web.UI.WebControls.Label megoldas;
        protected System.Web.UI.WebControls.Button Button1;

        private void Page_Load(object sender, System.EventArgs e)
        {
            // Put user code to initialize the page here
        }

        #region Web Form Designer generated code
        override protected void OnInit(EventArgs e)
        {
            //
            // CODEGEN: This call is required by the ASP.NET
            //                               Web Form Designer.
            //
            InitializeComponent();
            base.OnInit(e);
        }
    }
}
```