1. VS Code Extension For HTML

1. **Live Server**: This extension provides a live preview of your web pages in real-time as you edit them. It's especially useful for front-end development, allowing you to see changes instantly in the browser.

2. **HTML End Tag Labels**: This extension adds labels to closing HTML tags. It's particularly helpful in large HTML files to quickly identify which closing tag corresponds to which opening tag, improving code readability.

3. **Auto Close Tag**: Automatically adds HTML/XML close tags as soon as you type the opening tag. It speeds up your HTML coding by reducing the amount of typing required.

4. **Highlight Matching Tag**: Whenever you select an HTML tag, this extension highlights the matching opening or closing tag. This feature is useful for quickly navigating and understanding the structure of your HTML code.

5. **Auto Rename Tag**: When you rename one HTML/XML tag, this extension automatically renames the matching paired tag. It's a great time-saver and helps prevent errors in tag renaming.

6. **Code Spell Checker**: This extension helps to catch common spelling errors in your code. It supports multiple languages and programming languages, making it an essential tool for maintaining professional and error-free code.

7. **Indent Rainbow**: This extension colorizes the indentation in front of your text, alternating four different colors on each

step. It's visually appealing and makes it easier to distinguish different levels of indentation at a glance.

8. **Prettier - Code Formatter**: Prettier is a popular code formatter that supports many languages and integrates with most editors. It formats your code consistently, following a set of rules, which improves readability and reduces the chance of syntax errors.

9. **Axe Accessibility Linter**: This extension is focused on web accessibility, providing automated testing of HTML for accessibility issues. It's a vital tool for ensuring that web content is accessible to all users, including those with disabilities.

2. VS Code Emmet

- Visual Studio Code (VS Code) Emmet is a set of tools and functionalities integrated into the VS Code editor, designed to improve HTML and CSS workflow efficiency.

- Emmet uses a special syntax that allows you to quickly generate HTML and CSS code with minimal typing. It's particularly useful for web developers and front-end designers.

Common Abbreviations

- `!` or `html:5` : Basic HTML5 template.
- `a` : Creates an `<a href=""></a>` tag.
- `p` : Generates a `<p></p>` paragraph tag.
- `img` : Produces an `<img src="" alt="">` tag.
- `ul>li` : Creates a list, can be combined with `*` for multiple items.
- `table>tr>td` : Generates a table structure.
- `input:checkbox` : Creates a checkbox input.
- `form:post` : Generates a form with the method set to POST.
- `br` : Inserts a line break.

ID and Class Abbreviations

- `#header` : Generates `<div id="header"></div>`.
- `.container` : Produces `<div class="container"></div>`.

Attribute Abbreviations

`[attr=value]` : Any tag can be appended with attributes like `a[href="http://"]` .

3. Web Page Structure

1. **DOCTYPE Declaration**:
   - `<!DOCTYPE html>`: This declaration defines the document type and version of HTML. For HTML5, use `<!DOCTYPE html>`.
2. **HTML Element**:
   - `<html>`: This is the root element of an HTML page.
3. **Head Section**:
   - `<head>`: Contains meta-information about the document, like character encoding, title, linked CSS files, and scripts.
     - `<meta charset="UTF-8">`: Specifies the character encoding for the HTML document.
     - `<title>`: Title of the webpage, displayed in the browser's title bar or tab.
     - `<link rel="stylesheet" href="style.css">`: Link to an external CSS file.
     - `<script src="script.js"></script>`: Link to an external JavaScript file.
4. **Body Section**:
   - `<body>`: Contains the contents of the HTML document. This is where you place the content that will be visible to users.
     - **Headings**: `<h1>` to `<h6>` tags for headings (h1 for the main heading).
     - **Paragraphs**: `<p>` tag for paragraphs.

- **Links**: `<a href="url">` to create a hyperlink.
- **Images**: `<img src="image.jpg" alt="description">` for embedding images.
- **Lists**:
    - Ordered List: `<ol>` with `<li>` items.
    - Unordered List: `<ul>` with `<li>` items.
- **Tables**: `<table>`, with `<tr>` for rows, and `<td>` for cells.
- **Forms**: `<form>` for user input.
    - Input Elements: `<input>`, `<textarea>`, `<button>`, etc.
    -

5. **Comments**:
   - `<!-- Comment -->`: Comments are not displayed in the browser, but they can help document your HTML source code.

Here's a simple example of an HTML page:

4. Meta Tags

Basic Meta Tags

Title Tag: Defines the page title shown in browser tabs and search results.

```
<title>Page Title</title>
```

Description Meta Tag: A brief summary of the page's content.

```
<meta name="description" content="Short description of the page.">
```

SEO Meta Tags

Keywords Meta Tag: Lists keywords relevant to the page's content.

```
<meta name="keywords" content="keyword1, keyword2">
```

Robots Meta Tag: Directives for search engine crawling and indexing.

```
<meta name="robots" content="index, follow">
```

HTTP-Equiv Meta Tags

Content-Type: Specifies the character encoding for the HTML document.

```
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
```

Default-Style: Defines the default stylesheet.

```
<meta http-equiv="default-style" content="style.css">
```

Refresh: Sets an automatic page refresh interval.

```
<meta http-equiv="refresh" content="30">
```

Open Graph Meta Tags (For Social Media)

OG Title: The title for social media sharing.

```
<meta property="og:title" content="Social Media Title">
```

OG Type: The type of content (e.g., website, article).

```
<meta property="og:type" content="article">
```

OG Image: Image URL for social media.

```
<meta property="og:image" content="http://example.com/image.jpg">
```

Twitter Card Meta Tags

Twitter Card: The Twitter card type.

```
<meta name="twitter:card" content="summary">
```

Twitter Title/Description: Title and description for Twitter.

```
<meta name="twitter:title" content="Twitter Title">
```

Mobile Device Meta Tags

Viewport: Controls the page's viewport settings.

```
<meta name="viewport" content="width=device-width,
initial-scale=1.0">
```

Security Meta Tags

Content-Security-Policy: Security policy for resource loading.

```
<meta http-equiv="Content-Security-Policy"
content="default-src 'self'">
```

Web Application Meta Tags

1. **Application-Name**: The name of the web application.

```
<meta name="application-name" content="My Web App">
```

2. **Theme-Color**: Suggests a browser theme color.

```
<meta name="theme-color" content="#ffffff">
```

Verification Meta Tags

Google Search Console Verification: Verifies ownership for Google Search Console.

```
<meta name="google-site-verification"
content="verification_token">
```

Authorship Meta Tags

Author, Publisher: Defines the author or publisher of the page.

```
<meta name="author" content="Author Name">
```

Cache Control Meta Tags

Pragma, Cache-Control, Expires: Controls browser and server caching.

```html
<meta http-equiv="cache-control" content="no-cache">
```

5. Manifest.json

Structure of `manifest.json`

1. `name` : The full name of the application.
2. `short_name` : A shorter name for the application, used where space is limited.
3. `start_url` : The entry point of the application when launched. It's relative to the location of the manifest file.
4. `display` : Defines the preferred display mode, such as `fullscreen`, `standalone`, `minimal-ui`, or `browser`.
5. `background_color` : Specifies a background color for the splash screen when the app is launched.
6. `description` : A brief description of the application.
7. `lang` : The primary language of the app, specified using a language tag (`en-US`, `fr`, etc.).
8. `dir` : The text direction of the app, like `ltr` (left-to-right) or `rtl` (right-to-left).
9. `orientation` : Specifies the default orientation of the app, such as `portrait` or `landscape`.
10. `theme_color` : Defines the default theme color for the application.
11. `icons` : An array of image objects representing the app icon in different sizes. Each object typically includes `src`, `sizes`, and `type` properties.
12. `scope` : Specifies the set of URLs that the browser considers to be within your app, and navigations to those URLs stay within the app.

13. `related_applications` : An array of objects specifying native apps that are related to the web app, potentially with a `platform` and a `url` or `id` depending on the platform.

```json
{
  "name": "Example App",
  "short_name": "App",
  "start_url": "/",
  "display": "standalone",
  "background_color": "#ffffff",
  "description": "An example Progressive Web App",
  "lang": "en-US",
  "dir": "ltr",
  "orientation": "portrait",
  "theme_color": "#000000",
  "icons": [
    {
      "src": "icon/lowres.webp",
      "sizes": "48x48",
      "type": "image/webp"
    },
    {
      "src": "icon/hd_hi.ico",
      "sizes": "72x72 96x96 128x128 256x256",
      "type": "image/x-icon"
    }
  ],
  "scope": "/",
  "related_applications": [
    {
      "platform": "play",
      "url":
"https://play.google.com/store/apps/details?
id=com.example.app"
```

```json
    },
    {
      "platform": "itunes",
      "url": "https://itunes.apple.com/app/example-
app/id123456789"
    }
  ]
}
```