



- [Página Inicial](#)
- [Quem Somos »](#)
- [Membros](#)
- [Artigos / Tutoriais »](#)
- [Downloads](#)
- [Fale Conosco](#)

Manipulando o Web.Config – Armazenando e Recuperando Dados

Enviado por [Wellington Balbo de Camargo](#) em 21 de dezembro na categoria [ASP.NET](#), [C#](#), [Desenvolvimento](#), [Destaques](#)

[Home](#) » [ASP.NET](#) » Manipulando o Web.Config – Armazenando e Recuperando Dados

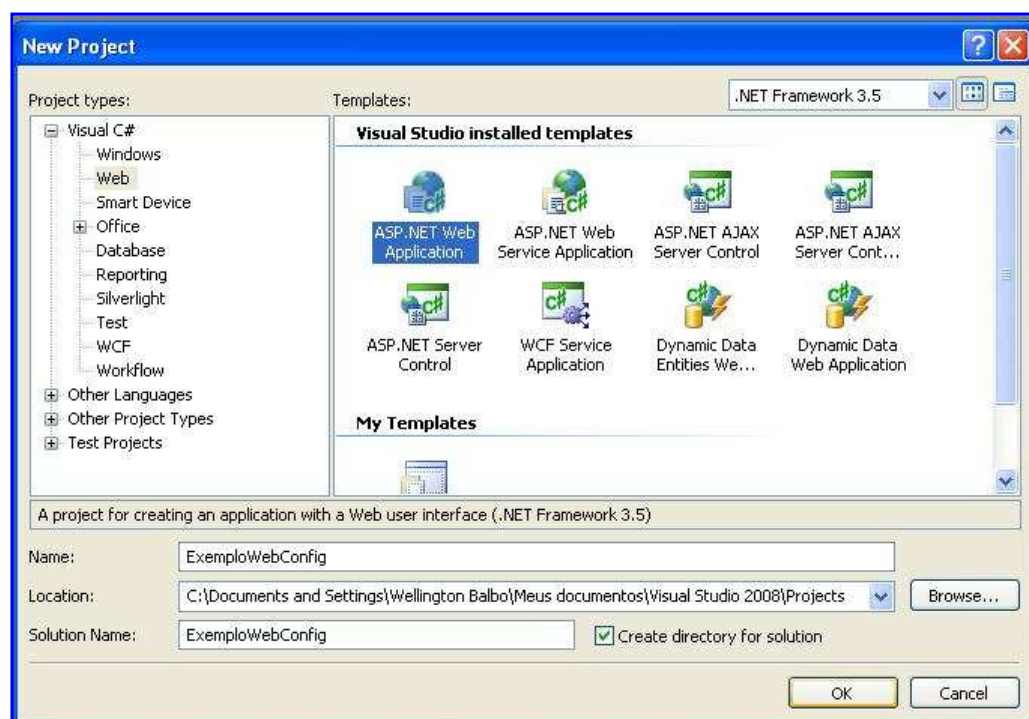


Olá pessoal, neste artigo mostrarei como armazenar e recuperar dados do arquivo **Web.Config** em uma aplicação **ASP.NET**.

Faço este artigo com base na videoaula de [Bruno Belizário](#), do [Portal Linha de Código](#), mediante autorização do mesmo. Acompanhem o passo-a-passo:

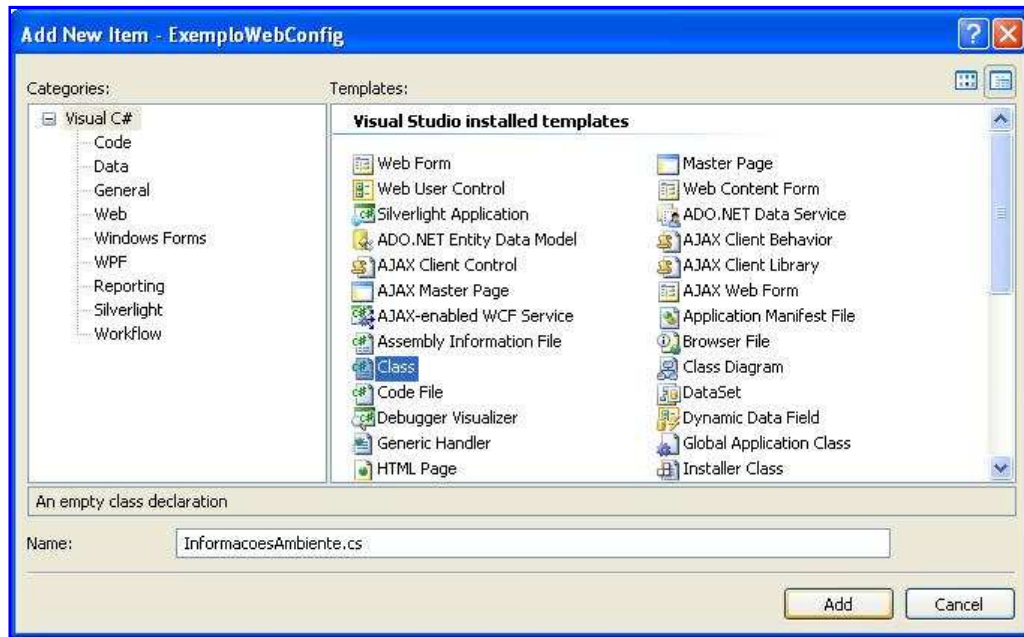
Como muitos devem saber, o **Web.Config** é o arquivo principal de configuração de uma aplicação **ASP.NET**. Este arquivo já vem com diversas configurações padrão, mais podemos adicionar novas seções dentro dele, criando assim novas opções de acordo com nossas necessidades.

Vá em **File > New Project** (CTRL + SHIFT + N), escolha o template **ASP.NET Web Application** e dê o nome de **ExemploWebConfig**, como a imagem abaixo nos mostra:



Após a criação da aplicação, abra o **Web.Config** e crie uma seção que será responsável por armazenar dados sobre o ambiente da aplicação com o cliente, se está em **homologação** ou **produção** (lembrando que este é um exemplo fictício, seguindo a videoaula de Bruno Belizário).

Crie uma classe que será o tipo de dados que irá estruturar minhas informações. Na **Solution Explorer** (CTRL + W + S), vá em **Add > New Class**, dê o nome de **InformacoesAmbiente.cs** e clique em OK.



Agora vá ao **Web.Config**, em cima de `</configSections>`, crie uma nova seção como mostra o código a seguir:

```
1 <section name="ConfigAmbiente" type="ExemploWebConfig.InformacoesAmbiente, ExemploWebC
```

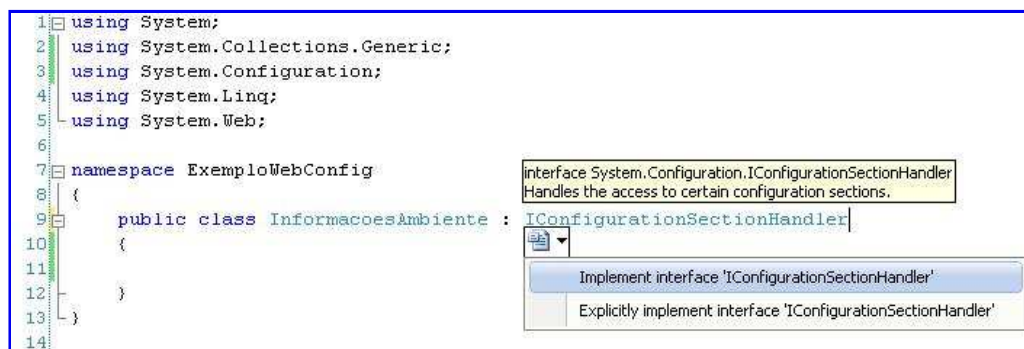
Como pode ser visto, uso os atributos **name**, para dar um nome à seção, **type**, para especificar o namespace + o nome da classe que acabamos de criar e o namespace novamente.

Agora, entre **appSettings** e **connectionStrings**, vamos descrever os dados da seção que acabamos de criar e criar alguns atributos:

```
1 <ConfigAmbiente Ambiente="Producao" Database="DBProducao"/>
```

Criei duas propriedades, uma informando qual é o ambiente da aplicação e outra informando qual é o database que estamos trabalhando. Neste exemplo é o de **Produção**.

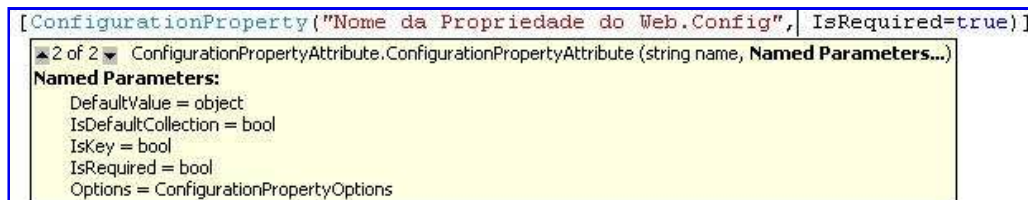
Agora vamos criar as propriedades referentes às declaradas no **Web.Config**. Volte à classe criada anteriormente. Temos que fazer com que nossa classe implemente uma interface específica (**IConfigurationSectionHandler**) para trabalhar com valores vindos do **Web.Config**:



Como você pode ver na imagem acima, quando implementamos esta interface, o Visual Studio já nos sugere que a declaremos em nosso código. Clique na 1ª opção para implementá-la.

*OBS: Se você não conseguir acessar esta interface, é porque não foi declarado o namespace **System.Configuration**, não se esqueça dele.*

Antes de criarmos as propriedades em nossa classe, temos que usar os atributos de configuração:



Uso o **ConfigurationProperty**, que espera o nome da propriedade, que no nosso exemplo é **Producao**, e o atributo **IsRequired** como **true**, que informa que ela é obrigatória.

Após isso, crie as seguintes propriedades, ambas já “setadas” com o **ConfigurationProperty**:

```
1 [ConfigurationProperty("Ambiente", IsRequired = true)]
2
3 public string Ambiente { get; set; }
4
5 [ConfigurationProperty("Database", IsRequired = true)]
6
7 public string Database { get; set; }
```

Agora no método que foi criado pela implementação, vamos alimentar as propriedades desta forma:

```
1 public object Create(object parent, object configContext, System.Xml.XmlNode section)
2 {
3     {
4     Ambiente = section.Attributes["Ambiente"].Value;
5     Database = section.Attributes["Database"].Value;
6     return this;
7     }
8 }
```

Resumindo, nossa classe **InformacoesAmbiente** é um tipo que irá armazenar e recuperar os dados do **Web.Config** para minha aplicação. Lá no **Web.Config** criei uma seção, com suas propriedades que estão descritas em minha classe, que implementa o tipo da seção do **Web.Config**, que usa a implementação **IConfigurationSectionHandler**.

Finalizando, dentro deste método **Create**, passamos as variáveis **Ambiente** e **Database**, criadas acima, que recebem os atributos referentes às propriedades criadas em minha seção do **Web.Config**.

Agora vá ao **Default.aspx** e crie dois labels e dois textboxes, para que fiquem dessa forma:

Os Textboxes irão exibir as informações armazenadas no Web.Config. Altere o ID deles para **txtAmbiente** e **txtBancoDados**, respectivamente.

Agorá vá a página de códigos da **Default.aspx** e crie um método chamado **RecuperaDados**:

```
1 private void RecuperaDados()
2
3 {
4
5 try
6
7 {
8
9 //Instancio a classe InformacoesAmbiente
10
11 InformacoesAmbiente objInfo = new InformacoesAmbiente();
12
13 //Faço um Cast(conversão) em meu objeto instanciado e uso o método GetSection
14
15 //para recuperar os dados da seção ConfigAmbiente que foi criada no Web.Config
16
17 objInfo = (InformacoesAmbiente)ConfigurationManager.GetSection("ConfigAmbiente");
18
19 //Com as informações armazenadas passo os valores das propriedades aos textboxes
20
21 txtAmbiente.Text = objInfo.Ambiente;
22
23 txtBancoDados.Text = objInfo.Database;
24
25 }
26
27 catch (Exception ex)
28
29 {
30
31 throw new Exception(ex.Message);
32
33 }
34
35 }
```

Agora chame o método no **Page_Load** da página:

```
1 protected void Page_Load(object sender, EventArgs e)
2
3 {
4
5 RecuperaDados();
6
7 }
```

Salve o projeto e compile. Irá aparecer a mensagem para você modificar o **Web.Config** para permitir a compilação, clique em **OK** e aguarde.

Ambiente:	<input type="text" value="Produção"/>
Banco de Dados:	<input type="text" value="DBProducao"/>

Dessa forma vemos sendo preenchido os textboxes com os valores recuperados do **Web.Config**.

Faça um teste e altere os valores do **Ambiente** e do **Database** para **Homologação** e **DBHomologacao**, respectivamente, compile e veja o resultado:

Ambiente:	<input type="text" value="Homologação"/>
Banco de Dados:	<input type="text" value="DBHomologacao"/>

Assim finalizo o artigo.

Para quem se interessar, disponibilizo o código fonte desse projeto [aqui](#).

Créditos à [Bruno Belizário](#), que fez a videoaula e ao [Portal Linha de Código](#), por onde pude baixá-la (mediante assinatura), estudá-la e posteriormente fazer este artigo.

Quaisquer dúvidas mandem emails para wellingtonbalbo@gmail.com ou deixem nos comentários deste artigo que responderei o mais rápido possível.

Até o próximo artigo!



Comentar

Nome (obrigatório)

E-mail (não ser publicado) (obrigatório)

Website

Enviar comentário

Últimas do site

- [ASP.NET MVC – Início, Meio e Fim – Parte 4](#)
- [Introdução a Parallel Programming](#)
- [Manipulando o Web.Config – Armazenando e Recuperando Dados](#)
- [Diferença entre Mocks e Stubs](#)
- [Como funciona o Azure – O Fabric](#)
- [Desenvolvimento para Windows Phone 7](#)
- [Road Show 2011 vem aí!](#)
- [Como funciona o Azure – O Hypervisor](#)
- [ASP.NET MVC – Criando relatórios com o Report Viewer](#)
- [Trabalhando com Cache](#)
- [O que é, e por que utilizar o ASP.NET MVC ?](#)
- [Utilizando Linq to jQuery](#)
- [ASP.NET MVC – Obter retorno Json pelo Controller](#)
- [ASP.NET MVC – Início, Meio e Fim – Parte 3](#)
- [Vídeo Aula: Trabalhando com SP's no Visual Studio 2010 com C#](#)

Mais da comunidade

- [Fórum .NET Interior](#)
- [Entre para a comunidade](#)
- [.NET Cast's](#)
- [.NET Videos](#)
- [Área do Membro \(Login\)](#)

Links

- [ASP.NET](#)
- [Fórum do MSDN](#)
- [Linha de Código](#)
- [MSDN Brasil](#)

- [TechNet Brasil](#)
- [Twitter – .NET Interior](#)

Navegação

- [Artigos / Tutoriais](#)
- [Downloads](#)
- [Fale Conosco](#)
- [Membros](#)
- [Quem Somos](#)

Mais da comunidade

- [Fórum .NET Interior](#)
- [Entre para a comunidade](#)
- [.NET Cast's](#)
- [.NET Videos](#)
- [Área do Membro \(Login\)](#)

Designed by [Elegant Themes](#) | Powered by [Free Premium Wordpress Themes](#)