Power of Visual Studio Code (VS Code)

Why VS Code for WTMA?

Visual Studio Code is a **professional code editor** used by developers across the world. In WTMA, it helps you write clean code, understand structure, and build real websites like industry professionals.

---

1. Smart Code Assistance ✨

VS Code helps you while typing: - Auto-completes HTML tags - Suggests attributes and values - Reduces typing mistakes

Example: Typing `<h` automatically suggests `<h1>` to `<h6>`.

---

2. Live Coding Experience ⚡

- Edit HTML/CSS files
- Save the file
- Open the file in Live server
- → see instant changes

This makes learning **fast and fun**.

---

3. Clean Folder Structure 📁

VS Code helps you organize files properly:

```
project-folder
 ├── index.html
 ├── style.css
 └── script.js
```

This teaches **real-world web development practice**.

---

4. Error Highlighting 🚨

- Missing tags
- Wrong syntax

- Spelling errors

VS Code highlights mistakes so you can fix them easily.

---

5. Multiple File Support 🧩

You can open: - HTML files - CSS files - JavaScript files

All in one place — no switching apps.

---

6. Extensions (Power Feature) 🔌

VS Code supports extensions that add extra features.

Popular for WTMA: - Live Server (auto refresh browser) - HTML CSS Support

(Extensions will be introduced later in the course)

---

7. Keyboard Shortcuts ⌨️

VS Code makes you faster: - `Ctrl + S` → Save file - `Ctrl + /` → Comment code - `Ctrl + Z` → Undo

---

8. Used by Professionals 🌍

VS Code is used by: - Web Developers - Software Engineers - Startups & IT Companies

Learning VS Code = **Industry-ready skill**.

Summary

Using VS Code in WTMA helps you: - Write better code - Learn faster - Think like a developer - Build real websites

> *“VS Code is not just an editor — it’s your coding workspace.”*

---

9. Configuring a Project Folder (Getting Started)

Before writing code, a developer always sets up a proper **project folder**. This helps you work neatly and professionally.

Step 1: Create a Project Folder

1. Create a new folder on your computer
2. Name it:

`WTMA_Project`

Step 2: Open Folder in VS Code

1. Open **VS Code**
2. Click **File → Open Folder**
3. Select your `WTMA_Project` folder

You will now see the folder on the **left sidebar** (Explorer).

Step 3: Create Required Files

Inside the folder, create these files:

```
WTMA_Project
 ├── index.html
 ├── style.css
 └── script.js
```

- `index.html` → Structure of the web page
- `style.css` → Design and layout
- `script.js` → Interactions (used later)

Step 4: Link Files Together

In `index.html`, connect CSS and JavaScript:

```
<link rel="stylesheet" href="style.css">
<script src="script.js"></script>
```

This teaches how files **work together** in real projects.

Step 5: Start Working

1. Write HTML in `index.html`
2. Add colors and fonts in `style.css`
3. `Add dynamic content in script.js`
4. Save files (`Ctrl + S`)
5. Open `index.html` in a browser

Make small changes and refresh the browser to see results.

Summary

By configuring a project folder, you learn: - File organization - Professional coding practice - How real websites are built

> **“Good developers write good code — great developers organize it well.”**

Next Step: Introduction to html.

HTML Basic Syntax – Lecture Notes

1. What is HTML?

HTML stands for **HyperText Markup Language**. It is the standard language used to create web pages. HTML describes the **structure** of a web page using elements called **tags**.

- HTML is **not** a programming language
- It is a **markup language**
- It works together with CSS (for styling) and JavaScript (for interactivity)

---

2. Basic Structure of an HTML Document

Every HTML page follows a basic structure.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>My First Web Page</title>
</head>
<body>
    <h1>Hello World</h1>
    <p>This is my first HTML page.</p>
</body>
</html>
```

Explanation:

- `<!DOCTYPE html>` – Defines the document type (HTML5)
- `<html>` – Root element of the HTML page
- `<head>` – Contains meta information (not visible on page)
- `<title>` – Title shown in browser tab
- `<body>` – Contains visible page content

---

3. HTML Tags

HTML uses **tags** to define elements.

Types of Tags:

6. **Paired Tags** – Have opening and closing tags

   ```
   <p>This is a paragraph</p>
   ```

7. **Empty (Self-closing) Tags** – No closing tag

```
<br>
<hr>
<img>
```

---

4. Commonly Used HTML Tags

Headings

```
<h1>Heading 1</h1>
<h2>Heading 2</h2>
<h3>Heading 3</h3>
```

Paragraph

```
<p>This is a paragraph.</p>
```

Line Break & Horizontal Line

```
<br>
<hr>
```

---

5. HTML Attributes

Attributes provide **additional information** about elements.

```
<tagname attribute="value">Content</tagname>
```

Example:

```
<p align="center">This text is centered</p>
```

Common attributes: - `id` - `class` - `style` - `src` - `href`

---

6. Links (Anchor Tag)

Used to create hyperlinks.

```
<a href="https://www.google.com">Visit Google</a>
```

---

7. Images

Used to display images on a web page.

```html
<img src="image.jpg" alt="Sample Image" width="200">
```

Attributes: - `src` – Image path - `alt` – Alternate text - `width` / `height`

---

8. Lists in HTML

Ordered List

```html
<ol>
    <li>HTML</li>
    <li>CSS</li>
    <li>JavaScript</li>
</ol>
```

Unordered List

```html
<ul>
    <li>Apple</li>
    <li>Banana</li>
</ul>
```

---

9. Tables

Used to display data in rows and columns.

```html
<table border="1">
    <tr>
        <th>Name</th>
        <th>Age</th>
    </tr>
    <tr>
        <td>Rahul</td>
        <td>20</td>
    </tr>
</table>
```

---

10. Forms

Forms are used to collect user input.

```html
<form>
    Name: <input type="text"><br><br>
    Password: <input type="password"><br><br>
    <input type="submit" value="Submit">
</form>
```

11. HTML Comments

Comments are not displayed in the browser.

```
<!-- This is a comment -->
```

---

12. Key Points for Students

- HTML files use `.html` extension
- Tags are **not case-sensitive** (but lowercase is recommended)
- Always close tags properly
- HTML defines **structure**, not design

---

13. Simple Practice Example

```
<!DOCTYPE html>
<html>
<head>
    <title>Student Page</title>
</head>
<body>
    <h1>Welcome Students</h1>
    <p>This is an HTML practice page.</p>
</body>
</html>
```

> ***"HTML is the skeleton of the web — once you learn it, every website starts to make sense."***

Next Step: Create a WTMA project folder and build your first multi-file webpage using VS Code.

Step 1 – Create a Basic index.html with below code

```
<!DOCTYPE html>
<html>
<head>
  <title>Student Record</title>
</head>

<body>

  <h2 align="center">WTMA Student Record</h2>

  <table border="1" cellpadding="10" cellspacing="0" align="center">
    <tr>
      <th>Sl.No</th>
      <th>Student Name</th>
      <th>USN</th>
      <th>Branch</th>
      <th>Semester</th>
      <th>Email</th>
    </tr>

    <tr>
      <td>1</td>
      <td>Rahul Kumar</td>
      <td>1CS21WT001</td>
      <td>CSE</td>
      <td>5</td>
```

```
      <td>rahul@gmail.com</td>
    </tr>

    <tr>
      <td>2</td>
      <td>Ananya S</td>
      <td>1CS21WT002</td>
      <td>CSE</td>
      <td>5</td>
      <td>ananya@gmail.com</td>
    </tr>

  </table>
</body>
</html>
```

Step 2 : style.css (Basic Styling for Student Table)

```
body {
  font-family: Arial, sans-serif;
  background-color: #f4f6f8;
}

h2 {
  text-align: center;
  color: #2c3e50;
}

table {
```

```
    width: 80%;
    margin: auto;
    border-collapse: collapse;
}

th {
    background-color: #34495e;
    color: white;
}

th, td {
    border: 1px solid black;
    padding: 10px;
    text-align: center;
}

tr:nth-child(even) {
    background-color: #e8e8e8;
}
```


```
<link rel="stylesheet" href="style.css">
```

Now see how your web page has visually improved by adding the css contents for style.

Step 3: script.js (Basic JavaScript for WTMA)

Add this to script.js

```
// Display welcome message
alert("Welcome to WTMA Student Record Page");

// Show total number of students
function showStudentCount() {
    let rows = document.getElementsByTagName("tr");
    let studentCount = rows.length - 1; // excluding header
    alert("Total Students: " + studentCount);
}

// Display current time on the page
function showCurrentTime() {
    let now = new Date();
    let time = now.toLocaleTimeString();
    document.getElementById("time").innerHTML = time;
}

// Update time every second
setInterval(showCurrentTime, 1000);
```

```
// Display welcome message

alert("Welcome to WTMA Student Record Page");

// Show total number of students

function  studentCount () {

   let rows = document.getElementsByTagName("tr");

   let studentCount = rows.length - 1; // excluding header

   alert("Total Students: " + studentCount);

}

// Display current time on the page

function showCurrentTime() {

   let now = new Date();

   let time = now.toLocaleTimeString();


document.getElementById("time").innerHTML = time;

}

// Update time every second

setInterval(showCurrentTime, 1000);
```

```
<script src="script.js"></script>
```

Now you need to add the scetion in index.html for diplayin the dynamic content used in Javascript

```
<p align="center">
    Current Time: <span id="time"></span>
</p>
```

Inside `<body>`, below the table:

```
html

<button onclick="showStudentCount()">Show Student Count</button>
```

And ensure JS is linked at the end:

```
html

<script src="script.js"></script>
```

Current Time:

<button type="button" onclick="studentCount()">Show Student Count</button>

Step 4 : Get explanation of Javascript/html using AI tool → workik Editor

https://workik.com/html-code-generator