



ROITRAINING
MAXIMIZE YOUR TRAINING INVESTMENT™

Cheat Sheet: Git Branching and Collaboration

Using Branches

- Try creating a branch called: new-feature-X
 - Make a change and commit in that branch
 - Switch back to the main branch
 - Look at code
 - Switch back to new-feature-X
 - Look at code
 - Merge the changes back to main
- For more detailed steps, use the next two slides for help

Branching Commands

- To list all branches:
git branch
- To create a branch:
git branch new-feature-X
- To select a branch:
git checkout new-feature-X
- Make changes to code and commit changes the normal way:
git commit -a -m "some message"
- Can switch between branches with:
git checkout main
git checkout new-feature-X

Merging Branches

- To merge a branch to main:
 - First checkout the main:
git checkout main
 - Then merge the branch:
git merge new-feature-X
- May need to resolve conflicts

Collaboration

- Nominate one team member to host the main repo in GitHub
 - That person needs to invite the other teammates
 - In the GitHub external repo: click **Settings**, click **Manage Access**, click **Invite a collaborator**, enter email of teammates
 - Other teammates need to check email and accept invite
- Teammates now need to clone repo
 - In Cloud Shell, issue the following commands:
`mkdir ~/environment/shared-repos`
`cd ~/shared-repos`
`git clone <GIT_HUB_URL_TO_SHARED_EXTERNAL_REPO>`
`ls`

Collaboration (continued)

- Each teammate should now try to:
 - Create a branch in the new repo
 - Make a change to the branch
 - Merge the branch with the main branch
- If another team member has pushed changes and changed the repo, you need to pull them
 - `git pull`
 - This will pull down the changes from the remote repo

Collaboration (continued)

- If you have more time...
- Do the same thing for the repo for the internal service

Success!

- **Congratulations!** You have successfully used git for branching and collaboration