

Getting Started with GIT

Notes

We are just getting going, and we are going to do it in small steps

what we will learn today will get you rolling, but is not sufficient for you to work professionally with GIT in a team

For the time being use the github GUI to see changes over time e.g. comparing changes to files over time

Configuring GIT

git config --list ○ we can find our current configuration

everyone on a computer ○ system

this user ○ global

current repository ○ local

in effect this allows us to have multiple identities e.g. work and public

git config --global user.name "My Extra Name" ○ configure global

git config --global user.name "My Own Name Here" ○ configure

ssh-keygen

copy the public component of the key to github ○ creating keys

create a ssh configuration and use your lab or your prod keys on github!!!

Host bitbucket.org
HostName bitbucket.org
IdentityFile ~/.ssh/id_git_rsa
Host gitlab.com
HostName gitlab.com
IdentityFile ~/.ssh/id_git_rsa
Host github.com
User git
HostName github.com
IdentityFile ~/.ssh/id_git_rsa ○ ssh keys

ssh config

Creating a repository

git init

[if you don't want in a public repository] >> .gitignore

echo "# repo-name" >> README.md

git status

git add .gitignore

git add README.md

git commit -m "created .gitignore and README.md"

lets step through this

for this example make sure you check the box that creates the README

Linking repository

SSH/HTTPS ○ You can use either https or ssh. I would suggest when possible stick to the ssh model

Cloning ○ git clone git@github.com:<repo>

Adding a remote repository ○ git remote add origin git@github.com:<repo>

git push -u origin master

this is situationally dependent

git pull/push ○ lets draw one i use - work/home

has anyone got a usage model they want to work through

Basic workflow

git status

git add/remove

git commit

git push