

Melhorando a usabilidade

Transcrição

Nesta aula veremos um último detalhe deste Capítulo!

Caso você acerte ou erre o número sorteado, faz sentido que o campo seja limpo, indicando que o usuário pode digitar uma nova tentativa.

Para fazer isso vamos dizer que o `input.value` deve receber uma string vazia. Portanto, teremos dentro da `function verifica()` o seguinte:

```
if(input.value == segredo) {  
  
    alert("Você ACERTOU!");  
} else {  
  
    alert("Você ERROU!!!!!!!!!!");  
  
}  
  
input.value = "";
```

E se quisermos que a caixa do número comece a piscar para indicar uma nova tentativa de adivinhar o número?

Para isso, acrescentamos abaixo do `input.value` o `input.focus()`. Ao fazer isso teremos:

```
input.value = "";  
input.focus();
```

Assim, já conseguimos mudar a experiência do usuário apenas com o recurso de focar na caixa.

O presente curso trabalha Lógica de Programação e a ideia é justamente desenvolver esse aspecto do aluno, por isso, abordamos o mínimo possível de Javascript e HTML, o que já faz com que o estudante perceba como o Javascript interage de maneira dinâmica com o HTML.

O detalhe é que não queremos que o segredo fique fixo, por isso vamos inserir o `Math.round(Math.random() * 10)`. Assim, quando a página for recarregada, um novo número aleatório será gerado. Teremos o seguinte no código:

```
var segredo = Math.round(Math.random() * 10);
```

Observe que quando abrimos a página, o campo no qual o número deve ser digitado não fica em destaque. Nosso objetivo é deixá-lo em foco assim que a página for aberta. Portanto, vamos acrescentar o `input.focus()`. Teremos:

```
var input = document.querySelector("input");  
input.focus();
```

Nessa aula acabamos de aprender a lidar melhor com a captura das informações informadas na caixa!