Setting Up Git

- $ git config --global user.name "your name"
- $ git config --global user.email "your github email"
- $ git config --global core.editor "nano -w"

- $ git config --list

Creating a Repository

- cd ~/Desktop
- mkdir test
- cd test
- git init
- ls, ls –a, gui: ctrl+h to see .git
- git status; open gui
- Delete .git
- git status
- git init again

Tracking Changes
1 round

- nano autodo.sh
- pwd > ../list.txt, ctrl+o, ctrl+x
- ls >> ../list.txt, ctrl+o, ctrl+x
- Bash autodo.sh
- git status
- git add autodo.sh
- git status
- git commit –m "list files in test" autodo.sh
- git status

2 round

- nano autodo.sh
- Change to ls -a >> ../list.txt, ctrl+o, ctrl+x
- Bash autodo.sh
- git add autodo.sh
- git commit –m "list all files in test" autodo.sh
- git status (all white)
- git log

3 round

- nano autodo.sh
- Add ls .. >> ../list.txt, ctrl+o, ctrl+x
- Bash autodo.sh
- git add autodo.sh
- git commit atodo.sh (w/out –m)
- git status (all white)
- git log

4 round

- Change your .sh file to list all files on desktop instead of just visible files.

- Commit it

- git diff shar shar

Exploring History

- git checkout second version
- git branch
- Bash autodo.sh
- git checkout master
- git branch
- Bash autodo.sh

Ignoring Things

1 Why ignore things?

- – Data to ignore (always the same)
- – Do not want to git upload your data to github

2 how to

- Create folder data in test
- Nano .gitignore
- Data/
- Ctrl o enter ctrl x
- git add .gitignore
- git commit -m "ignore data" .gitignore

3 check ignored stufff

- Copy list to data
- Make a data2 folder
- Cd data git add
- Cd .. Cd data2
- Git add git commit –m "autodo outcome" list.txt
- Git log
- Git log list.txt

Remotes in GitHub

- **Get an account on github**
- Walk through

- **Push autodo**
- Create a new repo right upper
- Copy the url
- Git remote add origin url
- Git remote –v
- Git push origin master
- Master branch is linked to origin
- Check on github

- git remote remove origin (in case)

Update autodo

- nano autodo.sh
- Add echo "done"
- Git add/commit
- Git push origin master

- git remote remove origin (in case)

Collaborating

- Download stuff
- Go to desktop
- Git clone url
- Change file, git add, git commit
- Delete app folder
- Git status
- Git add sandbox git commit –m "don't need sand box" sandbox
- Git push origin master

Collaborating

- Fork

- fork use vt account
- User:zhuoaprilfu
- Edit one file and commit it
- Check it (need to to the repo click commit)
- Creat a pull request
- The owner can deny by close the resquest or merge that pull request

- Explore settings