

## Lab 1: Foundations of Web Development (HTML, Web Basics & Tools)

### Prologue

Before starting today's lab, you should complete the following setup:

1. If you do not have it already, install [Microsoft Visual Studio Code](#)
2. Install Live Server [ritwickdey.LiveServer](#)

### Lab Overview

#### Purpose of this Lab

This lab helps you practice the core concepts covered in Week 1 lectures. You will apply your understanding of how the web works, basic HTML structure, common HTML elements, links, and development tools by creating your first simple webpage.

By the end of this lab, you will understand:

- How the web works (client, server, HTTP)
- The role of front-end development
- Basic file and folder organization
- HTML document structure
- Common HTML tags (headings, paragraphs, links)
- How to view your work in a browser

#### Learning Outcomes

After completing this lab, students will be able to:

- Explain the difference between the Internet and the World Wide Web
- Identify the role of a client and a server

- Create and structure an HTML document
- Use basic HTML elements correctly
- Create links using the <a> tag
- Open and preview a webpage using a browser
- Use Visual Studio Code as a development tool

## Part 1: Understanding How the Web Works (Concept Check)

Before writing any code, answer the following questions **in comments** at the top of your index.html file.

<!--

1. What is the Internet?
2. What is the World Wide Web?
3. What is a client?
4. What is a server?
5. What does HTTP stand for?

-->

Write your answers in **your own words** using simple sentences.

## Part 2: Project Setup (Developer Tools)

### Step 1: Folder Structure

1. Create a main folder for this course.
2. Inside it, create a folder called **Labs**.
3. Inside **Labs**, create a folder called **Lab1**.
4. Inside **Lab1**, create a file called index.html.

Your folder structure should look like this:

Labs

└── Lab1

    └── index.html

## Part 3: Creating Your First HTML Page

### Step 1: HTML Boilerplate

1. Open index.html in VS Code.
2. Ensure your file includes:
  - <!DOCTYPE html>
  - <html>
  - <head>
  - <title>
  - <body>

Set the <title> to: **My First Web Page**

## Part 4: Adding Page Content (HTML Elements)

Inside the <body> tag, add the following:

### Required Content

- 1. Main Heading (<h1>)**
  - Text: *Welcome to My First Website*
- 2. Subheading (<h2>)**
  - Text: *About This Page*
- 3. Paragraphs (<p>)**
  - Write two short paragraphs explaining:
    - What is front-end development
    - What are your takeaway goals from this course
- 4. Horizontal Line**
  - Add a horizontal rule (<hr>) between sections

## Part 5: Working with Links

Add the following links to your page:

1. A link to Google that opens in a new tab
2. A link to Wikipedia with a title attribute
3. An email link using mailto:

Example format:

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

## **Part 6: Line Breaks and Page Structure**

1. Add a paragraph that uses a line break (<br>).
2. Add at least three more paragraphs of placeholder or original text.

## **Part 7: Viewing Your Page in the Browser**

1. Save your file.
2. Right-click index.html.
3. Open it using:
  - A browser OR
  - Live Server (if installed)

Confirm:

- The page loads correctly
- Headings are larger than paragraphs
- Links are clickable

## **Part 8: Reflection (Required)**

At the bottom of your HTML file, add a comment answering:

```
<!--
```

Reflection:

1. What part of this lab was easiest for you?

2. What part was most confusing?
3. What is one thing you learned about how websites work?

-->

### **Lab Checklist (Self-Assessment)**

- HTML boilerplate added correctly
- Headings used in the correct order
- At least 5 paragraphs included
- Links work correctly
- Page opens in browser
- Concept questions answered in comments
- Reflection completed