Git Workflow Basics
Setting Up Your Git Repo

// student repo link format:
// https://gitlab.cs.umd.edu/cmsc436labsprivate/cmsc436-{YourDirID}.git
// upstream repo link
// https://gitlab.cs.umd.edu/cmsc436labsprivate/cmsc436f23students
% git clone httpsLinkToStudentRepo (or sshLinkToStudentRepo)
% cd cmsc436-DirectoryID
% git remote add upstream
https://gitlab.cs.umd.edu/cmsc436labsprivate/cmsc436f23students
% git pull upstream main
Or if "fatal: refusing to merge unrelated histories"
% git pull upstream main --allow-unrelated-histories
% git push origin main
Repo Structure

Upstream
(Gitlab repo for instructors)

Origin
(Gitlab repo for student)
Create Your Personal Repo

upstream

origin

Your Computer (With Git)

clone git.gitlab@ XXXXXXXX
Make Instructor Repo the Source of Starter Files

`git remote add upstream repoLocation`
Pull Starter Files to Your Local Repo

```
git pull upstream main
```

- **upstream**
- **origin Repo on GitLab**
- **Your Computer (With Git)**
Starter Files Flow to Student Local Repo

[saiyr@Sais-Macbook cmsc436-aporter %] git pull upstream main
remote: Enumerating objects: 32, done.
remote: Counting objects: 100% (32/32), done.
remote: Compressing objects: 100% (25/25), done.
remote: Total 32 (delta 5), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (32/32), 61.35 KiB | 1.92 MiB/s, done.
From https://gitlab.cs.umd.edu/cmsc436labsprivate/cmsc436f23students
 * branch         main      -> FETCH_HEAD
 * [new branch]   main      -> upstream/main
fatal: refusing to merge unrelated histories
Push Your Updated Files to Your Gitlab Repo

Your Computer (Master)

origin Repo on GitLab

git push origin main

upstream

Push Your Updated Files to Your Gitlab Repo
Important: Merge Errors

● Possibility of encountering one at some point is high

● Don’t panic
  ○ Save a copy of your work elsewhere for safety

```bash
[saiyr@Sais-Macbook cmsc436-aporter % git pull upstream main --allow-unrelated-hi stories
From https://gitlab.cs.umd.edu/cmse436labsprivate/cmse436f23students
 * branch     main       -> FETCH_HEAD
Auto-merging README.md
CONFLICT (add/add): Merge conflict in README.md
Automatic merge failed; fix conflicts and then commit the result.
saiyr@Sais-Macbook cmse436-aporter %]
```
Important: Merge Errors

- Resolve the merge conflict in an IDE
Important: Merge Errors

● Contact the TA or Instructor

● Nuclear Option
  ○ Clone your repository again elsewhere
  ○ Make sure it’s up-to date with upstream
  ○ Copy over your work to this new directory and push
Important: Merge Errors

- Not a merge error

```
merge branch 'main' of https://gitlab.cs.umd.edu/cmse436labsprivate/cmse436f23st udents
# Please enter a commit message to explain why this merge is necessary,
# especially if it merges an updated upstream into a topic branch.
#
# Lines starting with '#' will be ignored, and an empty message aborts
# the commit.
```

```
"~/Downloads/CMSC436 Fall 2023/cmse436-aporter/.git/MERGE_MSG" 6L, 313B
```
Important: Merge Errors

- Not a merge error

```
Merge branch 'main' of https://gitlab.cs.umd.edu/cmsc436labsprivate/cmsc436f23stdudents
# Please enter a commit message to explain why this merge is necessary,
# especially if it merges an updated upstream into a topic branch.
#
# Lines starting with '#' will be ignored, and an empty message aborts
# the commit.
```

Important: Merge Errors

- Not a merge error
Important Git commands

git status → To see if there are any changes to commit
Important commands

`git status` → To see if there are any changes to commit

`git fetch upstream master` → See if there is anything to pull

`git pull upstream master` → Actually pulling from upstream
How to turn in your work!

All on your local dev environment

```bash
  git status

  git add filesToCommit // (Don’t use the --all flag!!)

  git commit -m "exercise 1 implementation"

  git push origin main
```

Check status of repo through browser