

Análise e Desenvolvimento de Sistemas

Desenvolvimento de Software para Internet

2º Semestre/2019

Prof.º Eduardo Zied Milian

eduardo.milian@docente.unip.br

Análise e Desenvolvimento de Sistemas

CURSO: Tecnologia em Análise e Desenvolvimento de Sistemas

SÉRIE: 4º período (semestral)

DISCIPLINA: DESENVOLVIMENTO DE SOFTWARE PARA INTERNET

CARGA HORÁRIA SEMANAL:

03 horas/aula

CARGA HORÁRIA SEMESTRAL:

60 horas/aula

Desenvolvimento de Software para Internet

Plano de Ensino da Disciplina

I – Ementa

Conceito de desenvolvimento de aplicativos para web utilizando HTML e ASP.NET.

utilizando:

- ☐ *ADO.NET.*
- ☐ *AJAX.*
- ☐ *MVC. XML.*
- ☐ *ASP.NET e*
- ☐ *formulários Windows.*

Desenvolvimento de Software para Internet

Plano de Ensino da Disciplina

II – Objetivos

- ☐ Apresentação dos conceitos básicos do desenvolvimento de aplicações web.
- ☐ Apresentação da linguagem ASP.NET.
- ☐ Capacitação para a implementação aplicações (*sites*) utilizando a linguagem ASP.NET

Desenvolvimento de Software para Internet

Plano de Ensino da Disciplina

III – Conteúdo Programático

1. Conceitos básicos do desenvolvimento de aplicações *web*
2. Introdução ao *HTML 5*
3. Visão geral do *Microsoft .Net Framework*
4. Uso de Trace e Debug em aplicações *ASP.NET*
5. Validação de entrada de dados
6. Acesso a dados relacionais utilizando o *Microsoft Visual Studio .NET*
7. *ASP.NET MVC*
8. Acessando dados com *MS ADO.NET*

Critério de Promoção (Avaliação)

A avaliação será por meio de provas, trabalhos, estudos de caso, dentre outros, bem como pela participação do aluno durante as aulas e demais atividades, em conformidade com o *Plano de Ensino*. Serão feitas avaliações, assim distribuídas:

- Duas Notas do Professor (NP) para as atividades curriculares, com peso 4 (quatro) cada uma, na composição da nota semestral de cada disciplina;
- Uma nota referente ao Projeto Integrado Multidisciplinar (PIM), com peso 2 (dois) no cálculo da Média Semestral (MS) das disciplinas. Esse Projeto será desenvolvido durante o semestre.

A **MS** será: **$(NP1 \times 4 + NP2 \times 4 + PIM \times 2) / 10$** . Para a aprovação, a MS deverá ser igual ou superior a 5,0; é exigida a frequência mínima de 75%. O desempenho do aluno é avaliado numa escala de 0 (zero) a 10 (dez).

Bibliografia para a Disciplina

Bibliografia Básica

- DURÃES, R. Desenvolvendo para web usando o Visual Studio 2008. Rio de Janeiro: Brasport, 2011.
- LOTAR, A. Programando com ASP.NET MVC. São Paulo. Novatec, 2011.
- MANZANO, J. A. N.; TOLEDO, S. A. Guia de Orientação e Desenvolvimento de Sites - HTML, XHTML, CSS e JavaScript/JScript. 2. ed. Erica 2010.

Bibliografia Complementar

- BORGES JUNIOR, M. P. ASP.NET utilizando C# : de programador para programador. Rio de Janeiro: Ciencia Moderna, 2006.
- PROFFITT, B.; ZUPAN, A. XHTML: desenvolvimento Web. São Paulo: Makron Books, 2001.
- SHEPHERD, G. Microsoft ASP.NET 2.0: passo a passo. Porto Alegre: Bookman, 2008..
- CAMARA, F. Orientação a objeto com .NET. 2.ed. Santa Catarina: Visual Books. 2006.

Análise e Desenvolvimento de Sistemas

Desenvolvimento de Software para Internet

**** DEFINIÇÃO DOS GRUPOS:*

*– até 3 alunos - para os trabalhos em sala ****

HTML (1)

O que é o HTML ?

Hyper **T**ext **M**arkup **L**anguage

Linguagem de marcação de hipertexto

O que é hipertexto?

HTML (2)

Hipertexto são conjuntos de elementos (nós) ligados por conexões:

- ✓ palavras
- ✓ imagens
- ✓ vídeos
- ✓ áudio
- ✓ documentos
- ✓ entre outros...

HTML (3)

Navegadores interpretam HTML

O HTML possui o conteúdo do texto e marcadores (tags)

Os tags especificam formatos do texto e comandos de apresentação

HTML (4)

Diferentes navegadores nem sempre interpretam da mesma forma o HTML

Necessidade de padronização: w3c

Há no entanto as empresa que não seguem os padrões: Microsoft com o *Internet Explorer* (isto também ocorre com Microsoft *Edge*?)

Portanto cuidado!

Podem ocorrer diferenças entre *browsers*!

HTML (5)

O primeiro passo do projeto é a criação do leiaute das páginas

Para criar uma página HTML é necessário um editor de texto

O editor mais simples que pode ser utilizado é o *Notepad* do Windows.

Ao salvar o arquivo deve-se lembrar de colocar a extensão .htm ou .html para o navegador entender.

HTML (6)

A desvantagem do uso de editores simples como o *Notepad* é a falta de apoio específico para a linguagem.

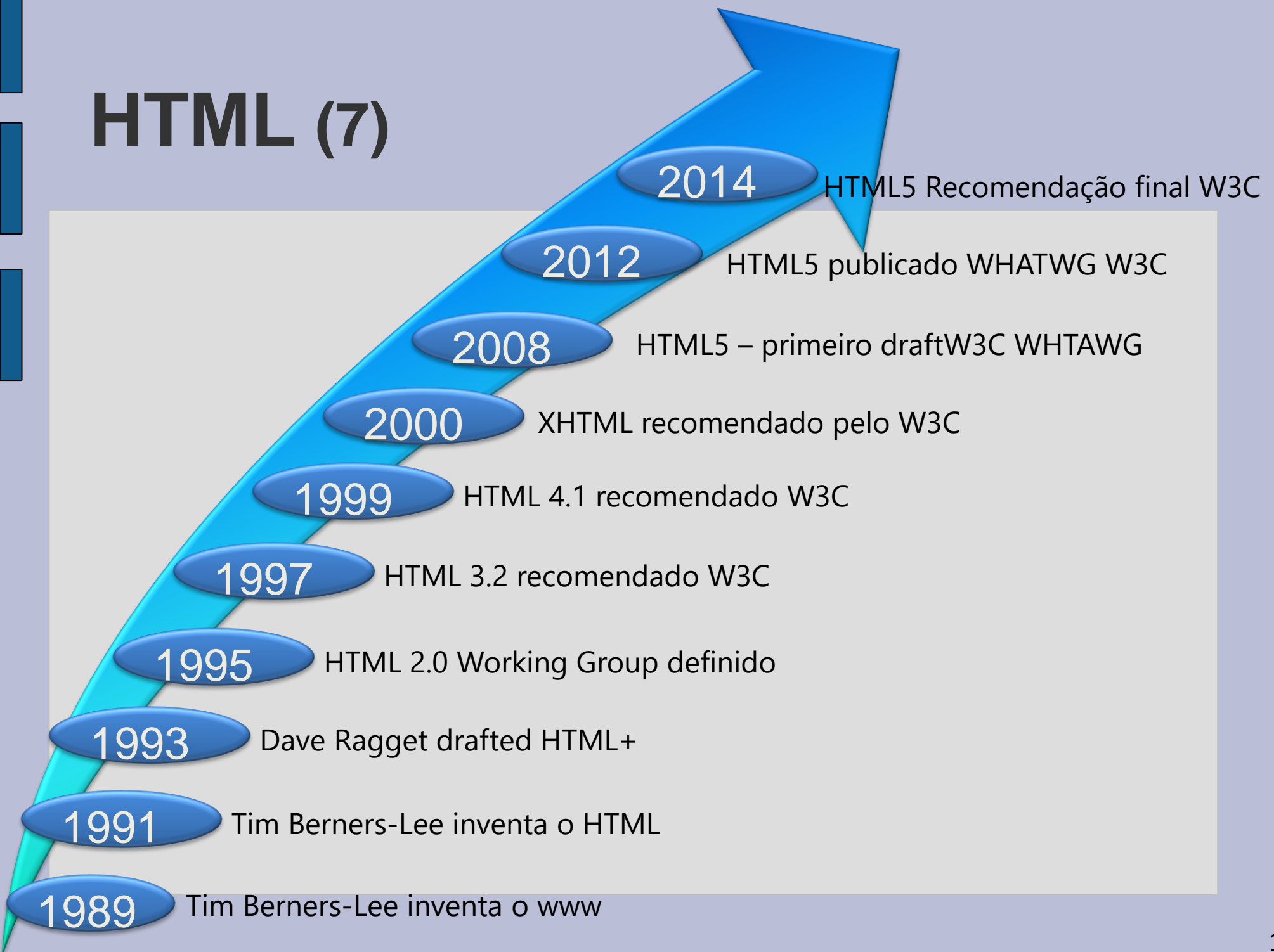
Recomenda-se utilizar o *Notepad++*, cujo link é <https://notepad-plus-plus.org/>.

Ou o *PSPad* cujo link é <http://www.pspad.com/en/download.php>.

Atenção: não utilize o Microsoft Word!

O Word armazena dentro do arquivo uma série de informações que não são visíveis e portanto vai atrapalhar.

HTML (7)



Tarefas

Atenção!

A partir de hoje, todas as aulas possuem trabalhos a serem entregues, realizados na própria aula

As instruções estarão nos slides das aulas

As entregas são feitas enviando um e-mail para o professor no endereço:

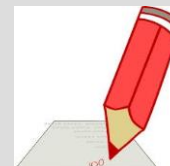
eduardo.milian@docente.unip.br

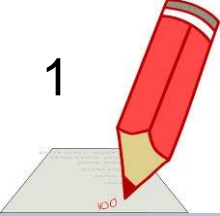
Todo o conteúdo do projeto deverá ser zipado e anexado ao e-mail de envio da entrega

Todos os trabalhos são individuais

Mesmo fazendo em dupla cada um deve fazer o envio.

Os entregáveis são marcados com o logo do lápis:





Novo Projeto no Visual Studio 2010

Após **iniciar o Visual Studio**, você deve ir ao menu “*File → New Project*”, onde existe a opção de iniciar um novo projeto usando ***Visual C#***, depois escolher o modelo de projeto que, no nosso caso, será ***Web***, conforme a Figura 8, escolhendo a opção ***ASP.NET Empty Web Application***.

Criando um Novo Projeto

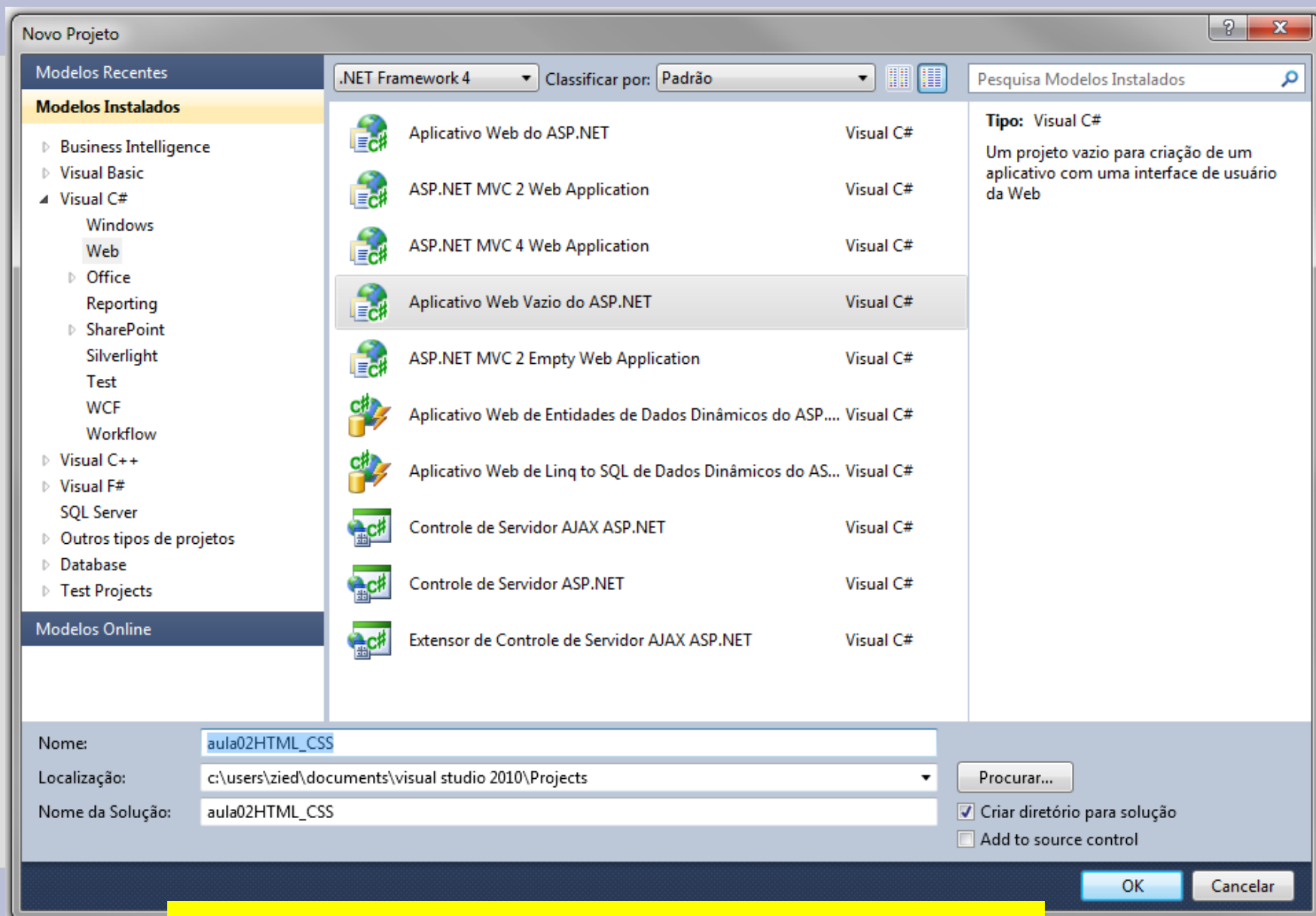
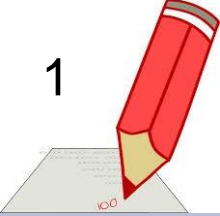
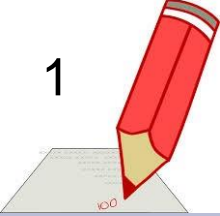


Figura 1 – Novo Projeto



Novo Projeto no Visual Studio 2010

- Nomear o projeto como “**aula02HTML_CSS**” e confirmar a sua criação.
- Vai aparecer uma tela com a visão do *ToolBox* (caixa de ferramentas) a esquerda e do *Solution Explorer* (Estrutura da solução) à direita.
- Caso eles não estejam aparecendo, vá ao *menu View → Solution Explorer* e ao *menu View → Toolbox*.



Primeiros passos - Novo Projeto no Visual Studio 2010

Para criar uma página HTML

- Como visto na **Figura 2**, clicar com o botão direito em cima de "aula02HTML_CSS" → *Add* → *New Item...*
- Abrirá um nova janela, conforme **Figura 3**, escolher "Página HTML"
- Preencher o campo "*Name:*" com "*hello_world.htm*" e clicar em "*Add*".

Primeiros Passos

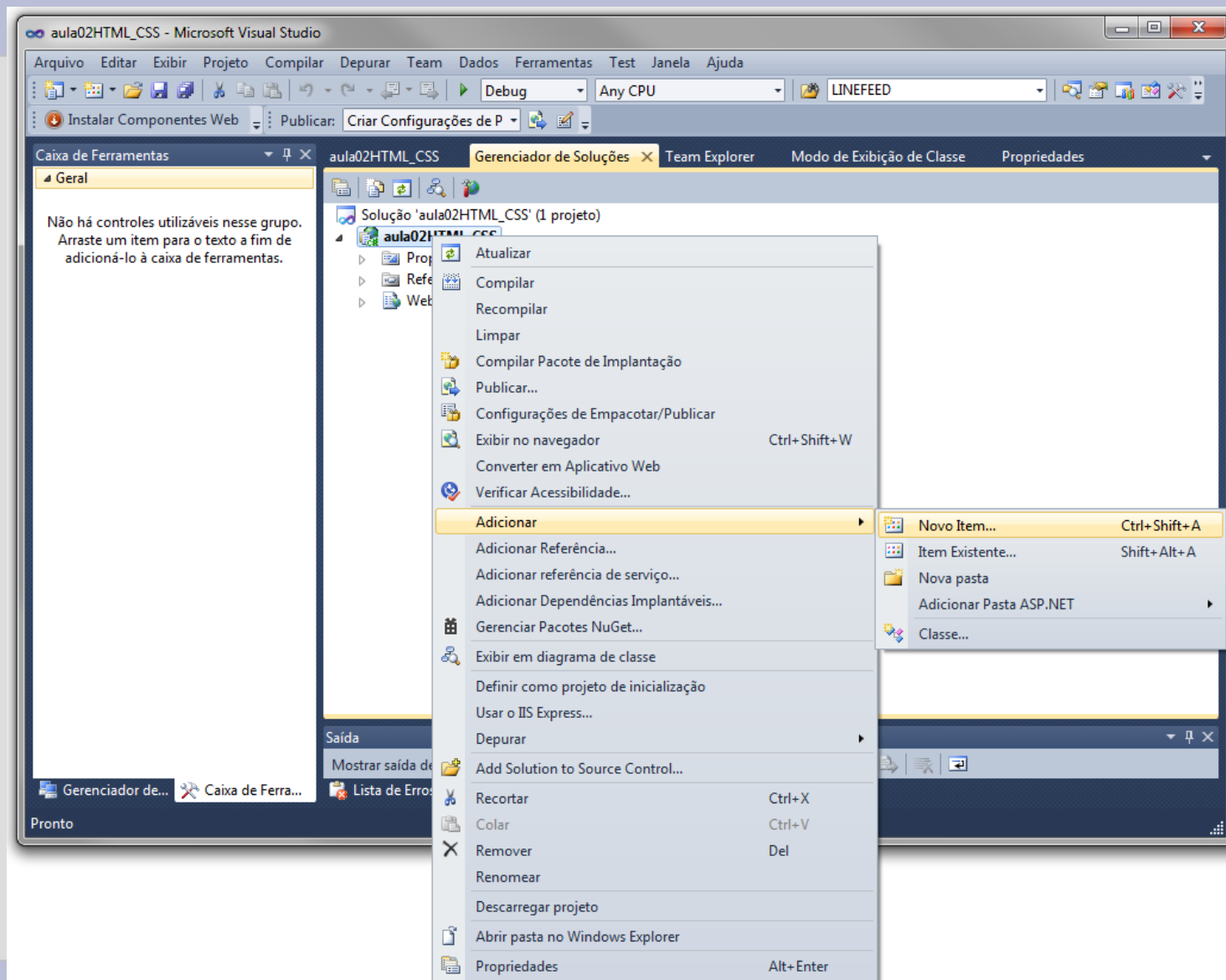


Figura 2 – Adicionar um novo item...

Primeiros Passos

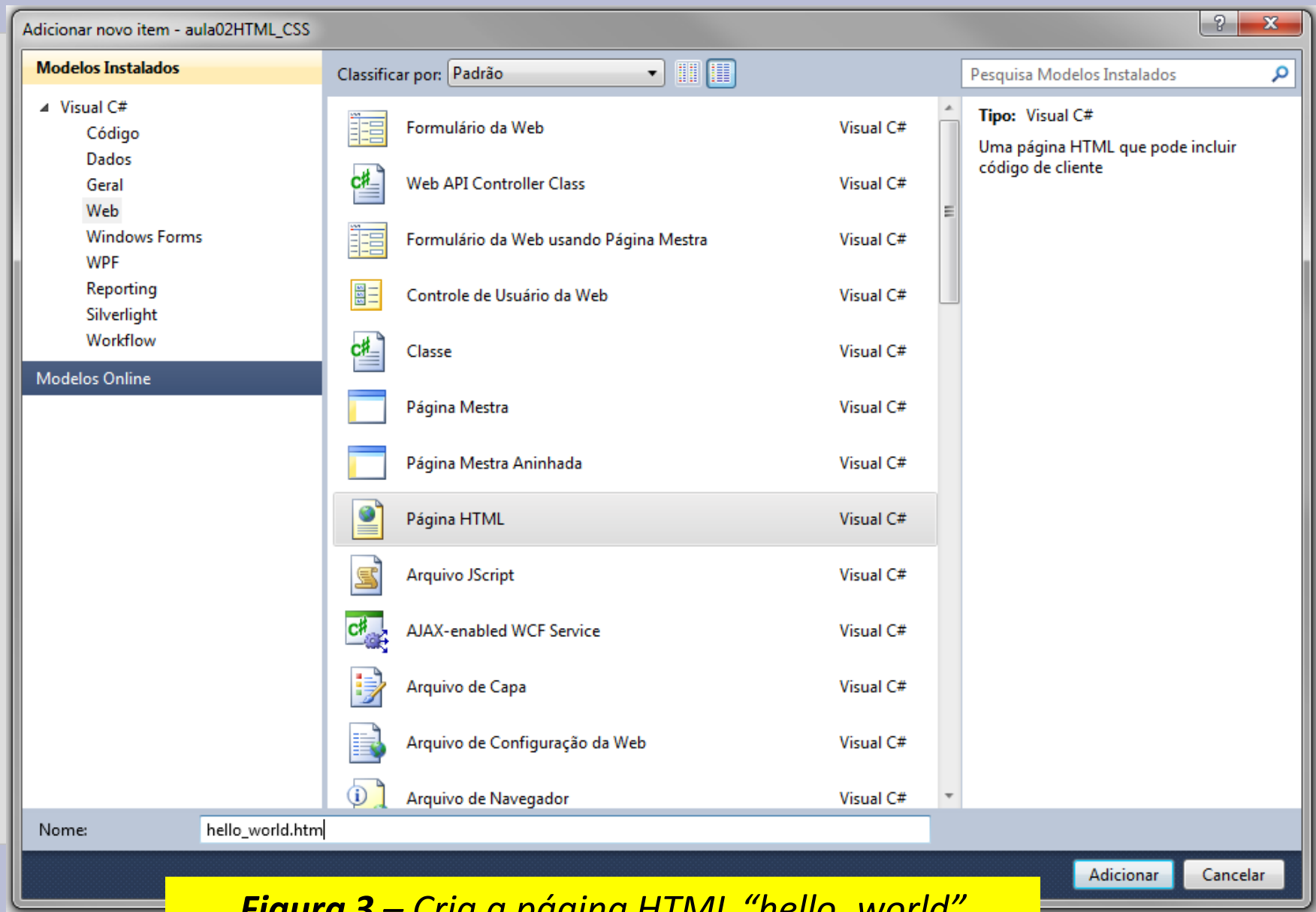
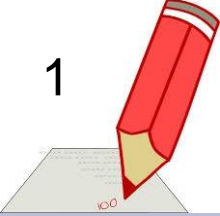


Figura 3 – Cria a página HTML “hello_world”



Primeiros passos – RESUMO

Iniciar o VISUAL STUDIO 2010

New Project...

Em "Installed Templates" → Visual C# → Web → ASP.NET Empty Web Application

Name: " aula02HTML_CSS"

Location: "c:\users\ \<<***user***>>\documents\visual studio 2010\Projects"

Solution name: " aula02HTML_CSS"

Manter SETADO: *"Create Directory for Solution"*

Manter RESETADO: *"Add to source control"*

Clicar em: <<<*** OK ***>>>

Botão direito em cima de " aula02HTML_CSS" → add → new item

→ HTML Page → Name: "hello_world.htm"

Clicar em: <<<*** Add ***>>>

Estrutura do HTML

Padrões definidos pelo
W3C – World Wide Web Consortium

- <http://www.w3.org/>
- <http://www.w3.org/Consortium/>



Estrutura do HTML (2)

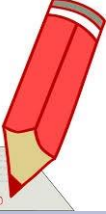
Sintaxe básica dos comandos:

`<marcador> conteúdo </marcador>`

Exemplo:

```
<html>
<head>
  <title>Título da página</title>
</head>
<body>
  <h1> Página simples HTML </h1>
  <p> Hello World! Minha primeira página HTML! </p>
  <p> parágrafo comum </p>
</body>
</html>
```

**Obs.: salvar no arquivo
"hello_world.htm"**



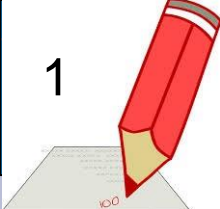
Estrutura do HTML (3)

Para que a página seja apresentada é necessário “publicar” na internet:

- 1) Botão direito na página que será exibida/testada.
- 2) Clicar na seta verde do lado esquerdo da expressão “*debug*” ou “*Debug*” → “*Start Debugging*”.
- 3) Verificar o resultado.
- 4) Para continuar o desenvolvimento é preciso parar a depuração em “*Debug*” → “*Stop Debugging*”.

Atenção DEVE ser ABERTO com o WINDOWS EXPLORER:

(Não é → file:///C:/Users/<<**user**>>/documents/
visual studio 2010/Projects/aula02HTML_CSS/hello_world.htm).



Para testar a Página “*hello_word.htm*”

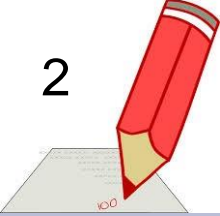
Outra maneira para testar sua página:

Na aba **Gerenciador de Soluções**

- Selecionar o arquivo “*hello_world.htm*”
- Acionar a tecla “**F5**” (Iniciar a Depuração)
- **Ver o resultado**
- Modificar o conteúdo da linha de
<p> parágrafo comum </p>, para
<h2> *header* do tipo 2 </h2> no arquivo
“*hello_world.htm*” e salvar.
- **Ver o resultado.**

Estrutura do HTML (4)

- texto ente `<html>` e `</html>` descreve a página web
- texto ente `<head>` e `</head>` descreve o cabeçalho
- texto ente `<title>` e `</title>` descreve a aba
- texto ente `<body>` e `</body>` é a parte visível do conteúdo
- texto ente `<h1>` e `</h1>` é apresentado como cabeçalho (header)
- texto ente `<p>` e `</p>` é um parágrafo



link

Atributo href – define um link

Aqui é apontado um site:

```
<a href="http://www.unip.br">Este é o link do UNIP </a>
```

Adicionar um novo arquivo denominado teste.htm com conteúdo *html* e apontá-lo aqui:

```
<a href='teste.htm'>Este é um link interno</a>
```

Pode ser colocado uma *tag* `
` entres o links acima para pular uma linha.

Observação: as aspas podem ser “ ou ‘ (simples ou duplas)

Atenção !

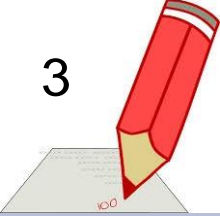
*Ao receber uma página, o navegador possui uma memória denominada **cache** que é verificada se ele já foi visitada.*

Caso positivo, é apresentada a página local e o navegador não recebe a página do servidor.

Isso é feito para economizar tráfego na rede.

Para desenvolvimento isso é ruim pois há o risco de uma nova página não ser apresentada!

*Para contornar isso, abrir a página com **Ctrl+F5***



imagens

1) Colocar uma imagem no projeto → parar a depuração (“debug”) → Adicionar → Item existente... (escolher uma imagem na Internet)

2) Publicá-la com o comando:

```

```

3) Alterar o tamanho da imagem

```

```

4) Modificar o nome do arquivo de imagem no diretório

```

```

O que ocorre quando o navegador não encontra o arquivo?

imagens

```

```

 define uma imagem

<map> define o mapeamento de uma imagem (vide componente 6)

<area> define uma área “clicável” dentro de uma imagem

Exemplo:

```
<!DOCTYPE html>
```

```
<body>
```

```
<h2><Minha foto preferida</h2>
```

```

```

```
</body>
```

```
</html>
```

Atributos:

src – endereço da imagem

alt – texto alternativo se a figura não for carregada

Comandos

Break (nova linha)

`
` é um elemento vazio sem o *tag* de fechamento e define um break. Nas versões futuras será necessário usar o fechamento `</br>`

Letras maiúsculas e minúsculas

O html não é sensível a maiúsculas ou minúsculas. No entanto é recomendável que seja usadas somente letras minúsculas pois a linguagem será no futuro sensível a essa característica.

Linha horizontal

`<hr>`

Comentários

`<!-- Este é um comentário -->`

Atributo de Tabelas

<u><table></u>	define uma tabela
<u><th></u>	define o cabeçalho de uma tabela
<u><tr></u>	define a linha de uma tabela
<u><td></u>	define uma célula de uma tabela
<u><caption></u>	define a captura de uma tabela
<u><colgroup></u>	define um grupo de colunas de uma tabela
<u><col></u>	atribui valores a uma ou mais colunas
<u><thead></u>	define o cabeçalho de uma tabela
<u><tbody></u>	define o corpo da tabela
<u><tfoot></u>	define o rodapé da tabela

Atributo de Tabelas

```
<!DOCTYPE html>
<body>
```

```
<table width="100%" border="1">
```

```
<tr>
```

```
  <th>ISBN</th>
```

```
  <th>Title</th>
```

```
  <th>Price</th>
```

```
</tr>
```

```
<tr>
```

```
  <td>3476896</td>
```

```
  <td>My first HTML</td>
```

```
  <td>$53</td>
```

```
</tr>
```

```
<tr>
```

```
  <td>2489604</td>
```

```
  <td>My first CSS</td>
```

```
  <td>$47</td>
```

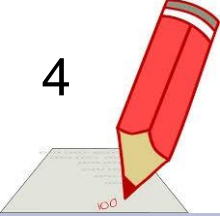
```
</tr>
```

```
</table>
```

```
</body>
```

```
</html>
```

ISBN	Title	Price
3476896	My first HTML	\$53
2489604	My first CSS	\$47



Criação de tabela

Criar uma tabela com 4 colunas e pelo menos duas linhas

As colunas são:

- RA
- Nome
- Sobrenome
- Número do celular

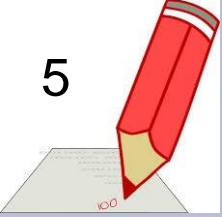
Listas

<u><table></u>	define uma tabela
<u></u>	define uma lista ordenada (com numeração)
<u></u>	define uma lista sem numeração mas com <i>tick marks</i>
<u></u>	define um item de lista
<u><dl></u>	estabelece a definição de uma lista
<u><dt></u>	define um termo (um item) na definição de uma lista
<u><dd></u>	estabelece a descrição de um termo em uma lista de definição

Listas

```
<dl>  
<dt>Café</dt>  
<dd>Bebida preta quente</dd>  
<dt>Leite</dt>  
<dd>Bebida branca fria</dd>  
</dl>
```

```
<ol>  
<li>Café</li>  
<li>Chá</li>  
<li>Leite</li>  
</ol>
```



Listas

Criar uma lista com a relação de disciplinas que vocês estão fazendo no momento

Formulário

<u><form></u>	define um formulário para entrada do usuário
<u><input></u>	define um campo de entrada
<u><textarea></u>	define uma área de texto (controle de entrada multilinhas)
<u><label></u>	define o rótulo de um controle
<u><fieldset></u>	define um conjunto de campos
<u><legend></u>	define a captura para um conjunto de campos
<u><select></u>	define uma lista de seleção (<i>drop-down box</i>)
<u><optgroup></u>	define um grupo de opção
<u><option></u>	define uma opção no grupo de opção
<u><button></u>	define um botões tipo “ <i>push button</i> ”

Formulário – Exemplo GET

```
<body>
<p> Exemplo de entrada de dados com o método HTTP GET </p>
<form name="input" action="exemplo_recebe_get.aspx" method="get">
  Username:
  <input type="text" name="user" />
  <input type="submit" value="entra" />
</form>
</body>
```

Username:

**Obs.: Adicionar o arquivo “*exemplo_get.htm*”,
copiar e salvar o trecho acima.**

Formulário – Recepção do dado

Para a recepção do dado é necessário que haja um arquivo com linguagem de programação. No caso será usado um arquivo .aspx indicando que nele há código em linguagem *Visual C#*.

Para criar uma página .ASPX

- Como visto na **Figura 2**, clicar com o botão direito em cima de "aula02HTML_CSS" → *Add* → *New Item...*
- Abrirá uma nova janela, conforme **Figura 3**, e escolher "*Web Form*".
- Preencher o campo "*Name:*" com "exemplo_recebe_get.aspx" e clicar em "*Add*".

Formulário – Recepção do dado

```
<!DOCTYPE html >  
  
<html xmlns="http://www.w3.org/1999/xhtml">  
<head runat="server">  
    <title> Recepção de dados com GET </title>  
</head>  
<body>  
<p> Seja bemvindo! </p>  
<%  
    string fusuario = Request.QueryString["user"];  
    Response.Write("O usuário digitado foi " + fusuario + ". <br>");  
%>  
</body>  
</html>
```

**Obs.: Copiar salvar no arquivo
“*exemplo_recebe_get.aspx*”**

Formulário – exemplo POST

```
<body>
<p> Exemplo de entrada de dados com o método HTTP POST </p>
<form name="input" action="exemplo_recebe_post.aspx" method="post">
  Nome: <input type="text" name="nome" /> <br />
  Cidade: <input type="text" name="cidade" />
         <input type="submit" value="entra" />
</form>
</body>
```

Exemplo de entrada de dados com o método HTTP POST

Nome:

Cidade:

Obs.: Salvar no arquivo
"exemplo_post.htm"

Formulário – recepção do dado

```
<!DOCTYPE html >

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title> Recepção de dados com POST </title>
</head>
<body>

<p> Seja bem vindo! </p>
<%
    string fnome = Request.Form["nome"];
    string cidade = Request.Form["cidade"];
    Response.Write("Prezado " + fnome + ". <br>");
    Response.Write("Desejo que você seja feliz em " + cidade + "!<br>");
%>
</body>
</html>
```

**Obs.: Copiar e salvar no arquivo
“exemplo_recebe_post.aspx”**

Formulário – exemplo (1)

```
<form>
```

I have a bike:

```
<input type="checkbox" name="vehicle" value="Bike" />
```

```
<br />I have a car:
```

```
<input type="checkbox" name="vehicle" value="Car" />
```

```
<br />
```

I have an airplane:

```
<input type="checkbox" name="vehicle" value="Airplane" />
```

```
</form>
```

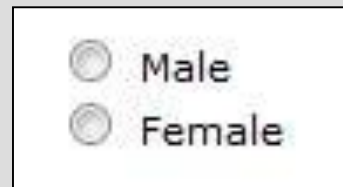
I have a bike: ☐

I have a car: ☐

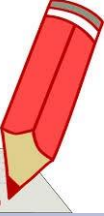
I have an airplane: ☐

Formulário – exemplo (2)

```
<form>  
  <input type="radio" name="sex" value="male" /> Male  
  <br />  
  <input type="radio" name="sex" value="female" /> Female  
</form>
```



☒ Male
☐ Female

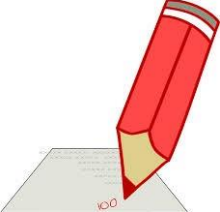


Formulário

Escolher um dos formulários anteriores e fazer as alterações para a entrada dos seguintes dados:

- Nome
- RA
- Idade

Criar uma página ASPX para recepcionar os dados.



Tarefas de hoje



1. Criação da primeira página HTML
2. Página com link
3. Página com imagem
4. Tabela
5. Lista
6. Formulário de entrada de dados

Validação

Um site publicado pode ser validado no W3C com relação ao atendimento às normas definidas por esta instituição

<http://validator.w3.org/>

Nestes site é colocado o endereço web do site e, após análise, é emitido um relatório de erros

Referências

Cursos grátis de HTML:

<http://www.w3schools.com/html/default.asp>

Análise e Desenvolvimento de Sistemas

Desenvolvimento de Software para Internet

2º Semestre/2019

Prof.º Eduardo Zied Milian

eduardo.milian@docente.unip.br