

[Cursos](#)[Todos os cursos](#)[Formações](#)[Projetos práticos](#)[Direto ao ponto](#)[Quanto custa?](#)[Vantagens](#)[Artigos](#)[Login](#)[Matricule-se](#)

CSS

HTML

Media queries, o que são e como usar no CSS?

Neste artigo, nós iremos aprender sobre o uso de media queries no CSS e iremos aprender como utilizá-los em nossas aplicações web.

Redação Treinaweb • há 9 meses 4 dias

Quer receber
conteúdos
exclusivos sobre
programação?

Você sabia que a TreinaWeb é a mais completa escola para desenvolvedores do mercado?

O que você encontrará aqui na TreinaWeb?



Mentoria
de carreira



**Suporte
direto** com
os professores



**Plano de
estudos** simples
e direto



Projetos
voltados
ao **mercado de
trabalho**

[Cursos](#)[Todos os cursos](#)[Formações](#)[Projetos práticos](#)[Direto ao ponto](#)[Quanto custa?](#)[Vantagens](#)[Artigos](#)[Login](#)[Matricule-se](#)

pelos mais variados dispositivos, como computadores, celulares, tablets e até mesmo tv's. Cada um desses dispositivos possui diferentes resoluções e formatos de tela.

E para conseguirmos que nossos usuários possuam uma boa experiência nossas páginas devem ser capazes de se adequar à esses dispositivos, daí a necessidade de desenvolver páginas que são responsivas. Um dos recursos do CSS que podemos utilizar para alcançar esse objetivo é o recurso conhecido como media queries.

Neste artigo iremos aprender o que são os media queries e também sobre a sua sintaxe, para que você possa acompanhar este artigo sem dificuldades é necessário possuir alguns conhecimentos básicos sobre HTML e CSS. Fica aqui a recomendação dos seguintes artigos caso precise refrescar a memória sobre esses tópicos:

- [O que é e como começar com HTML e CSS?](#)
- [Seletores básicos no CSS](#)
- [Unidades de medida no CSS](#)

O que são Media Queries?

Media query, nada mais é do que uma estrutura do CSS que permite aplicar estilizações específicas para uma página web de acordo com certas condições, fazendo assim com esta página possa se adequar ao layout de tela em diferentes tamanhos e tipos de mídia.

A sintaxe básica ao utilizarmos media query é a seguinte:

[Copiar](#)

```
@media (condição) {  
    /* Estilizações CSS */  
}
```

No exemplo acima, podemos observar a sintaxe dos media queries, primeiramente utilizamos a expressão `@media` e logo em seguida definimos a condição dentro dos parênteses.

[Cursos](#)[Todos os cursos](#)[Formações](#)[Projetos práticos](#)[Direto ao ponto](#)[Quanto custa?](#)[Vantagens](#)[Artigos](#)[Login](#)[Matricule-se](#)

- **print:** Para impressoras;

[Copiar](#)

```
@media print {  
    /* Estilizações CSS */  
}
```

- **screen:** Para monitores ou outros dispositivos como computadores, tablets e smartphones com telas coloridas e com resolução adequada;

[Copiar](#)

```
@media screen and (max-width: 768px){  
    /* Estilizações CSS */  
}
```

- **handheld:** Para dispositivos de mão.

[Copiar](#)

```
@media handheld and (min-width: 700px) and (orientation: landscape){  
    /* Estilizações CSS */  
}
```

- **all:** Para todos os tipos de mídia.

[Copiar](#)

```
@media all and (orientation: landscape){  
    /* Estilizações CSS */  
}
```

Media features no media query

Um media feature é a largura da janela do documento, normalmente usado para atribuir uma condição que vai testar se o que foi definido é verdadeiro ou falso, ou seja, são

[Cursos](#)[Todos os cursos](#)[Formações](#)[Projetos práticos](#)[Direto ao ponto](#)[Quanto custa?](#)[Vantagens](#)[Artigos](#)[Login](#)[Matricule-se](#)

■ height

O **orientation** possui dois possíveis valores, os quais são **portrait** e **landscape**, onde o primeiro define a orientação em retrato, ou seja, se a altura é maior ou igual à largura e o segundo, orientação paisagem seguindo o mesmo resultado citado antes

Já o **width** define a largura do elemento na horizontal. E por último o **height** que define a altura do elemento na vertical.

Muitos media features podem ter prefixado **min-** ou **max-**, ou seja, servem para definir a largura mínima ou máxima de uma condição, assim definindo se nossas estilizações serão aplicadas conforme o tamanho da tela. Vamos ver alguns exemplos:

[Copiar](#)

```
@media (max-width:600px){
  /* Estilizações CSS */
}

@media (max-width:600px){
  .Card-TreinaWeb{
    flex-direction: column;
  }
}
```

Neste exemplo, podemos observar que foi definido um tamanho máximo de tela, segundo o tamanho que se encontra dentro dos parênteses definido com **600px**, chamado de breakpoints, fazendo com que essa estilização só seja aplicada em dispositivos com telas maiores que **600px** de largura.

Então, partimos para o segundo exemplo de código que mostrará o seu uso com media query para mínimo:

[Copiar](#)

```
@media (min-width:500px){
  /* Estilizações CSS */
}
```

[Cursos](#)[Todos os cursos](#)[Formações](#)[Projetos práticos](#)[Direto ao ponto](#)[Quanto custa?](#)[Vantagens](#)[Artigos](#)[Login](#)[Matricule-se](#)

Assim, observamos que foi definido o tamanho mínimo de tela para as nossas alterações. Como exemplo foi usado o `flex-direction: column`. Que indica que a partir deste tamanho a classe aplicada sofrer uma alteração e ficará em coluna.

Nos próximos tópicos aprenderemos como adicionar operadores lógicos que facilitará e simplificará o uso desta explicação.

O que é um operador lógico no media query e quais seus tipos?

Um operador lógico pode ser uma comparação entre elementos ou resultados, e pode conter, uma negação entre os mesmos e, neste artigo falaremos sobre os operadores lógicos de media query que são: `and`, `or`, `only` e `not`.

Operador lógico and no media queries:

Este operador lógico é usado para combinar múltiplas media features em uma mesma media query, requerendo que cada sequência de características, retorne verdadeiro na ordem para que a query seja verdadeira, simplificando isso, no `and` às duas condições precisam ser verdadeiras para que sua media query funcione. Como no exemplo a seguir:

[Copiar](#)

```
@media (min-width: 769px) and (max-width: 1024px){  
  /* Estilizações CSS */  
}
```

No exemplo anterior, você pode perceber que para a condição entrar no media seria necessário ter o mínimo de largura de `769px` e a máxima de `1024px` para o tamanho da tela. Está é uma pequena demonstração de como usar o operador `and`.

[Copiar](#)

```
@media screen and (max-width: 600px){  
  /* Estilizações CSS */
```

[Cursos](#)[Todos os cursos](#)[Formações](#)[Projetos práticos](#)[Direto ao ponto](#)[Quanto custa?](#)[Vantagens](#)[Artigos](#)[Login](#)[Matricule-se](#)

query.

Como aplicar no CSS:

[Copiar](#)

```
@media only screen and (max-width: 768px){  
  /* Estilizações CSS */  
}
```

Neste exemplo usamos o operador **only** e o **and**, para definir o tamanho máximo que sua tela terá que atingir para serem feitas alterações.

Operador lógico or no media queries:

Este operador lógico é usado com o caractere **,**, que simboliza que para ocorrer alteração no media, uma das condições definidas precisa ser verdadeira.

Exemplo da condição:

[Copiar](#)

```
@media (max-width: 30.5em), print and (orientation: portrait) {  
  /* Estilizações CSS */  
}
```

Para este exemplo, usamos a virgula para definir uma das condições aplicadas e como exemplo atribuímos o **orientation**.

Operador lógico not no media queries:

Usamos este operador lógico para negar uma característica na query, porém, ele não pode ser usado sozinho, pois necessita de algo a ser negado para cair na condição aplicada.

[Copiar](#)

```
@media not all and (monochrome) {...}
```



- Cursos
- Todos os cursos
- Formações
- Projetos práticos
- Direto ao ponto
- Quanto custa?
- Vantagens
- Artigos

Login

Matricule-se

Neste artigo aprendemos sobre medias types, media queries e seus operadores lógicos. Além dos exemplos citados, existem outros métodos de media mais avançados, por isso recomendo fortemente a leitura da [documentação disponibilizada pela Mozilla](#).

#CSS #HTML #media types #media queries

Autor(a) do artigo



Redação Treinaweb

Todos os artigos

Artigos relacionados

Ver todos →



- Cursos
- Todos os cursos
- Formações
- Projetos práticos
- Direto ao ponto
- Quanto custa?
- Vantagens
- Artigos

Login

Matricule-se

Media Query no CSS3, na prática aprendendo como uti...

CSS poderia ser trabalhoso, dependendo do que fosse...

tema HTML, nele irá conter desde sua história, quem o criou e com...

CSS

CSS Grid – Um Guia Interativo (Parte 1 – Containers)

Aprenda definitivamente a trabalhar com CSS Grid com este Guia Interativo para facilitar o seu traba...

CSS

CSS Flexbox - Um Guia Interativo (Parte 2 - Itens)

Aprenda a trabalhar com o CSS Flexbox com este guia definitivo e interativo! Vamos ver as propriedad...

CSS

CSS Filters - Tratamento de imagens em sua página

Com a evolução do CSS3 passou a ser possível criar muitos dos efeitos dos softwares de edição de ima...

[Cursos](#)[Todos os cursos](#)[Formações](#)[Projetos práticos](#)[Direto ao ponto](#)[Quanto custa?](#)[Vantagens](#)[Artigos](#)[Login](#)[Matricule-se](#)

Você sabe quando utilizar "float" e "display: inline-block"? Veja as características de cada um e ap...

aprenderemos a criar um navbar usando Tailwind CSS, e entender melhor como funciona seu...

veremos alguns dos principais frameworks CSS do mercado e algumas de suas característic...

CSS

O que é e como começar com HTML e CSS?

Quer iniciar seus estudos em HTML e CSS e não sabe por onde começar? Veja o que preparamos pra você.

Desenvolvimento
Front-end

Entendendo os atributos HTML

No artigo de hoje, veremos algo que está muito presente no mundo da programação, atributos. Por que...

Desenvolvimento
Front-end

Seletores avançados do CSS

Nesse artigo você irá entender alguns tipos de seletores pouco utilizados no CSS, porém extremamente...



Escola online para desenvolvedores



Inscreva-se e receba nossos lançamentos, promoções e novidades

[Inscreva-se](#)[Cursos](#)[A empresa](#)[Contato](#)[Artigos](#)[Baixe nosso aplicativo](#)



- Cursos ▾
- Todos os cursos
- Formações
- Projetos práticos
- Direto ao ponto
- Quanto custa?
- Vantagens
- Artigos

Login

Matricule-se