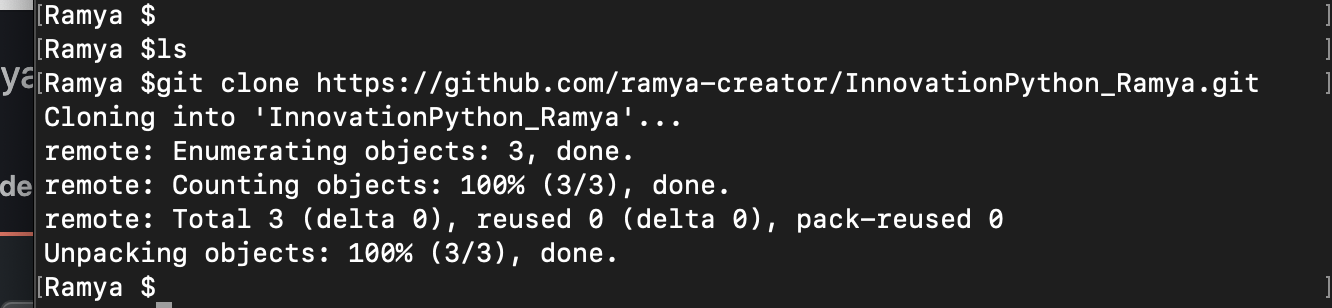
**Ramya Vemuru**

1. Make a repository on GitHub with the name “InnovationPython\_yourname”

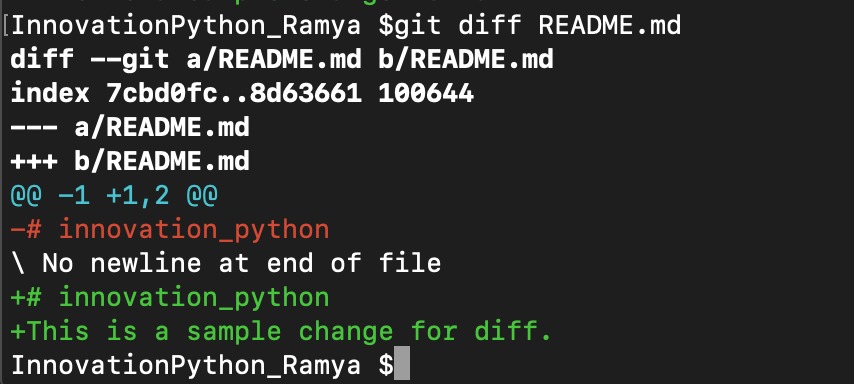
eg: “InnovationPython\_Ankush”.

Practice on following commands:

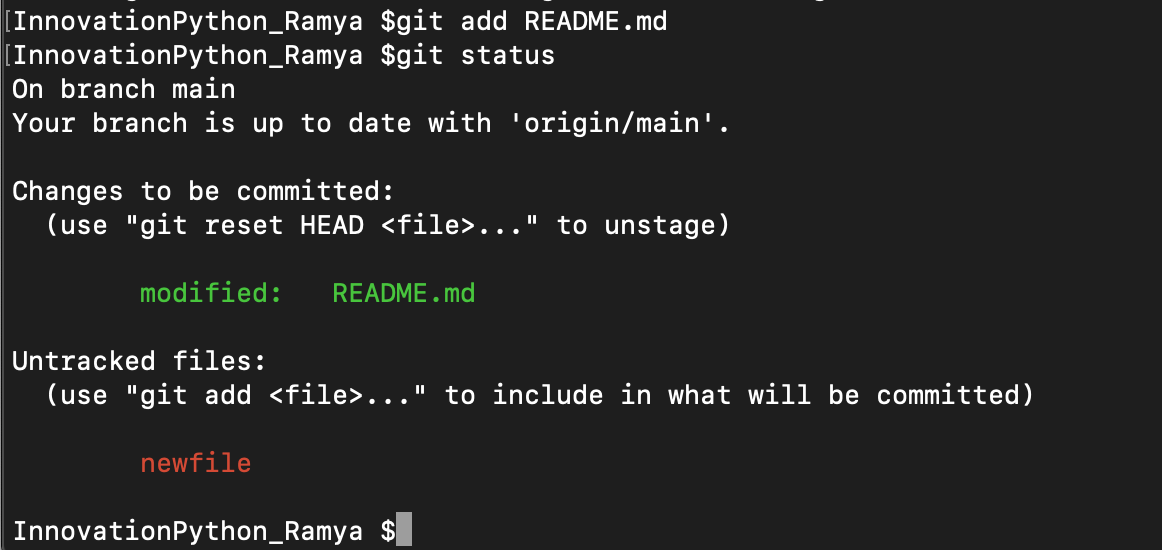
Git Clone:



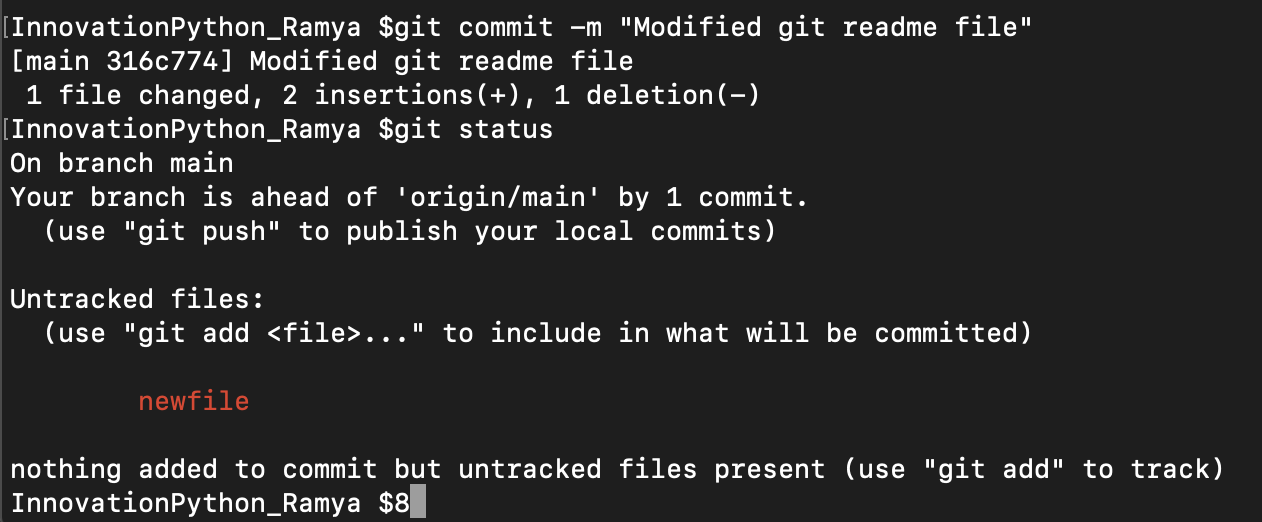
Git Diff:



Git Add, Git Status:



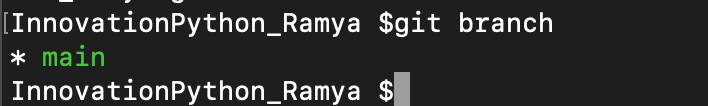
Git Commit:



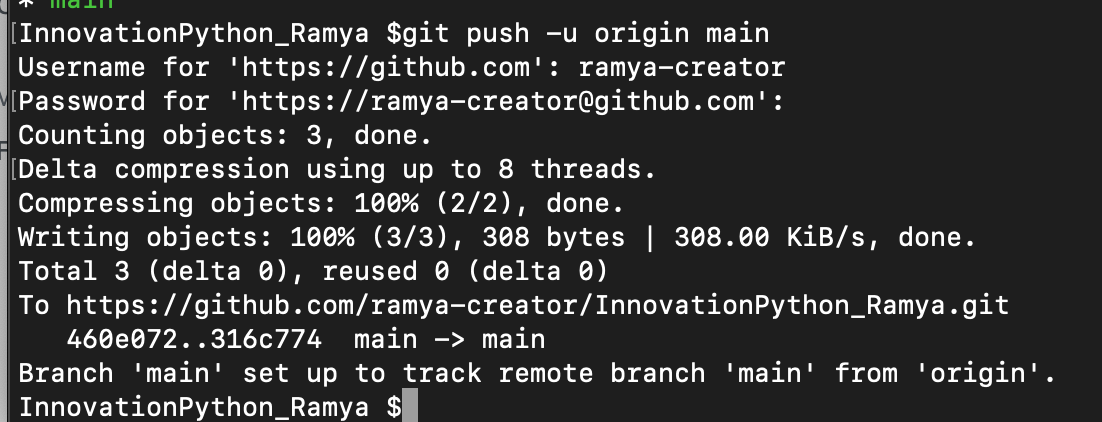
Git Log:



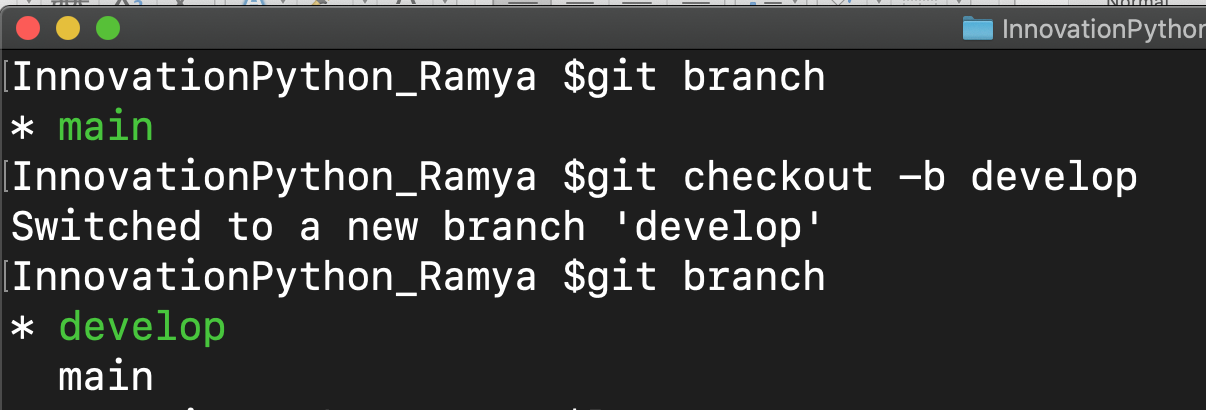
Git Branch:



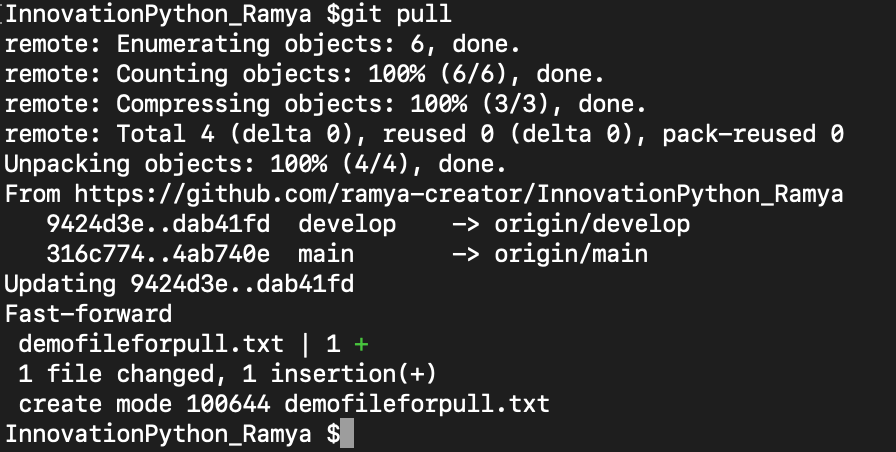
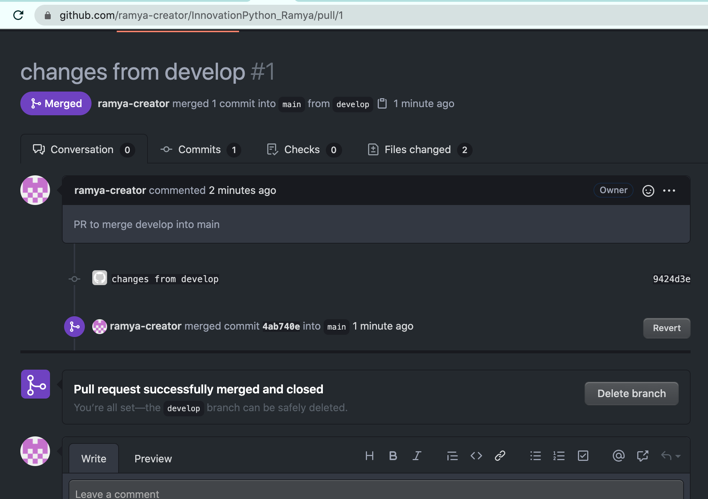
Git Push:



Git Checkout:



Git Pull:



(2) Read about difference between Git and Github.

**Git** is a revision control system, a tool to manage your source code history. **GitHub** is a hosting service for Git repositories.

(3) Read about Git Workflow.

Git Workflow is a recipe or recommendation for how to use Git to accomplish work in a consistent and productive manner. Git workflows encourage users to leverage Git effectively and consistently. Git can be used effectively and consistently by the developers because of the Git Workflows.

(4) How many types of version control systems are there?

-Local version control system

-Centralized version control system

-Distributed version control system

(5) Explain Branching concept in Git.

A branch in Git is simply a lightweight movable pointer to one of these commits. The default branch name in Git is master. As you start making commits, you're given a master branch that points to the last commit you made.

(6) Explain Forking Workflow in Git.

The Forking Workflow is fundamentally different than other popular Git workflows. Instead of using a single server-side repository to act as the “central” codebase, it gives every developer their own server-side repository. This means that each contributor has not one, but two Git repositories: a private local one and a public server-side one. The Forking Workflow is most often seen in public open source projects.