

# HTML

Elementos  
Atributos  
Nesting  
Hierarquia  
Semântica

# Elementos

<nav></nav>

<aside></aside>

<article></article>

<article></article>

<section></section>

# Elementos

<nav></nav>

<aside></aside>

<article></article>

<article></article>

<section></section>

<nav></nav>

<aside></aside>

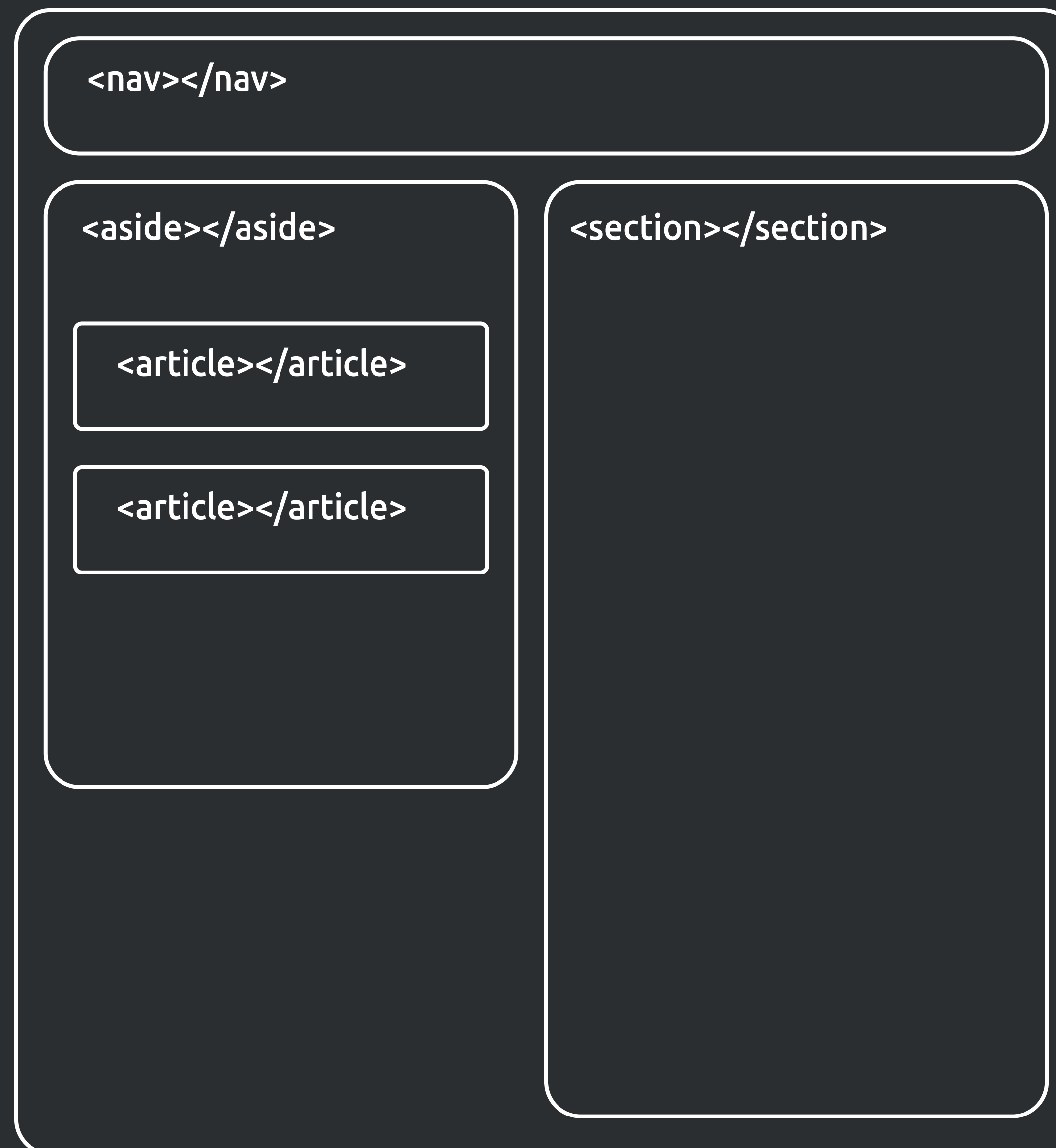
<article></article>

<article></article>

<section></section>

# Elementos

Podem ser visualmente alterados com a introdução do CSS ou pela sua própria natureza.



# Atributos

São propriedades que podemos adicionar aos nossos elementos.

`<tagname>` conteúdo aqui... `</tagname>`

# Atributos

São propriedades que podemos adicionar aos nossos elementos.

Abertura



`<div>` conteúdo aqui... `</div>`

# Atributos

São propriedades que podemos adicionar aos nossos elementos.

Attr



`<div class="box">` conteúdo aqui... `</div>`

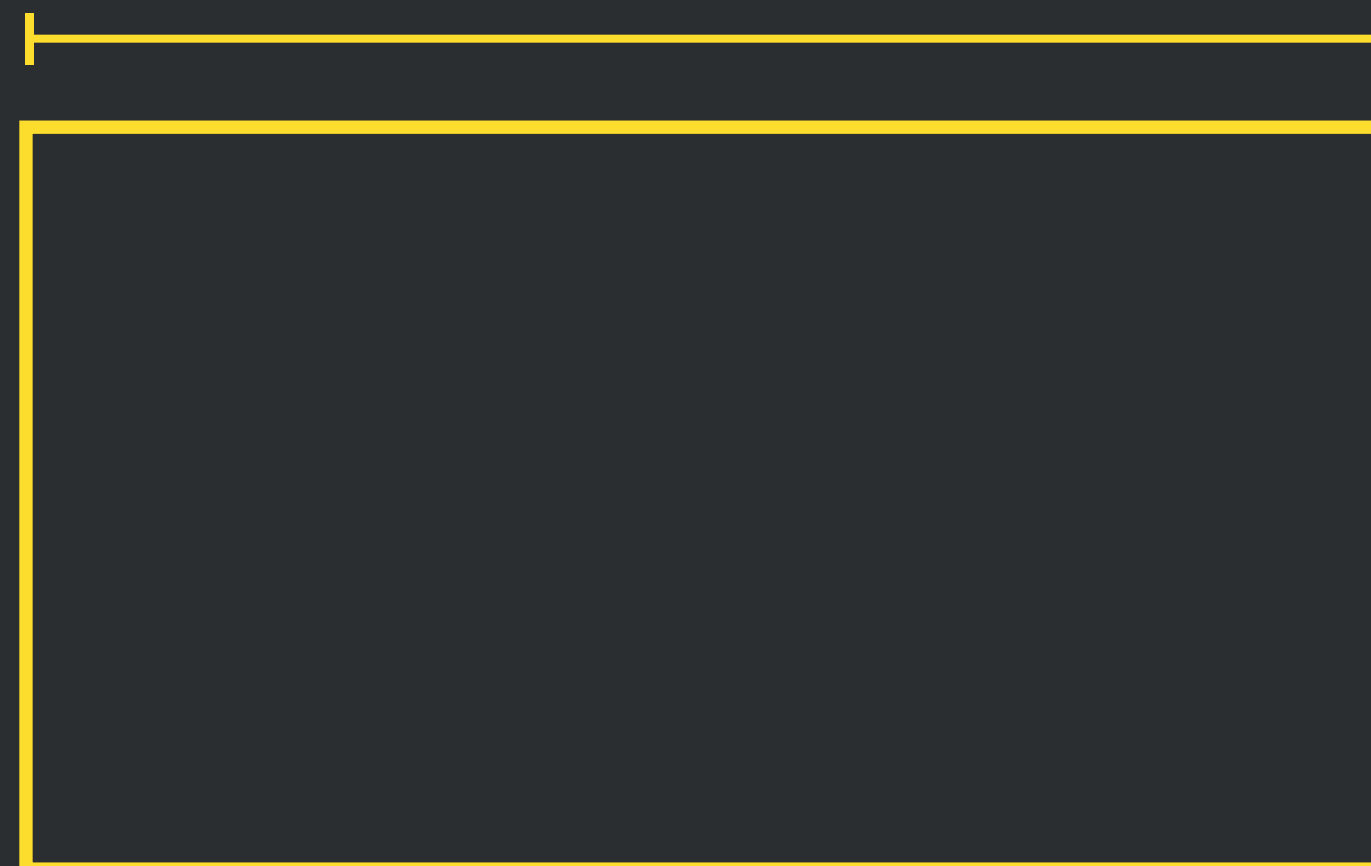
# Atributos

São propriedades que podemos adicionar aos nossos elementos.

Attr

`<img width="200">` conteúdo aqui... `</div>`

200px de largura





# Atributos

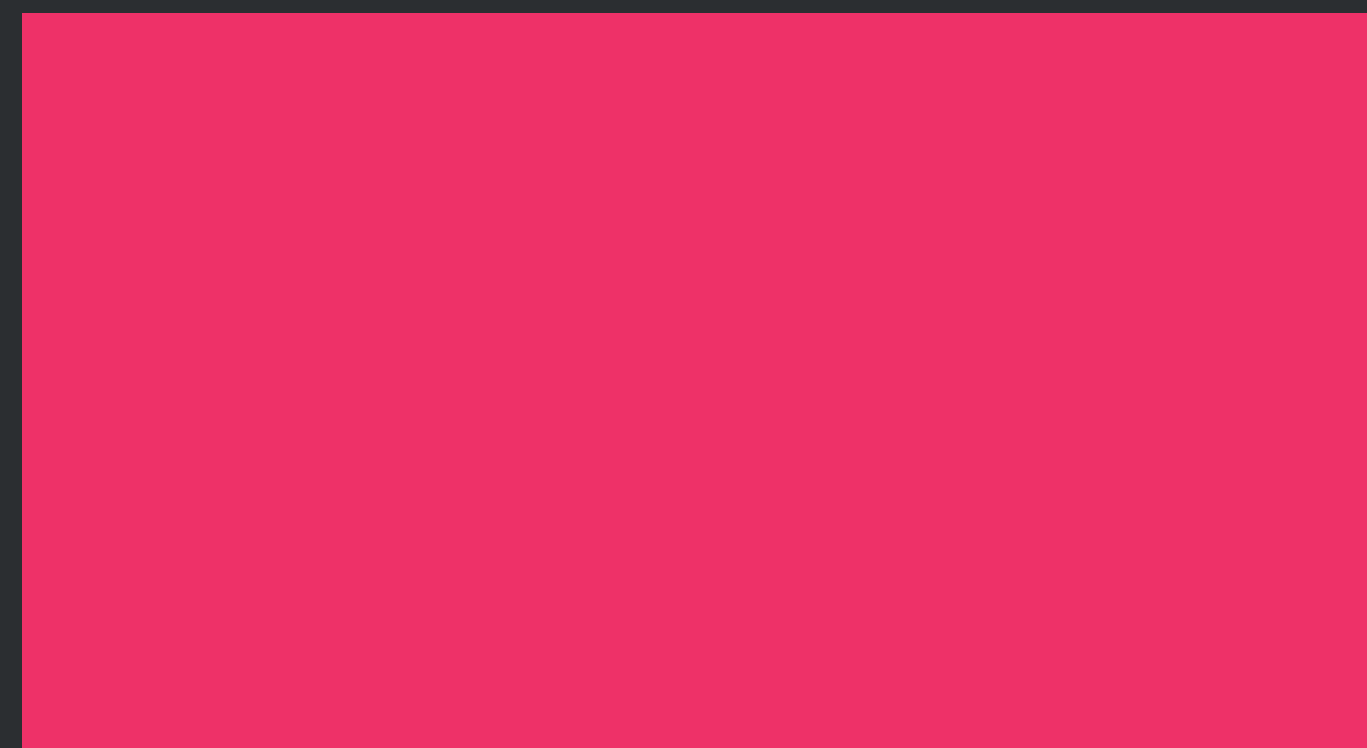
São propriedades que podemos adicionar aos nossos elementos.

Style Attribute

`<img style="width: 200px; background: red"> conteúdo aqui.. </div>`

Prop.

Prop.



# Atributos

Há atributos que são globais, podem ser usados por todos os elementos.  
Existem outros não, depende do contexto. Tanto do elemento como do atributo.

## Atributos Globais

class

data-\*

id

style

title

etc..

## Atributos Específicos

src

No caso de uma imagem <img>.

src serve para associar uma  
ficheiro a um elemento.

href

No caso de um link <a>.

Serve para associar um destino a  
um link.

# Considerações nos atributos

```
<p class="my-class" id="my-id" style="color: red">conteúdo aqui.</p>
```

Todos os elementos de HTML podem ter atributos

Atributos anexam informação adicional sobre os elementos

Atributos são sempre adicionados na tag de abertura.

Atributos normalmente, alias, devem, ser adicionados em name/value pairs

ex: name="value"

# Nesting

Nesting é quando agrupamos os nossos elementos dentro de outros elementos.  
Criando relações entre pais e filhos e irmãos.

```
<nav></nav>
```

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

```
<a></a>
```

```
<a></a>
```

```
<a></a>
```

# Nesting

Nesting é quando agrupamos os nossos elementos dentro de outros elementos.  
Criando relações entre pais e filhos e irmãos.

```
<nav></nav>
```

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

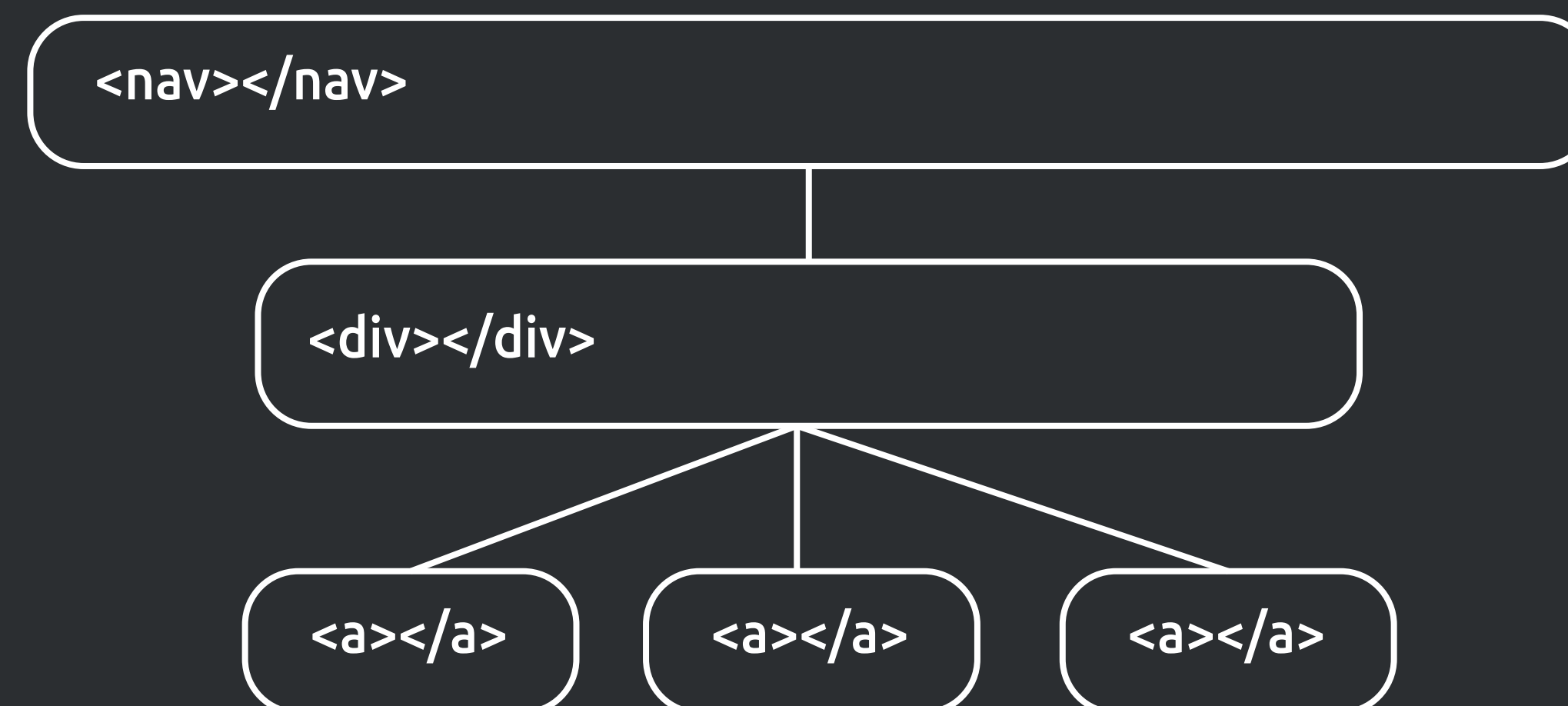
```
<a></a>
```

```
<a></a>
```

```
<a></a>
```

# Nesting

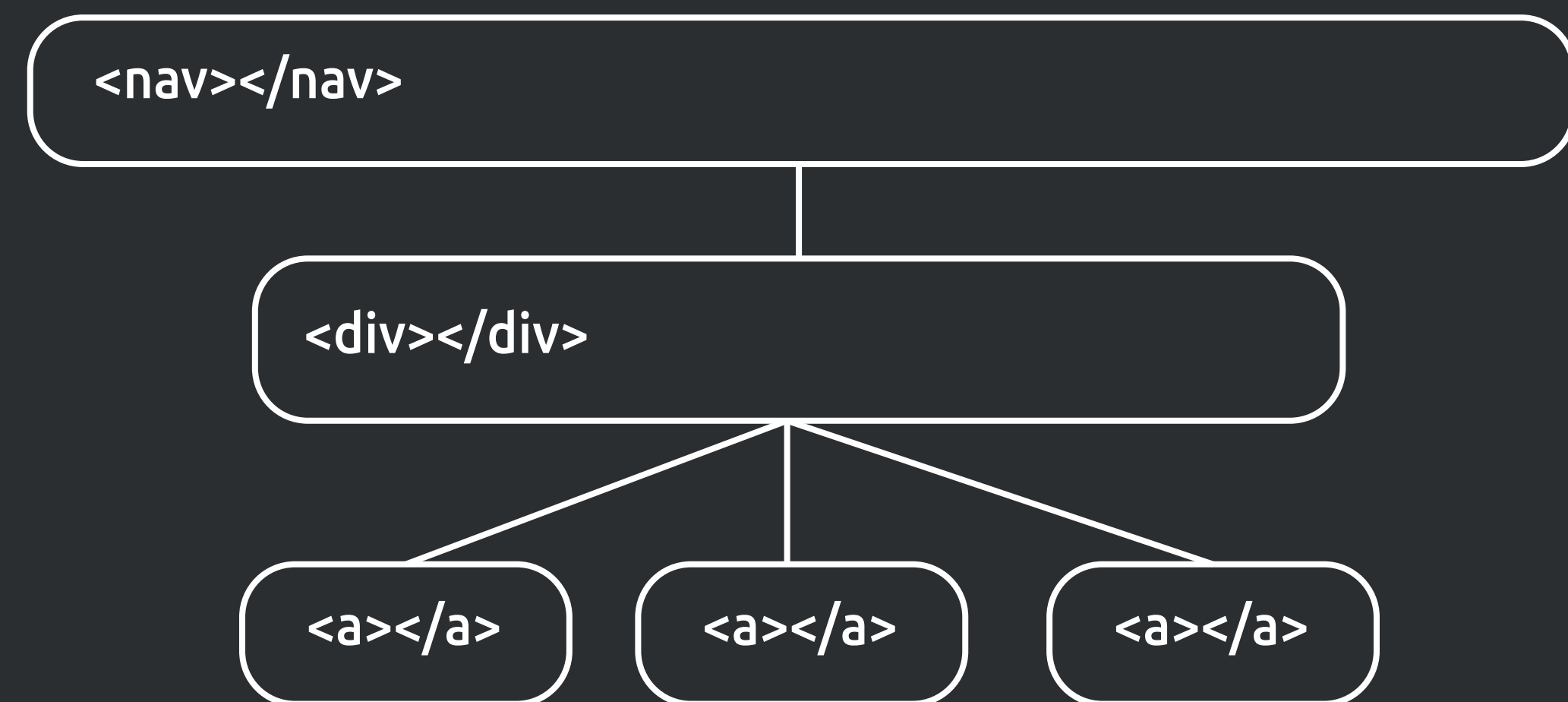
Nesting é quando agrupamos os nossos elementos dentro de outros elementos.  
Criando relações entre pais e filhos e irmãos.



# Nesting

Nesting é quando agrupamos os nossos elementos dentro de outros elementos.  
Criando relações entre pais e filhos e irmãos.

```
<nav>  
  <div>  
    <a href="">  
    <a href="">  
    <a href="">  
  </div>  
</nav>
```

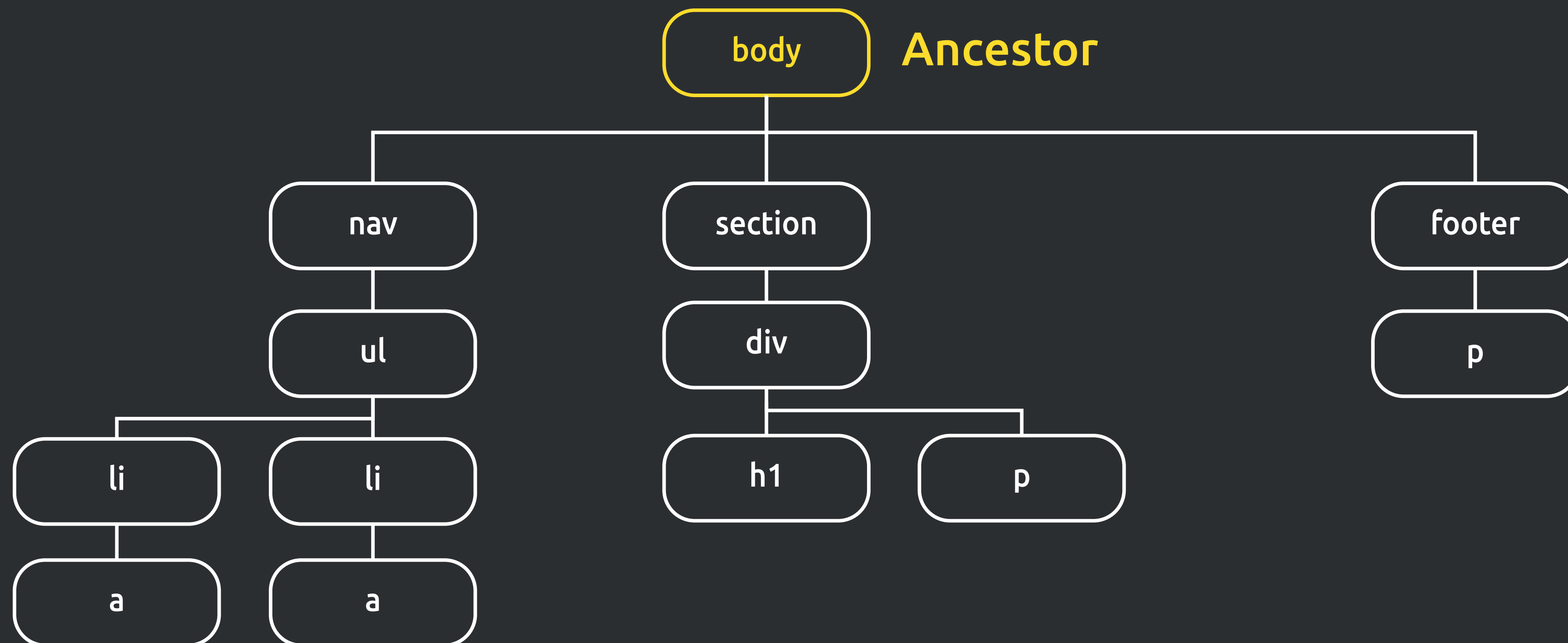


# Nesting e a Árvore de um documento HTML

Nesting é também a introdução a um conceito de HTML Document Tree.

Esta “árvore” tem a função de relacionar os nossos elementos.

A sua representação da DOM (Documento Object Model) será algo como isto:



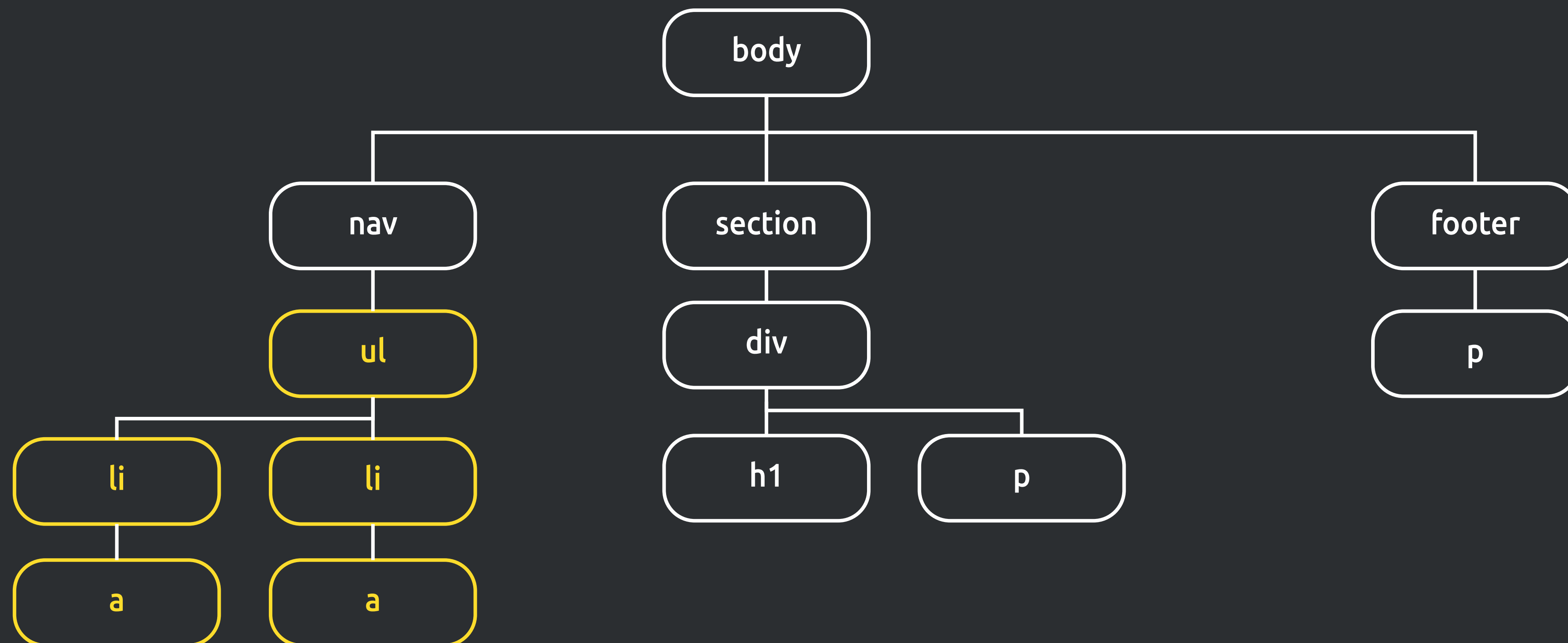


# Nesting e a Árvore de um documento HTML

Nesting é também a introdução a um conceito de HTML Document Tree.

Esta “árvore” tem a função de relacionar os nossos elementos.

A sua representação da DOM (Documento Object Model) será algo como isto:



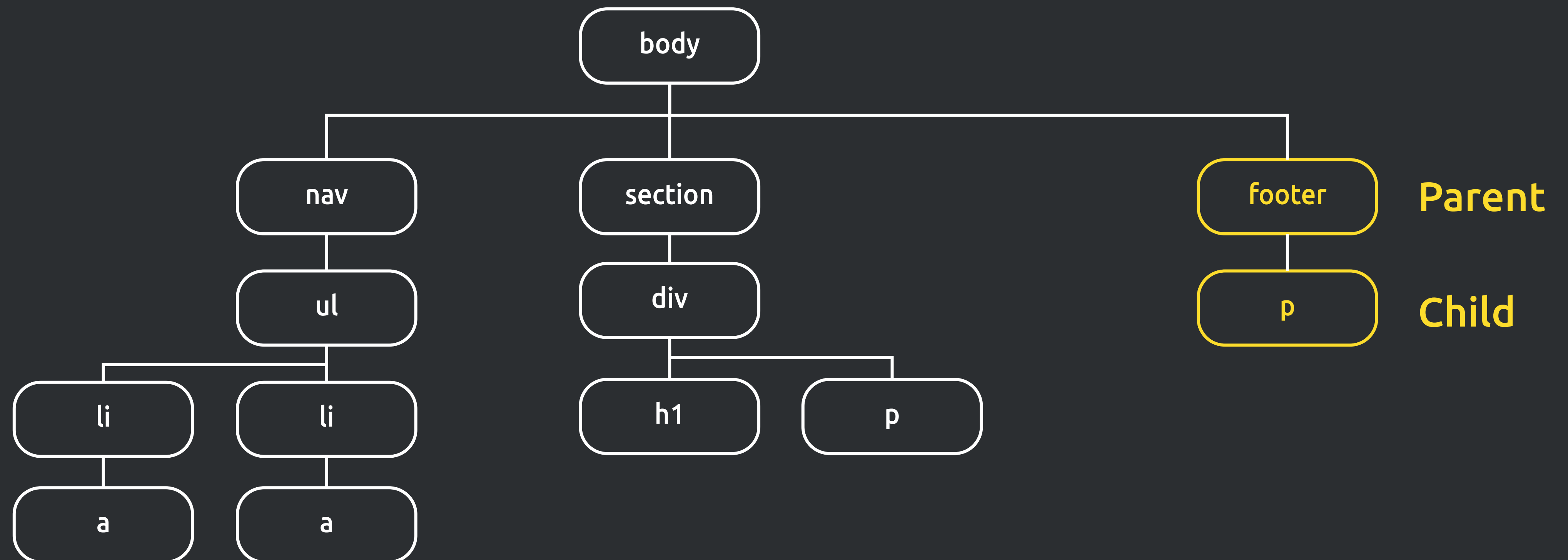
Descendants  
da nav

# Nesting e a Árvore de um documento HTML

Nesting é também a introdução a um conceito de HTML Document Tree.

Esta “árvore” tem a função de relacionar os nossos elementos.

A sua representação da DOM (Documento Object Model) será algo como isto:

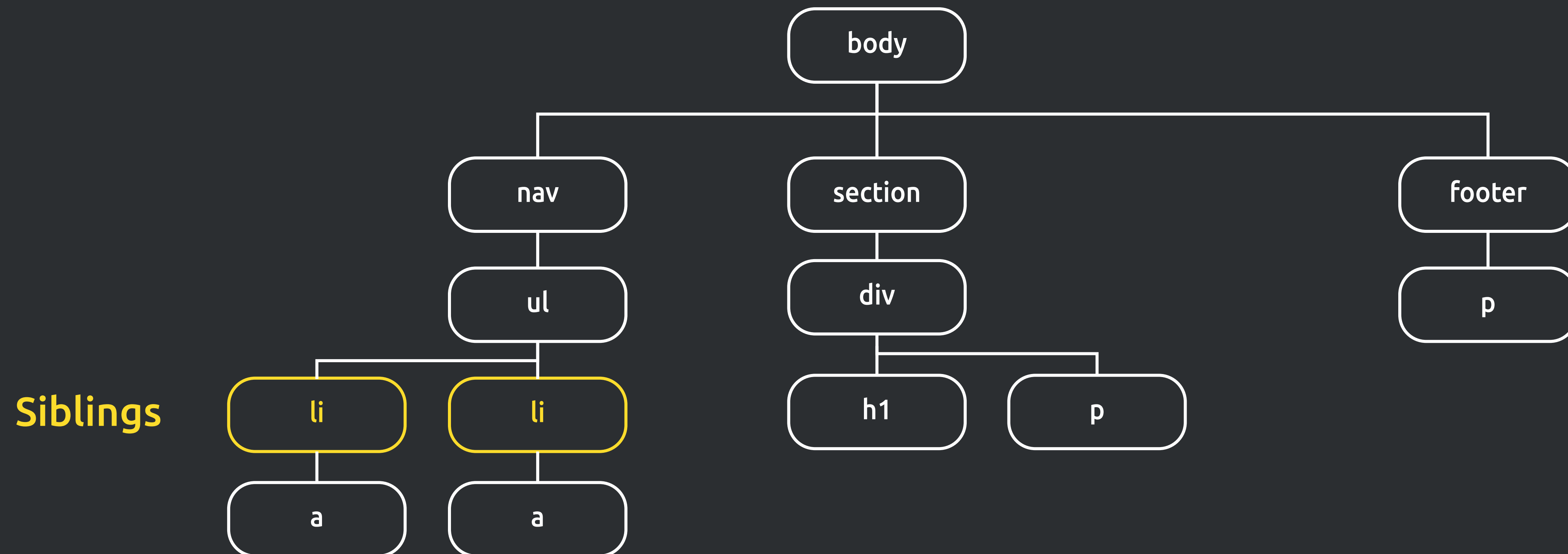


# Nesting e a Árvore de um documento HTML

Nesting é também a introdução a um conceito de HTML Document Tree.

Esta “árvore” tem a função de relacionar os nossos elementos.

A sua representação da DOM (Documento Object Model) será algo como isto:



# Hierarquia

Alguns elementos tem a sua própria hierarquia. Seguem “a sua própria linha” de como são representados.

```
<h1>Heading 1</h1>
```

```
<h2>Heading 2</h2>
```

```
<h3>Heading 3</h3>
```

```
<h4>Heading 4</h4>
```

```
<h5>Heading 5</h5>
```

```
<h6>Heading 6</h6>
```

```
<p>Paragraph</p>
```

ooo

# Heading 1

## Heading 2

### Heading 3

#### Heading 4

##### Heading 5

###### Heading 6

Paragraph

# Hierarquia

Alguns elementos tem a sua própria hierarquia. Seguem “a sua própria linha” de como são representados.

```
<ul>
  <li>List Item 1</li>
  <li>List Item 2</li>
  <li>List Item 3</li>
</ul>
```

ooo

List Item 1

List Item 2

List Item 3

# Hierarquia

Alguns elementos tem a sua própria hierarquia. Seguem “a sua própria linha” de como são representados.

```
<ol>
  <li>List Item 1</li>
  <li>List Item 2</li>
  <li>List Item 3</li>
</ol>
```

ooo

1. List Item 1
2. List Item 2
3. List Item 3

# Semântica

## Semântica?

Cada elemento tem o seu significado.

Uma `<nav>` é uma nav. Define o propósito de navegação.

Uma `<div>` não tem significado semântico, mas estruturalmente é uma divisão, agrupamento.

Uma `<section>` é uma secção.

O `<h1>` está relacionado com hierarquia tipográfica. Deve ser usado num contexto tipográfico.

Um `<p>` é um parágrafo, um bloco de texto.

Semântica introduz significado ao código que escreves.

Pensa antes de escrever.

Irás poupar-te muito tempo e terás menos dores de cabeça.

Um código com pés e cabeça.



# Exercício

