Let's delve into each of these topics with explanations and examples:

5. Media Queries
6. Positioning

5. Media Queries:

Media queries are a crucial part of responsive web design. They allow you to apply specific styles to a webpage based on the characteristics of the device or screen it is being viewed on. This helps optimize the layout and appearance for various screen sizes and resolutions.

Example:

Suppose you want to change the font size when the screen width is less than 600 pixels. You can use a media query in your CSS like this:

```css
/* Default styles for larger screens */
body {
  font-size: 16px;
}

/* Media query for screens with a width of 600 pixels or less */
@media only screen and (max-width: 600px) {
  body {
    font-size: 14px;
  }
}
```

In this example, the font size is set to 16 pixels by default, but when the screen width is 600 pixels or less, it changes to 14 pixels. You can customize various CSS properties within media queries to create a responsive design.

6. Positioning:

Positioning is about determining the layout and placement of elements on a webpage. CSS provides several positioning properties that allow you to control the position of an element relative to its normal position in the document flow or to the containing element.

Example:

Let's say you have a navigation bar at the top of your webpage that you want to stick to the top even when users scroll down. You can use the `position: fixed` property.

```css
/* Styling for the navigation bar */
nav {
  background-color: #333;
  color: #fff;
  padding: 10px;
  /* Set initial position */
  position: relative;
}

/* Set position to fixed when scrolling */
nav.fixed {
  position: fixed;
  top: 0;
  width: 100%;
  /* Add a shadow for better visibility */
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
}
```

In this example, initially, the navigation bar has a relative position. When the user scrolls down, a JavaScript function (not shown here for brevity) adds a class of "fixed" to the navigation bar, making it fixed at the top of the page.

These examples provide a glimpse into the use of media queries and positioning in web development. In real-world scenarios, you would often use these concepts together to create a fully responsive and visually appealing layout.