AKPOTU AUGUSTINE NTISHOR

Setting Up Development Environment with VS Code, GitHub, and Browser Developer Tools: Documentation

Installed VS Code and Extensions

- Downloaded and installed VS Code from the official website
- Installed the following extensions:
    - GitLens: Provides Git version control features, including repository management, commit history, and branch switching
    - Live Server: Enables live preview of HTML/CSS/JS changes in the browser
    - Debugger for Chrome: Allows debugging of JavaScript code in Chrome
    - Prettier: Formats code for consistency and readability
    - ESLint: Lints JavaScript code for errors and best practices

Configured VS Code Settings

- Set default editor settings:
    - Font size: 14
    - Theme: Dark+
- Set up Git credentials:
    - Username: [Your GitHub username]
    - Email: [Your GitHub email]

Set-Up GitHub

- Created a GitHub account if not already existing
- Set up GitHub profile with username, email, and password
- Installed GitHub Desktop app (optional)
- Connected VS Code to GitHub using GitLens extension

Enabled Browser Developer Tools

- Choose Chrome as the preferred browser

- Enabled developer tools in Chrome:
    - Pressed F12 or Ctrl+Shift+I (Windows/Linux) or Cmd+Opt+I (Mac)
- Familiarized myself with the browser developer tools:
    - Elements panel (HTML/CSS inspection)
    - Console panel (JavaScript debugging)
    - Sources panel (JavaScript debugging)
    - Network panel (network request inspection)

Set-Up Project Repository

- Created a new project folder
- Initialized a Git repository using git init
- Connected project folder to GitHub repository using git remote add origin
- Committed initial changes and pushed to GitHub using git push

Necessary Extensions and Their Functions

- GitLens: Git version control, repository management, commit history, and branch switching
- Live Server: Live preview of HTML/CSS/JS changes
- Debugger for Chrome: Debugging of JavaScript code in Chrome
- Prettier: Code formatting for consistency and readability
- ESLint: Linting of JavaScript code for errors and best practices

Conclusion

By following these steps, I set up a comprehensive development environment with VS Code, GitHub, and browser developer tools. The installed extensions enhance productivity, version control, and debugging capabilities. This setup enables efficient coding, testing, and collaboration for web development projects.