

Learn Responsive Design

CSS unit em

`em` is a CSS unit used to create relatively-sized content. 1 `em` unit represents the base font size of the current element.

In the example code block, `` elements nested inside of elements with class `nav-container` will have a font size of `15px`.

```
.nav-container {  
  font-size: 10px;  
}  
  
.nav-container span {  
  font-size: 1.5em; /* 10 x 1.5 = 15px */  
}
```

Usage of Percentages in CSS

Instead of defining a fixed width using units like `px`, or `cm`, percentages can be used in CSS to set the height and width of child elements relative to the dimensions of their parent. The child elements will grow or shrink according to the set percentage values if the parent element changes in size.

```
.news-row {  
  height: 300px;  
  width: 500px;  
}  
  
.news-row .news-column {  
  height: 80%; /* 240px */  
  width: 50%; /* 250px */  
}
```

background-size: cover;

The CSS `background-size` property is used to specify the size of the background image. When given the value `cover`, like `background-size: cover`, the image covers the complete background of the container in which it is being displayed.

The proportions of the image are maintained, so if the dimensions exceed the container's, then some parts of the image will be left out.

CSS unit rem

The CSS unit `rem` can be used to set the font size of HTML elements relative to the font size of the root element. 1 `rem` represents the size of the base font within the root element - the `<html>` tag.

In the example code block, the font size for all the `` elements will be twice the size of the font size declared for the root element.

```
html {  
  font-size: 18px;  
}  
  
span {  
  font-size: 2rem;  
}
```

When writing media queries in CSS we use the `only` keyword to specify the type of device. Initially, CSS incorporated a variety of media types such as `screen`, `print` and `handheld`. In order to ensure responsive design and to accommodate a variety of screen sizes the keyword `screen` is now always used for displaying content.

And Operator Media Queries

Through using the `and` operator when writing media queries in CSS, we can specify that multiple media features are required. For example we could require a certain width as well as a specific resolution for a CSS ruleset to apply. The `and` operator when chaining together multiple media features allows us to be more specific with the screens that we are targeting as we build responsive designs.

CSS Media Features in Media Queries

Media queries in web development ensure that a website design is responsive. It does so through creating tailored style sheets based on the resolution of the device that it is displayed upon. The browser will detect the screen size and apply the CSS styles for that screen size based on specified media features. Media queries can set rules based on features like the screen width, the screen resolution and device orientation. These media rules can be separated by commas. When one of the media rules is true then the accompanying CSS will apply.

css media query

A CSS *media query* can be used to adapt a website's display and layout to different screen sizes. A media query begins with the `@media` keyword and is followed by one or more conditions that check screen size, orientation, resolution, and/or other properties. If these conditions are met, all CSS rules within the media query will be applied to the page. Media queries are used for responsive web design by tailoring specific stylesheets to different types of devices such as laptops, tablets, mobiles, and more.

```
@media only screen and (min-width: 600px)
{
    /* ruleset for >= 600px */
}
```

```
@media only screen and (max-width: 480px)
and (min-resolution: 300dpi) {
    /* CSS ruleset */
}
```

```
/* For screen sizes less than or equal to
480px (most likely a mobile device), the
body element's font size will be set to
12px and the #photo element's width will
be set to 100% */
@media only screen and (max-width: 480px)
{
    body {
        font-size: 12px;
    }

    #photo {
        width: 100%;
    }
}
```

Ranges in Media Queries

Media queries can use CSS to target screen sizes within a certain range through using multiple widths and/or heights. This is an effective tool for responsive design that will address a variety of screen sizes in one CSS media query. In order to set a range for width or the height, set the minimum screen size through using `min-width` and/or `min-height` and then set the maximum through using `max-width` or `max-height`. These properties are used in combination with the `and` operator.

```
@media only screen and (min-width: 480px)
and (max-width: 600px) {
    /* ruleset for 480px - 600px */
}
```