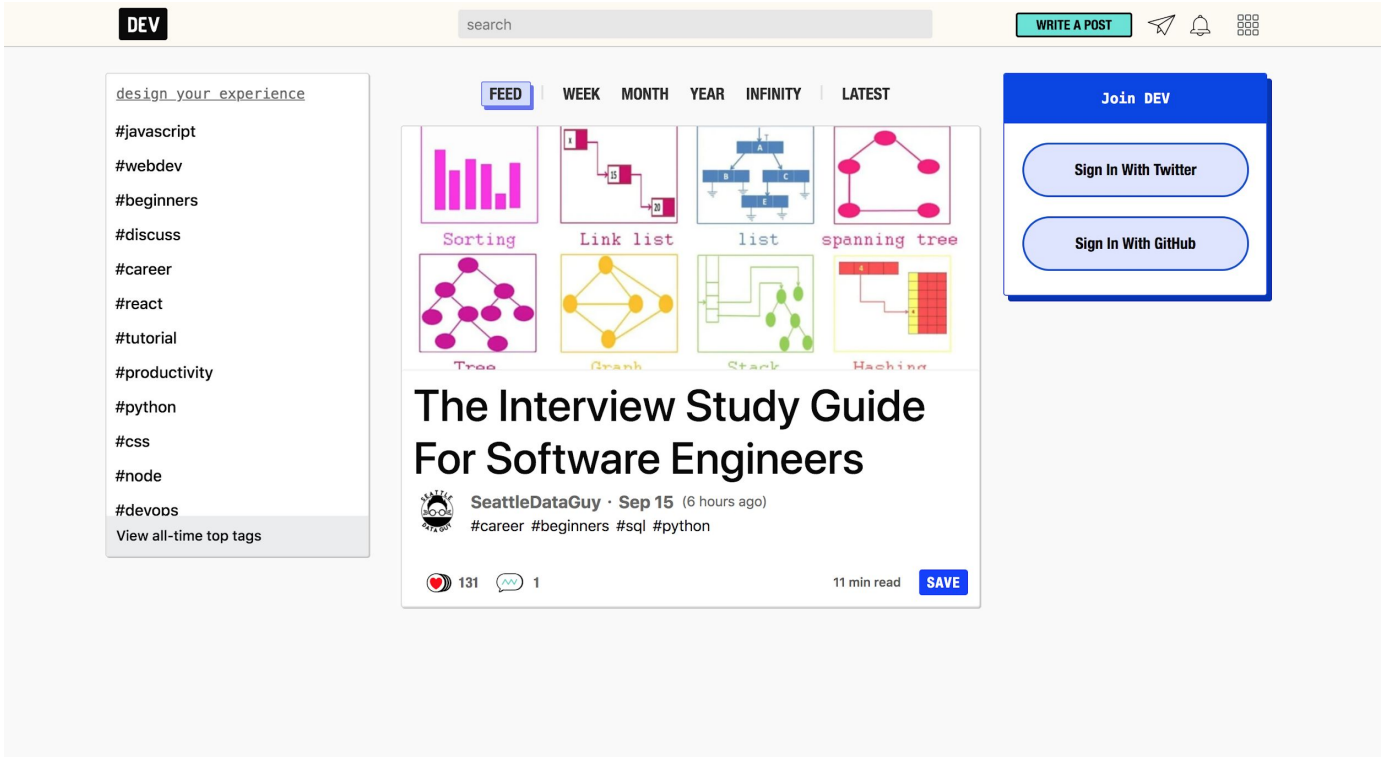
Responsividade

FUTURE4

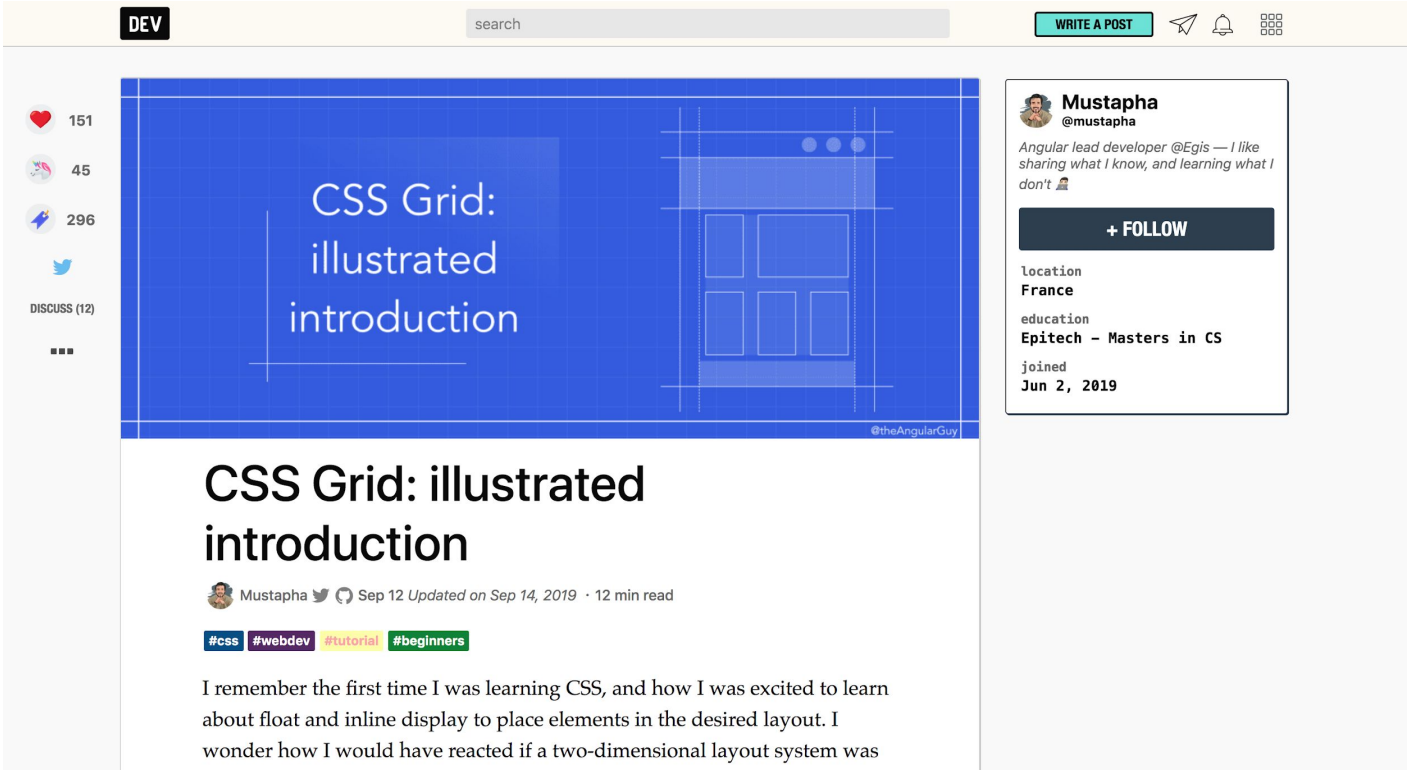
Adendo: “correção” dev.to

Como reproduzimos esta página?



FUTURE4

Como reproduzimos essa página?



FUTURE4

Voltando: Responsividade

FUTURE4

O que é?

FUTURE4

Definição

“Responsividade” - substantivo feminino

Capacidade de responder rápida e adequadamente ao que lhe é perguntado, adaptando-se às circunstâncias.

FUTURE4

Princípios

- Cada tipo de dispositivo tem uma estrutura mais apropriada para apresentação de conteúdo
- Uma estrutura responsiva é a que trata o conteúdo como se fosse água
  - Se você serve água em:
    - Um copo, ela assume o formato do copo
    - Uma garrafa, ela assume o formato da garrafa
    - Uma xícara, ela assume o formato da xícara

Estratégias comuns

FUTURE4

Mobile First

- Quando usar?
  - A maior parte do acesso à sua plataforma vier de dispositivos mobile.
- Vantagens
  - O Google **valoriza** mais sites que dão suporte para mobile.
  - Maior chance de cobrir os usuários *
- Desvantagens
  - Pode ser difícil de transpor funcionalidades do seu site/aplicação para o desktop

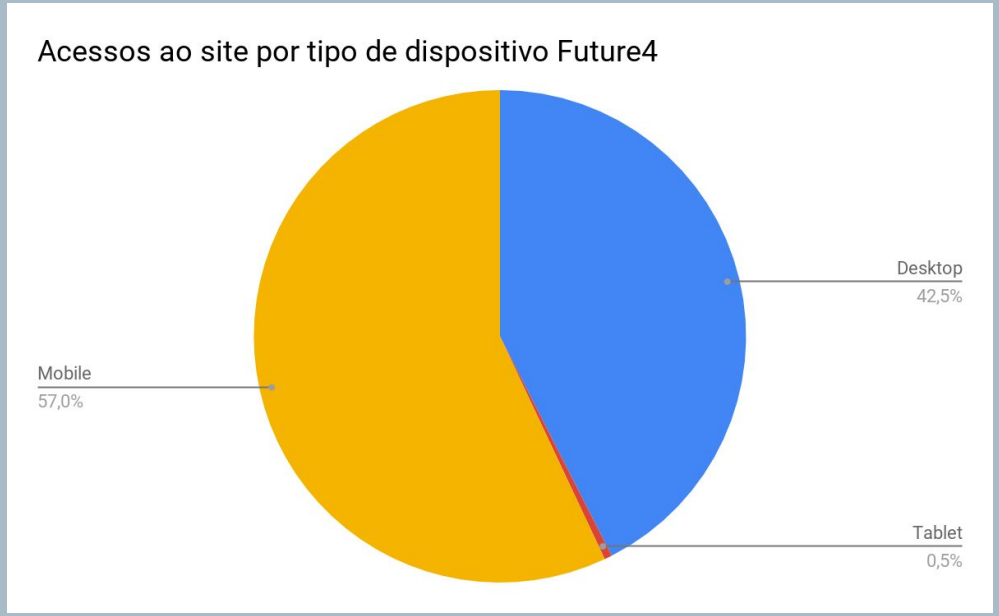
FUTURE4

Desktop First

- Quando usar?
  - Quando a maior parcela dos seus usuários acessam o site pelo desktop.
  - Quando você já tiver aplicações nativas para iOS/Android, ou outra alternativa nativa.
- Vantagens
  - Permite interfaces rebuscadas com muitas funcionalidades
- Desvantagens
  - A **adaptação** para mobile pode se tornar impossível, fazendo com que outra versão diferente tenha que ser criada

FUTURE4

Como decidimos por onde começar?



FUTURE4

Pausa: 5 min 

FUTURE4

Vamos para a parte técnica!

```
        }).done(function(response) {
            for (var i = 0; i < response.length; i++) {
                var layer = L.marker(
                    [response[i].latitude, response[i].longitude]
                    // ,{icon: myIcon}
                );
                layer.addTo(group);

                layer.bindPopup(
                    "<p>" + "Species: " + response[i].species + "<
                    "<p>" + "Description: " + response[i].descript
                    "<p>" + "Seen at: " + response[i].latitude + '
                    "<p>" + "On: " + response[i].sighted_at + "</p
                )
            }

            $('select').change(function() {
                species = this.value;
            });
        });
    }
$.ajax({
            url: queryURL,
            method: "GET"
        }).done(function(response) {
            for (var i = 0; i < response.length; i++) {
                var layer = L.marker(
                    [response[i].latitude, response[i].longitude]
                    // ,{icon: myIcon}
                );
                layer.addTo(group);
```

FUTURE4

HTML

- tag meta viewport

```
<meta name="viewport" content="width=device-width, user-scalable=no, initial-scale=1">
```

- *user-scalable* → se o site permite que o usuário dê zoom usando dois toques ou com o gesto de pinça
- *width* → tamanho da viewport, podendo ser um valor em px ou valores especiais como ***device-width***
- *initial-scale* → Estabelece uma relação 1:1 entre pixels CSS e pixels independentes de dispositivo.

- Quando 1 pixel não é 1 pixel
  - Permite que lidemos com DPIs diferentes, bem comuns entre dispositivos móveis

HTML

- ausência do viewport

ZOOM ENABLED

ZOOM DISABLED

CSS

- `@media`
  - Define para qual tipo de dispositivo as regras CSS devem ser aplicadas
    - Screen → Qualquer tipo de tela
    - Print → Somente para impressão
  - Permite a criação de condições, ou *queries* que quando cumpridas, “ativam” um conjunto de regras CSS.

CSS

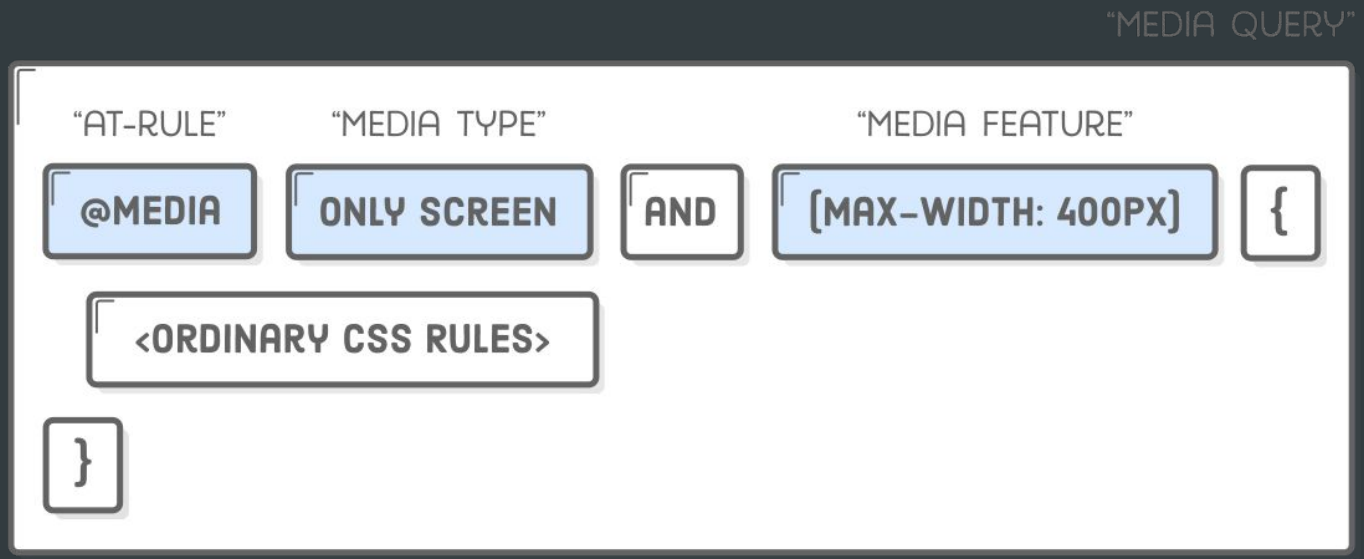
- `@media`
  - Exemplo de uso inline:

```
<link rel="stylesheet" media="(max-width: 800px)" href="example.css"/>
```

  - Faz com que a folha de estilo (inteira) referenciada só entre em vigor quando a tela tiver um tamanho **menor ou igual** a 800px

CSS

- `@media` estrutura:



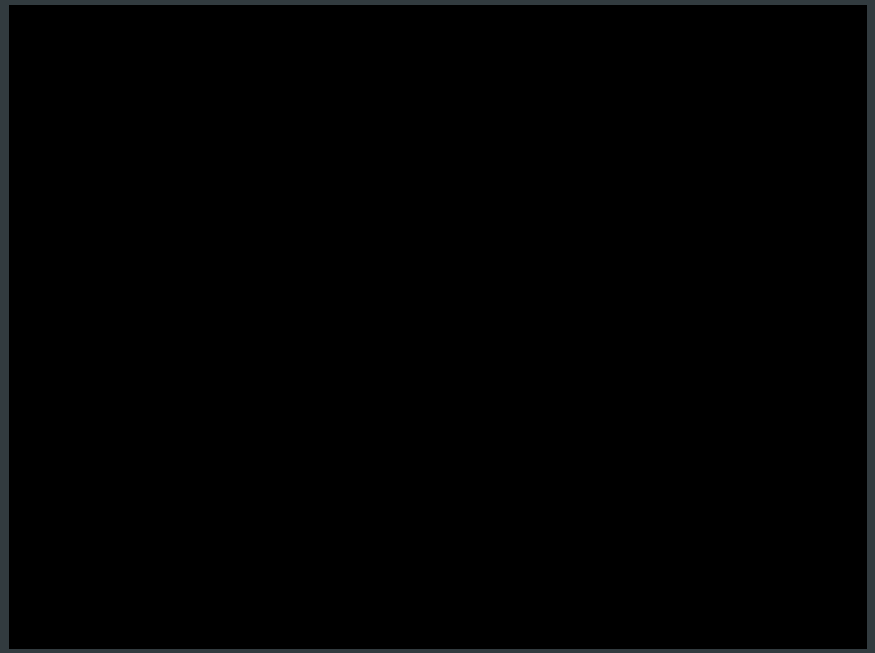
CSS

- @media
  - Exemplo de uso no CSS:
    - As regras abaixo só entrarão em vigor quando a condição for satisfeita.

```
@media (max-width: 800px)
{
   .main-container {
     display: none;
   }
}
```

CSS

- Uso prático no layout:



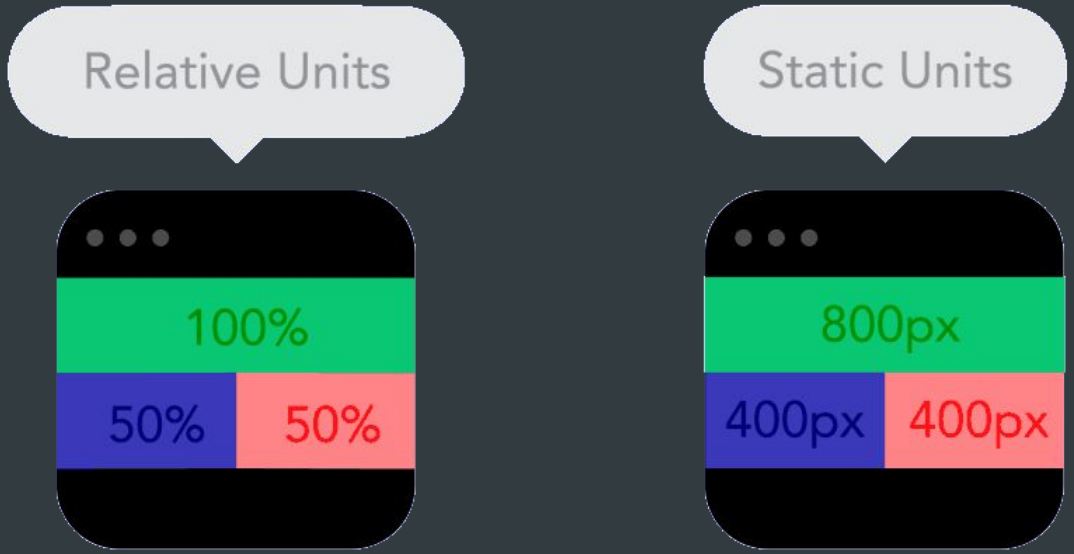
CSS

- Uso prático no layout:

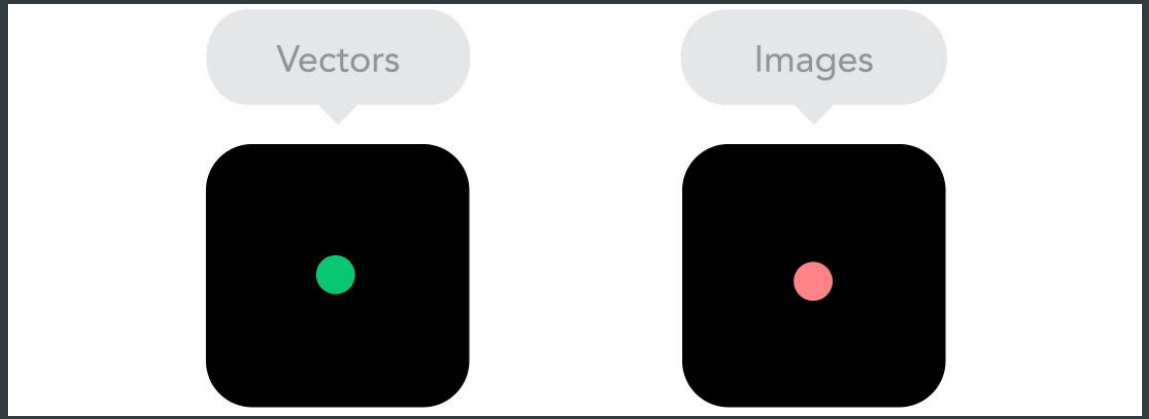
```
@media screen and (min-width: 960px) {
  body { background-color: green; }
}

@media screen and (min-width: 500px) and (max-width: 960px) {
  body { background-color: pink; }
}

@media screen and (max-width: 500px) {
  body { background-color: yellow; }
}
```

Disponível aqui

Responsividade e unidades relativas



Imagens escalares e vetoriais



Pausa: 10 min 

FUTURE4

Coding Together

Vamos adaptar o exercício do coding together de ontem!

- Crie regras para telas menores do que 800px que:
  - Divida igualmente o espaço para o HEADER e MENU
  - Divida o espaço vertical entre todos os elementos
  - Troque a cor do content1 para vermelho
  - Esconda o menu e faça o HEADER ocupar o espaço restante

Usando o resultado do exercício anterior adicione:

- Crie regras para telas menores do que 500px que:
  - O elemento com “CONTENT 1” seja escondido
  - Faça com que o elemento FOOTER tenha posição fixa, arrume a largura e altura e posicione em bottom: 0
  - Aumente a altura do “CONTENT2” para 1800px

Pausa: 5 min 

FUTURE4

Review

FUTURE4

Tópicos

- Correção dev.to
- Definição conceitual de responsividade
- Estratégias de aplicação
- HTML e CSS
- Media queries

Dúvidas?

FUTURE4

Obrigado!

FUTURE4