w3schools.com

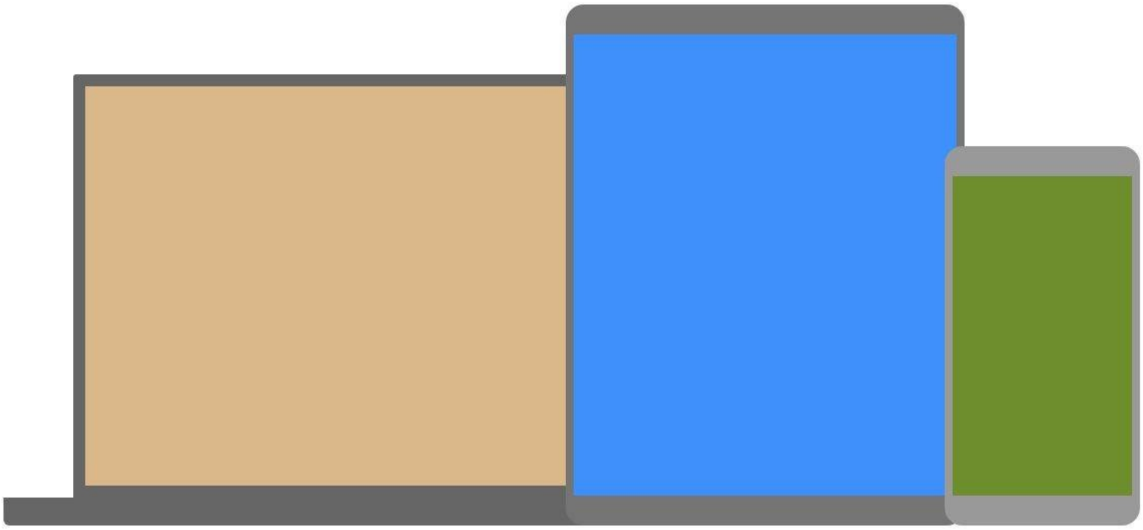
# CSS3 Media Queries - Examples

5-6 minutos

## CSS Media Queries - More Examples

Let us look at some more examples of using media queries.

Media queries are a popular technique for delivering a tailored style sheet to different devices. To demonstrate a simple example, we can change the background color for different devices:



### Example

```
body {
  background-color: tan;
}
```

```
@media screen and (max-width: 992px) {
  body {
    background-color: blue;
  }
}


@media screen and (max-width: 600px) {
  body {
    background-color: olive;
  }
}
```

Try it Yourself »

Do you wonder why we use exactly 992px and 600px? They are what we call "typical breakpoints" for devices. You can read more about typical breakpoints in our Responsive Web Design Tutorial.

## Media Queries For Menus

In this example, we use media queries to create a responsive navigation menu, that varies in design on different screen sizes.

### Example

```
.topnav {
  overflow: hidden;
  background-color: #333;
}


.topnav a {
  float: left;
  display: block;
  color: white;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
}


@media screen and (max-width: 600px) {
  .topnav a {
    float: none;
    width: 100%;
  }
}
```

[Try it Yourself »]

---

## Media Queries For Columns

A common use of media queries, is to create a flexible layout. In this example, we create a layout that varies between four, two and full-width columns, depending on different screen sizes:

**Example**

```
.column {
  float: left;
  width: 25%;
}


@media screen and (max-width: 992px) {
  .column {
    width: 50%;
  }
}


@media screen and (max-width: 600px) {
  .column {
    width: 100%;
  }
}
```

Try it Yourself »

**Tip:** A more modern way of creating column layouts, is to use CSS Flexbox (see example below). However, it is not supported in Internet Explorer 10 and earlier versions. If you require IE6-10 support, use floats (as shown above).

To learn more about the Flexible Box Layout Module, read our CSS Flexbox chapter.

To learn more about Responsive Web Design, read our Responsive Web Design Tutorial.

**Example**

```
.row {
  display: flex;
  flex-wrap: wrap;
}


.column {
  flex: 25%;
  padding: 20px;
}


@media screen and (max-width: 992px) {
  .column {
    flex: 50%;
  }
}


@media screen and (max-width: 600px) {
  .row {
    flex-direction: column;
  }
}
```

Try it Yourself »

## Hide Elements With Media Queries

Another common use of media queries, is to hide elements on different screen sizes:

I will be hidden on small screens.

**Example**

```
@media screen and (max-width: 600px) {
  div.example {
    display: none;
  }
}
```

Try it Yourself »

## Change Font Size With Media Queries

You can also use media queries to change the font size of an element on different screen sizes:

## Variable Font Size.

### Example

```
@media screen and (min-width: 600px) {
  div.example {
    font-size: 80px;
  }
}


@media screen and (max-width: 600px) {
  div.example {
    font-size: 30px;
  }
}
```

[Try it Yourself »]

---

## Flexible Image Gallery

In this example, we use media queries together with flexbox to create a responsive image gallery:

---

## Flexible Website

In this example, we use media queries together with flexbox to create a responsive website, containing a flexible navigation bar and flexible content.

## Orientation: Portrait / Landscape

Media queries can also be used to change layout of a page depending on the orientation of the browser.

You can have a set of CSS properties that will only apply when the browser window is wider than its height, a so called "Landscape" orientation:

### Example

Use a lightblue background color if the orientation is in landscape mode:

```
@media only screen and (orientation: landscape) {
  body {
    background-color: lightblue;
  }
}
```

Try it Yourself »

## Min Width to Max Width

You can also use the `(max-width: ..) and (min-width: ..)` values to set a minimum width and a maximum width.

For example, when the browser's width is between 600 and 900px, change the appearance of a <div> element:

### Example

```
@media screen and (max-width: 900px) and (min-width: 600px)
{
  div.example {
    font-size: 50px;
    padding: 50px;
    border: 8px solid black;
    background: yellow;
  }
}
```

Try it Yourself »

**Using an additional value:** In the example below, we add an additional media query to our already existing one using a comma (this will behave like an OR operator):

**Example**

```
@media screen and (max-width: 900px) and (min-width: 600px),
(min-width: 1100px) {
  div.example {
    font-size: 50px;
    padding: 50px;
    border: 8px solid black;
    background: yellow;
  }
}
```

Try it Yourself »

## CSS @media Reference

For a full overview of all the media types and features/expressions, please look at the @media rule in our CSS reference.

**Tip:** To learn more about responsive web design (how to target different devices and screens), using media query breakpoints, read our Responsive Web Design Tutorial.