

Responsividade

Labenu_



O que vamos ver hoje?

- O que é Responsividade?
- HTML Responsivo
 - meta-viewport
- CSS Responsivo
 - unidades
 - media query
- Estratégias



O que é Responsividade?

Labenu_



O que é responsividade?

- **Responsividade** é a capacidade de **responder rápida e adequadamente** ao que lhe é perguntado, adaptando-se às circunstâncias
- No contexto de programação Web, responsividade significa a capacidade de responder rápida e adequadamente às **mudanças de tamanho das telas dos dispositivos** no qual o site está sendo visualizado



O que é responsividade?

- Classificamos as telas em: telas de **desktop** e **mobile**
- Ser uma tela de **desktop** significa que a tela é de um **computador**
 - A largura maior que a altura
- Ser uma tela de **mobile** significa que a tela é de um **celular**, que pode rodar Android ou iOS
 - A altura maior que a largura



HTML Responsivo: meta-viewport

Labenu_



HTML Responsivo: meta-viewport

- **Viewport** é o nome que damos para a área visível de uma página
- A tag **meta** (colocado no head), se tiver, como name, **viewport**, pode ser usada para definir algumas configurações referentes a viewport

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



content

- **width=device-width**
 - Este atributo define o quanto a área visível
- **user-scalable=no**
 - Diz se o usuário pode, ou não, dar zoom na tela (em um device mobile)
- **initial-scale=1**
 - Diz o quanto os pixels do CSS devem se adaptar para tela(em um device mobile)



CSS Responsivo: Unidades

Labenu_



CSS Responsivo: Unidades



- As unidades podem ser divididas em
 - **Absolutas**
 - px, pt, cm, mm
 - **Relativas**
 - em, rem, %, vh, vw



Unidades Absolutas

- São unidades cujo **tamanho final** que será renderizado é **fixo**, independente do dispositivo
- **px**
 - Representa os **pixels**
 - 1 pixel representa **o menor elemento de disposição** de um dispositivo



Unidades Absolutas



- **pt**
 - Representa os **points**
 - 1 point equivale a **0.35cm**
- **cm, mm**
 - Representam, respectivamente, os **centímetros** e os **milímetros**

Vamos ver na prática!



Unidades Relativas

- São unidades cujo **valor depende de outro propriedade** relacionada a tamanhos
- **em**
 - Relativo ao tamanho da **fonte do elemento**
 - 2em significa "2 vezes o tamanho da fonte do elemento"

Vamos ver na prática! 



Unidades Relativas

- **rem**
 - Relativo ao tamanho da **fonte do elemento root** (tag **html** - normalmente 16px)
 - Normalmente, o HTML coloca o tamanho 16px
 - 2rem significa "2 vezes o tamanho da fonte do elemento root"

Vamos ver na prática! 



Unidades Relativas

- %
 - O tamanho terá uma **porcentagem** em relação a **mesma propriedade do elemento pai**
- vh
 - viewport-height
 - Porcentagem relativa à **altura do device** em si
 - Se o device tem 50cm de altura, $100vh = 50cm$ e $1vh = 0.5cm$



Unidades Relativas

- **vw**
 - viewport-width
 - É uma porcentagem **relativa ao comprimento do device** em si
 - Se o device tem 50cm de comprimento, $100vw = 50cm$ e $1vw = 0.5cm$

Vamos ver na prática! 



Extra: max e min-width



- A propriedade **max-width** estabelece a largura máxima de um elemento, e a propriedade **min-width**, a largura mínima.
- Interessante para **limitar** até qual valor a largura pode expandir ou diminuir dependendo do tamanho da tela.

Vamos ver na prática!





Exercício 1

- Use HTML e CSS para criar o esquema ao lado
- Cumpra o que cada elemento diz sobre si próprio

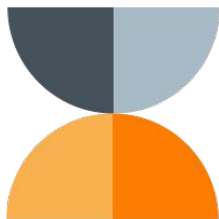
- A minha fonte tem 3 vezes o tamanho da fonte do meu root.
- Os meus paddings verticais possuem 2 vezes o tamanho da minha fonte.
- A minha altura assume vinte por cento do comprimento da tela toda.
- Já o meu comprimento assume metade da altura da tela toda.

- A minha fonte é setenta por cento da fonte do meu pai.
- O meu comprimento é dez por cento do comprimento do meu pai.
- A minha altura é dez por cento da altura do meu pai.



Pausa para relaxar 🧘

10 min



- **Viewport** é a área visível pelo usuário
- A tag **meta** pode ser usada para fazermos algumas configurações da viewport
- Unidades:
 - **Absolutas:** px, pt, cm, mm
 - **Relativas:** em, rem, %, vh, vw



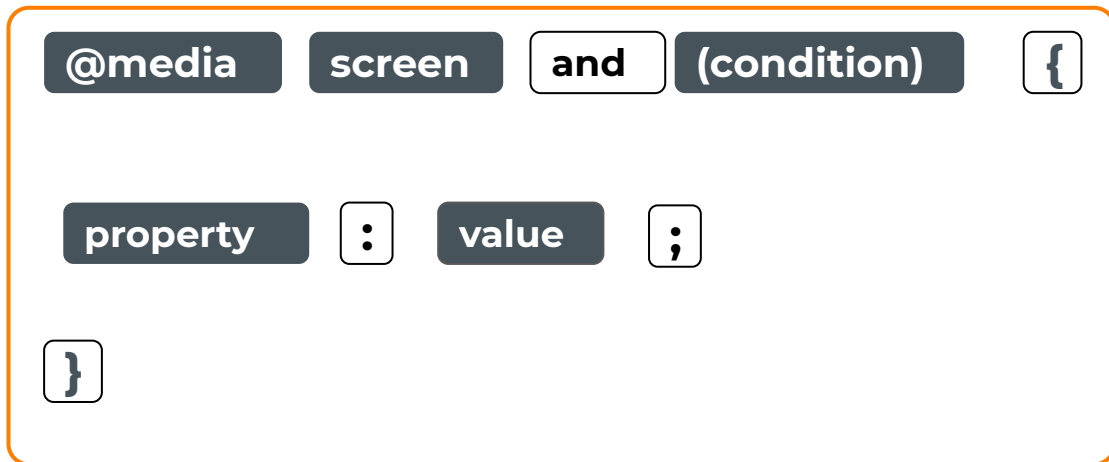
CSS Responsivo: media query

Labenu_



CSS Responsivo: media query

- **@media** é uma das estruturas do CSS que nos permite indicar **regras diferentes** de CSS dependendo de certas **condições referentes à tela**



CSS Responsivo: media query

- Sobre os tamanhos da tela, recomenda-se o seguinte:

```
@media screen and (min-device-width : 320px) and (max-device-width : 480px) {  
    /* celulares */
```

```
}
```

```
@media screen and (min-device-width : 481px) and (max-device-width : 800px) {  
    /* tablets */
```

```
}
```

```
@media screen and (min-device-width : 1200px) {  
    /* computadores */
```

```
}
```

Vamos ver na prática! 



CSS Responsivo: media query

- A seguinte regra faz com que o elemento com a classe **main-container** desapareça quando o comprimento da tela for, no máximo, 800px

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





Exercício 2

- Crie o site ao lado
- A primeira imagem é o site para desktop e a segunda é o site para mobile

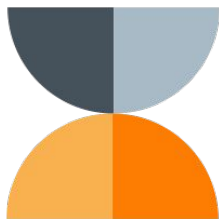
Item 1	Item 2	Item 3
Item 4	Item 5	

Item 1
Item 2
Item 3
Item 4
Item 5



Pausa para relaxar 🧘

5 min



- **Media queries** são uma estrutura que permite alterar o comportamento inteiro do nosso layout dependendo do tamanho das telas
- `@media screen and (max-width: 800px)`
 - um exemplo de uso de media query



Estratégias de Responsividade

Labenu_



Estratégias de Responsividade

- Existem **várias estratégias** para organizar a responsividade. Aqui, falaremos de duas que são opostas uma em relação a outra:
 - **Desktop First**
 - **Mobile First**



Estratégias de Responsividade

- **Desktop First:** Fazemos o site pensando, **primeiro** na versão para **computador**
 - Vantagens: permite interfaces robustas com muitas funcionalidades
 - Desvantagens: a adaptação para mobile pode se tornar impossível, fazendo com que tenhamos que criar outra versão separada para mobile



Estratégias de Responsividade

- **Desktop First:** Fazemos o site pensando, **primeiro** na versão para **computador**
 - Quando usar?
 - Quando a maior parcela dos seus usuários acessam o site pelo computador
 - Quando já temos um app mobile e queremos disponibilizar as funcionalidades na Web



Estratégias de Responsividade

- **Mobile First:** Fazemos os site pensando, **primeiro** na versão para **celular**
 - Vantagens: maior chance de cobrir usuários, buscadores (google) valorizam mais sites que dão suporte mobile
 - Desvantagens: pode ser difícil de transpor funcionalidades do site para a versão desktop



Estratégias de Responsividade

- **Mobile First:** Fazemos o site pensando, **primeiro** na versão para **celular**
 - Quando usar?
 - Quando a maior parcela dos seus acessos for feita através de dispositivos mobile



Estratégias de Responsividade



- **Estratégias envolvendo CSS:**
 - Breakpoints
 - Unidades Relativas
 - Valores máximos e mínimos



Breakpoints

- É recomendável que tentemos separar o nosso layout para telas de **tamanhos diferentes**
- Para não ter que colocar para cada um dos tamanhos na mão, normalmente, definimos **intervalos** e colocamos o layout da tela para cada um deles (**media queries**)
- Esses intervalos chamamos de **breakpoints**



Estratégias de responsividade 🚔

- Breakpoints:

With Breakpoints



Without Breakpoints



Estratégias de responsividade



- Unidades relativas:

Relative Units

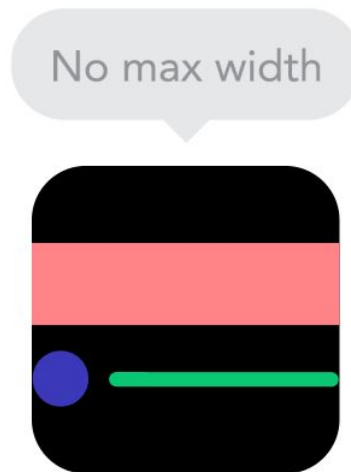
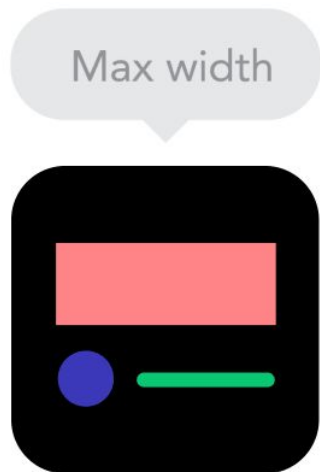


Static Units



Estratégias de responsividade

- Tamanhos mínimos e máximos



Resumo

Labenu_



Resumo

- Responsividade é o conceito de adaptar o seu site a diversos tipos de dispositivos
- A tag **meta** permite que façamos configurações na área visível do usuário (chamada de **viewport**)
- Um grande aliado nosso na responsividade são as unidades
 - Absolutas: px, pt, cm
 - Relativas: em, rem, %, vh, vw



Resumo

- Outro aliado nosso são as **media queries**, que permite **definir regras** de layout totalmente diferentes dependendo do tamanho da tela

```
@media screen and (min-device-width : 320px) and (max-device-width : 480px) {  
    /* celulares */  
}
```

```
@media screen and (min-device-width : 1200px) {  
    /* computadores */  
}
```



Resumo

- Estratégias de responsividade
 - Mobile first VS Desktop first
 - Breakpoints
 - Unidades Relativas
 - Tamanhos mínimos e máximos



AVISO IMPORTANTE

Labenu_



AVISO IMPORTANTE 🚒

AS AULAS DE CSS ACABARAM!!!



AVISO IMPORTANTE

- **Mas, pera, acabou, mesmo?** Não!
- Há mais coisas de CSS para aprender:
 - **Transforms**
 - **Transactions**
 - **Animations**
 - **Outros tipos de estilizações**



Dúvidas? 🧐

Labenu_





Obrigado(a)!