GOMYCODE.

Create a Website with code

Setting Up Your Developer Environment

For this course, we'll need three tools that every web developer uses:

1. A text editor to write code.

2. A web browser to preview what we're building.

3. A version control and Repository Manager (Git & Github).

While you can use any web browser or text editor, we recommend using Google Chrome and VSCode for this course.

- **Google Chrome** can be downloaded from https://www.google.com/chrome/

- **Visual Studio Code** can be downloaded from https://code.visualstudio.com/.

Like any good craftsperson, it's important that you become familiar with your tools. In order to do so, we'll have a look at what is at our disposal.

Google Chrome : is a web browser. You may or may not already be using Google Chrome for browsing the web. A web browser allows us to view web pages (similar to the one you're reading this very text on). We'll use Google Chrome to view the results after writing and making changes to our code.

Visual Studio Code : is a popular code editor. A code editor is a very aptly named software; it helps us edit text. It is similar to the default text editor that comes preinstalled on Mac or Windows, but has additional features like code highlighting to enhance our coding and debugging experience.

Code Editor (IDE)

If you're going to do any kind of development, you're going to spend a lot of time in front of a text editor. There are many coding languages on the web, but let's focus on the big three (HTML, CSS & JavaScript) and setup our environment to manipulate them.

Visual Studio Code (or VSCode) is the most used text editor.

VSCode provides a ton of plugins that make a developer's life as easy as possible. When you're using it, you have the ability to make it fit your preferences by installing the right plugins.

The plugins that you're using have a huge impact on your productivity and the way you work.

VSCode Extensions

Here are some examples of what you can achieve with an Extension API:

Prettier

Prettier is one of the best plugins for developers who need to follow a well-laid set of rules when developing. It's a robust opinionated code formatter that allows developers to format their codes in a structured way.
Prettier works together with JavaScript, HTML, CSS, Markdown, and other modern tools and allows you to properly format your code. Link

Themes

Since you'll be looking at your editor every day, why not make it as pretty as possible? There are tons of customization plugins that change the color scheme and the icons in the sidebar. Some popular themes that are available for free are One Monokai, One Dark Pro, and Material Icon. link

Bracket Pair Colorizer

This extension allows matching brackets to be identified with colours. The user can define which characters to match, and which colours to use. Link

Create Your First Sample Web Page

Follow the next steps to create your first web page in this track:
Create a file and name it whatever you want with the extension .html: which is a markup language of a web page. We will learn more about HTML in the next super skill.

Open this file with VSCode, copy and paste the next few lines:

```
<!DOCTYPE html>

<html>
        <head>
                <title>Page Title</title>
        </head>
        <body>
                <h1>Woooohooow this is my first page :)</h1>
                <p>This is a paragraph.</p>
        </body>
</html>
```

Save the changes and then open it with your browser and this is your first basic web page. There is still a long way to go since we have a lot more to add. Our next steps would be to make it look better and style it using CSS. Afterward, we will make it interactive and user-friendly using Javascript. You'll learn this in the upcoming super skills.

Organizing Your Work

One of the fundamentals of web development is organization. In order to achieve that, here is the folder creation step-by-step that we'll use during our session:

1. Create a folder named "GoMyCode" on your Desktop.

2. Create a new folder in **GoMyCode** folder and name it whatever you want.

3. Create a new folder inside that one and name it "**res**".

4. Create a new folder inside that one and name it "**css**".

5. Create a new folder inside that one and name it "**js**".

6. Create a new text file, name it "**index**", and change its extension to ".**html**".

7. Inside the "**css**" folder, create a new text file, name it "style", and change its extension to

   ".**css**".

8. Inside the "**js**" folder, create a new text file, name it "main" and change its extension to ".**js**".