Revisando o que vimos nesta aula:

* Aprendemos que desenvolver **não** exige um ambiente complexo;
* É possível utilizar o próprio navegador, e todo sistema operacional possui um. Ao longo do curso utilizaremos o **Google Chrome**;
* Podemos utilizar um editor de texto padrão. Nós usaremos o **Sublime Text**, disponível para todas as plataformas;
* Compreendemos que o HTML é uma linguagem de marcação (linguagem estática!), pois ela possibilita marcar conteúdos por meio de *tags*;
* Algumas *tags* não possuem marcação, como o <br>, que serve para pular uma linha;
* A *tag* <a> nos permite sair de uma página e mudar para outra;
* O HTML é estático, isto é, não muda. Não podemos fazer nada muito sofisticado com ele. Por esse motivo o navegador entende também a linguagem JavaScript, que é uma linguagem de programação dinâmica, com a qual podemos desenvolver de maneira mais avançada;
* Para o navegador interpretar uma instrução JavaScript, não basta colocarmos a instrução JavaScript direto no HTML. Temos que utilizar a *tag* <script>. Dessa forma o navegador saberá que deve processar essa parte do código como linguagem JavaScript e não como HTML;
* A primeira instrução que vimos do JavaScript foi o alert, que recebe como parâmetro um texto;
* Por último, aprendemos que todo texto em JavaScript vem entre aspas.

<meta charset="UTF-8"> ‘Linha padrão para interpretar caracteres especiais.

<h1>Meu primeiro teste!</h1> ‘<h1> e </h1>: tipo de header

<br> ‘pula linha

<br>

Seria isso um programa? Descubra visitando o Alura

<a href="http://www.alura.com.br">aqui</a>! ‘<a></a> ancora um link na palavra

<script> ‘inicia o ambiente para javascript

alert("Isso sim é um programa"); ’função alert: pop-up com ok

</script>

Tag Meta

Quando salvamos um arquivo texto no disco ele é salvo usando uma cadeia de caracteres (*character set encoding*). Se no editor de texto salvamos o arquivo contendo charset UTF-8, precisamos dar uma pista para o navegador de como ele deve ser processado. Se não fizermos isso, ele não conseguirá exibir corretamente qualquer texto acentuado.

Contudo, muito cuidado! Vamos supor que acidentalmente seu editor de texto não salvou o arquivo como UTF-8, mas em latin1. Se colocarmos a tag <meta charset="UTF-8"> daremos uma dica errada para o navegador e isso nos trará problemas na acentuação. Para resolver esse tipo de situação, podemos usar <meta charset="latin"> ou mudar nosso arquivo para UTF-8, o que é mais difícil.

Pode ser que o editor de texto escolhido não siga o padrão UTF e utilize outro qualquer que nem eu ou você sabemos.

<meta charset="UTF-8">

<meta> =: tag

charset =: atributo

"UTF-8" =: valor atribuído

document.write =: função javascript que consegue escrever no mundo HTML

<meta charset = "UTF-8">

<script>

document.write("aloy <br> aloy2");

document.write("<br><br>");

document.write("18" + "20");

document.write("<br><br>");

document.write(18 + 20);

document.write("<br><br>");

document.write("18" + 20);

document.write("<br><br>");

document.write(18 + "20");

</script>

Output:

aloy

aloy2

1820

38

1820

1820

<meta charset = "UTF-8">

<script>

document.write("soma errada " + 20 + 10);

document.write("<br>");

document.write("soma certa " + (20 + 10));

document.write("<br>");

</script>

Output:

soma errada 2010

soma certa 30

declaração de variável: var

<meta charset = "UTF-8">

<script>

var ano = 2020;

document.write("ano passado: " + (ano - 1));

document.write("<br>");

document.write("ano que vem: " + (ano + 1));

</script>

Função sem parênteses, mostra o código somente da função.