**01 Um pouquinho de posicionamento**

## Transcrição

Na barra lateral, ainda estão faltando os ícones que servirão de links para os nossos perfis nas redes sociais.



Podemos representar esses ícones como uma lista de links dentro do nosso <aside>. Dentro de cada tag <a>, podemos colocar, em vez de texto, a imagem com o logotipo da rede social:

<ul>

<li>

<a href="https://github.com/joaodasilva">

<img src="github.png" alt="Github">

</a>

</li>

<li>

<a href="https://twitter.com/joaodasilva">

<img src="twitter.png" alt="Twitter">

</a>

</li>

<li>

<a href="https://br.linkedin.com/pub/joão-da-silva/32/4/508">

<img src="linkedin.png" alt="LinkedIn">

</a>

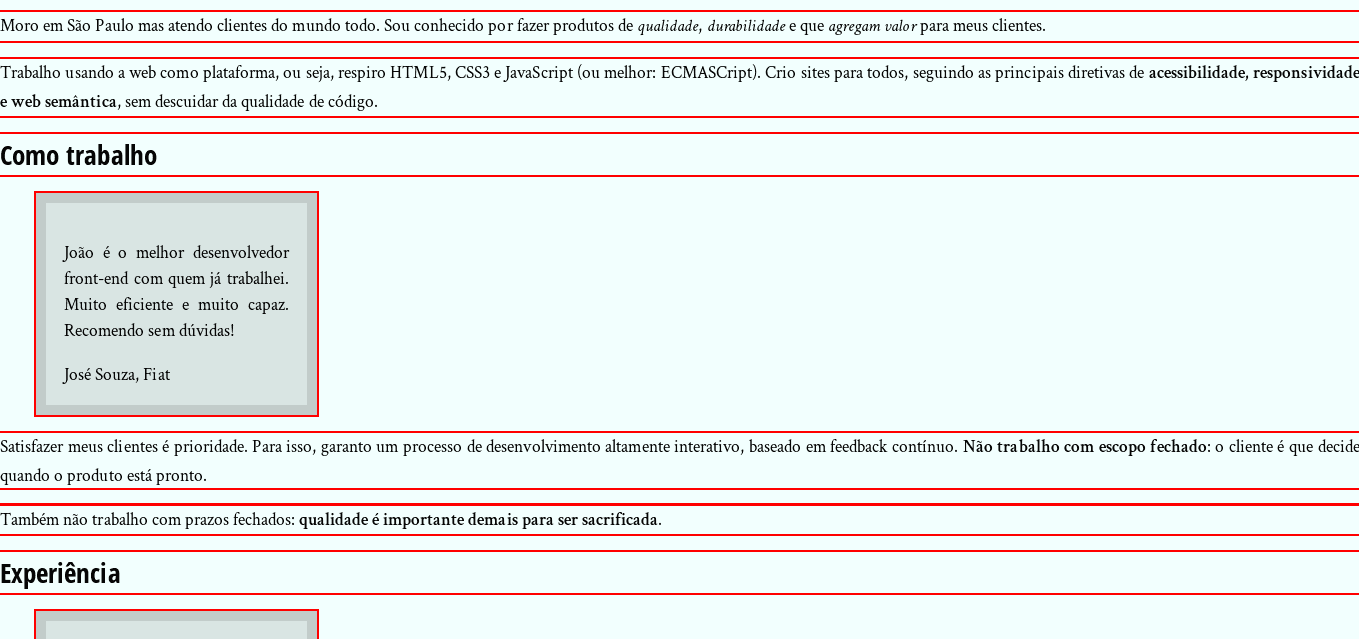
</li>

</ul>

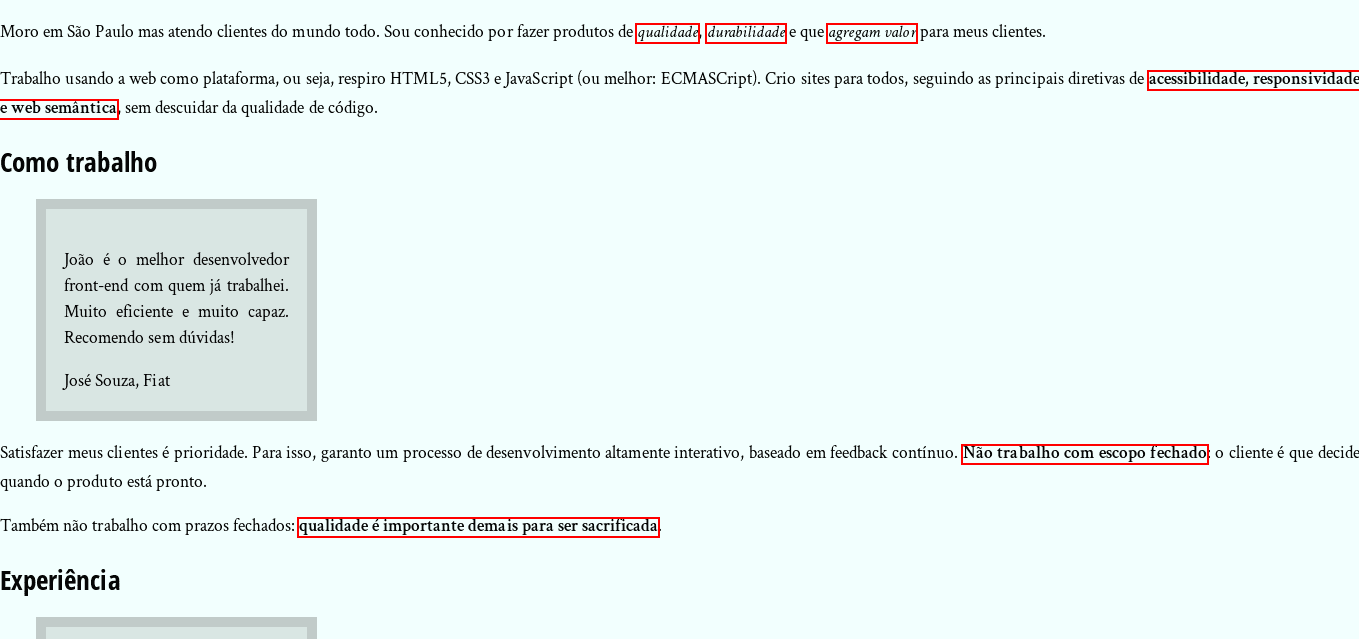
Isso já nos dá um resultado próximo do esperado: as imagens aparecem, são clicáveis mas ainda estão uma em cima da outra. Por que isso acontece e como resolver isso?



Se olharmos nosso layout atual e compararmos com o nosso HTML, veremos que, de modo geral, as tags que criamos – <p>, <header>, <aside>, <footer> – ficam uma embaixo da outra. Ou seja, esse é o comportamento padrão do navegador: empilhar as tags como caixas, uma em cima da outra.



Mas nem toda tag tem esse comportamento. Repare, por exemplo, nas tags <a>, <strong> e <em>. Elas ficam no meio do texto, não empilhadas.



Então, na verdade, temos esses dois comportamentos possíveis: empilhar ou ficar no meio do texto. Ou, em outras palavras, temos tags que são **block**, que empilham, e tags que são **inline**, que ficam no meio do texto.

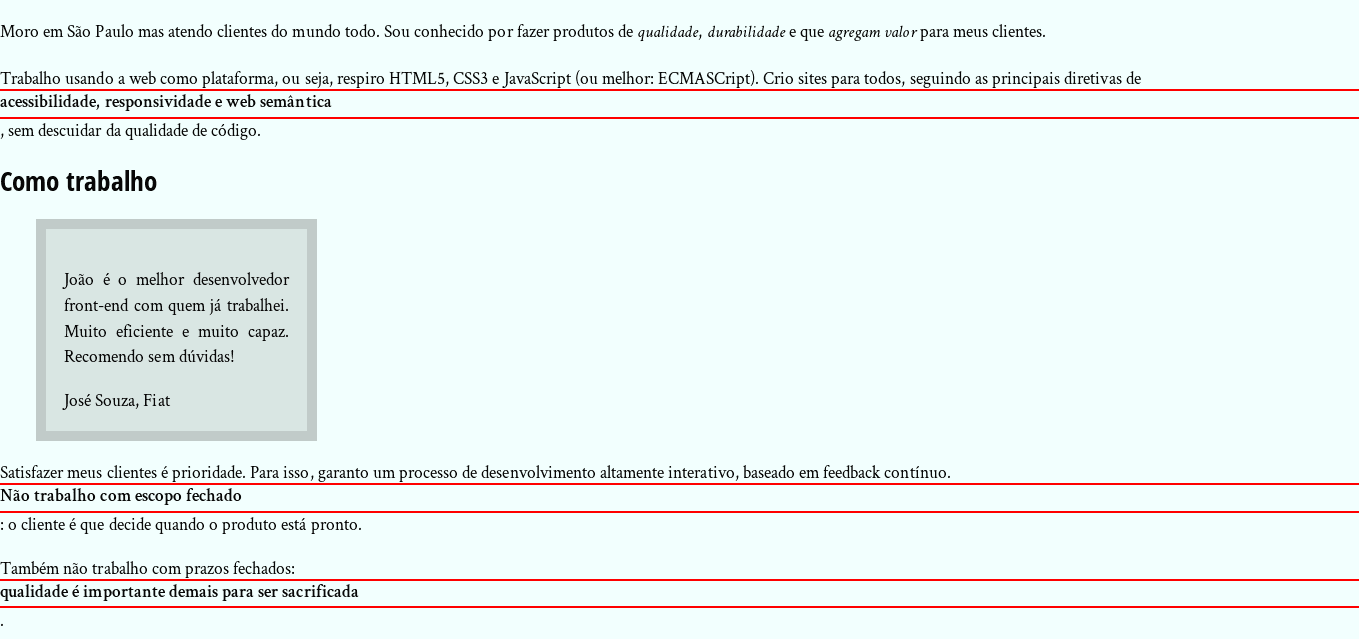
No caso da nossa lista de ícones, as tags <li> têm o comportamento de empilhar, são block, por isso os ícones aparecem um abaixo do outro. Se as removermos, deixando apenas os links com as imagens, conseguiremos o resultado desejado. Mas será que faz sentido mexer no HTML por uma questão **visual**? Faz sentido alterar a **semântica** da nossa página por isso?

Como é uma questão visual, devemos resolver usando a linguagem responsável por esse aspecto da nossa página: a linguagem CSS. No CSS, conseguimos controlar esse aspecto de um elemento por meio da propriedade display. Se quisermos fazer com que os <strong> tenham comportamento block, podemos escrever no CSS:

strong {

display: block;

}



No nosso caso, como queremos deixar os ícones das redes sociais em linha, como se fossem texto, podemos mudar o comportamento padrão do <li>, tornando-o inline:

li {

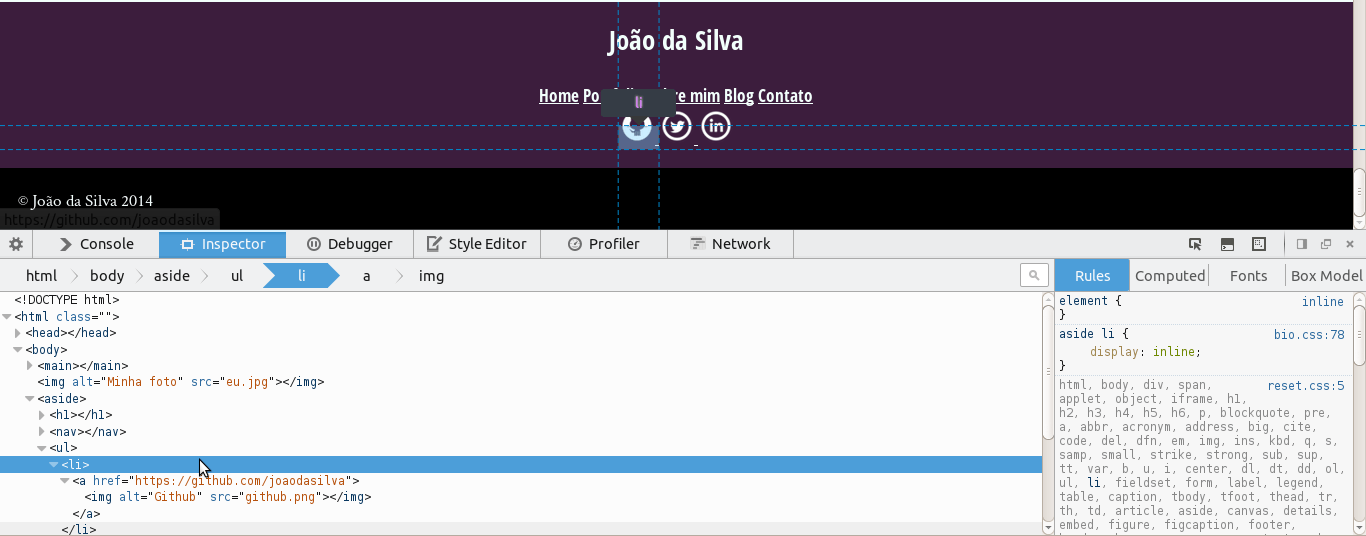
display: inline;

}

Com isso, conseguimos o efeito desejado. Ou quase; aparecem uns sublinhados estranhos do lado das imagens:



Se olharmos como o navegador "enxerga" nossos elementos, veremos que há algo estranho: os itens da lista estão **menores** que as imagens dentro deles:



O problema é que **não podemos mudar a altura de um elemento inline**, e é isso que a imagem tenta fazer com o <li>. O elemento inline **sempre** terá a altura da fonte. Ou seja, se quisermos que a imagem fique dentro do <li>, precisamos definir um font-size para ele igual à altura da imagem. Isso não é muito prático e, ainda por cima, não resolve o problema do sublinhado.

O que precisávamos, nesse caso, é que o <li> tivesse a altura e a largura correta automaticamente sem perder o comportamento do inline, de ficar em linha. Para resolver esse cenário, temos o **inline-block**, mais um valor da propriedade display. Usando-o nos <li>, conseguimos o resultado que queríamos.



Acabamos alterando o layout do menu logo acima das redes sociais, também. Resolveremos esse problema na próxima aula.

# 02 Tudo inline

É possível mudar o valor da propriedade display para qualquer elemento da página, inclusive de **todos** os elementos da página! Experimente colocar o código abaixo no seu arquivo site.css:

\* {

display: inline;

}

O que acontece?

### Opinião do instrutor

Quando colocamos essa regra no nosso arquivo CSS, **todos** os elementos se comportarão como elementos inline, ou seja, todos os elementos ficarão **na mesma linha**. Veremos, por exemplo, os cabeçalhos ficarem lado a lado com o texto e as citações (blockquotes) quebrarem entre as linhas.

# 03 inline versus inline-block

**Qual a diferença entre elementos inline e inline-block?**

Top of Form

* Alternativa correta



Elementos inline não podem ter padding

Bottom of Form

  Alternativa correta



Nenhuma, são apenas suportados em navegadores diferentes

  Alternativa correta



Elementos inline podem ter altura definida

  Alternativa correta



Elementos inline não podem ter altura definida

O valor inline-block para a propriedade display surgiu para podermos definir **altura** e **largura** em elementos *inline*. Fora isso, ambos os elementos podem ter um padding definido, ambos ficam em linha e ambos são suportados por todos os navegadores modernos.