**ATIVIDADE 12-1. TRABALHANDO COM WEB PAGES E SERVER CONTROLS**

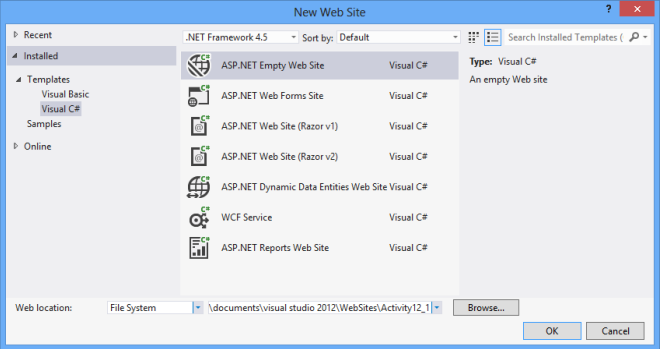
Nesta atividade você irá começar a familiarizar com o seguinte:

* Criando uma aplicação Web Form-baseado em GUI.
* Trabalhando com página e eventos server control.

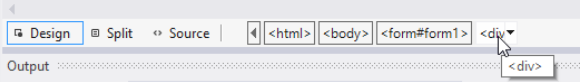
**Criando uma aplicação Web**

Para criar uma aplicação Web, siga estes passos:

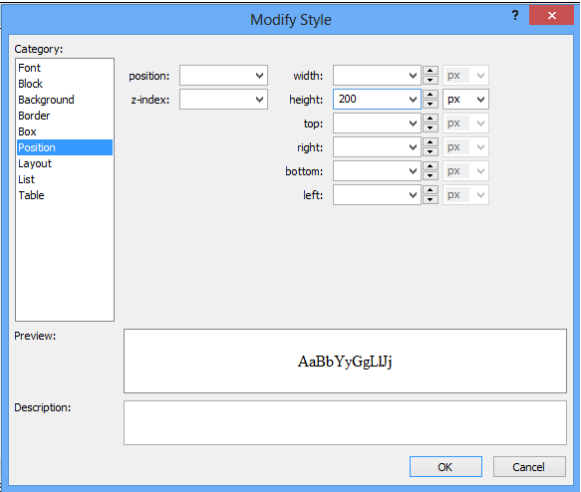
1. Inicie o Visual Studio. No menu *File*, escolha *New Web Site*.
2. Escolha o *ASP.NET Empty Web Site*. Troque o nome do web site na localização final do arquivo *Activity12\_1*. Por padrão, web sites são armazenados no caminho *Documents\Visual Studio 2012\WebSites*. Na figura abaixo mostra a caixa de diálogo *New Web Site* para criar um *Web Site* ou *Service*.



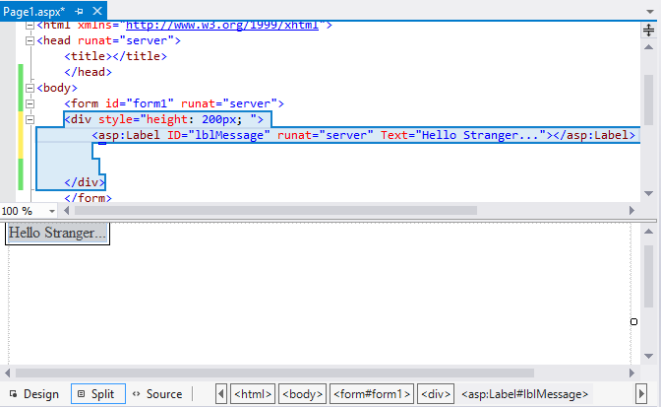
1. Clique no botão *OK* na caixa de diálogo *New Web Site*.
2. Depois de criado o *Web Site*, você precisa adicionar um *web form*. Embaixo do menu *Website*, selecione *“Add new item”* e adicione um *Web Form* nomeando como *Page1.aspx*.
3. Será apresentado para você um *Web Page Designer* no *view Source*. Troque para o *view Design* clicando o *tab* no canto esquerdo embaixo da janela.
4. Selecione a tag *<div>* no rodapé da janela *Design*. Veja na figura abaixo:



1. Na janela *Properties*, selecione a propriedade *style* e clique nas elipses para lançar a caixa de diálogo *Modify Style*. Embaixo da categoria *Position*, seta para altura para *200 pixels*. Veja na figura abaixo e clique no botão *OK*.



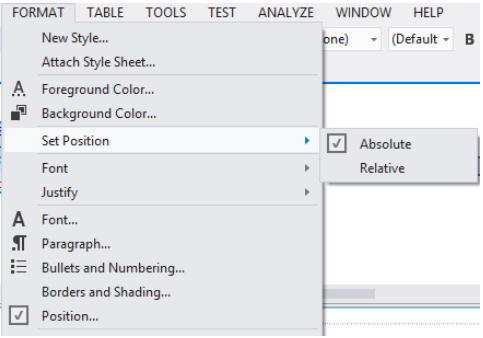
1. Troque a janela *Page Design* para o modo split. Arraste o controle *Label* do *toolbox* e jogue este entre as *tags div* na *source pane*. Troque o texto para *“Hello Stranger...”* e o ID para *lblMessage*.
2. Você pode obter uma mensagem perguntando para sincronizar a *view design* e a *view source*. Depois de sincronizar as *view*, você pode desenhar conforme a figura abaixo:



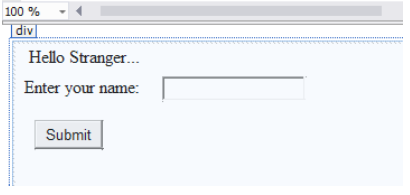
1. Em um processo similar do passo 9, adicione os controles na tabela baixo entre as *div tags*. Não se preocupe sobre o posicionamento da forma ainda.

|  |  |  |
| --- | --- | --- |
| Control  (Controle) | Properties (Propriedades) | Value  (Valores) |
| Label | Id  Text | lblName  Enter your name: |
| Textbox | ID  Text | txtName  [Blank] |
| Button | ID  Text | btnSubmit  Submit |

1. Para a posição um controle, selecione o controle no *designer* e o *menu format*, selecione *Set Position* e selecione *Absolute*.



1. Volte no *absolute positioning* para o resto dos controles. Você deverá agora ser hábil para arrastar eles em torno no designer visual para a posição dele. Disponha os controles no formulário assim, como na figura abaixo.



**Criando Controle de Evento Handlers server-side**

Para criar um controle de evento handlers server-side, siga esses passos:

1. No pane Design, dê um duplo clique no botão Submit. Esta ação irá abrir a página do *code behind* (atrás do código) *Page1.aspx.cs* e cria um método evento *handlers* para o evento *click* do botão.
2. No *evento handler* do botão, verifique para ver se a página que solicita é o resultado de um *postback*, Se este é, troque a *text* da *lblMessage* para falar *Hello* usando o texto contido no *control txtName*:

*protected void btnSubmit\_Click(object sender, EventArgs e)*

*{*

*if (IsPostBack)*

*{*

*lblMessage.Text = “Hello ” + txtName.Text;*

*}*

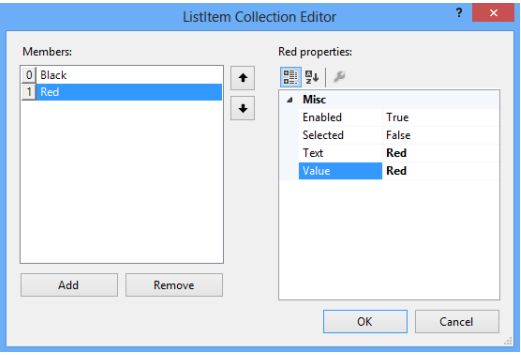
*}*

1. Usando o *menu Debug*, lança a página no *modo Debug*. O primeiro tempo você lança o depurador para um projeto web, você irá ver uma caixa de diálogo perguntando se você quer habilitar o depurador no arquivo *Web.config*. Clique no botão *OK* para habilitar o depurador. Quando a página mostra no navegador, entre com o seu nome o clique no botão *Submit*. Verifique que a página irá mostrar novamente com o seu nome incluído no *Hello... message*. Depois de testar, feche o navegador.

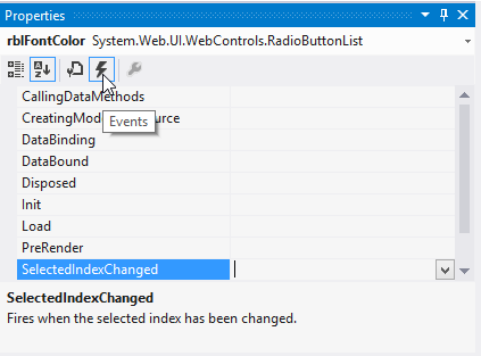
**Criando Controle Eventos Non-Postback Server**

Para criar o controle de eventos *non-postback server*, siga os passos abaixo:

1. Usando o *Page Designer*, adicione um controle *RadioButtonList* para a *Page1*. Troque este posicionamento para *absolute* e localize isto debaixo do *textbox*.
2. Troque a propriedade *ID* para *rblFontColor*. Clique em *Items collections* na janela *Properties* para mostrar a caixa de diálogo *ListItem Collection Editor*. Adicione um item *Black* e um item *Red* para a lista.



1. Na janela *Properties* para o *rblFontColor*, clique no evento *buttons* para mostrar o evento *list*.



1. Dê um duplo clique no evento *SelectedIndexChanged*. Isto irá levar você para a página do *code-behind* e setar os métodos para manusear o evento. Adicione o seguinte código para trocar a cor da fonte do controle *lblMessage* no evento *handler*.

*public void rblFontColor\_SelectedIndexChanged(object sender, EventArgs e)*

*{*

*if(rblFontColor.SelectedItem.Value == “Red”)*

*{*

*lblMessage.ForeColor = System.Drawing.Color.Red;*

*}*

*else*

*{*

*lblMessage.ForeColor = System.Drawing.Color.Black;*

*}*

*}*

1. Usando o menu *Debug*, lança a página no modo *Debug*. Quando é mostrado a página no navegador, clique no *radio button Red*. Aviso que o evento não causará um *postback* para ocorrer. Este evento irá ficar na fila até que um evento *AutoPostBack* provocará um *post back*.
2. Entre com um nome no controle *txtName* e clique no botão *Submit*. Quando o postback é processado, ambos o evento *SelectedIndexChanged* do *rblFontColor* e o evento *Click* do botão *Submit* são processados.
3. Após testar, feche o navegador e saía do Visual Studio.