DAY 14

Git Commands

Git commands are essential for version control and collaboration in software development. Here are some key commands:

- **git init**: Initializes a new Git repository locally.
- **git clone**: Copies a remote repository to your local machine.
- **git add**: Adds changes in the working directory to the staging area.
- **git commit**: Records changes in the repository with a message.
- **git push**: Uploads local repository content to a remote repository.
- **git pull**: Fetches and integrates changes from a remote repository to a local repository.
- **git branch**: Lists, creates, or deletes branches.
- **git merge**: Combines changes from different branches.
- **git checkout**: Switches branches or restores working tree files.

Forking Repositories

Forking a repository creates a copy of a project under your GitHub account. It allows you to freely experiment with changes without affecting the original project. Key benefits include:

- **Experimentation**: Test new features or modifications without affecting the main project.
- **Contributions**: Make contributions to open-source projects through pull requests.
- **Collaboration**: Collaborate with others by sharing your fork or merging changes back upstream.

TOTP Apps for Authentication (e.g., Authy)

Time-Based One-Time Password (TOTP) apps generate secure authentication codes that expire after a short period. Authy is a popular TOTP app that provides multi-device support and backup options for added security. Here’s how it works:

1. **Setup**:
   - Install Authy on your mobile device or desktop.
   - Enable two-factor authentication (2FA) on services that support TOTP.
2. **Usage**:
   - When logging in, enter your username and password.
   - Open Authy to generate a temporary authentication code.
   - Enter the code within the time limit provided by the service.
3. **Security**:
   - TOTP codes are generated locally on your device and expire after a short duration, enhancing security.
   - Authy offers encrypted backups and multi-device sync for seamless access across devices.

Implementing Git Commands, Forking, and TOTP Apps

1. **Git Commands**:
   - Practice using essential Git commands (git clone, git add, git commit, git push, git pull) to manage version control effectively.
   - Explore branching (git branch, git checkout, git merge) for concurrent development and feature isolation.
2. **Forking Repositories**:
   - Fork repositories on GitHub to contribute to open-source projects or experiment with changes.
   - Create pull requests to propose changes and collaborate with project maintainers.
3. **TOTP Apps (e.g., Authy)**:
   - Secure your accounts with two-factor authentication using TOTP apps like Authy.
   - Enable backup and sync features to ensure access to authentication codes across devices.

By mastering Git commands, utilizing forking for collaboration, and securing accounts with TOTP apps like Authy, you can enhance your development workflow, contribute effectively to projects, and strengthen account security practices.