

A decorative graphic on the left side of the slide, consisting of a network of white lines and small circles on a dark blue background, resembling a circuit board or a neural network.

DESENVOLVIMENTO WEB

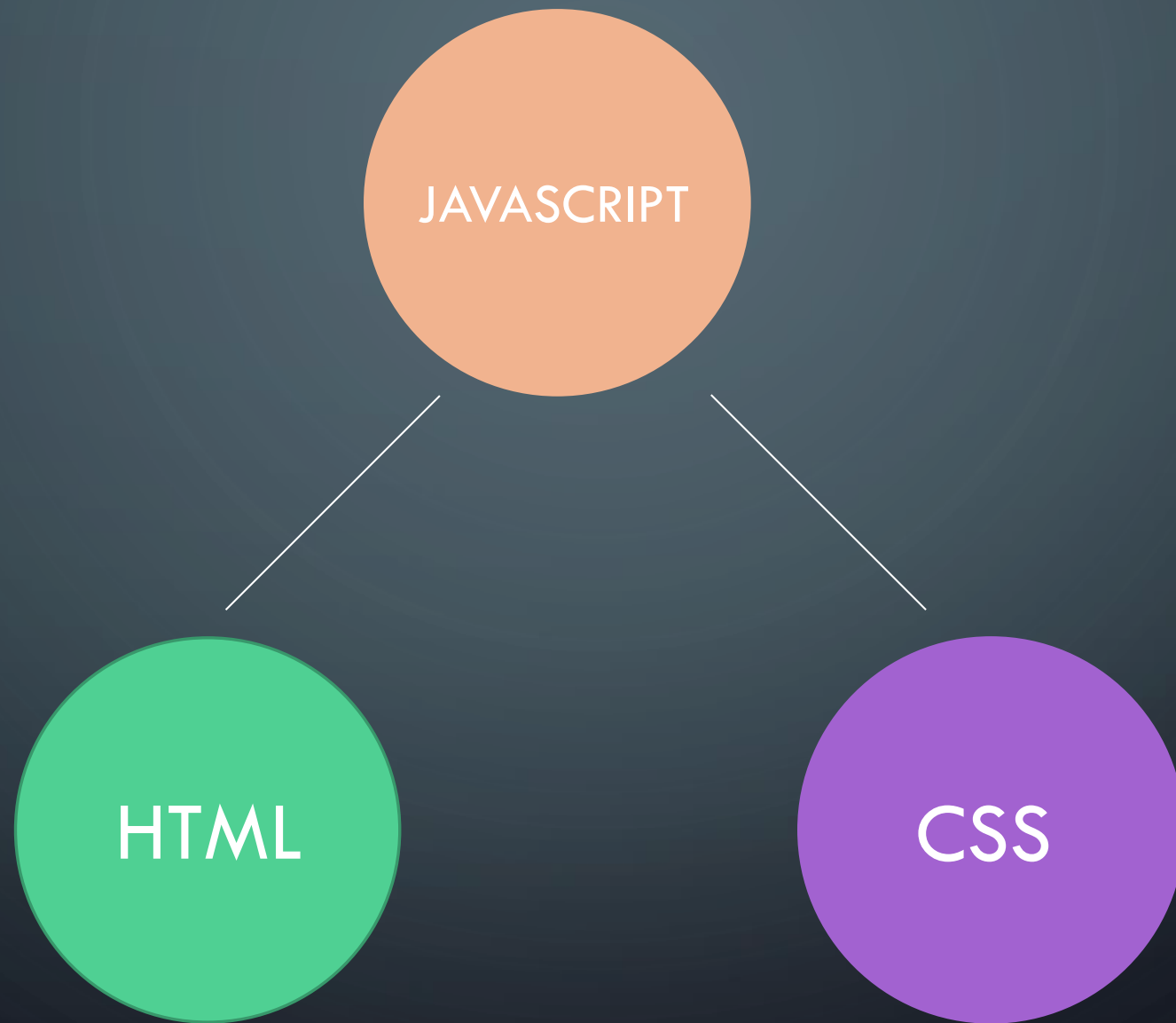
FULL STACK

CSS

- CSS é a abreviatura para Cascading Style Sheets.
- Define o layout do documento HTML atribuindo estilos para eles
 - Fontes
 - Cores
 - Margens
 - Altura, largura
 - Entre outros

CSS

- HTML – estruturar conteúdo
- CSS – formatar o conteúdo
- Javascript – interagir com o conteúdo (ações)



ESTRUTURA DE DIRETÓRIOS - CONVENÇÃO

- index.html – Página Inicial
 - styles – Diretório dos estilos CSS
 - estilo1.css
 - estilo2.css
 - scripts – Diretório dos scripts
 - script1.js
 - script2.js

CSS - SINTAXE

```
seletor {  
    background: #ff0000;  
}
```

SELETORES

- Seletores servem para indicar qual elemento HTML vai receber determinado estilo.
- Eles podem ser dos seguintes tipos:
 - Através da tag. Ex: `div { background: red; }`
 - Através do ID do elemento. Ex: `#meu-id { background: blue }`
 - Através de um atributo. Ex: `input[type="text"] { color: red; }`
 - Através de uma classe. Ex: `.minha-classe { border-radius: 20px }`

```

<DOCTYPE html>
<html>
  <head>
    <title>Meu form</title>
    <meta charset="utf-8" />
    <link href="example1.css" rel="stylesheet" type="text/css" media="screen" />
  </head>
  <body>
    <seletor>
      Meu background é vermelho
    </seletor> <br />

    <seletor id="rosa">
      Meu background é rosa
    </seletor> <br />

    <seletor class="amarelo">
      Meu background é amarelo
    </seletor>

    <form>
      <label>E-mail
        <input type="email" placeholder="E-mail" required />
      </label>

      <button type="submit">Enviar</button>
    </form>

  </body>
</html>

```

Meu background é vermelho
 Meu background é rosa
 Meu background é amarelo

E-MAIL


```

1  * {
2    font-family: Arial, Helvetica, sans-serif;
3    font-size:14px;
4  }
5  seletor {
6    background: #ff0000;
7  }
8
9  input[type="email"] {
10   color: black;
11   border-radius: 30px;
12   border: 1px solid gray;
13   padding: 15px;
14 }
15
16 input[type="email"]:invalid {
17   color: red;
18 }
19
20 label {
21   text-transform: uppercase;
22 }
23
24 #rosa {
25   background: pink;
26 }
27
28 .amarelo {
29   background: yellow;
30 }
31
32 form {
33   padding: 20px 10px;
34 }
35

```


PSEUDO ELEMENTOS E CLASSES

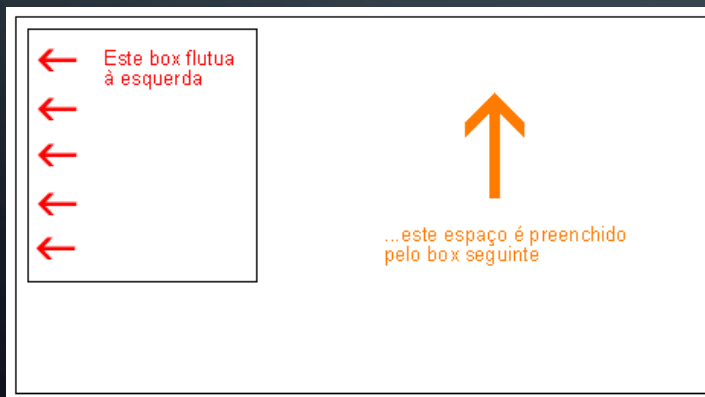
<http://bit.ly/1vEEYnj>

EXERCÍCIO 1

- Pegar o formulário feito na última aula e aplicar estilos nos elementos. Usar os seletores de tag, id classe e atributo.
- Plus 1: colorir em vermelho quando um formulário estiver inválido
- Plus 2: Adicionar um degrade no botão que envia o formulário

FLUTUANDO ELEMENTOS

- Um elemento pode ser 'flutuado' à esquerda ou à direita com uso da propriedade float.
- `div { float:left; }`
- `img { float:left; }`





Uma imagem flutuada

Iste quidem veteres inter ponetur honeste, qui vel mense brevi vel toto est iunior anno. Utor permissio, caudaeque pilos ut equinae paulatim vello unum, demo etiam unum, dum cadat elusus Interdum volgus rectum videt, est ubi peccat. Si veteres ita miratur laudatque poetas, ut nihil anteferat, nihil illis comparet, errat. Si quaedam nimis antique, si peraque dure Interdum volgus rectum videt, est ubi peccat. Si veteres ita miratur laudatque poetas, ut nihil anteferat, nihil illis comparet, errat. Si quaedam nimis antique si peraque dure

EXERCÍCIO 2

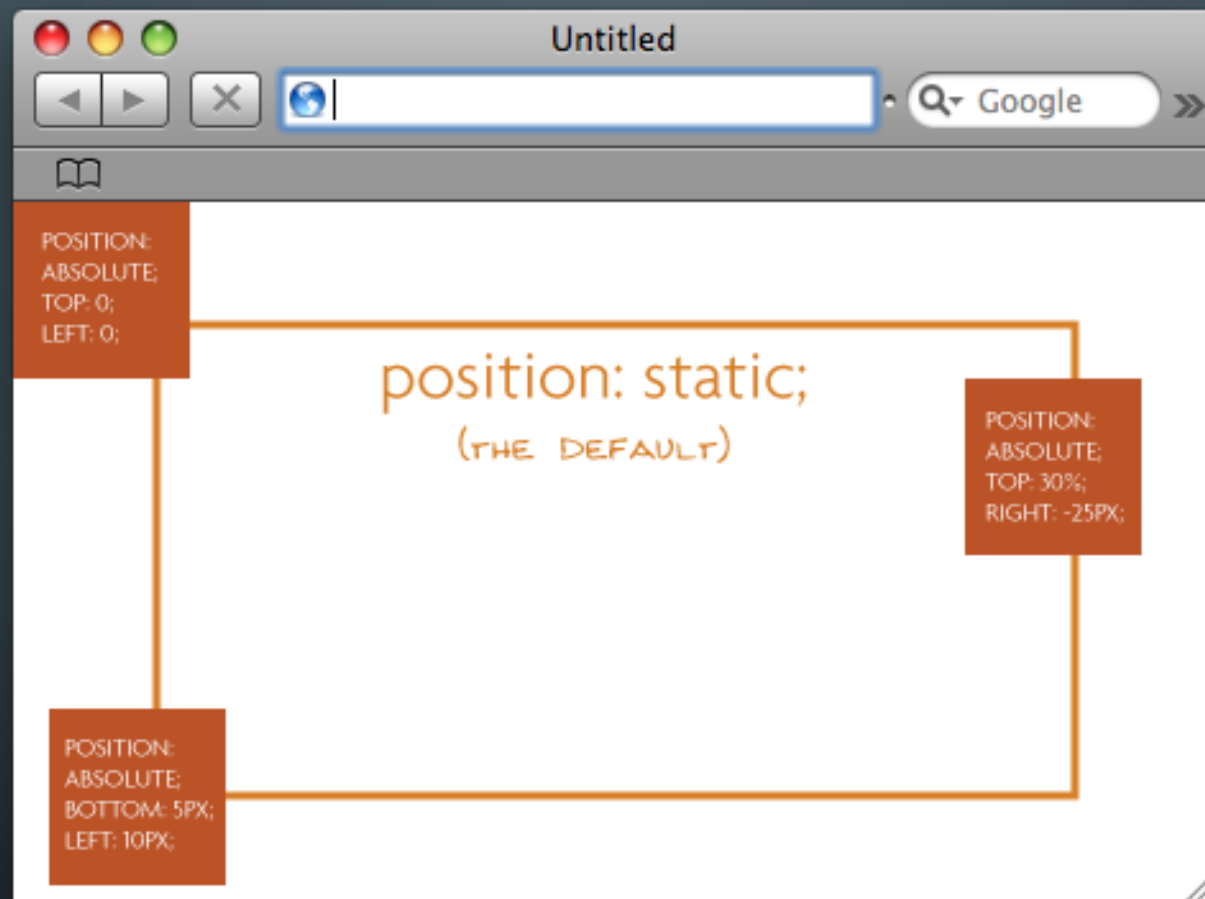
- Criar uma lista `` de 15 elementos com `<div>` dentro delas com caixas de tamanhos variados, utilizando seletores de classes. Essas caixas deverão ser flutuantes.
- Plus 1: O 4 elemento deverá apresentar um comportamento diferente das demais.
- Plus 2: definir uma altura máxima para lista e adicionar uma barra de rolagem.

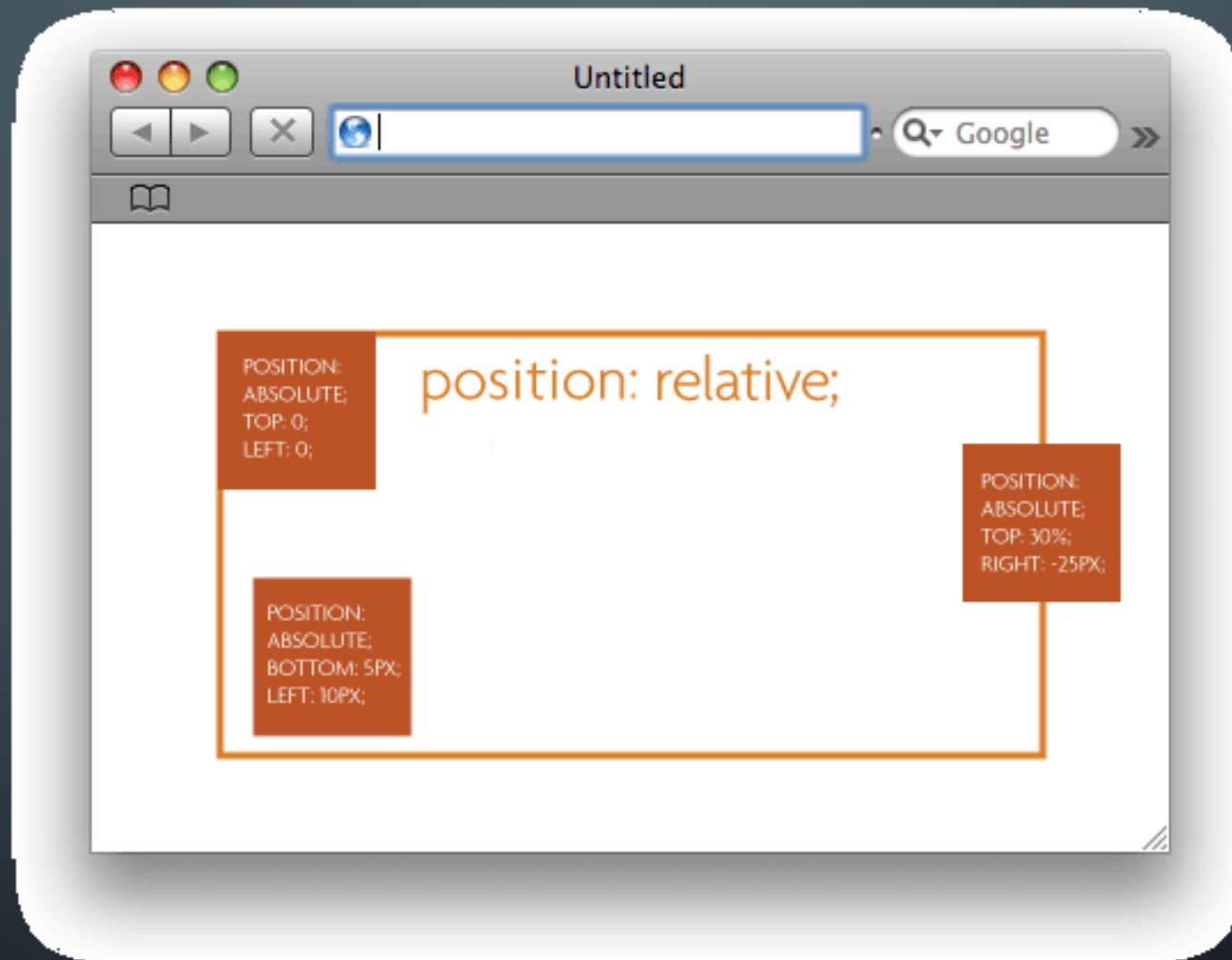
CAMADAS

- Você pode definir um comportamento de camadas para elementos do layout HTML.
- Para isso você precisa adicionar uma dessas propriedades:
- `div { position: relative; }` ou `div {position: absolute}`
- Para ordenar a posição delas adicionamos o atributo `z-index` a classe CSS.
- `Div#atrás { position: relative; z-index: 1 }` `div#frente {position:relative; z-index: 2}`

RELATIVE VS ABSOLUTE

- Relativo vai pegar parâmetros e posicionamento relativos ao seu pai.
- Absolute vai pegar informações e posicionamento do documento raiz. *





EXERCÍCIO 3

- Desenvolver o front-end baseado no layout que está dentro da pasta `exercio/cartas` no repositório do Github.
- PS: Interações de mouse e focus são um diferencial.