Version Control with Git and GitHub

Description:

This guide is designed to introduce web developers to version control systems, focusing on Git and GitHub. Learn how to track changes in your code, collaborate with other developers, and deploy projects more efficiently using Git.

Key Topics Covered:

1. What is Version Control?

- Definition of version control and why it is crucial in modern web development.
- An overview of different version control systems, with a focus on Git.

2. Setting Up Git

- A step-by-step guide to installing Git on your local machine.
- Configuring your Git username and email for version control tracking.

3. Basic Git Commands

- Common Git commands such as git init, git add, git commit, and git push.
- Explanation of each command with examples.

Example:

```
git init
git add .
git commit -m "Initial commit"
git push origin master
```

4. Understanding Branching and Merging

- How to create and switch between branches using git checkout and git branch.
- Merging branches and handling merge conflicts.

5. Using GitHub for Collaboration

- How to push your local repository to GitHub and collaborate with others by forking and pulling requests.
- Setting up a GitHub repository and managing permissions for team collaboration.

6. Reverting Changes and Using Git Logs

- How to undo changes with git revert and git reset.
- Using git log to track changes and view the history of commits.

7. Git Workflows

- Popular Git workflows like Feature Branch, GitFlow, and Forking Workflow.
- Best practices for managing collaborative projects using GitHub Issues, Pull Requests, and Project boards.

Practice Exercises:

1. Set Up a Git Repository

- Create a new Git repository, make some changes, and push them to a GitHub repository.
- Invite a friend to collaborate, and practice pulling and merging changes.

2. Resolve a Merge Conflict

- Simulate a merge conflict in a test project and resolve it using Git.

Next Steps:

- **Further Learning**: Explore more advanced Git features like stash, rebase, and cherry-pick to enhance your workflow.

- **Further Resources**:
  - *Pro Git by Scott Chacon and Ben Straub* (Book)
  - *GitHub Learning Lab* (Website)
  - *Git Documentation* (Website)