GitHub Workflow Tutorial: Cloning, Branching, and Pushing Code

This professional tutorial document guides students or new developers through the essential steps of working with GitHub in a collaborative setting. It includes how to clone a repository, create a branch, and push your changes to the remote repository.

Prerequisites

- Git must be installed on your system.
- You need a GitHub account.
- You should have access to a GitHub repository (either your own or one shared with you).

⚡ Step 1: Clone the Repository

Cloning a repository means downloading the project from GitHub to your local machine.

1. Copy the Repository URL

- Go to the GitHub page for the repository (e.g., https://github.com/username/MOP-Code)
- Click the green **Code** button
- Copy the **HTTPS** URL (e.g., `https://github.com/username/MOP-Code.git`)

2. Open Command Prompt or Terminal and run:

```
cd C:\Users\YourName\Documents\GitHub

git clone https://github.com/username/MOP-Code.git
```

Replace **username** with the actual GitHub username. This creates a folder called MOP-Code.

⚡ Step 2: Navigate Into the Repository

```
cd MOP-Code
```

Now you're inside the working directory where you'll manage all your Git work.

⚡ Step 3: Create and Switch to a New Branch

Working on a branch keeps your changes isolated from the `main` or `master` branch.

```
git checkout -b feature-branch-name
```

Example: `git checkout -b update-readme`

This command both **creates** and **switches** you to the new branch.

To verify:

```
git branch
```

You should see:

```
* update-readme
  main
```

⚡ Step 4: Make Changes

Open the project folder in your code editor (VS Code, PyCharm, etc.) and make any required edits.

⚡ Step 5: Stage and Commit Your Changes

```
git add .
git commit -m "Descriptive message about what you changed"
```

Example:

```
git commit -m "Updated the README with setup instructions"
```

⚡ Step 6: Push Your Branch to GitHub

```
git push origin feature-branch-name
```

Example: git push origin update-readme

This uploads your branch and commits to GitHub.

⚡ Step 7: Create a Pull Request (PR)

1. Visit the repository on GitHub
2. You will see a prompt: **"Compare & pull request"**
3. Click the button to open a PR
4. Add a title and description, then click **"Create pull request"**

This allows your changes to be reviewed and merged into the main project.

Summary of Key Git Commands

Action	Command
Clone a repo	git clone <repo-url>

Navigate to folder	cd repo-name
Create & switch branch	git checkout -b new-branch-name
Stage changes	git add .
Commit changes	git commit -m "message"
Push to GitHub	git push origin new-branch-name

Best Practices

- Always work in a separate branch, not **main**.
- Use clear, descriptive commit messages.
- Open pull requests for review before merging.
- Keep branches focused on single features or fixes.

By following this tutorial, you now understand how to safely and professionally contribute to a GitHub repository using branching workflows.