

<labmm2></labmm2>

<section>

ua.deca.ntc.2018.2019.t12

</section>

<footer>

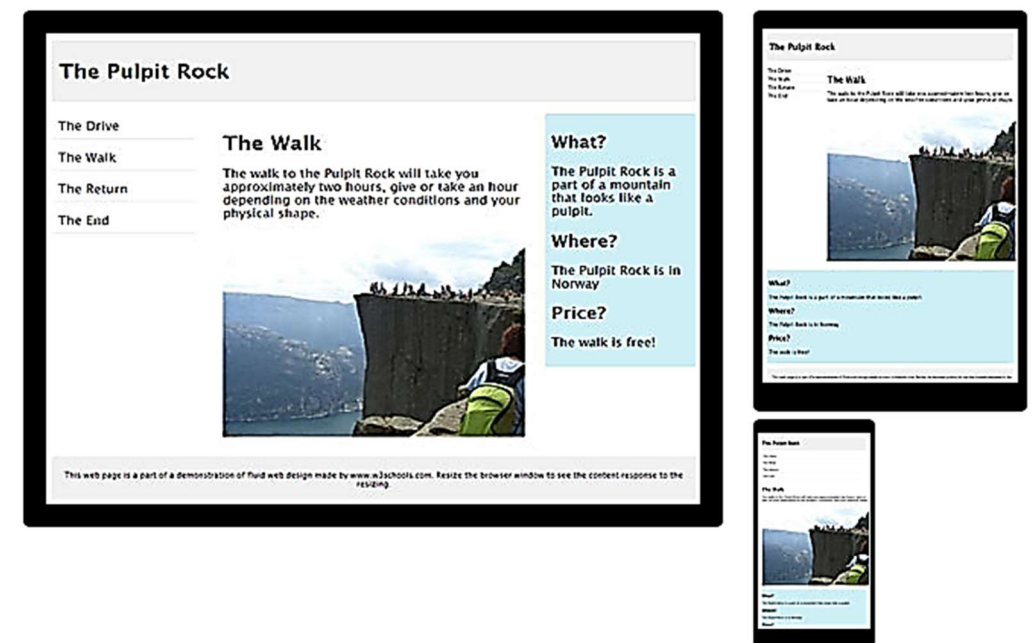
helder caixinha | caixinha@ua.pt

</footer>

# CSS – RESPONSIVE WEB DESIGN (RWD)

Multiplicidade de dispositivos e de ecrãs → Desafios acrescidos para o design das páginas Web (a definição dos layouts das páginas)

Responsive Web Design (RWD) → Estratégias e técnicas, que utilizando o HTML e o CSS (e o JavaScript 😊) permitem construir Layouts para as páginas Web que se adequam a uma visualização otimizada nessa multiplicidade de dispositivos e de ecrãs



# RWD – MEDIA TYPES & MEDIA FEATURES

Permitem delimitar a aplicação dos estilos a diferentes dispositivos (Media Types) e/ou a diferentes características desses dispositivos (Media Features), por exemplo dos seus ecrãs, etc

## Media Types (Dispositivos)

- **all** → todos os dispositivos
- **print** → impressoras
- **screen** → ecrãs
- **speech** → leitores de ecrãs (utilizadores com necessidades especiais)

Media Features (Características desse dispositivo) → ver lista completa em: <https://goo.gl/HZ6UTA>

- **min-width** → largura mínima da área de visualização (por exemplo, a janela do browser)
- **max-width** → largura máxima da área de visualização (por exemplo, a janela do browser)
- **min-height** → altura mínima da área de visualização (por exemplo, a janela do browser)
- **max-height** → altura máxima da área de visualização (por exemplo, a janela do browser)
- **orientation** → orientação do dispositivo (**portrait** ou **landscape**)
- **min-resolution** → resolução mínima permitida pelo dispositivo (em **dpi** ou **dpcm**)
- **max-resolution** → resolução máxima permitida pelo dispositivo (em **dpi** ou **dpcm**)

# RWD – MEDIA QUERIES

**Media Queries** → Regras/condições para a aplicação dos estilos, definidas com base em **Media Types** e **Media Features**

```
body {  
    background-color: lightblue;  
    font-size: 28px;  
}
```

Estilos genéricos para a página

```
@media screen and (min-width: 400px) {  
    body {  
        background-color: lightsalmon;  
        font-size: 16px;  
    }  
}
```

Estilos que serão aplicados quando a página for visualizada num ECRÃ e a LARGURA DA ÁREA DE VISUALIZAÇÃO FOR DE 400PX OU SUPERIOR (400px é o BREAKPOINT)

```
@media print {  
    body {  
        background-color: white;  
        font-size: 12px;  
    }  
}
```

Estilos que serão aplicados quando se IMPRIME a página

# RWD – MEDIA QUERIES (OUTROS EXEMPLOS)

@media **not** print {

**not** – negação da condição

}

@media **only** screen **and** (max-width: 600px) {

**only** – impede que browsers antigos, que não suportem as medias queries, mostrem os estilos definidos

}

@media screen **and** (min-width: 600px) **and** (max-width: 900px) **and** (min-width: 1100px) {

}

A página ser visualizada num **ECRÃ**

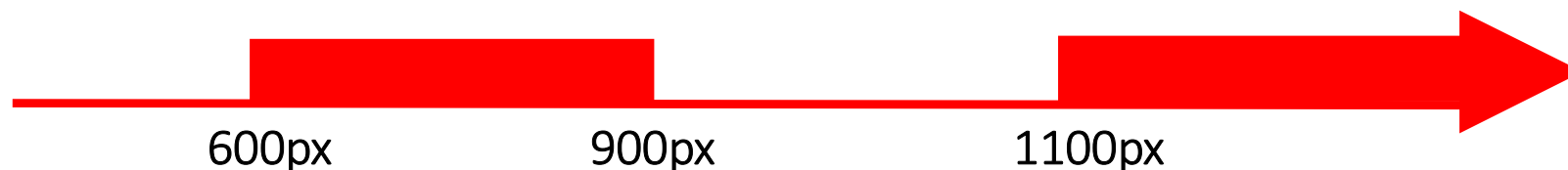
**E**

a largura da área de visualização ser de 600px ou superior **E**

a largura da área de visualização ser de 900px ou inferior

**OU**

a largura da área de visualização ser de 1100px ou superior



# RWD – MEDIA QUERIES & MOBILE FIRST

Estratégia Mobile First na definição das Media Queries e respetivos breakpoints

```
/* Extra small devices (0px --- 599px) */  
@media only screen and (max-width: 599px) {  
    ← 599  
}  
/* Small devices (600px --- 767px) */  
@media only screen and (min-width: 600px) {  
    600 →  
}  
/* Medium devices (768px --- 991px) */  
@media only screen and (min-width: 768px) {  
    768 →  
}  
/* Large devices (992px --- 1199px) */  
@media only screen and (min-width: 992px) {  
    992 →  
}  
/* Extra large devices (1200 e superior) */  
@media only screen and (min-width: 1200px) {  
    1200 →  
}
```

A ordem de definição das media queries é feita dos ecrãs MENORES para os ecrãs MAIORES

Breakpoints típicos (largura da área de visualização)

600px  
768px  
992px  
1200px



# CSS – VIEWPORT

**Viewport** – área onde é feita a visualização da página Web (por exemplo, a janela do browser)

“Antigamente” ao visualizar num ecrã pequeno, uma página Web pensada para um ecrã grande, esta era reduzida para caber nesse ecrã de menores dimensões



Para controlar como é feita essa visualização no viewport e evitar essa redução deve inserir-se no **HEAD** da página

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

**width=device-width** → a largura da página adapta-se à largura do viewport  
**initial-scale=1.0** → escala inicial com que a página é mostrada no viewport





# CSS – FLEXBOX

Permite a definição de Responsive Layouts SEM o recurso a elementos flutuantes e/ou posicionamentos ☺

Estilos na caixa MAIN

**display: flex;** → display flexível (os elementos filhos do main serão flexíveis também)

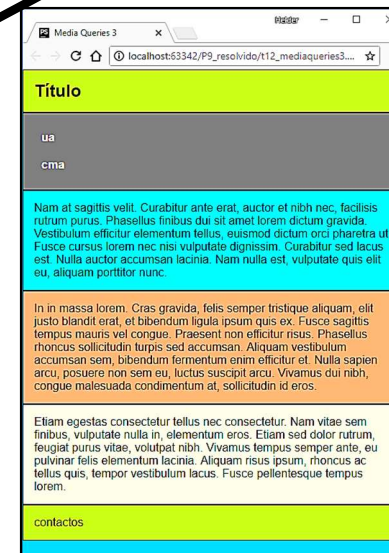
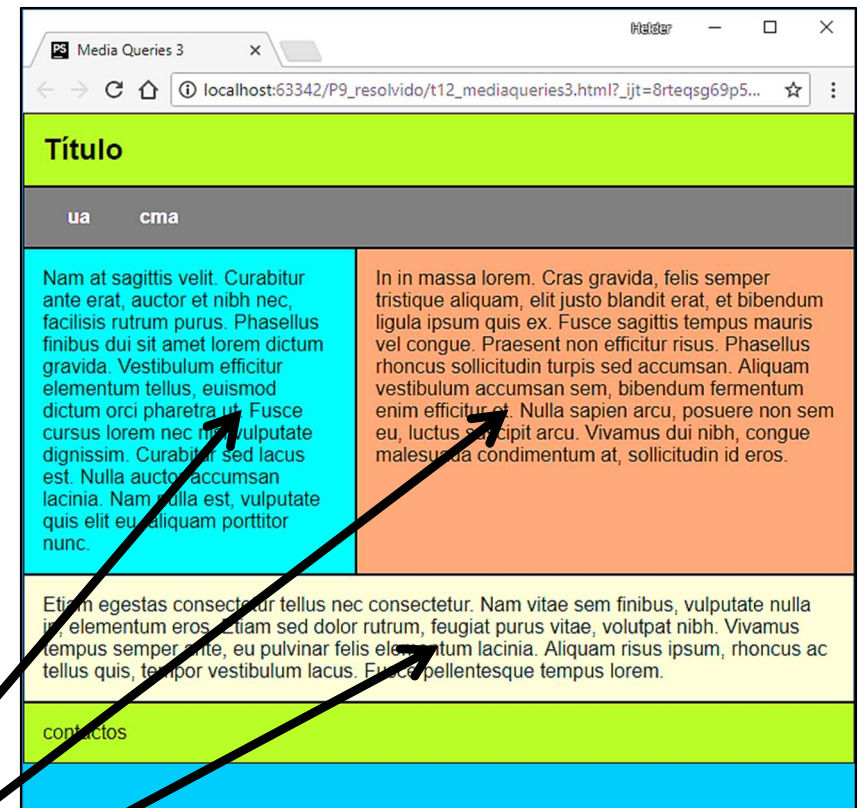
**flex-wrap: wrap;** → os elementos filhos do main farão o “wrap” dentro do container

Nos elementos filhos do main definem-se as suas larguras base (iniciais)

**aside → flex-basis: 40%;**  
**section → flex-basis: 60%;**  
**article → flex-basis: 100%;**

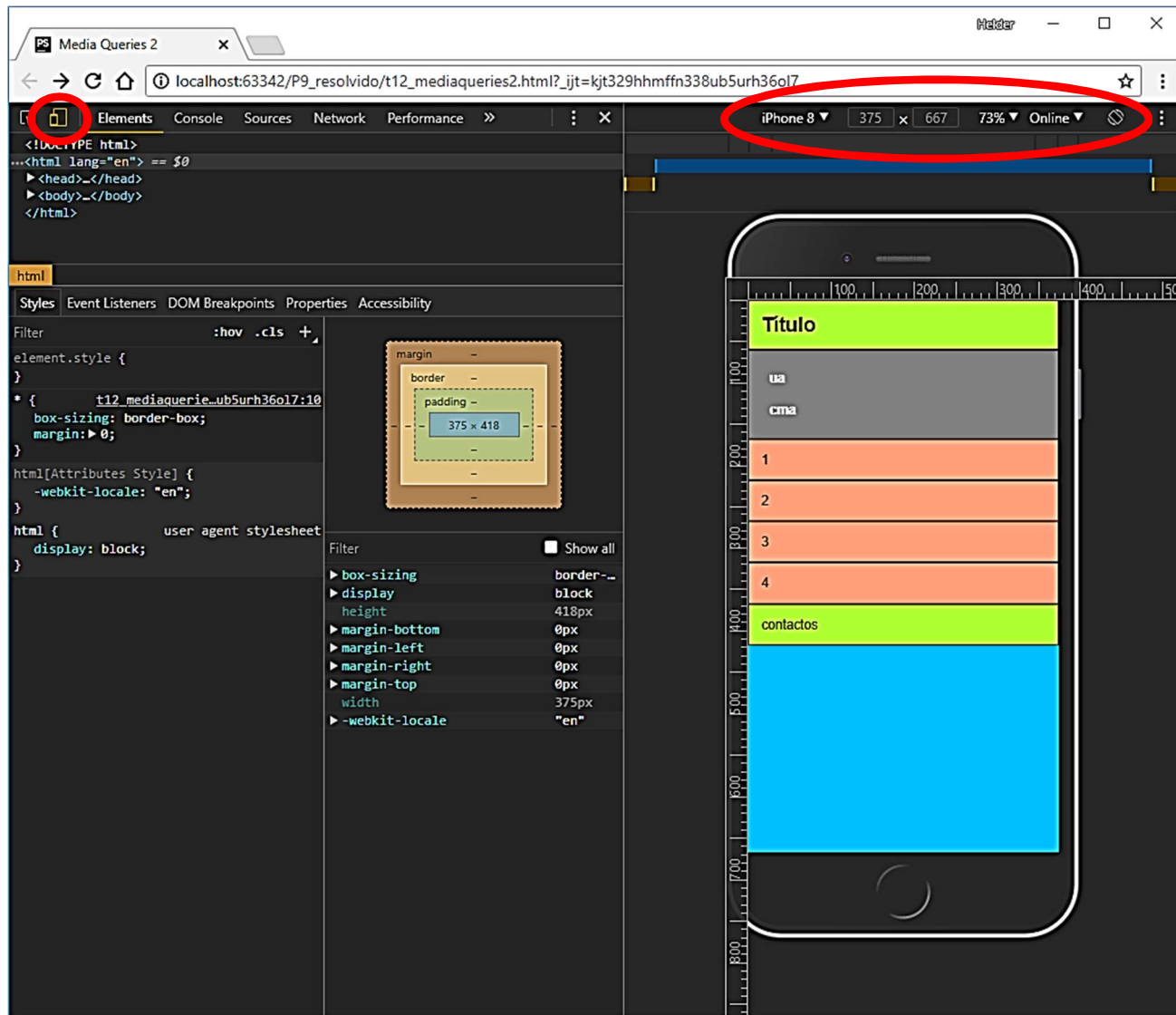
Em ecrãs de reduzidas dimensões (até 599px de largura) os elementos filhos serão dispostos em coluna

**flex-direction: column;** → Estilo definido dentro da media querie e aplicado ao main



# RWD & MOBILE FIRST

Análise de um exemplo de layout Responsive e Mobile First  
Breakpoint definido → 599px



# CSS – FLEXBOX

Alinhar horizontalmente e verticalmente os elementos filhos do container flexível

```
main {
```

```
display: flex;
```

alinhar horizontalmente

```
justify-content: center;
```

```
align-items: center;
```

```
height: 200px;
```

```
background-color: salmon;
```

alinhar verticalmente

```
}
```

```
<main>
```

```
  <article>LabMM2</article>
```

```
</main>
```

