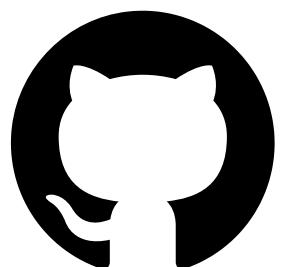
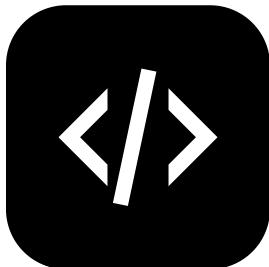


# Version Control

---

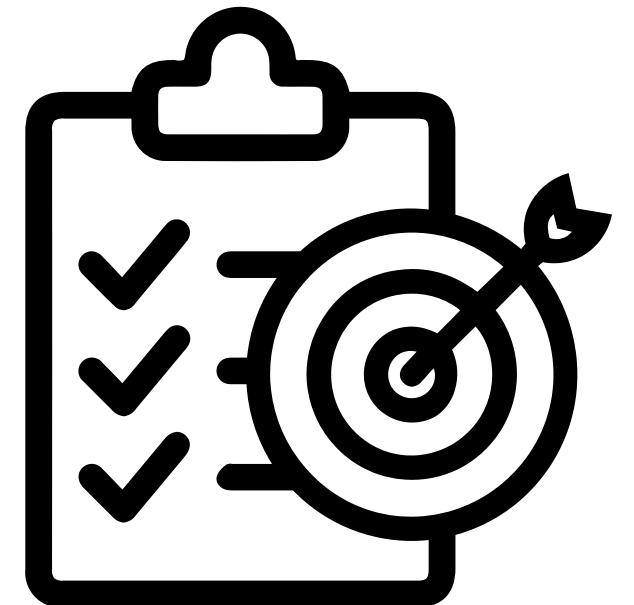
**VERSION CONTROL USING GIT AND GITHUB**



# AGENDA

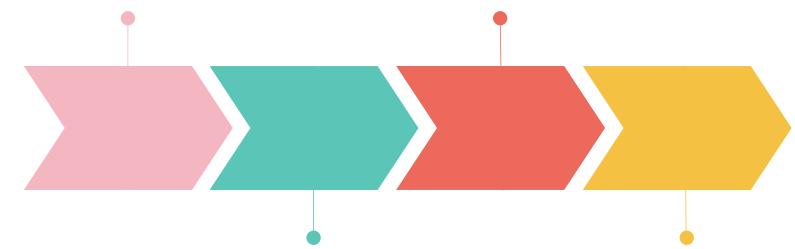
---

- Version Control
- Git and Github
- Basic Workflow
- Collaborating with Git
- Resources



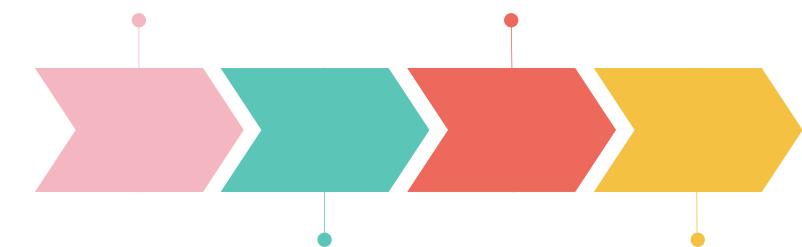
# OLD VERSION CONTROL

Name
main-final-final.go
main-final.go
main.go



# VERSION CONTROL

- A system that records changes to a file or set of files over time so you can recall specific versions later.
- **Benefits:**
  - Tracks history of changes
  - Collaboration with multiple contributors
  - Backup and restore capabilities

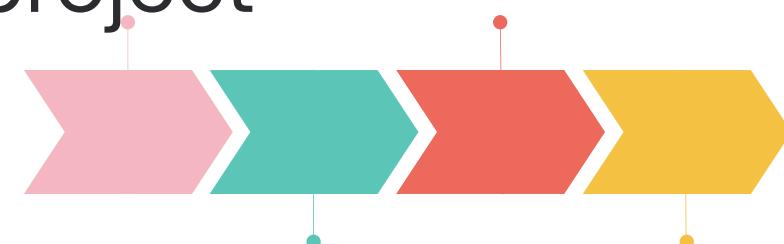


- **Git**

- Git is a version control system that allows you to track changes in your code and collaborate with others.
- Key Features:
  - Local repository for every user
  - Fast performance, branching and merging capabilities

- **Github**

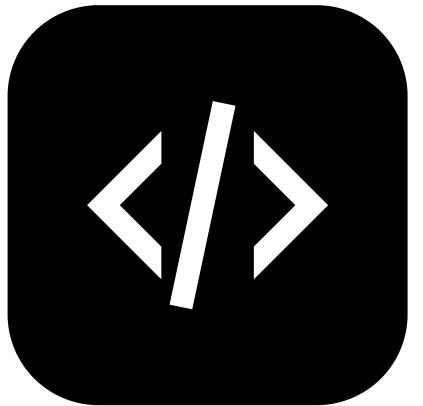
- GitHub is a web-based platform that provides hosting for Git repositories and offers additional features such as issue tracking and project management.



# KEY CONCEPTS

---

- Repository:
  - A directory or folder that contains files and metadata for a project under version control.
- Stage:
  - The area where changes are prepared for a commit.
- Commit:
  - A recorded change to the repository.
- Push:
  - Sending local commits to a remote repository.
- Pull:
  - Fetching and integrating changes from a remote repository



# BASIC WORKFLOW WITH GIT AND GITHUB



- Set up a repository:
  - git init or clone from GitHub
- Make changes:
  - Edit files in your project
- Stage changes:
  - git add <file>
- Commit changes:
  - git commit -m "Describe changes"
- Push changes:
  - git push to GitHub
- Collaborate:
  - Pull requests and code reviews



# INITIALIZING A REPOSITORY USING GIT



- On your computer, create a directory with the name “git-tutorial”.
- Open your command prompt or terminal window.
- Navigate to the directory.
- Run the command “git init” to initialize a new Git repository in the directory.
- Create a README.md file using the command “git add README.md”
- Run the command “git remote add origin <URL>”



# KEY GIT COMMANDS

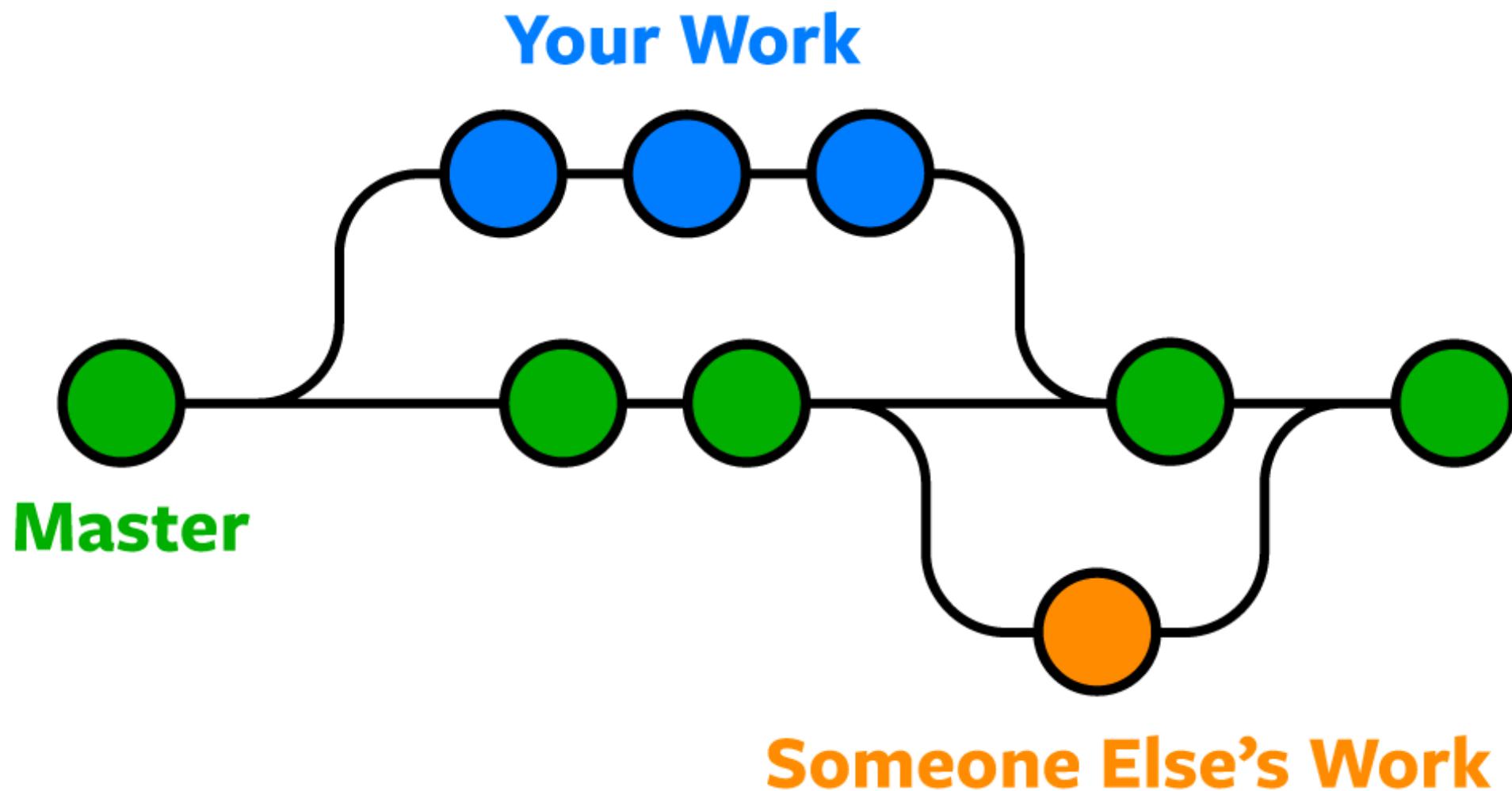


- **git init:** Initialize a new Git repository
- **git clone:** Copy an existing repository
- **git status:** Show the working directory status
- **git add:** Add files to staging area
- **git commit:** Commit changes to the repository
- **git push:** Push changes to a remote repository
- **git pull:** Fetch and integrate changes from a remote repository



# COLLABORATING WITH GIT

- Using Git's features to manage and integrate changes from multiple contributors.



- Branch:
  - A parallel version of the repository to work on different features.
- Merge:
  - Combining changes from different branches.
- Conflict:
  - When changes from different branches collide and need resolution.
- Pull Request:
  - A request to merge changes from one branch to another, often used for code review.



# PRACTICAL EXAMPLE

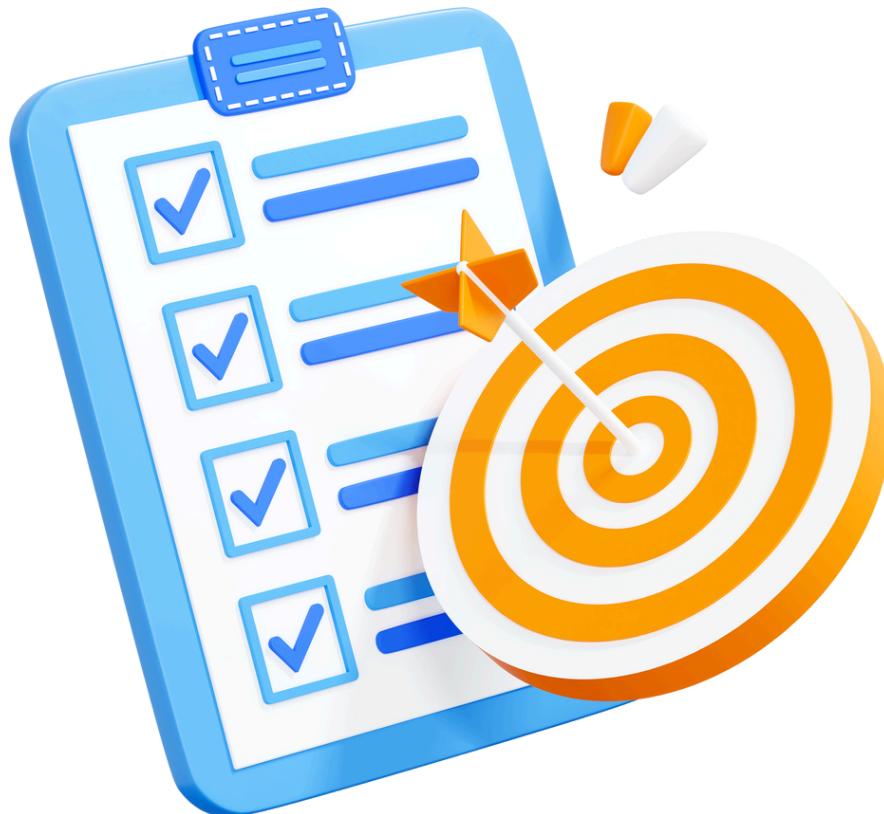
- Clone a Repository:
  - `git clone "https://github.com/Mahleth/code-review-practice.git"`
- Create a Branch:
  - `git branch new-feature` and `git checkout new-feature`
  - Make changes to `README.md`
  - Then `git commit -m "New feature"`
- Merge
  - `git checkout main` and `git merge new-feature`
- Create a Pull Request:
  - Push a branch to GitHub and create a pull request for review



# BEST PRACTICES

---

- Use branches for new features or bug fixes
- Regularly pull updates from the main branch
- Resolve conflicts through merging
- Use pull requests for code reviews and discussions



- **Git and Github**

- Git Documentation: [git-scm.com/doc](https://git-scm.com/doc)
- GitHub Guides: [guides.github.com](https://guides.github.com)
- Interactive Tutorial: [learngitbranching.js.org](https://learngitbranching.js.org)

- **Code Review**

- [Revised Review Guideline](#)

- **Documentation templates and style guides:**

- [6 Best Practices to Manage Pull Request Creation and Feedback](#)
- [Using templates to encourage useful issues and pull requests - GitHub Docs](#)

# Thank you!

---

