

INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
RIO GRANDE DO NORTE

# Desenvolvimento Web

## Introdução a XHTML

Prof. Bruno Gomes

2014



# Introdução

## ▶HTML e XHTML

- ▶ Estrutura básica do documento
- ▶ DTDs
- ▶ Validação do documento

## ▶Exercícios



# HTML e XHTML

## ▶ HTML (Hypertext Markup Language)

- ▶ Padrão para a publicação de documentos na Web
- ▶ Linguagem de marcação simples
- ▶ Camada de informação da página
- ▶ Cada marcação HTML dá significado a informação nela contida

## ▶ XHTML (eXtensible HTML)

- ▶ Esforço para padronizar a notação HTML



# XHTML

## ▶ Melhoria da linguagem HTML

- ▶ Torna a linguagem mais “enxuta”
- ▶ Elimina ambiguidades da notação HTML

É baseada em XML, que é o padrão para *troca* de informações na Web

▶ XHTML = XML + HTML

▶ XML define uma sintaxe (forma) e semântica (significado) das suas *tags* de forma rígida,

- ▶ Reduz as falhas da notação HTML.



# Por que XHTML?

## XHTML separa claramente “informação” e “apresentação”

- ▶ Informação – textos, imagens, vídeos que compõem a sua página
  - ▶ Apresentação ou formatação – A forma como o conteúdo da página é exibido para o usuário
  - ▶ Desenvolvimento cliente: XHTML (informação) + CSS (formatação) + JavaScript (Programação do lado cliente)
- 
- ▶ Todos os navegadores que suportam HTML 4.01, também suportam XHTML
    - ▶ XHTML nada mais é que um HTML reduzido
- 
- ▶ HTML é mais permissivo
    - ▶ Deixar tags abertas
    - ▶ Usar tags para mexer com o visual da página, etc.



# W3C (*World Wide Web Consortium*)

- ▶ Órgão que normatiza algumas das principais linguagens e tecnologias utilizadas na Web
  - ▶ HTML, XML, XHTML, CSS, AJAX, Web Services, etc.
- ▶ No sítio <http://www.w3c.org> você encontrará
  - ▶ Especificações e recomendações sobre essas tecnologias
  - ▶ Documentação
  - ▶ Artigos relacionados a novidades com relação as tecnologias
  - ▶ Programas que validam documentos segundo a especificação W3C (veremos adiante um validador para XHTML)



# Um pouco de HTML para começar

- ▶ Todo arquivo HTML (e também XHTML) deve ser um arquivo de texto puro e possuir a extensão `.htm` ou `.html`
  - ▶ Exemplos: `index.htm`, `index.html`, `futebol.html`, `trabalhos.htm`, etc.
  - ▶ O padrão de “três” letras para a extensão normalmente é “herança” do Dos/Windows, mas o `.html` também é aceito no Windows.
- ▶ Inicialmente, utilizaremos o “bloco de notas” (Windows) ou `gedit` (Linux)

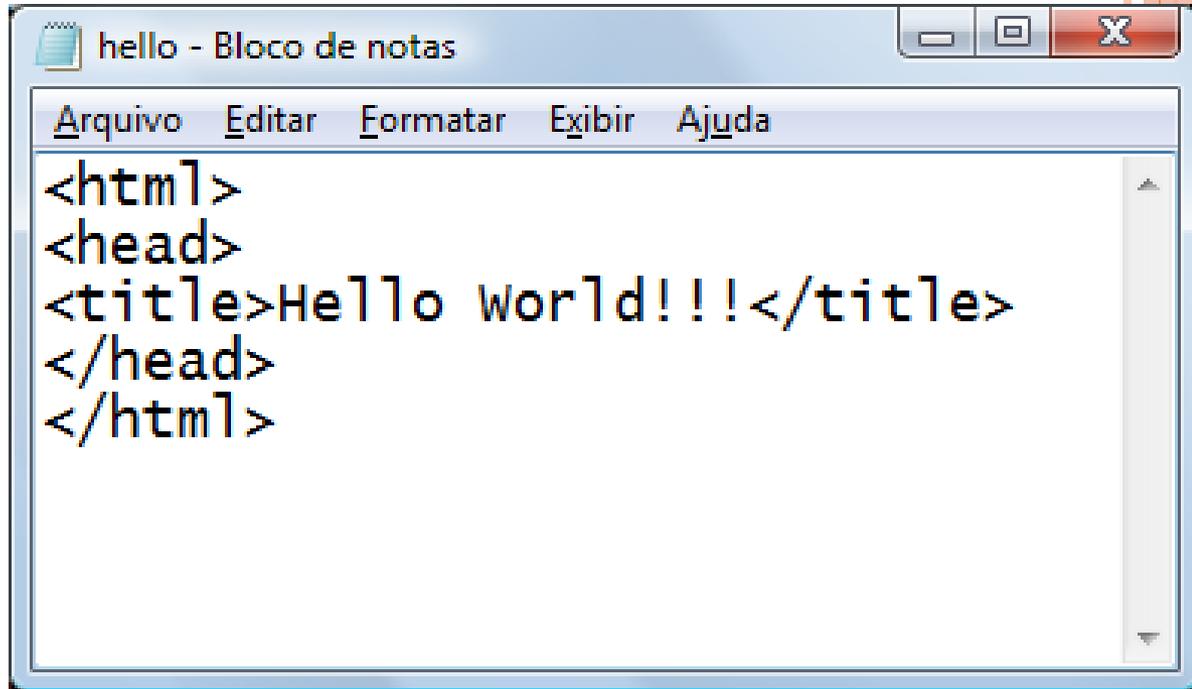


# Criando o Primeiro HTML

►Vejam os passos necessários

1.Abrir o editor de texto

2.Digitar o nosso primeiro HTML (Hello World!!!)

A screenshot of a Windows Notepad window titled "hello - Bloco de notas". The window has a menu bar with "Arquivo", "Editar", "Formatar", "Exibir", and "Ajuda". The text area contains the following HTML code:

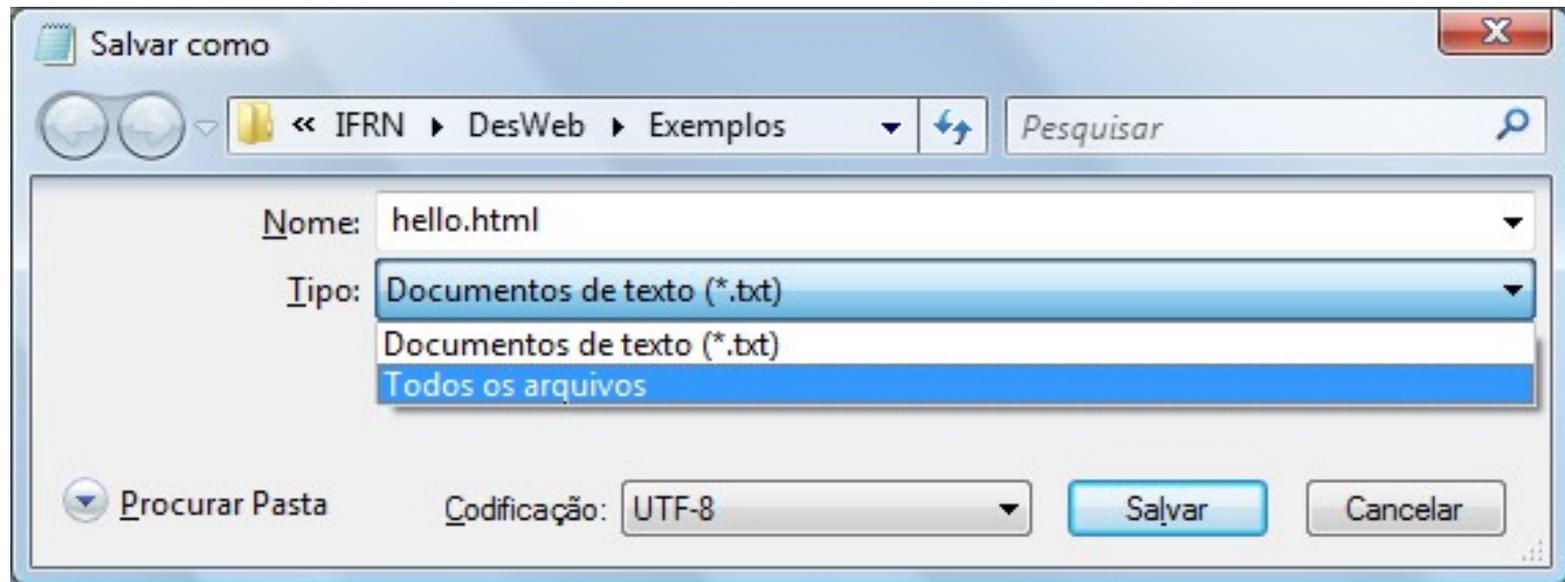
```
<html>
<head>
<title>Hello World!!!</title>
</head>
</html>
```



# Criando o Primeiro HTML

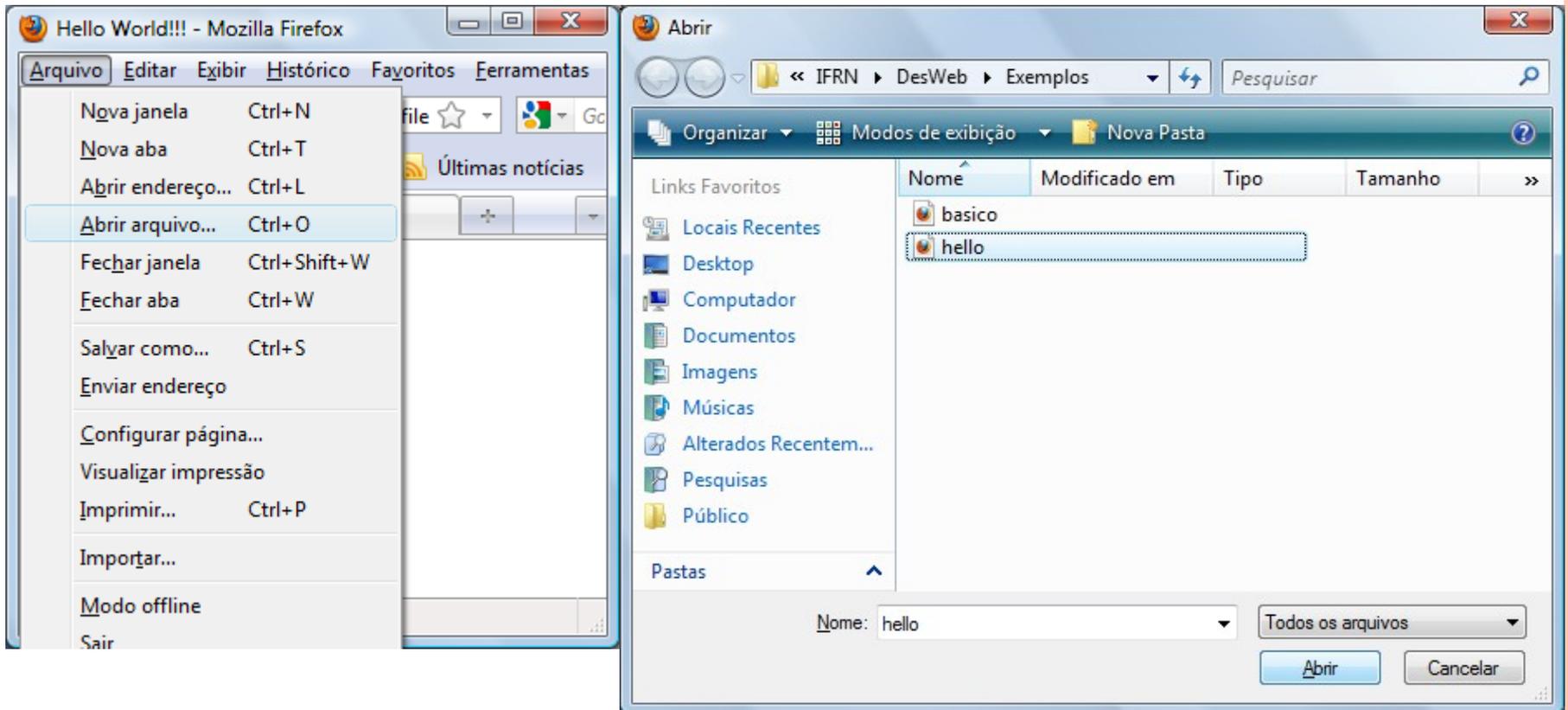
1. Salvar o arquivo com algum *nome* e *extensão* `.html`,  
p. ex.: `hello.html`

**Atenção:** no Windows, não salve o arquivo como texto (`.txt`) e sim como “*todos os arquivos*”



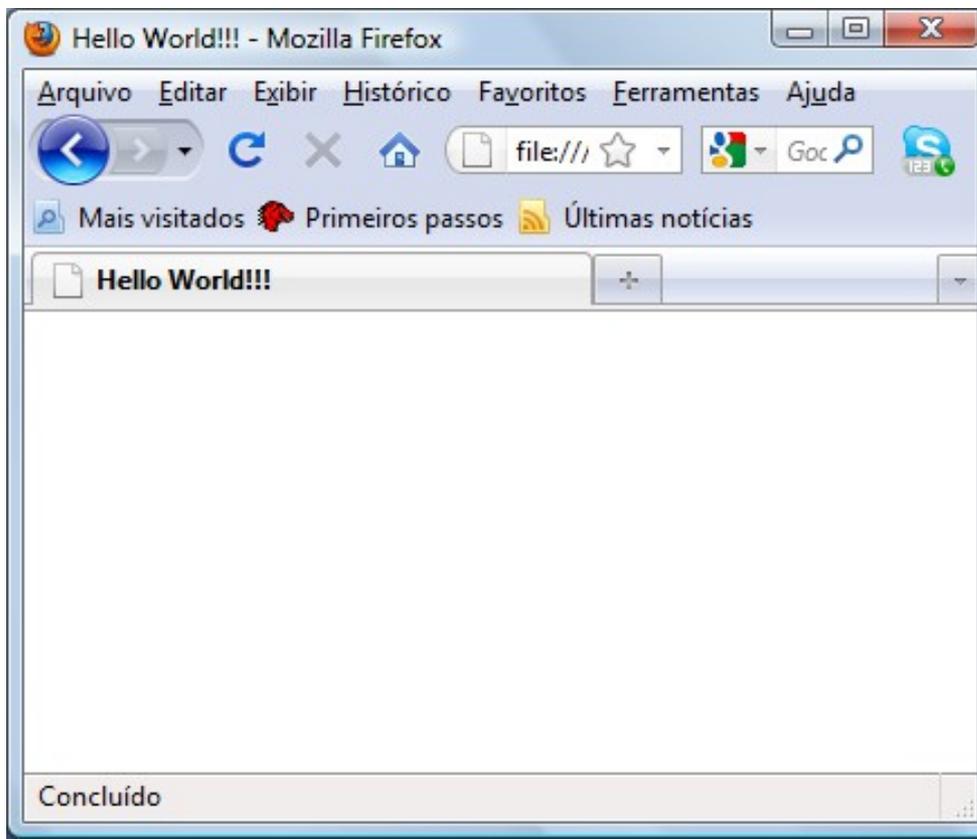
# Criando o Primeiro HTML

1. Após ter salvo o arquivo, abra-o com o seu navegador Internet (Firefox, Opera, Internet Explorer, etc.)



# Primeiro HTML

- ▶ O resultado será o nome “Hello World” aparecendo no título da página
  - ▶ *Nada muito emocionante, mas o que você queria para um primeiro exemplo???*



# Primeiro HTML – O que aconteceu neste primeiro exemplo??

▶ O seu navegador é capaz de interpretar as *tags* HTML digitadas e exibir os resultados dessa interpretação

```
<html>
<head>
  <title>Hello World!!!</title>
</head>
</html>
```

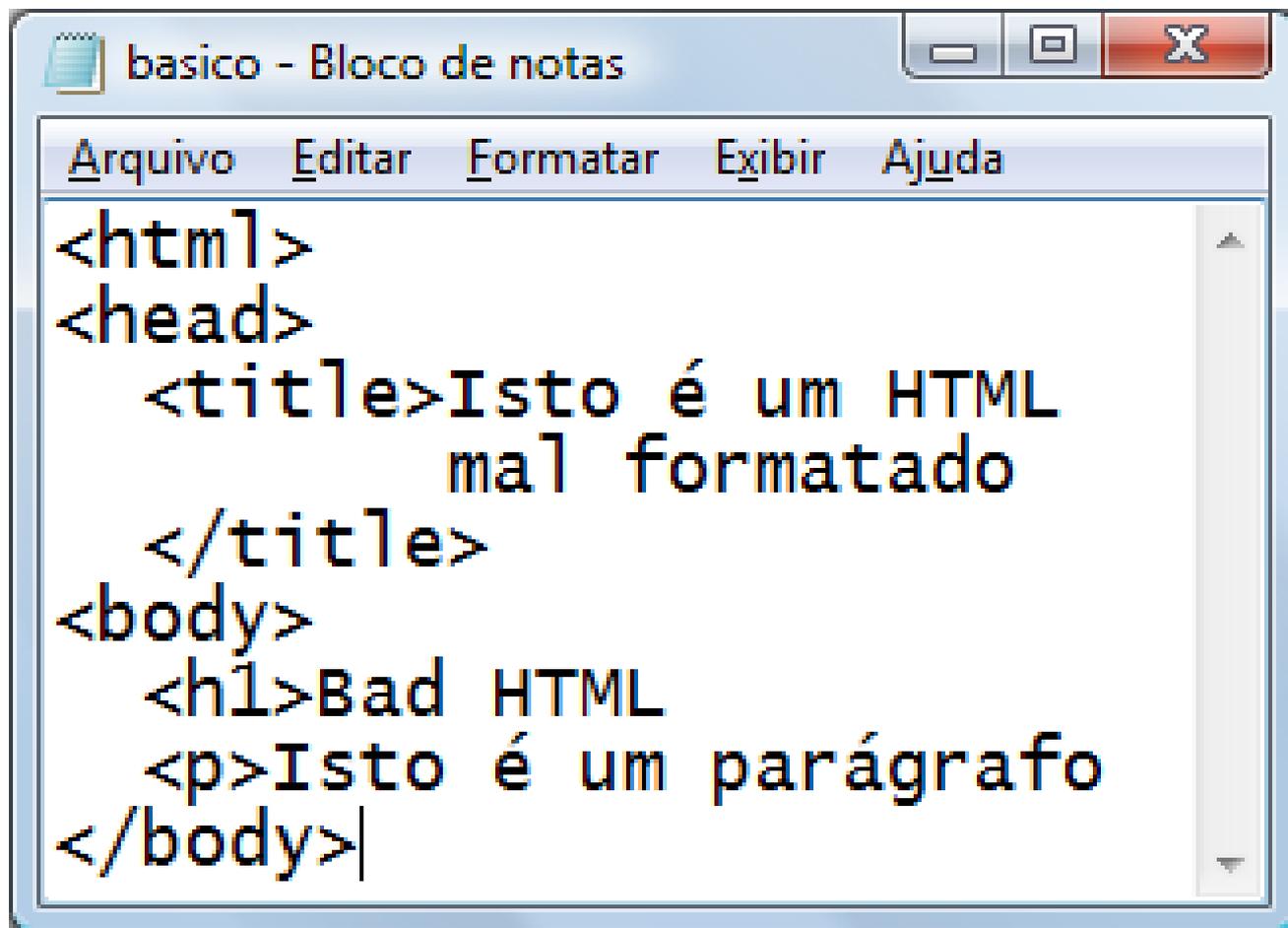
▶ Neste caso, temos apenas três *tags*, `<html>`, `<head>` e `<title>`

- ▶ Cada *tag* deve vir em par, com uma *tag* de abertura `<xxx>` e outra de fechamento `</xxx>`
- ▶ `<html>` é a *tag* que abre o documento HTML e `</html>` é a *tag* que fecha o documento
- ▶ `<head>` é a *tag* que inicia a seção de cabeçalho
- ▶ Os textos contidos entre `<title>` e `</title>` serão exibidos no título da janela da página



## “Melhorando” um Pouco

► Quantas “irregularidades” você consegue identificar no HTML abaixo (mesmo que você ainda não saiba o significado das *tags*)????



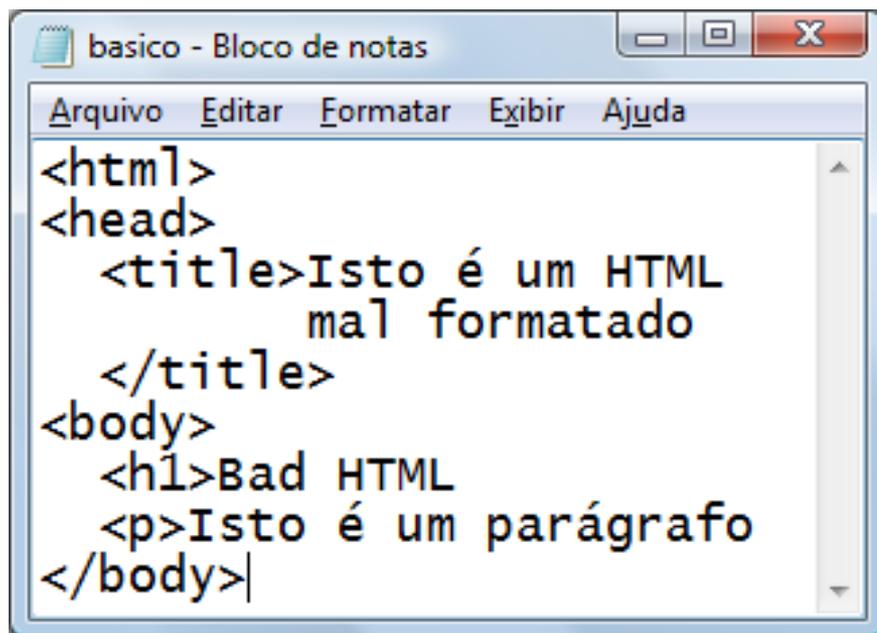
The image shows a screenshot of a Notepad window titled "basico - Bloco de notas". The window contains the following HTML code:

```
<html>
<head>
  <title>Isto é um HTML
    mal formatado
  </title>
<body>
  <h1>Bad HTML
  <p>Isto é um parágrafo
</body>
```

The code is displayed in a monospaced font. The window has a menu bar with "Arquivo", "Editar", "Formatar", "Exibir", and "Ajuda". The title bar includes standard window controls (minimize, maximize, close).



# “Melhorando” um Pouco



```
basico - Bloco de notas
Arquivo  Editar  Formatar  Exibir  Ajuda
<html>
<head>
  <title>Isto é um HTML
    mal formatado
  </title>
<body>
  <h1>Bad HTML
  <p>Isto é um parágrafo
</body>
```

## •Vejam os:

- A tag **<html>** não possui tag de fechamento
- A tag **<head>** não possui tag de fechamento
- A tag **<h1>** (tamanho do texto grande) e **<p>** (parágrafo) não possuem fechamento

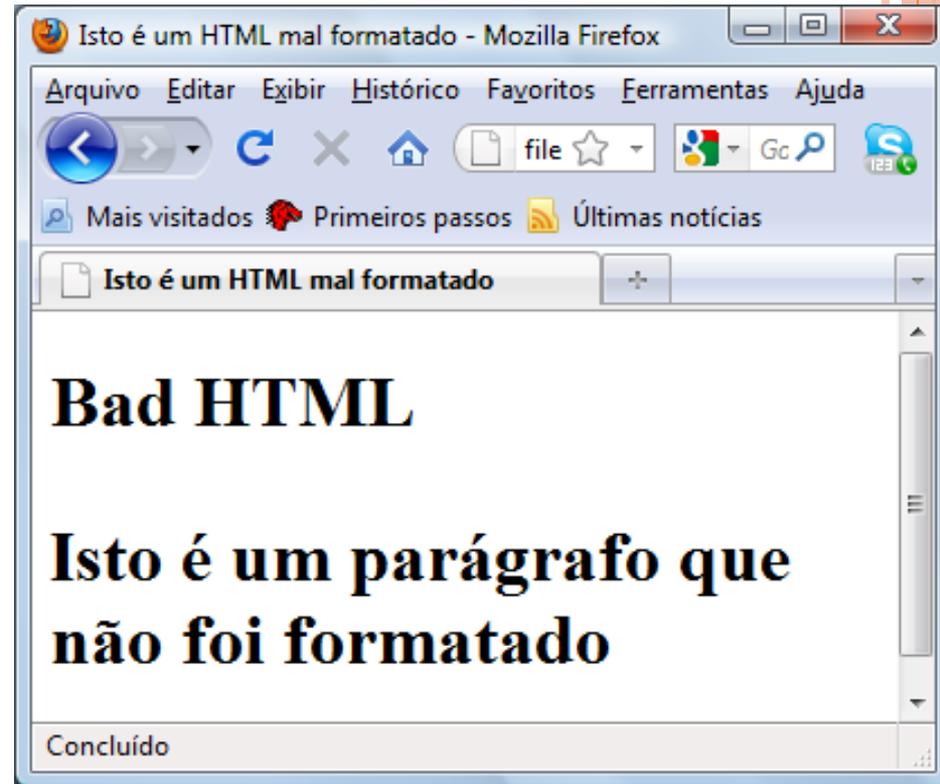
○ Esse exemplo vai resultar em algum texto apresentado no navegador? qual????



# “Melhorando” um Pouco

▶ Resposta: Sim. Mas, por quê?

- ▶ O navegador, tenta ao máximo “corrigir” as inconsistências deixadas pelo desenvolvedor, exibindo o texto
- ▶ No caso de um exemplo mais complexo, podem acontecer inconsistências ou maior tempo de carregamento da página
- ▶ Essa facilidade pode não estar disponível para navegadores de equipamentos com menos recursos (celulares, p.ex)



# XHTML

▶ XHTML adiciona mais rigidez na formatação do documento

▶ Um XHTML bem formado obedece a algumas regras, por exemplo:

- ▶ *Elementos (tags) devem ser definidas dentro do elemento principal*  
`<html> </html>`
- ▶ *Elementos devem ser fechados*
- ▶ *Elementos devem ser escritos em letra minúscula*
- ▶ *Elementos devem ser aninhados corretamente*

▶ As definições das regras encontram-se descritas no padrão W3C para XHTML



# XHTML – Estrutura Básica do Documento

```
<html>  
<head>  
  <title> ... </title>  
</head>  
  <body>  
  ...  
  </body>  
</html>
```

**<html> </html>** é o elemento “raiz” do documento, delimita o início e o fim da página

**<head> </head>** é o elemento de cabeçalho da página. Por exemplo, é nele que o título será inserido utilizando-se as *tags* (**<title> </title>**)

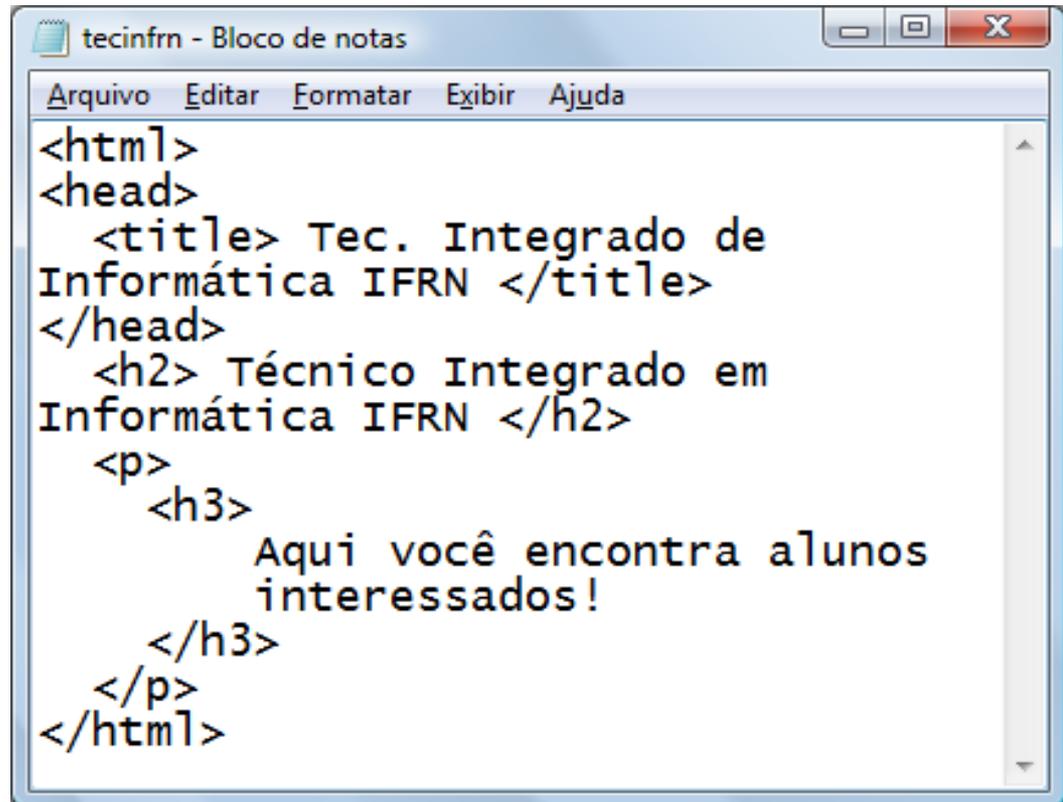
**<body> </body>** delimitam o corpo do documento. É no corpo onde iremos ter o conteúdo em si da página



# Exercício

1.1. Digite o exemplo de página XHTML abaixo em arquivo, salve-o com algum nome e extensão **.html** e depois abra-o no seu navegador e veja o resultado

2.2. Modifique o arquivo, por exemplo, tirando uma *tag* e veja o que acontece

A screenshot of a Notepad window titled "tecinfrn - Bloco de notas". The window has a menu bar with "Arquivo", "Editar", "Formatar", "Exibir", and "Ajuda". The text area contains the following XHTML code:

```
<html>
<head>
  <title> Tec. Integrado de
Informática IFRN </title>
</head>
  <h2> Técnico Integrado em
Informática IFRN </h2>
  <p>
    <h3>
      Aqui você encontra alunos
      interessados!
    </h3>
  </p>
</html>
```

# Detalhamento das restrições de XHTML

*Tags devem ser aninhadas apropriadamente*

- ▶ As marcações como deste exemplo podem não funcionar como se espera

```
<b><i>This text is bold and italic</b></i>
```

- ▶ O “correto” seria:

```
<b><i>This text is bold and italic</i></b>
```



# XHTML : Restrições

*Todas as tags devem ser fechadas*

## ○ Incorreto:

`<p>` Isto é um parágrafo  
`<p>` Isto é outro parágrafo

## ○ Correto:

`<p>` Isto é um parágrafo `</p>`  
`<p>` Isto é outro parágrafo `</p>`



# XHTML : Restrições

*Tags “sem corpo” devem ser fechadas com <nome-da-tag/>.*

## ○ Incorreto:

A break: **<br>**

A horizontal rule: **<hr>**

An image: ****

## ○ Correto:

A break: **<br />**

A horizontal rule: **<hr />**

An image: ****

○ **< />** esse tipo de fechamento é parte da notação XML para o fechamento de tags “vazias”

# XHTML : restrições

*“Nomes” das tags devem ser digitados em minúscula.*

## ▶ Incorreto

```
<BODY>  
<P>This is a paragraph</P>  
</BODY>
```

## ▶ Correto

```
<body>  
<p>This is a paragraph</p>  
</body>
```



# XHTML: RESTRICÇÕES

*Todos as outras tags devem vir entre a tag <html>*

## ▶ Exemplo

```
<html>  
<head> ... </head>  
<body> ... </body>  
</html>
```



# XHTML “Mínimo”

- ▶ Na verdade, um XHTML mínimo deve incluir uma declaração de tipo de documento (DOCTYPE), antes da tag <html>.
- ▶ Assim, teríamos o XHTML mínimo como visto abaixo

```
<!DOCTYPE ...>  
<html xmlns="http://www.w3.org/1999/xhtml">  
  <head>  
    <title> ... </title>  
  </head>  
  <body> ...  
  </body>  
</html>
```

**Atenção:** A declaração de DOCTYPE não é parte do documento XHTML em si, ela não é uma tag. No entanto, ela é obrigatória em um documento XHTML bem formado



# DTD (Document Type Definition)

▶ A declaração de tipo de documento (DOCTYPE) deve ser sempre a primeira do documento

*Obs.: Não se preocupe em decorar, as ferramentas inserem automaticamente a declaração*

▶ Um DTD determina qual a sintaxe de uma página na linguagem (no caso XHTML)

▶ Em XHTML, o DTD diz quais as regras permitidas para os elementos da linguagem

- ▶ Quais elementos são permitidos
- ▶ Como eles devem ser declarados, etc.



# Três tipos de DTD (Strict, Transitional e Frameset)

▶ **Strict** – Utilizando quando se quer uma linguagem de marcação limpa, livre de detalhes de “apresentação”

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

▶ **Transitional** – Utilizado quando se quer misturar “apresentação” em HTML

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

▶ **Frameset** – Quando queremos utilizar o recurso de frames HTML

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">
```



# XHTML Mínimo revisitado

O documento, usando o DTD transicional, ficaria como abaixo. Não é preciso “decorar” o texto do DTD, você pode simplesmente “copiar” o DTD apropriado para o seu documento.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <title> ... </title>
  </head>
  <body> ...
  </body>
</html>
```

Vemos aqui também um atributo, denominado de “declaração de espaço de nomes” (NAMESPACE) no elemento `<html>`:

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

**Mas, para que serve isso????**

# Espaço de nomes

- ▶ O atributo ***xmlns*** (XML namespace) especifica o espaço de nomes *xml* para o documento
  - ▶ É obrigatório em documento baseados em XML, como o XHTML
  - ▶ Serve para delimitar o escopo de definição das tags. Por exemplo, suponha que eu defina uma tag como o mesmo nome de outra que já exista. Se esta nova tag estiver em outro *namespace*, é possível utilizarmos as duas *tags*, pois não haverá conflito de “nomes” de tags
- ▶ No caso, o namespace **`xmlns=http://www.w3.org/1999/xhtml`** é padrão, não sendo obrigatório que você o coloque.



# Introdução à validação do documento no W3C

▶ *Tudo bem, mas se alguns navegadores deixam passar XHTML's mal-formatados, como eu vou saber se o meu XHTML está correto ???*

▶ Você pode utilizar o **Validador** oferecido pelo **W3C**.

- ▶ Documento é validado de acordo como o DTD utilizado

- ▶ Inconsistências são apontadas para que você possa corrigir o documento

- ▶ Endereço: <http://validator.w3.org/>



# Exercício

1. Verifique, manualmente, as inconsistências no documento abaixo
2. Submeta o documento ao validador do W3C e veja o que acontece

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

```
<html xmlns="http://www.w3.org/1999/xhtml">  
  <title>Documento XHTML inconsistente</title>  
  <body>  
    <H2>Corrija esta página, por favor</h2>  
    <p>Aqui temos um parágrafo.</p>  
    E aí consegui identificar algo?<hr>  
    uma <br> quebra de linha aqui.  
  </body>  
</html>
```

