

# Using Git and GitHub for Version Control and Project Collaboration

Haseeb Ali – Applied Economics

11/8/2023



UNIVERSITY OF MINNESOTA

**Driven to Discover®**

# Goals

- Introduce git concepts
- Gain exposure to a common development workflow
- Share resources for further knowledge



# Before Starting

- Install+Download R+Rstudio
- Set up github account (<https://github.com/join>)
- Set up git/github with R/Rstudio  
(<https://rfortherestofus.com/2021/02/how-to-use-git-github-with-r>)



# What is version control?

- The practice of tracking and managing changes to software code over time
  - What if you make a change but later want to go back to a previous version?

## git

- Distributed version control software
- Command-line tool
- Installed locally on computer (or server)

## GitHub

- Web-based hosting service for git repositories
- Graphical user interface (GUI)
- [github.com](https://github.com)



# What is a repository?

- A repository is a collection of files and folders
  - It also stores the history of changes (commits) made to your files and folders
- You can think of a repository like a project
- A typical repository will contain:
  - Code files – may include multiple languages
  - README.md
  - Other adjacent project documentation

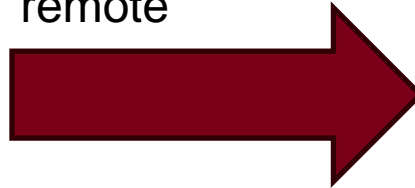


# Local vs Remote Repository

Personal Development Environment

Code development and testing

Push changes to remote



Github.com

Share code with teammates



# What is a 'commit'

- Staging: adds a change in the working directory to the staging area.
  - Main benefit: Helps control the size of your commits/snapshots. You can choose how large your commits are by selecting which files you would like to be included in the next commit/snapshot.
- Commit: Once you're happy with the snapshot you've created in the staging area, you commit it to the project. This is making it so that git is now tracking your changes locally.



# Branches

- Branches are a way to facilitate independent code development
  - They contain one or more commits
- Master/main branch: source of truth
  - Currently used, most up to date codebase. Changes made from here as a starting point.
- Independent branches should not contain conflicts
  - Collaborators should work on parts of the code base that are mutually independent.
  - This can still lead to problems if code files have interdependencies.





# What is a feature branch

- A feature is an individual update or enhancement to one component of the repository.
- A feature branch keeps development of features separate and independent from each other
  - Mostly used to distinguish from “main branch”.



# What does it mean to 'push' changes?

- Manually tell git to bring the changes that you have developed on your local feature branch to your remote feature branch
- Team is made aware of what I am developing and can clone my version of repository from remote.
- The 'pull' command can clone this branch (or changes to a branch), from remote to local.

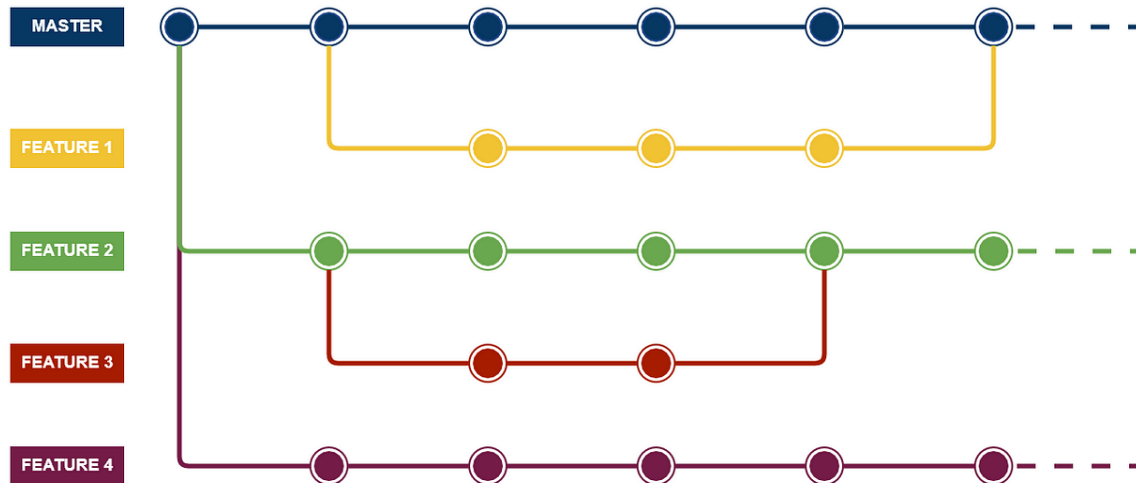


# How are the changes integrated?

- When I am done with my code development, I need to get feedback from my collaborators
  - Peer review and GitHub Pull Request
- When we are all happy with the updates, we integrate them with our original codebase.
  - Merge with master/main so the feature becomes the source of truth
  - Or, if working of a sub-branch, merge with the upper-branch.



# The development process



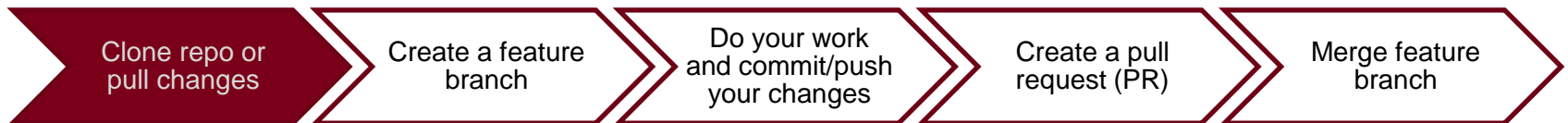
# These are the main commands you will use

- `$ git status` – see current state of local repository
- `$ git checkout` – change the branch the local repository is pointing to
- `$ git add` – stage new or changed files to be included in next commit
- `$ git commit` – save snapshot of repository
- `$ git push` – publish local changes to the remote repository
- `$ git pull` – get latest changes to the remote repository into local



# 1. Clone repository or pull changes

- Repository exists on remote (collaborators account on github.com) but you need a copy of the repository in your local development environment to do development and testing.

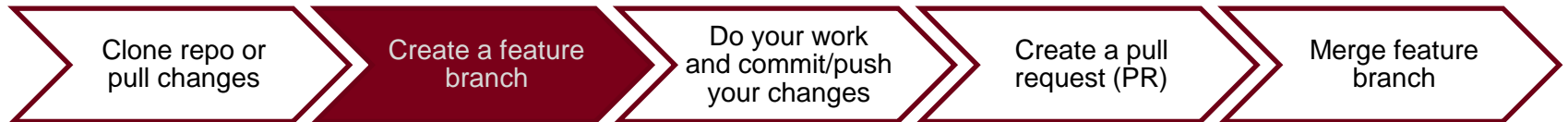


## 2. Create your feature branch

Switch to base branch (master/main)    `$ git checkout <base-branch>`

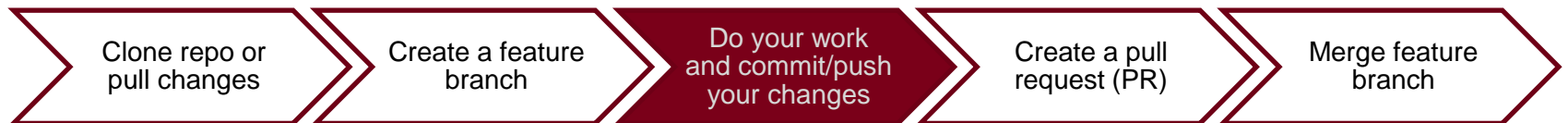
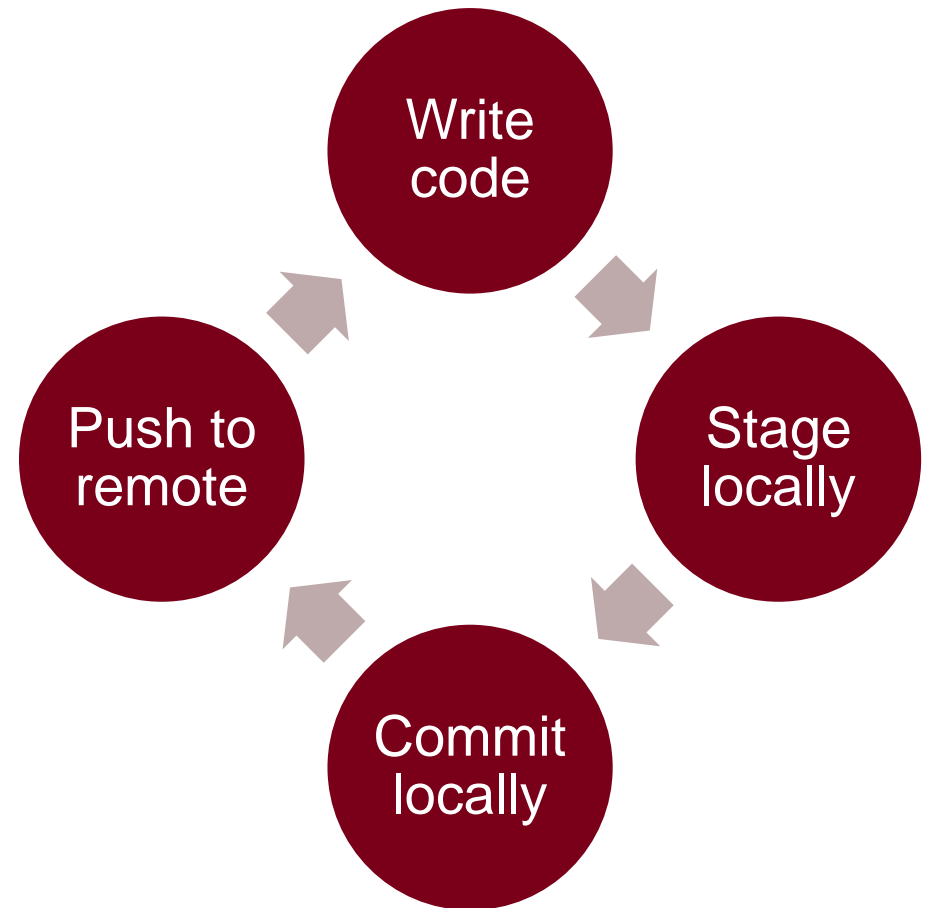
Create and switch to feature branch    `$ git checkout -b <lanid-feature-branch>`

Push new branch to remote    `$ git push -u origin <lanid-feature-branch>`



### 3. Start development work and commit/push often

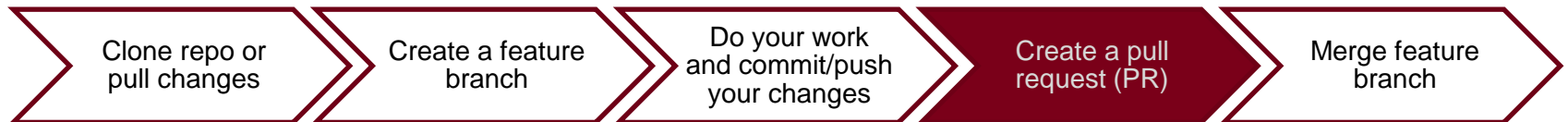
1. See files that have been created or changed:
  - `$ git status`
2. Stage created or changed files:
  - `$ git add <filename>`
3. Commit staged files (creating a snapshot):
  - `$ git commit -m "Your commit message here"`
4. Push committed changes to remote feature branch:
  - `$ git push`
5. Repeat until feature is complete





## 4. Create a pull request (PR)

- This step is a whole separate process facilitated through GitHub
- Assign a “Reviewer”, i.e. the person you want to examine and provide feedback for your work.



# For the Reviewer of Pull Request

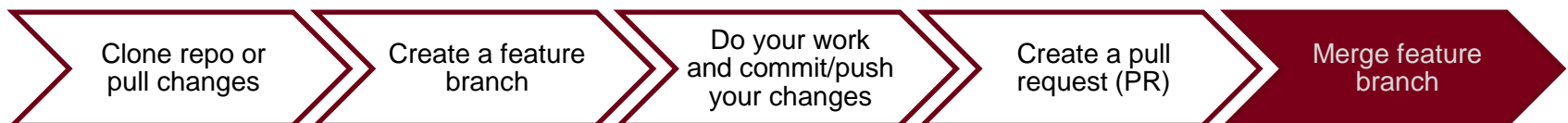
## Three Options

1. Provide comments only on github.com
2. Pull feature branch and test locally.
3. Pull feature branch, create your own branch off this branch, and create changes.
  - Submit your own pull request to merge your changes into the original feature branch.



# 5. Merge feature branch

- After the Pull Request is complete, update your local repository:
  - `$ git checkout master`
  - `$ git pull`



# Closing thoughts

- In practice, multiple people are developing at the same time.
- That means that a repository will have multiple feature branches
  - A feature branch should contain **independent** work
  - Git doesn't know how to handle multiple people editing the same file and conflicts will occur
- It is a best practice to commit and push your changes at least daily
- You should be merging and creating new feature branches regularly
- Utilizing Git/GitHub is not a replacement for communication



# Some Online Tutorials and Resources

- <https://rfortherestofus.com/2021/02/how-to-use-git-github-with-r>
- <https://www.youtube.com/@Riffomonas/videos>
- <https://happygitwithr.com/>





UNIVERSITY OF MINNESOTA

**Driven to Discover®**

Crookston Duluth Morris Rochester Twin Cities

The University of Minnesota is an equal opportunity educator and employer.