

HTML5 - Tipos de dados e validadores

abr 24, 2012 às 11:10 por [DiegoBlanco](#)

2 classificações 29 comentários

Descrição



Este treinamento é homologado pela organização W3C que trabalha no desenvolvimento de padrões para web.

TIPOS DE DADOS E VALIDADORES

Formulários vitaminados

Conforme você deve ter percebido no último capítulo, o HTML5 avançou bastante nos recursos de formulários, facilitando muito a vida de quem precisa desenvolver aplicações web baseadas em formulários. Neste capítulo vamos avançar um pouco mais nesse assunto e, você vai ver, a coisa vai ficar ainda melhor.

autofocus

Ao incluir em um campo de formulário o atributo autofocus, assim:

```
<input name="login" autofocus >
```

O foco será colocado neste campo automaticamente ao carregar a página. Diferente das soluções em Javascript, o foco estará no campo tão logo ele seja criado, e não apenas ao final do carregamento da página. Isso evita o problema, muito comum quando você muda o foco com Javascript, de o usuário já estar em outro campo, digitando, quando o foco é mudado.

Placeholder text

Você já deve ter visto um "placeholder". Tradicionalmente, vínhamos fazendo isso:

```
<!DOCTYPE html>
<html lang="en-US">
<head>
<meta charset="UTF-8" />
<title>Placeholder, the old style</title>
</head>

<body>
<input name="q" value="Search here" onfocus="if(this.value=='Search here')this.value=''">
</body>

</html>
```

HTML5 nos permite fazer isso de maneira muito mais elegante:

```
<!DOCTYPE html>
<html lang="en-US">
<head>
<meta charset="UTF-8" />
<title>Placeholder, HTML5 way</title>
</head>

<body>
<input name="q" placeholder="Search here">
</body>

</html>
```

required

Para tornar um campo de formulário obrigatório (seu valor precisa ser preenchido) basta, em HTML5, incluir o atributo required:

```
<input name="login" required>
```

maxlength

Você já conhecia o atributo maxlength, que limita a quantidade de caracteres em um campo de formulário. Uma grande lacuna dos formulário HTML foi corrigida. Em HTML5, o elemento textarea também pode ter maxlength!

Validação de formulários

Uma das tarefas mais enfadonhas de se fazer em Javascript é validar formulários. Infelizmente, é também uma das mais comuns. HTML5 facilita muito nossa vida ao validar formulários, tornando automática boa parte do processo. Em muitos casos, todo ele. Você já viu que pode tornar seus campos "espertos" com os novos valores para o atributo type, que já incluem validação para datas, emails, URLs e números. Vamos um pouco além.

pattern

O atributo pattern nos permite definir expressões regulares de validação, sem Javascript. Veja um exemplo de como validar CEP:

```
<!DOCTYPE html>
<html lang="pt-BR">
<head>
<meta charset="UTF-8" />
<title>O atributo pattern</title>
</head>

<body>

<form>
<label for="CEP">CEP:
<input name="CEP" id="CEP" required pattern="\d{5}-?\d{3}" />
</label>
<input type="submit" value="Enviar" />
</form>

</body>

</html>
```

novalidate e formnovalidate

Podem haver situações em que você precisa que um formulário não seja validado. Nestes casos, basta incluir no elemento form o atributo novalidate.

Outra situação comum é querer que o formulário não seja validado dependendo da ação de submit. Nesse caso, você pode usar no botão de submit o atributo formnovalidate. Veja um exemplo:

```
<!DOCTYPE html>
<html lang="pt-BR">
<head>
<meta charset="UTF-8" />
<title>Salvando rascunho</title>
<style>
label{display:block;}
</style>
</head>

<body>

<form>
<label>nome: <input name="nome" required></label>
<label>email: <input name="email" type="email" required></label>
<label>mensagem: <textarea name="mensagem" required></textarea></label>
<input type="submit" name="action" value="Salvar rascunho" formnovalidate>
<input type="submit" name="action" value="Enviar">
</form>

</body>

</html>
```

Custom validators

É claro que as validações padrão, embora atendam a maioria dos casos, não são suficientes para todas as situações. Muitas vezes você vai querer escrever sua própria função de validação Javascript. Há alguns detalhes na especificação do HTML5 que vão ajudá-lo com isso:

1. O novo evento oninput é disparado quando algo é modificado no valor de um campo de formulário. Diferente de onchange, que é disparado ao final da edição, oninput é disparado ao editar. É diferente também de onkeyup e onkeypress, porque vai capturar qualquer modificação no valor do campo, feita com mouse, teclado ou outra interface qualquer.
2. O método setCustomValidity pode ser invocado por você. Ele recebe uma string. Se a string for vazia, o campo será marcado como válido. Caso contrário, será marcado como inválido.

Com isso, você pode inserir suas validações no campo de formulário e deixar o navegador fazer o resto. Não é mais preciso capturar o evento submit e tratá-lo.

Este treinamento faz parte do Microsoft Virtual Academy: <http://www.microsoftvirtualacademy.com>

marcas: [Brazilian Portuguese](#), [HTML 5](#), [HTML5](#), [Microsoft](#)

Episódios relacionados

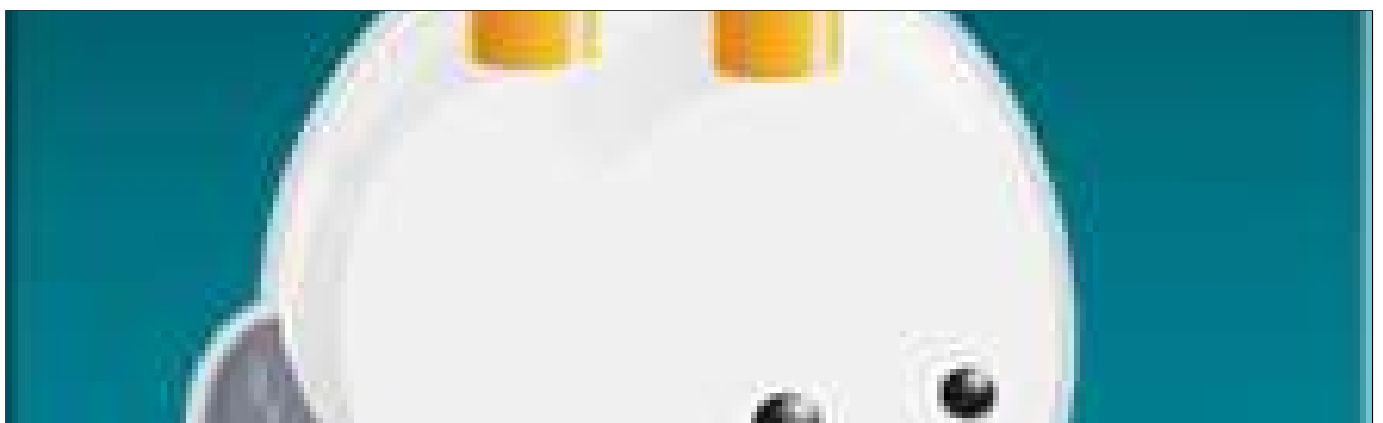




HTML5 - Undo



HTML5 - Geolocation API



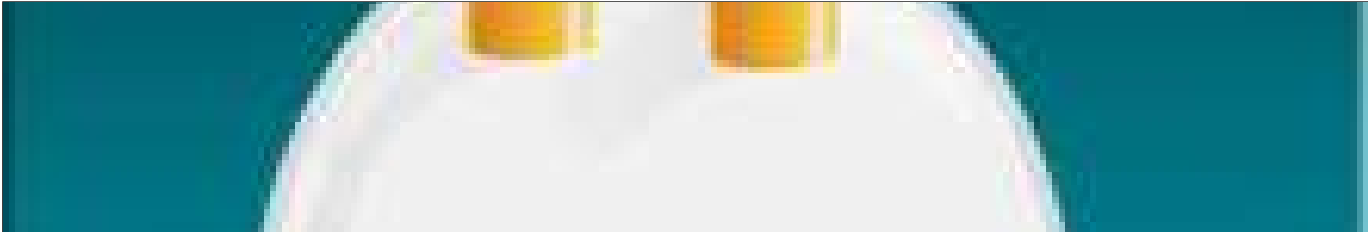


HTML5 - Scroll Into View e Hidden



HTML5 - Aplicações Offline



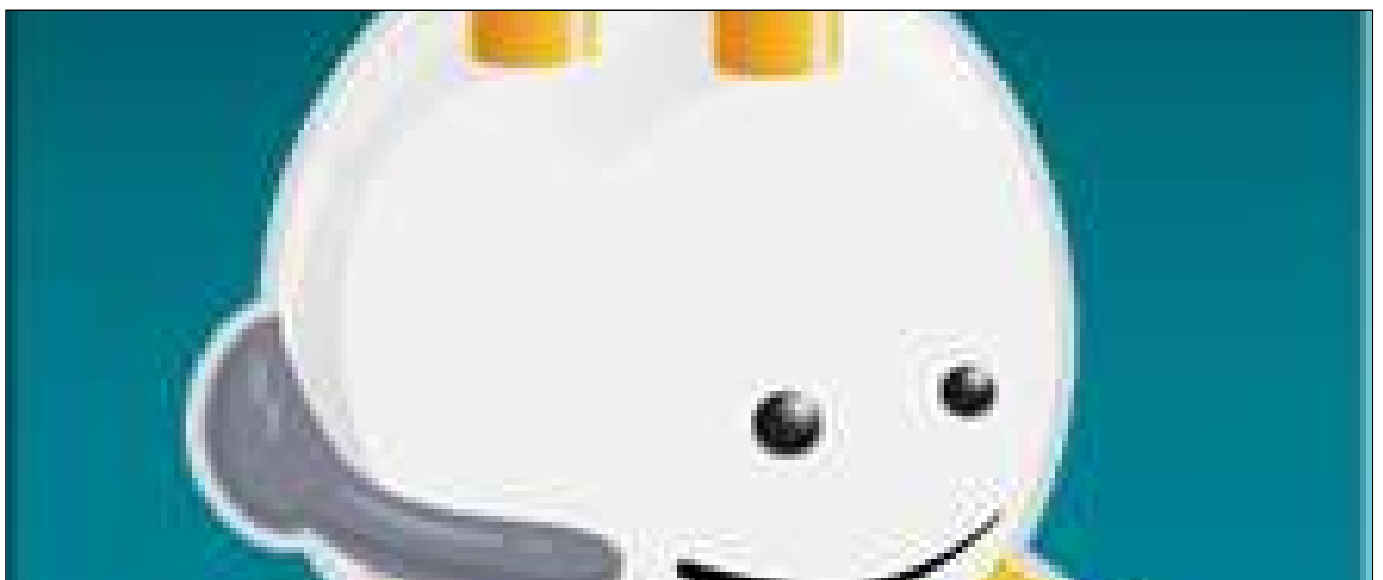


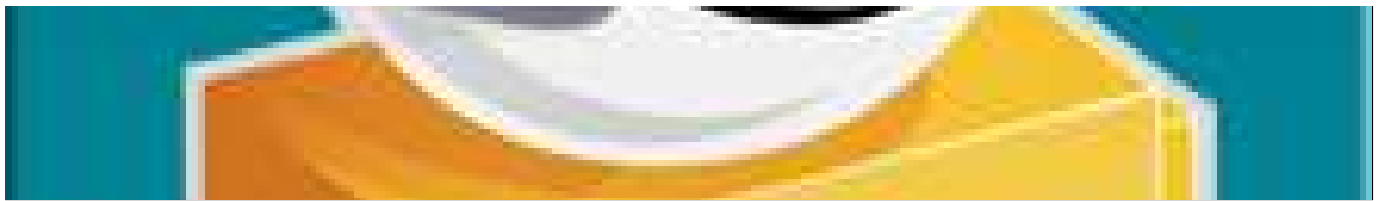


HTML5 - Novos Eventos DOM



HTML5 - DOM e HTML5





HTML5 - Server-Sent Events

Explicação



Rafael

De boa facilito um monte!

Última modificação em [jul 10, 2012 às 6:46](#)



Fernando Cardoso Ribeiro

Excelente informação sobre TIPOS DE DADOS E VALIDADORES.

Última modificação em [jul 12, 2012 às 7:24](#)



Pablo

Não pude copiar os códigos pq os parenteses não eram compatíveis, tive que apagar e colocas parenteses ("") novos. =)

Última modificação em [jul 13, 2012 às 12:01](#)



Eduardo Gatz

Os formulários realmente é a parte mais chata, com evento oninput que valido dentro do proprio campos antes de perder o foco é de alta valia!

Última modificação em [jul 18, 2012 às 6:38](#)



Thiago

aaaaaaaaaaaaaooooo blza, html 5 veio pra nos salvar \O/

Última modificação em [jul 18, 2012 às 6:33](#)



Leandro Siqueira

O HTML5 está me surpreendendo a cada aula. Grato!

Última modificação em [jul 27, 2012 às 4:58](#)