

Adding Media Style Sheets

In this lesson, we learn how to add media style sheets to our webpage.

WE'LL COVER THE FOLLOWING ^

- Listing 9-12: Using `@media`
- Media queries

Media-dependent style sheets use exactly the same syntax as any other CSS declarations. When attaching an external style sheet to your page, the media attribute of the `<link>` tag lets you specify the media type.

This sample shows how you can add three style sheets to your web page:

 index.html

```
<link rel="stylesheet" href="allStyle.css" />
<link rel="stylesheet" media="screen, projection, handheld, tv"
      href="screenStyle.css" />
<link rel="stylesheet" media="print"
      href="printStyle.css" />
```



The first `<link>` tag loads a CSS file that contains all style rules that are suitable for every media type. The second and third `<link>` tags target specific media types, the second targets devices with screens, and the third targets printing devices.

Using the media attribute, you can attach the same style sheet file to different media types. This method cannot work with internal style sheets, because there is no media attribute to set up.

With the `@media` at-rule, you can define sections within the style sheet and put style rules into them, as demonstrated in Listing 9-12 below:

Listing 9-12: Using `@media` #


```

<!DOCTYPE html>
<html>
<head>
  <title>Using @media</title>
  <style>
    @media all {
      h1 {
        background-color: blue;
        color: white;
      }
    }

    @media print {
      h1 {
        background-color: white;
        color: blue;
      }
    }
  </style>
</head>
<body>
  <h1>Introduction</h1>
  <p>
    Lorem ipsum dolor sit amet, consectetur
    adipiscing elit fusce vel sapien elit
    in malesuada semper mi, id sollicitudin
    urna fermentum.
  </p>
</body>
</html>

```

Here, the all and print media type identifiers define the target type of the section. As you can see, the `<h1>` tag is defined to use a blue background with a white text color when the page is displayed with screen-like devices, but it is instructed to use blue color with white background on printers.

 **NOTE:** Technically, it doesn't matter whether you put all styles in a single file and use the `@media` method or put media-specific styles in their own external style sheets.

Media queries

CSS3 added an extension to media types that allows for more sophisticated control over defining styles for different devices. This extension is called media queries, and it allows building logical expressions in conjunction to media types.

You can test the features of the output device you intend to render the output for, and can create styles that accommodate your “screens” better than with simple media types.

Media queries are composed from simple test expressions that are combined with logical AND and logical OR operators. Here is a sample:

```
@media handheld and (min-width: 640px) {  
  /* Add rules here */  
}
```



This media query is pretty easy to read and understand. The section defines style rules that should be applied for handheld type devices with a horizontal resolution of *640pixels* or above.

Media queries can be added to the `@media` and `@import` at-rules, and they can also be used in the media attribute of `<link>` tags. You can use multiple comma separated media queries in a single rule where the comma acts as the logical OR.

For example, this media query imports the **myStyle.css** file for screens with at least 8-bit color per component and for projectors that support colors:

```
@import url(myStyle.css)  
  screen and (min-color: 8),  
  projection and (color);
```



The media queries extension defines a number of features that can be tested, as listed in Table 9-6. Most features can be used with the min- and max- prefix to test for minimum and maximum values.

Achievement unlocked! 🎉

Congratulations! You’ve learned to make use of media style sheets.

Good job! Give yourself a round of applause!



Let's go on to the usage of media queries in the *next lesson*.