

Spark! - Git Level 3

Part 1: Local Repository and Basic Git Operations

1. **Create a Local Repository**
 - Initialize a new Git repository in a local folder.
 2. **Create and Edit Files**
 - Create a simple README.md
 - Write a brief description of the project.
 - Add the file to the staging area using `git add` and commit it with the message "**Initial commit**".
 3. **Branching and Merging**
 - Create a branch called **feature**.
 - In the **feature** branch, create a file called **sum_of.py** and add a function named **sum_of** that takes in two integers and returns the sum.
 - Add and commit the change in the **feature** branch with the message "Create adding function".
 - Switch back to the main branch.
 - Merge the **feature** branch into the main branch.
 4. **Push to a Remote Repository**
 - Create a repository on GitHub.
 - Set up the remote repository.
 - Push your local changes to the remote repository.
-

Part 2: Resolve Merge Conflict and Push to Remote Repository

1. **Change main branch**
 - In the **main** branch, make a change to the **sum_of.py** file so that it takes in an **array of ints** and returns the sum of the ints.
 - Add and Commit the change in the **main** branch with the message "Update sum_of so it takes in array".
2. **Change feature branch**
 - Switch to the **feature** branch and make a conflicting change to **sum_of.py** by returning a tuple of the sum of ints and the length of the array
 - Add and Commit the change in the **feature** branch with the message "Create conflicting change".
3. **Resolve Merge Conflict**
 - Attempt to merge **main** into **feature** and resolve the merge conflict that arises.
 - Commit the merge resolution with the message "Merge Resolved"

Part 3: Updating from Upstream Repository

1. Fork a Remote Repository

- Navigate to this [repository](#) on GitHub.
- Fork the repository to your personal GitHub account.
- Clone the forked repository to your local machine.

2. Wait for the Repository to update

- In this part, we will be pulling changes made to the original repository. We have set up automation so that the repository updates everyday at midnight.
- Wait until after midnight to ensure the update has occurred.
- Confirm the repository update by checking the commit history or last updated time.

3. Pull Changes from the Original (Upstream) Repository

- Set the original repository (the one you forked from) as the upstream remote.
- Fetch any new changes from the upstream repository.
- Merge the upstream changes into your local branch.

4. Resolve Merge Conflicts

- If any merge conflicts arise during the merge, resolve them manually by editing the conflicting files.
- Once conflicts are resolved, stage the resolved files and complete the merge with the message "Resolve merge conflicts from upstream"

5. Push Changes to Your Forked Repository

- Once all changes are complete and the merge conflicts are resolved, push your branch to your forked repository(Your copy repository).

6. Create a Pull Request

- Navigate to your forked repository on GitHub.
- Create a pull request from your feature branch to the main branch of the original repository.
- Add a title and include your name. Also, add a description of what changes you saw were made when you fetched from the upstream repo.
- Submit the pull request for review.

Submission

For your submission, follow this [template](#) to create an Issue in the Git Level 3 repo.