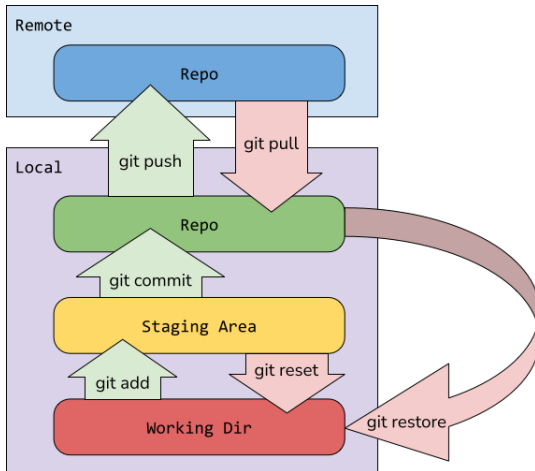


# CS 136L Lecture 8

## Version Control (git)

# Workflow Diagram



# Demo 1

```
mkdir new_dir
cd new_dir
git init
ls -la
git status
vi README
git add README
git status
git rm --cached README
#(or . or git restore --staged README or
    git reset HEAD README)
git status
git add README
git commit -m "Added README"
git status
```

## Demo 2

```
vi README
git add README
git commit -m "Added Authors"
vi README
git add README
git commit -m "Added Project Title"
git diff HEAD HEAD~2
git log
git log --oneline
git revert <insert_hex_code>
git checkout <insert_hex_code> <
    insert_file_name>
git add README
git commit -m "Reverting README"
```

## Demo 3

```
git branch program
git checkout program
vi main.c
git add main.c
git commit -m "adding main"
git checkout master
git merge master program
```

# Collaboration

- To collaborate on a project, *on a remote server* go to the repository containing it, and type `git init --bare`
- You can clone this remote repository, say, from the server `linux.student.cs.uwaterloo.ca`, to your computer:  
`git clone ssh://[userid]@linux.student.cs.uwaterloo.ca/[absolutepath]`
- After the commit, you can push the changes to the remote repo by typing `git push`
- We can download changes from the remote repo by typing `git pull`

## Cheat Sheet (for the Lab)

<code>git init</code>	Initialize a git repo
<code>git status</code>	see the status of files
<code>git log</code>	see previous actions
<code>git add</code>	add file(s) to a staging area
<code>git commit</code>	commit file(s)
<code>git branch</code>	list all branches
<code>git branch [branchname]</code>	creates a new branch
<code>git checkout [branchname]</code>	switch to a specified branch
<code>git merge [branchname]</code>	while in master, merges master with branchname
<code>git clone ssh//[server]/[addr]</code>	clone remote repo