There are three main ways to apply styles to a webpage using CSS (Cascading Style Sheets):

1. Inline Styles:

- **How it works:** Styles are applied directly to individual HTML elements using the style attribute.
- **Example:**

```
<p style="color: blue; font-size: 16px;">This text is blue and 16
    pixels.</p>
```

- **Pros:** Easy for quick, one-off styling.
- **Cons:** Not maintainable for larger projects. Styles are scattered throughout the HTML, making it difficult to update or change styles consistently. Also, inline styles have the highest specificity, making them hard to override. Generally, inline styles are best avoided except for very specific, dynamic styling done by JavaScript.

2. Internal Style Sheet (Embedded Styles):

- **How it works:** Styles are placed within a <style> tag inside the <head> section of the HTML document.
- **Example:**

```
<head>
    <style>
      p {
        color: blue;
        font-size: 16px;
      }
      h1 {
        color: green;
      }
    </style>
  </head>
```

- **Pros:** Better than inline styles for maintainability, as styles are grouped together. Useful for page-specific styles.

- **Cons:** Not ideal for large websites with many pages, as styles are duplicated across pages. External style sheets are generally preferred for larger projects.

3. External Style Sheet (Linked Styles):

- **How it works:** Styles are written in a separate .css file, and then linked to the HTML document using the <link> tag. This is the most common and recommended approach for larger websites.
- **Example (in styles.css):**

```css
p {
  color: blue;
  font-size: 16px;
}

h1 {
  color: green;
}
```

- **Example (in your HTML file):**

```html
<head>
  <link rel="stylesheet" href="styles.css">
</head>
```

- **Pros:** Best for maintainability and organization. Styles are centralized, making it easy to update them across all pages. Browsers can cache external style sheets, leading to faster page load times. This is the preferred method for almost all projects.

Which Method to Use:

- **Small, simple projects:** Internal style sheets might be okay.
- **Large, complex websites:** External style sheets are strongly recommended.
- **Dynamic, JavaScript-driven styling:** Inline styles (but set by JavaScript, not directly in the HTML) can be appropriate in very specific cases.
- **Quick, one-off styling (rarely):** Inline styles, but generally best avoided.

Best Practices:

- **Use external style sheets whenever possible.** This keeps your HTML clean and your CSS organized.

- **Organize your CSS files logically.** Use folders and meaningful names for your CSS files.
- **Use a consistent naming convention** for your CSS classes and IDs.
- **Keep your CSS as concise as possible.** Avoid unnecessary or redundant styles.
- **Use a CSS preprocessor (like Sass or Less) if your project gets large.** Preprocessors can help you write more maintainable CSS.
- **Validate your CSS.** There are online tools that can help you find errors in your CSS.

By following these guidelines, you can create and maintain clean, efficient, and scalable CSS for your web projects. Using external style sheets is the most important practice for writing maintainable and reusable CSS.