

FULL

UMA “NOVA” ESTRUTURA PARA CRIAR LAYOUTS DE SITES

STACK

EPISÓDIO IV

O que aprendemos na aula anterior?

- **Formatando textos**

strong, b, em, i, cite, abbr, mark

- **Formulários HTML**

form, label, input, textarea, select, button

- **Imagens no CSS**

background-image

- **Pseudo Seletores**

:hover, :focus, :visited, :active

Seções semânticas

`<section> </section>`

Seção de conteúdo monotemático.

`<article> </article>`

Informações dentro de uma seção.

`<header> </header>`

Cabeçalho do conteúdo ou de um documento.

`<footer> </footer>`

Rodapé do conteúdo ou de um documento.

Seções não semânticas

```
<div> </div>
```

Divisão de conteúdo.

```
<span> </span>
```

Tag multifuncional
(não é um bloco).

Vamos praticar?



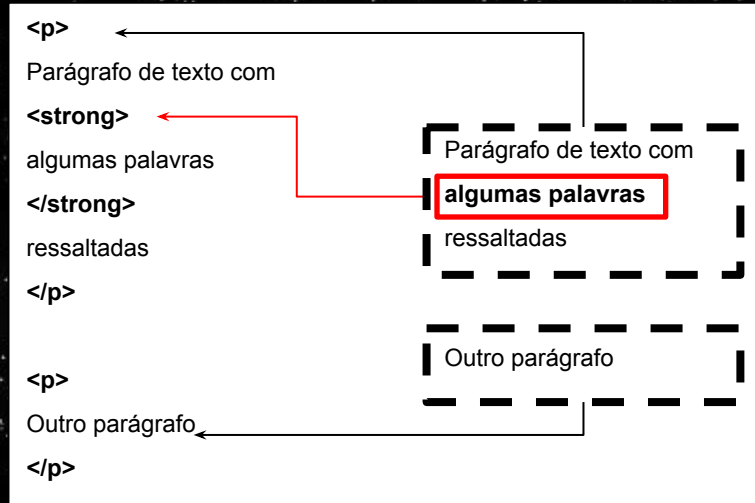
Questão 1 da folha de exercícios

Modelo de caixa



Sem sombra de dúvidas, a **característica mais importante do CSS**.

Modelo de caixa



- É o comportamento que faz com que todos os elementos de um documento HTML sejam representados por caixas retangulares.
- Condiciona o design de todas as páginas web.

Tipos de elementos

== display: block, inline ==

inline

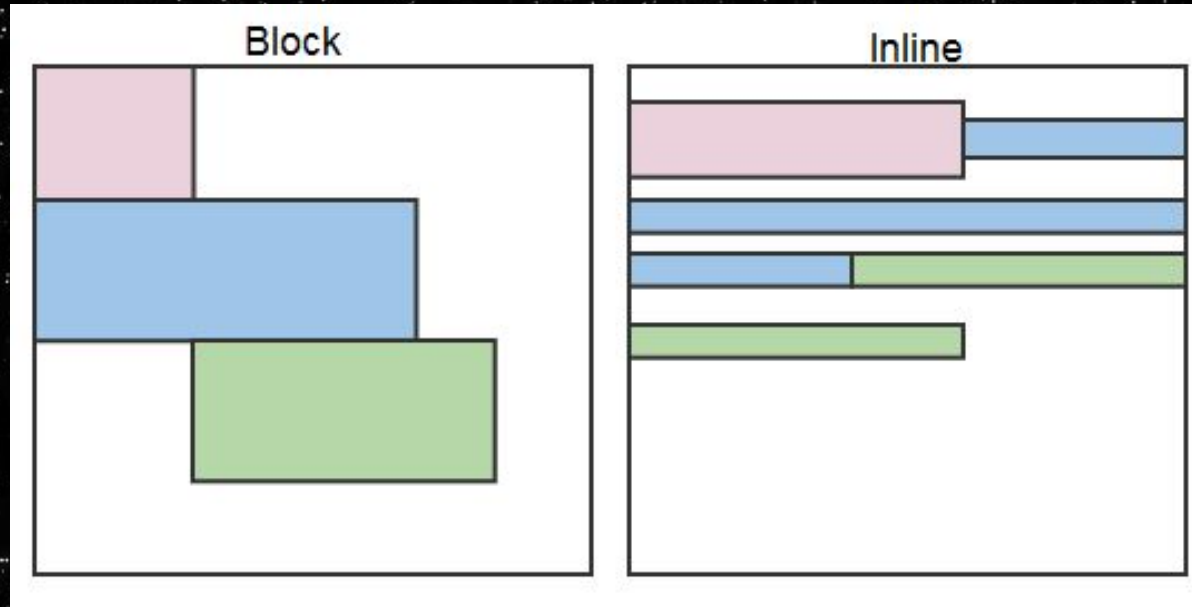
Define um elemento com comportamento de linha. Não recebe algumas propriedades do modelo de caixa.

block

Define um elemento com comportamento de bloco. Recebe “facilmente” propriedades do modelo de caixa.

Tipos de elementos

== display: block, inline ==



Utilizar o DEVTOOLS (F12) para visualizar melhor as diferenças.

Tipos de elementos

== display: inline-block, none ==

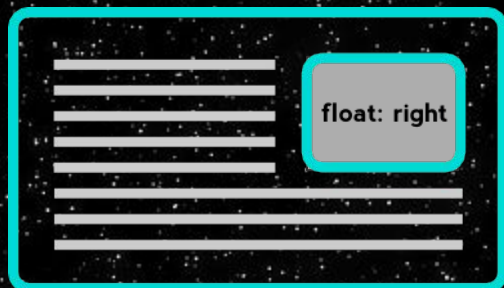
inline-block

Define um elemento com comportamento de semibloco, recebe “facilmente” propriedades do modelo de caixa. Também tem propriedades do elemento de linha.

none

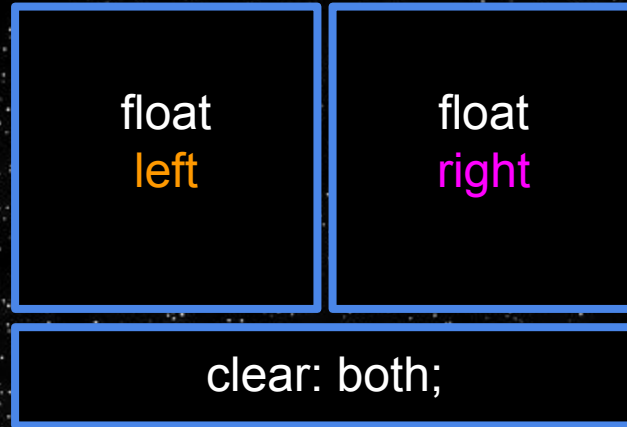
Oculto visualmente um elemento sem eliminá-lo da estrutura de HTML. O efeito é apenas visual.

Posicionamento por flutuação



- Permite que os elementos de bloco “compartilhem a fila” em que estão com outros elementos que também flutuem.
- Os elementos abaixo do elemento flutuante presumem que ele não existe e tendem a ocupar o lugar vazio “deixado” pelo elemento flutuante. Isso NÃO acontece com elementos de texto.

Posicionamento por flutuação



Não flutua, tem clear: both;

Clear: é a forma mais simples de “limpar” flutuações. Basta aplicar essa propriedade ao elemento que está abaixo das caixas flutuantes para que ele mantenha sua posição.

Propriedades CSS

== overflow ==

Com a propriedade overflow, é possível controlar o “fluxo” interno de um container.

Em outras palavras, ela permite manipular o que acontece quando o conteúdo interno supera o tamanho do container pai.

A propriedade overflow tem 4 valores possíveis:

- visible
- hidden
- scroll
- auto

Propriedades CSS

== overflow ==

Lorem ipsum ad his scripta blandit partiendo, eum fastidii accumsan euripidis in, eum liber hendrerit an. Qui ut wisi vocibus suscipiantur, quo dicit ridens inciderint id. Quo mundi lobortis reformidans eu, legimus senserit definiebas an eos. Eu sit tincidunt incorrupte definitionem, vis mutat affert percipit cu, eimod

consectetur signiferumque eu per. In usu latine equidem dolores. Quo no falli viris intellegam, ut fugit veritus placerat per.

Ius id vidit volumus mandamus, vide veritus democritum te nec, ei eos debet libris consulatu. No mei ferri graeco dicunt, ad cum veri accommodare. Sed at malis omnesque delicata, usu et iusto zzril meliore. Dicunt maiorum eloquentiam cum eu, sit summo dolor essent te. Ne quodsi nusquam legendos has, ea dicit voluptua eloquentiam pro, ad sit quas qualisque. Eos vocibus deserunt quaestio ei.

overflow:
visible;

Lorem ipsum ad his scripta blandit partiendo, eum fastidii accumsan euripidis in, eum liber hendrerit an. Qui ut wisi vocibus suscipiantur, quo dicit ridens inciderint id. Quo mundi lobortis reformidans eu, legimus senserit definiebas an eos. Eu sit tincidunt incorrupte definitionem, vis mutat affert percipit cu, eimod

overflow:
hidden;

Lorem ipsum ad his scripta blandit partiendo, eum fastidii accumsan euripidis in, eum liber hendrerit an. Qui ut wisi vocibus suscipiantur, quo dicit ridens inciderint id. Quo mundi lobortis reformidans eu, legimus senserit definiebas an eos. Eu sit tincidunt incorrupte

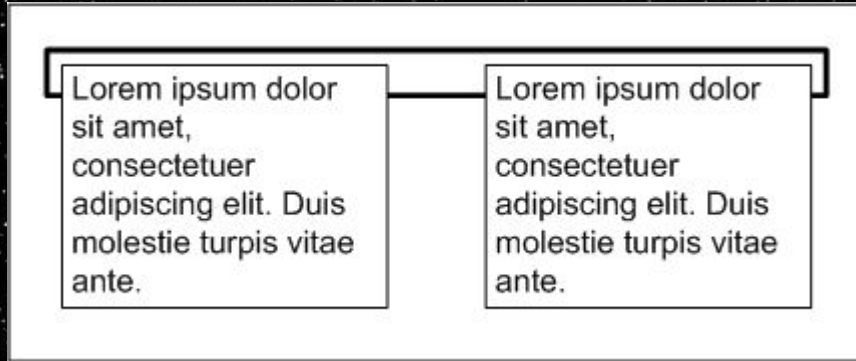
overflow:
scroll;

Lorem ipsum ad his scripta blandit partiendo, eum fastidii accumsan euripidis in, eum liber hendrerit an. Qui ut wisi vocibus suscipiantur, quo dicit ridens inciderint id. Quo mundi lobortis reformidans eu, legimus senserit definiebas an eos. Eu sit tincidunt incorrupte definitionem, vis mutat affert

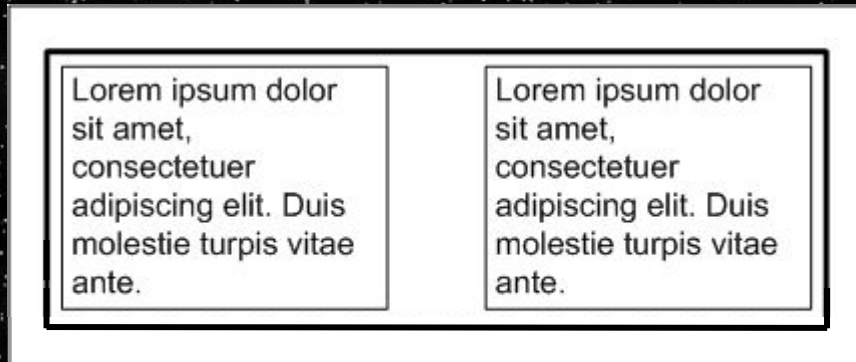
overflow:
auto;

Propriedades CSS

== overflow para limpar flutuação ==



Comportamento natural do container pai e de filhos com flutuação.



Comportamento do container pai com a propriedade overflow: hidden;

Propriedades CSS

== Width ==

```
elemento {  
    width: 560px;  
  
    /* podemos usar % */  
    /* width: 50% */  
}
```

Atenção: se um elemento não tiver a width declarada, ela será igual a 100% do container pai, desde que seja um bloco.

Propriedades CSS

== Height ==

```
elemento {  
    height: 200px;  
  
    /* NÃO podemos usar % */  
    /* width: 20% */  
}
```

Atenção: se um elemento não tiver a height declarada, ela será igual à altura do conteúdo interno, seja ele um bloco ou uma linha.

width

height

elemento

display:block; ou display:inline;

Propriedades CSS

== Padding ==

```
elemento {  
    padding-top: 10px;  
    padding-right: 20px;  
    padding-bottom: 30px;  
    padding-left: 40px;  
  
    /* shorthand */  
    padding: 10px 20px 30px 40px;  
}
```

padding

width

height

elemento

display:block; ou display:inline;

Propriedades CSS

== Border ==

```
elemento {  
    border-style: solid;  
    border-width: 2px;  
    border-color: #FF0000;  
  
    /* shorthand */  
    border: solid 2px #FF0000;  
}
```

border

padding

width

height

elemento

display:block; ou display:inline;

Propriedades CSS

== Margin ==

```
elemento {  
    margin-top: 10px;  
    margin-right: 20px;  
    margin-bottom: 30px;  
    margin-left: 40px;  
  
    /* shorthand */  
    margin: 10px 20px 30px 40px;  
}
```


margin

border

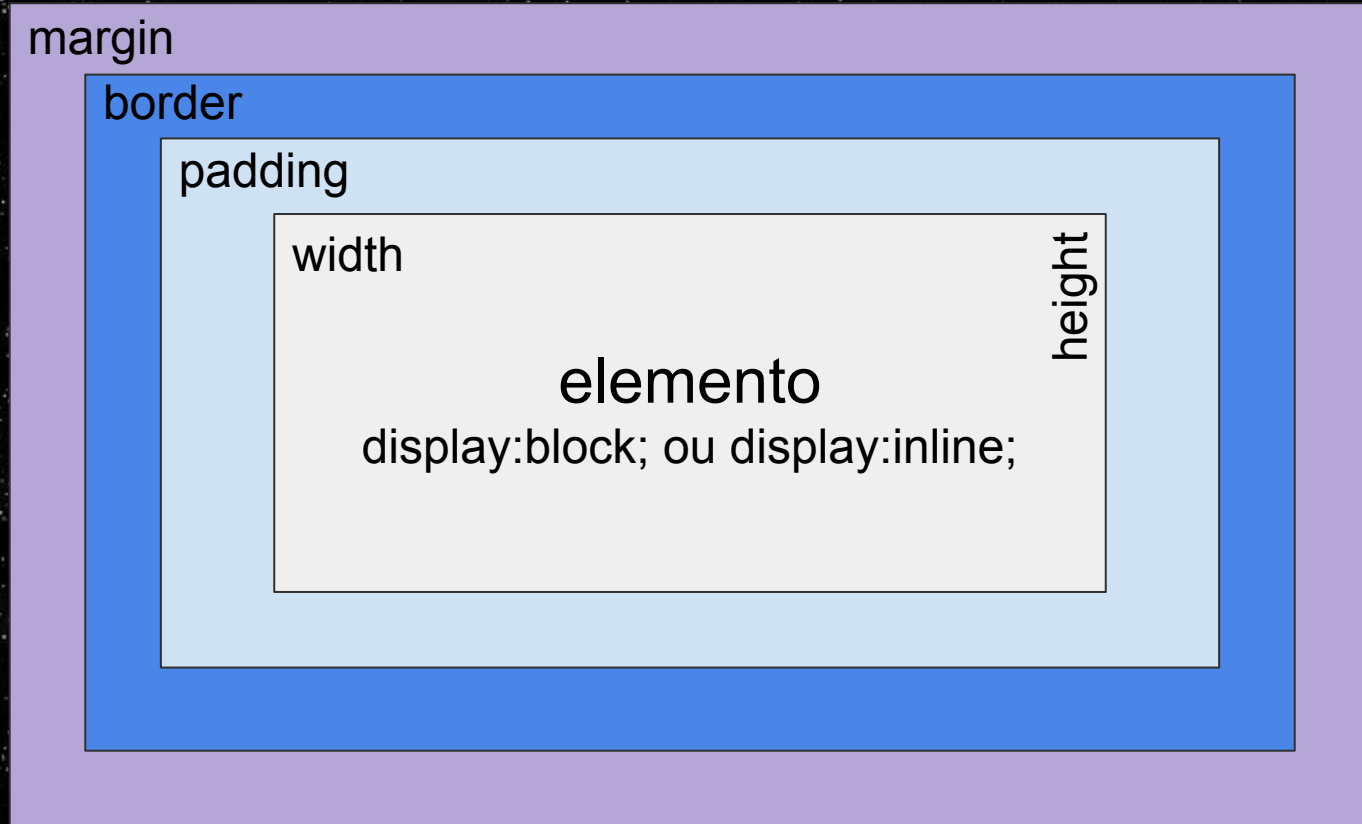
padding

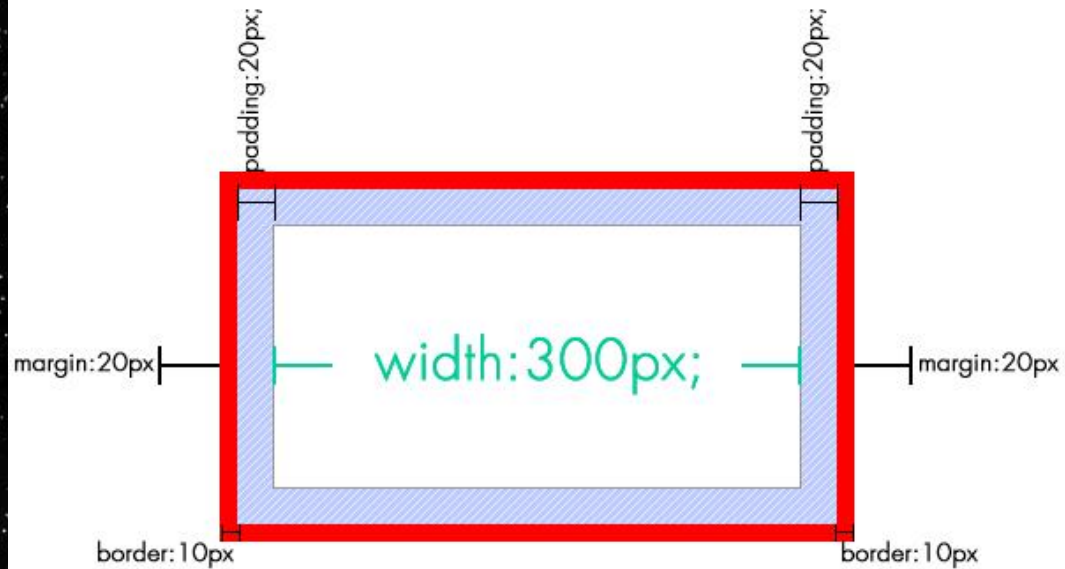
width

height

elemento

display:block; ou display:inline;





$$20 + 10 + 20 + 300 + 20 + 10 + 20$$

400px

Propriedades CSS

== Box-sizing ==

```
elemento {  
    box-sizing: border-box;  
    /* outros valores: */  
    /* content-box (padrão) */  
}
```

Definição: esta propriedade facilita o uso do **modelo de caixa**, pois desconta automaticamente da largura e da altura o que será somado ao preenchimento e à borda. **Atenção: a margem ainda precisa ser somada.**

Vamos praticar?



Questão 2 da folha de exercícios

Perguntas?



Vamos tirar as dúvidas!

Na próxima aula...

Caixas de Layout