Frontend Assignment Set

Module 1 – Foundation

• What is a HTTP?

⇨ **HTTP** stands for **HyperText Transfer Protocol**. It's the foundation of data communication on the **World Wide Web**.

⇨ **In Simple Terms:**

⇨ HTTP is the protocol used by **web browsers and servers** to communicate. When you type a website URL into your browser (like `https://www.example.com`), your browser uses HTTP to request that page from the website's server.

• What is a Browsers?

⇨ A **browser** is like a translator between you and the internet. It takes code written in languages like **HTML, CSS, and JavaScript** and turns it into what you see: text, images, videos, buttons, etc.

Component	Purpose
User Interface	The part you interact with (address bar, back button)
Browser Engine	Coordinates actions between UI and rendering engine
Rendering Engine	Displays the content (turns code into visuals)
Networking	Handles HTTP requests and data from the internet
JavaScript Engine	Runs JavaScript code (animations, interactions)
Data Storage	Stores cookies, cache, local storage

How they works?

⇨ A **domain name** is like the **name of a house**, while the **IP address** (like `192.0.2.1`) is the actual **location** of that house on the internet.

⇨ Instead of remembering numbers, you use a name like:
⇨ `www.google.com`
⇨ `www.wikipedia.org`

• What is Domain Name?

⇨ A **domain name** is like the **name of a house**, while the **IP address** (like `192.0.2.1`) is the actual **location** of that house on the internet.
⇨ Instead of remembering numbers, you use a name like:
⇨ `www.google.com`
⇨ `www.wikipedia.org`

• What is hosting?

⇨ **Hosting** (or **web hosting**) is a service that **stores your website's files and makes them available on the internet**.
⇨ Think of it this way:
⇨ **A domain name is your website's address**, and
Hosting is the house where your website lives.

Module 2 – Fundamentals of World Wide Web

• Difference between Web Designer and Web Developer

Feature	Web Designer	Web Developer
Main Focus	Visual look and feel of the website	Functionality and technical structure
Works With	Colors, layout, typography, user experience	Code, databases, servers, logic
Key Skills	UI/UX design, graphic design, Adobe XD, Figma	Programming languages like HTML, CSS, JS, PHP

⇨ Feature	⇨ Web Designer	⇨ Web Developer
⇨ **Tools Used**	⇨ Photoshop, Figma, Sketch, Canva	⇨ VS Code, Git, Terminal, Chrome DevTools
⇨ **Output**	⇨ Website mockups or prototypes	⇨ Fully functional website
⇨ **Goal**	⇨ Make it **beautiful and user-friendly**	⇨ Make it **work properly and efficiently**

• What is a W3C?

⇨ **W3C** stands for the **World Wide Web Consortium**.
⇨ It is the **main international organization** that **develops web standards** to ensure the **long-term growth, consistency, and accessibility** of the internet.

• What is Domain?

⇨ A **domain** (short for *domain name*) is the **human-friendly address of a website** on the internet.
⇨ Instead of typing a complicated IP address like `192.0.2.1`, you type something like:
⇨ `www.google.com`
⇨ `www.example.org`
⇨ That's the **domain** — it's how people find your website online.

• What SEO?

⇨ **SEO** stands for **Search Engine Optimization**.
It's the process of **improving a website** so it ranks **higher on search engines** like Google, Bing, or Yahoo — meaning **more people can find it**.

• What is SDLC life cycle?

⇨ **SDLC** stands for **Software Development Life Cycle**.
It is a **step-by-step process** used to **design, develop, test, and deploy software**.
⇨ The goal of SDLC is to ensure that software is built **systematically, efficiently, and with high quality**.

Phase	Description
1. **Planning**	Understand the project goal and create a plan
2. **Requirement Analysis**	Gather and analyze what the user needs
3. **Design**	Plan how the software will look and function (architecture, UI/UX)
4. **Development**	Write the actual code for the software
5. **Testing**	Test the software to find and fix bugs or issues
6. **Deployment**	Release the software to users or the public
7. **Maintenance**	Fix issues, make updates, and improve over time