**Programação ADVPL Web**

O treinamento de ADVPL WEB mostra como funciona do Protheus como um servidor HTTP e integração do ADVPL com **HTML, Java Script e Asp**

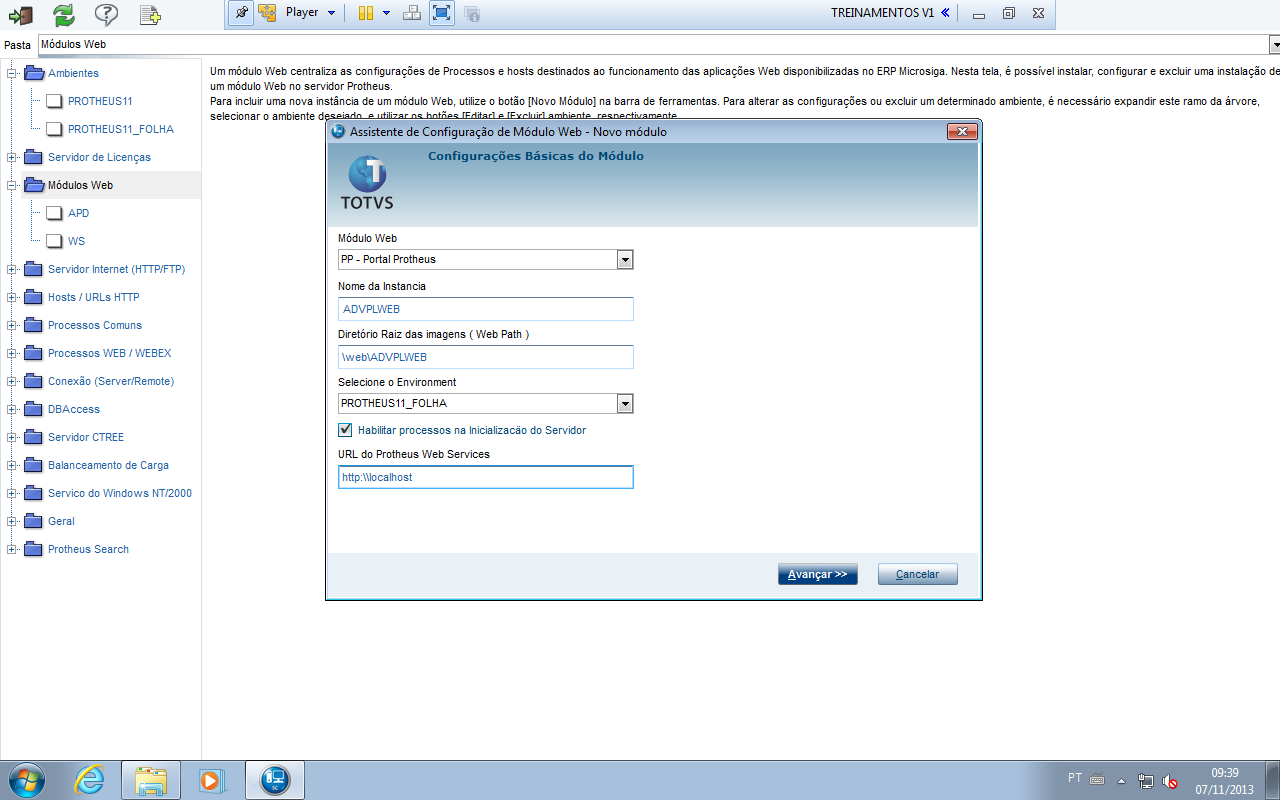
Iremos ver um case de um portal que iremos configurar juntos.

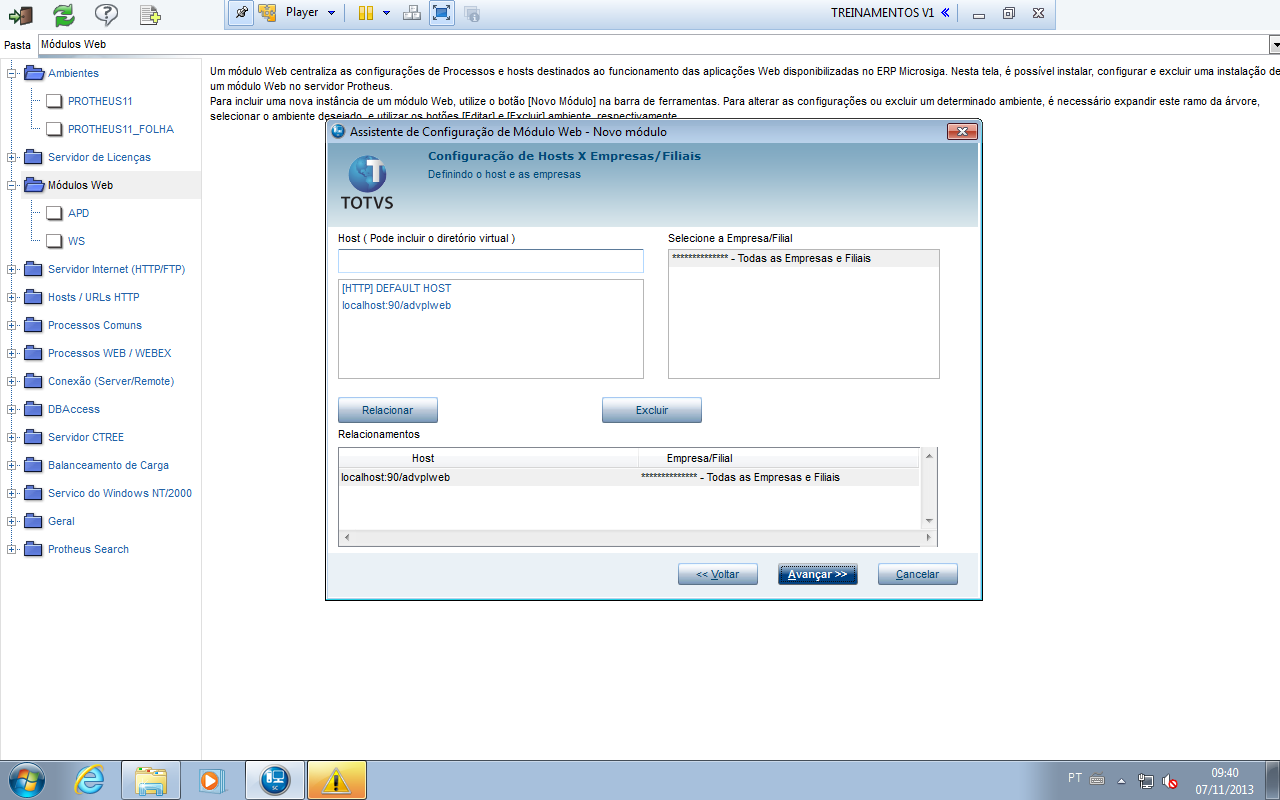
É necessário subir um portal para configurar um Servidor HTTP com nosso ambiente de desenvolvimento.

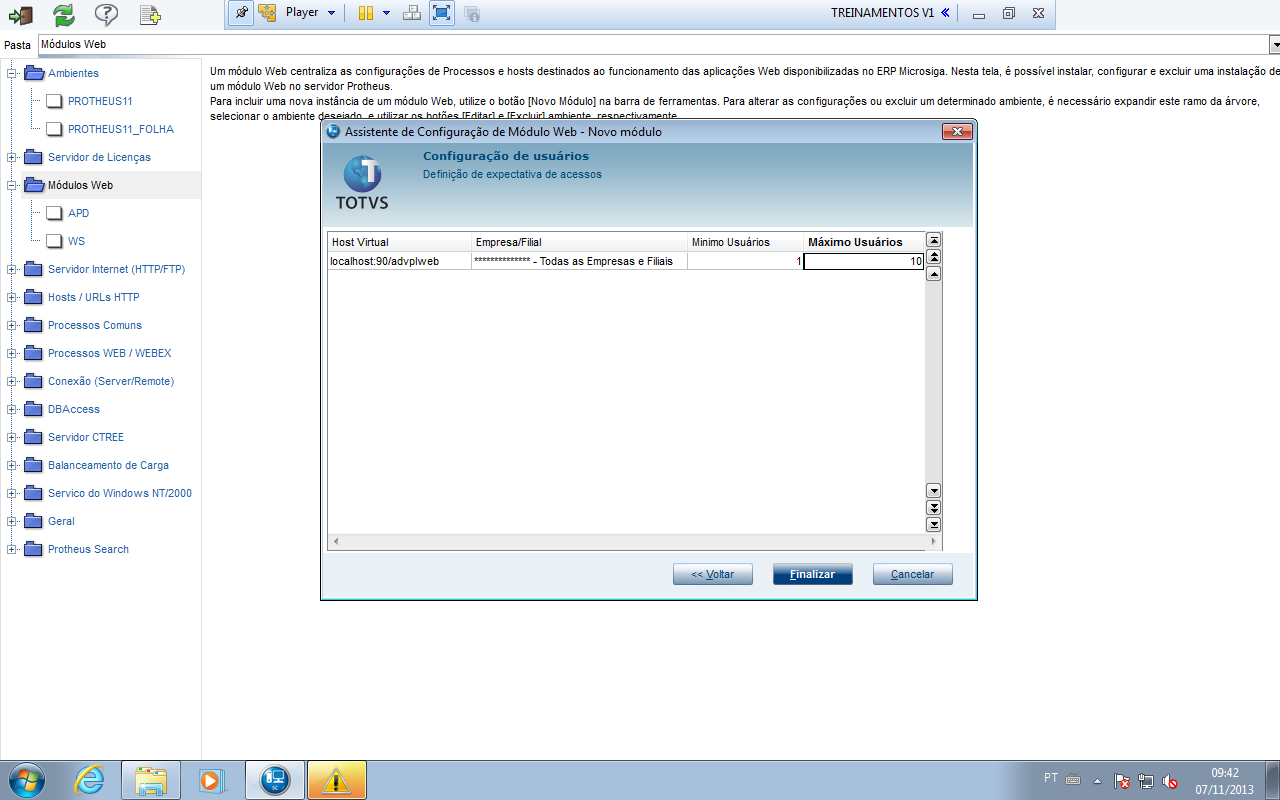
Para subir o portal é necessário:

|  |  |
| --- | --- |
| 1. | Abre o Wizard localizado na pasta smartclient |

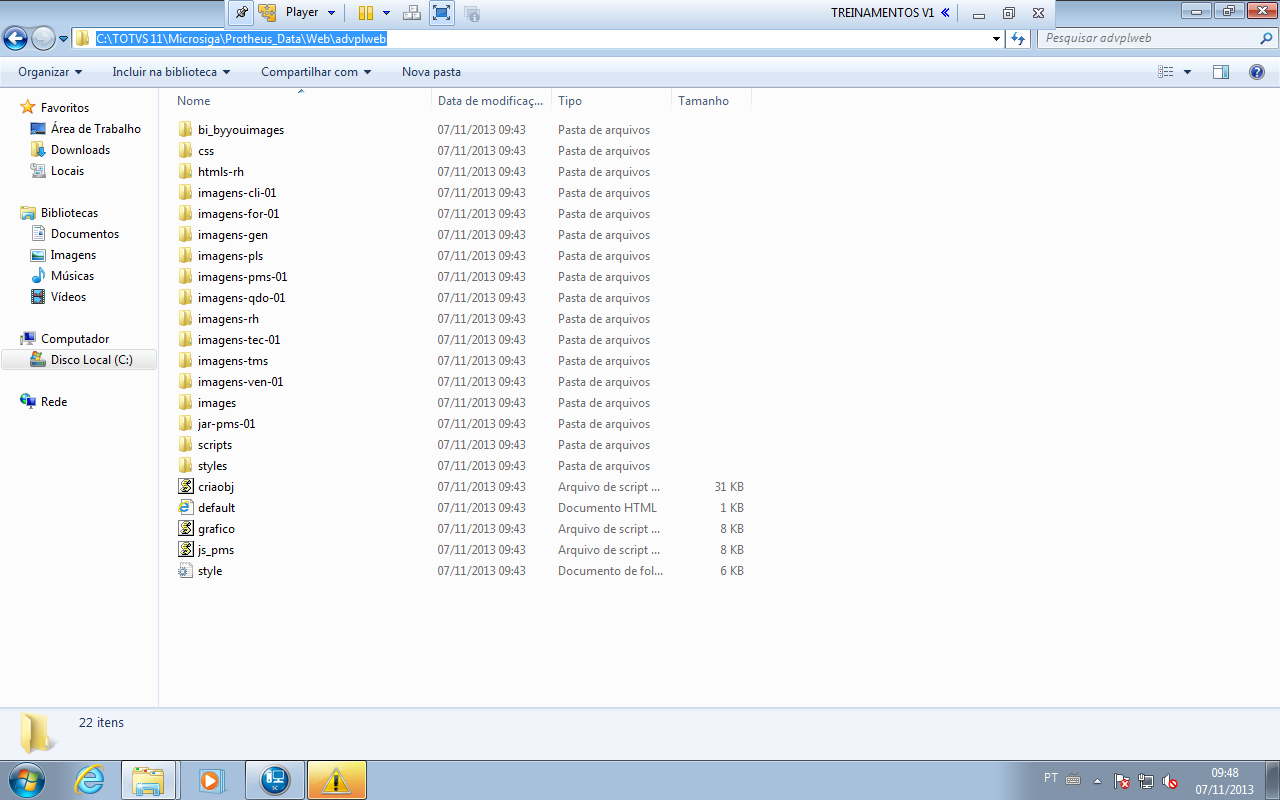
2.Acessar Modulo Web





3º Quantidades de acessos simultâneo

Após criar o portal iremos verificar se criou nossa pasta no diretório do RootPath\web\



Agora iremos verificar o que foi criado no appserver.ini e parar o serviço e rodar em modo console.

Colocar no APPSARVER.INI

Refreshrate = 120

Iremos verificar se vai funcionar nosso portal padrão.

E fazer o teste ex:<http://localhost:81/pp/time.apl>

Montar um fonte e fazer das extensões .APW e .APH

Exemplo1.APW

#INCLUDE "PROTHEUS.CH"

#INCLUDE "APWEBEX.CH"

User Function Exemplo1()

Local cret := ""

Private cTitulo := "Pagina 01"

Private aDados := {}

Web Extended Init cRet // Estrutura de função para Web

For i := 1 to 10

aadd( aDados, { "Nome", Date(), Time() } )

Next i

cRet := H\_Exemplo1() // Chamada da funcao do APH

Web Extended End

Return(cRet)

Exemplo1.APH

<HTML>

<HEAD>

<TITLE>

<%= cTitulo %>

</TITLE>

</HEAD>

< BODY>

<TR>

<% For n := 1 To Len(aDados) %>

<Td> <%= aDados[n,1] %> </Td>

<Td> <%= aDados[n,2] %> </Td>

<Td> <%= aDados[n,3] %> </Td>

<BR>

<% Next n %>

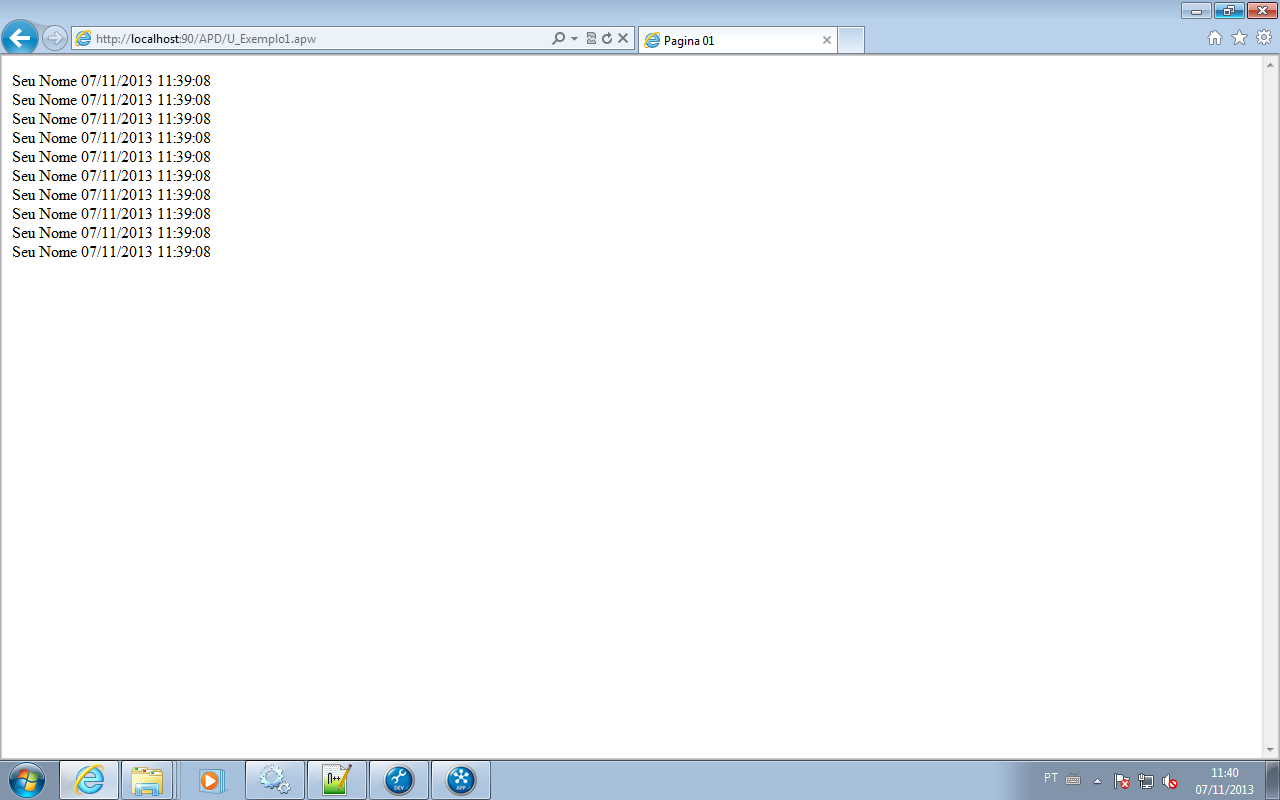
</TR>

</BODY>

</HTML>

Toda estrutura para desenvolvimento vai possuir dois fontes APW chamando as estrutura de Web do APH.

Para testar o que foi feito iremos acessar o link: http://localHost:90/APD/U\_Exemplo1.apw



Falar da diferencia

*<%= %> Aparecer na tela*

*<% %> Não aparece na tela*

**Exemplos**

**Não pode abrir duas estrutura na mesma linha ou abrir e fechar a mesma estrutura na linha de baixo**

**CERTO**

|  |
| --- |
| <% IF !lOk nErro++ Endif %> |

**CERTO**

|  |
| --- |
| <% IF !lOk %> <% nErro++ %> <% Endif %> |

**CERTO**

|  |
| --- |
| <% IF !lOk ; nErro++ ; Endif %> |

**ERRADO**

|  |
| --- |
| <% IF !lOk %><% nErro++ %> -- 2 aberturas e fechamentos na mesma linha <% Endif %> |

2. Quanto aos delimitadores de avaliação <%= ... %> , podemos ter várias aberturas e fechamentos na mesma linha , porém não podemos ter uma abertura e seu respectivo fechamento em linhas diferentes.

3. Uma linha qualquer em um arquivo .APH nao deve conter mais do que 150 Caracteres, pois o Compilador insere caracteres de controle em cada linha do mesmo durante a pré-compilação . e a linha final resultante não pode ultrapassar 254 caracteres, pois neste caso isto impossibilita a compilação do APH.

Agora montar o case para os alunos ver funcionando o projeto.

Seguir a Apostila

**Exemplos**

METHOD="GET"

**Como Funciona o Método *GET***

O método *GET* utiliza a própria *URL* para enviar dados ao servidor, quando enviamos um formulário pelo método *GET*, o navegador pega as informações do formulário e coloca junto com a *URL*

*Exemplo:* Quando você busca algo no Google, ele faz uma requisição utilizando o método *GET*, você pode ver na barra de endereço do seu navegador que o endereço ficou com um ponto de interrogação no meio, e depois do ponto de interrogação você pode ler, dentre outros caracteres, o que você pesquisou no Google.

Metodo GET é utilizado sempre que a requisição que você faz não altere o estado do servidor, tipo uma consulta, uma busca ou qualquer outra coisa que, que não alterar o estado do servidor.

METHOD="POST"

**Como Funciona o Método *POST***

O método *POST* envia os dados colocando-os no corpo da mensagem os dados no servidor.

Ficando com a URL separada dos dados que serão enviados e com isso podemos enviar qualquer tipo de dados por esse método

Exemplo: Utilizamos o método POST sempre que vamos alterar alguma coisa no servidor,