

media query

CSS Media Queries are a fundamental tool in responsive web design, enabling you to apply CSS rules conditionally based on the characteristics of the device or viewport displaying the content. This lets you tailor layouts, font sizes, colors, and other styles to different screen sizes, orientations, resolutions, and more.

Basic Syntax:

```
@media media-type and (media-feature) {  
  /* CSS rules here */  
}
```

- **@media**: starts the media query block.
- **media-type**: optional, specifies the device type (e.g., `screen`, `print`, or `all`).
- **media-feature**: one or more expressions in parentheses that check properties like `min-width`, `max-width`, `orientation`, etc.

Example:

```
/* Default styling for all devices */  
body {  
  background-color: tan;  
}  
  
/* For screens 992px wide or less */  
@media screen and (max-width: 992px) {  
  body {  
    background-color: blue;  
  }  
}  
  
/* For screens 600px wide or less */  
@media screen and (max-width: 600px) {  
  body {  
    background-color: olive;  
  }  
}
```

In this example, the background color changes based on viewport width, demonstrating *responsive design*^{[1] [2]}.

Key Concepts:

- **Breakpoints:** Specific viewport widths where the design adapts, e.g., 600px or 992px.
- **Logical Operators:**
 - `and`: combines conditions.
 - `not`: negates a query.
 - Multiple queries separated by commas act as "or" (any matches apply).
- **Media Features:** Examples include
 - `width`, `min-width`, `max-width`
 - `orientation` (portrait or landscape)
 - `resolution`, `hover`, `prefers-color-scheme`, etc.

Advanced Usage Example:

```
@media screen and (min-width: 600px) and (orientation: landscape) {  
  body {  
    color: blue;  
  }  
}  
  
@media screen and (max-width: 576px) {  
  header {  
    display: block;  
  }  
  
  nav ul {  
    display: block;  
  }  
}
```

This applies styles only if the screen is at least 600px wide and in landscape mode, or changes the layout for very small devices like mobiles [\[3\]](#) [\[4\]](#).

Different ways to use media queries in HTML:

You can link different CSS files based on viewport size or media type directly in HTML `<head>`:

```
<link rel="stylesheet" href="style.css" media="all" />  
<link rel="stylesheet" href="mobile.css" media="screen and (max-width: 600px)" />  
<link rel="stylesheet" href="print.css" media="print" />
```

This allows serving optimized stylesheets for different conditions [\[5\]](#) [\[6\]](#).

In summary:

- Media queries let you apply CSS conditionally based on device characteristics.
- They enable responsive designs that adapt to different screen sizes and orientations.

- Syntax involves @media, optional media types, and feature expressions.
- Common features include min-width and max-width for breakpoints.
- Queries can be combined with and, negated with not, or expanded with multiple queries separated by commas.

This flexible system is essential in modern web design to ensure usability and aesthetics across desktops, tablets, and mobiles^{[7] [3] [4]}.

**

1. https://www.w3schools.com/css/css3_mediaqueries_ex.asp
2. <https://www.programiz.com/css/media-query>
3. https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_media_queries/Using_media_queries
4. https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/CSS_layout/Media_queries
5. <https://css-tricks.com/a-complete-guide-to-css-media-queries/>
6. https://www.w3schools.com/cssref/css3_pr_mediaquery.php
7. https://www.w3schools.com/css/css3_mediaqueries.asp