Git Workshop

Christian Vaughn

Agenda

- What is Git?
- Clone repository
- Add/Modify files
- View Diff
- Commit/Push/Pull
- Create branch
- Pull request & merge

What is Git

Git is an open-source version control system, that allows us to make collaboration easier. It keeps history of code changes, and lets multiple people work on the same codebase more seamlessly

Cloning Repositories

Set-up

Configure user

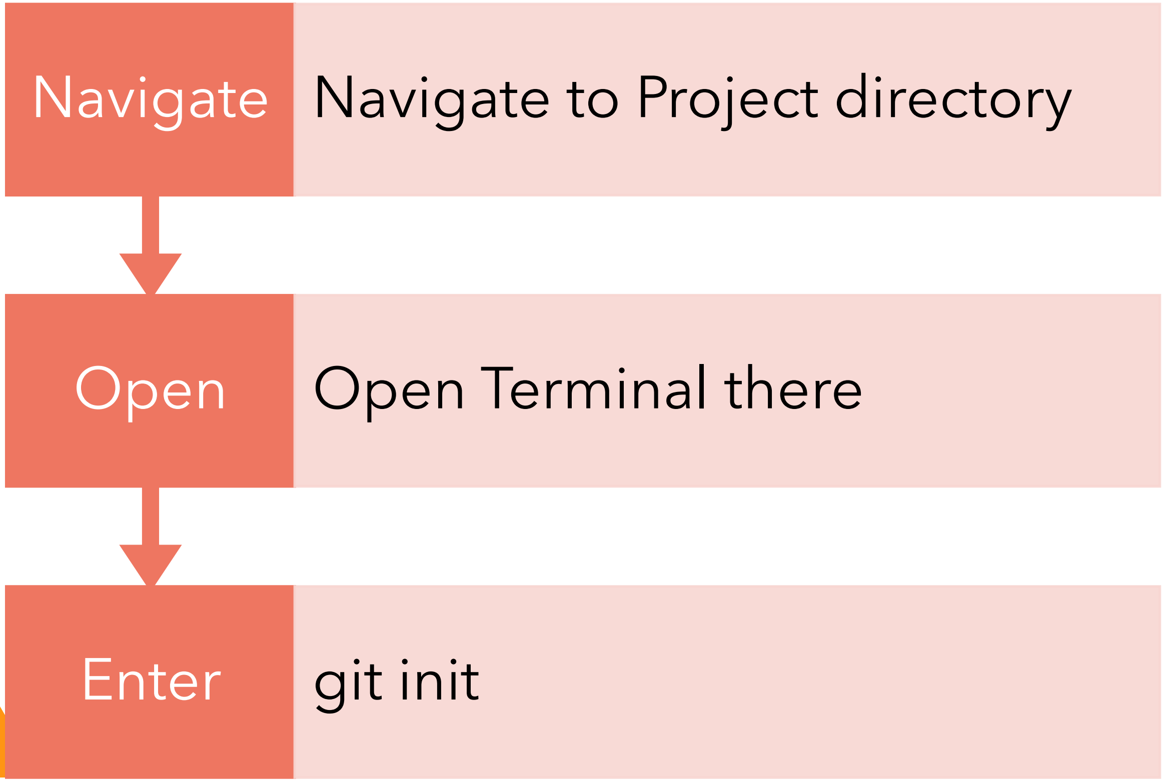
- On your computer, open the Git Shell application.
- Enter these lines (with appropriate changes):
  - git config --global user.name “John Smith”
  - git config –global user.email jsmith@example.com

Notes

- Only needs to be done once
- Password will be asked on commits/pushes and similar operations

Create/Clone Directory

Create

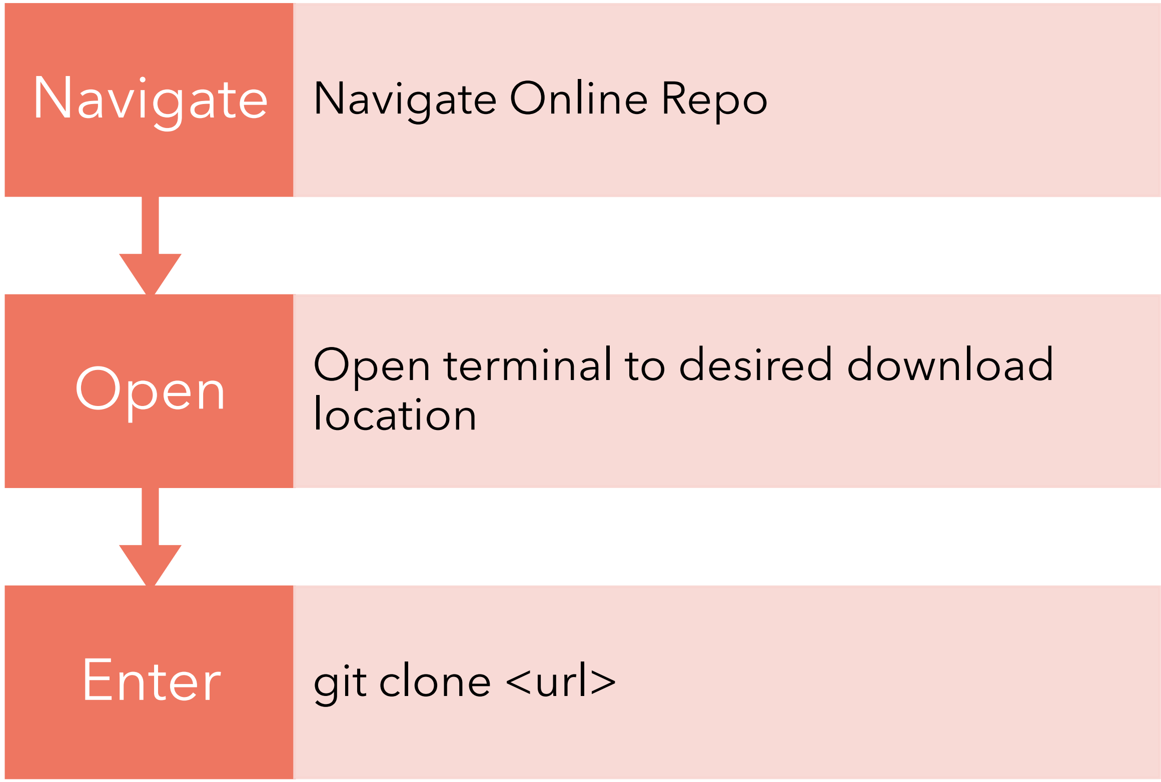


Navigate — Navigate to Project directory

Open — Open Terminal there

Enter — git init

Create



Navigate — Navigate Online Repo

Open — Open terminal to desired download location

Enter

git clone <url>

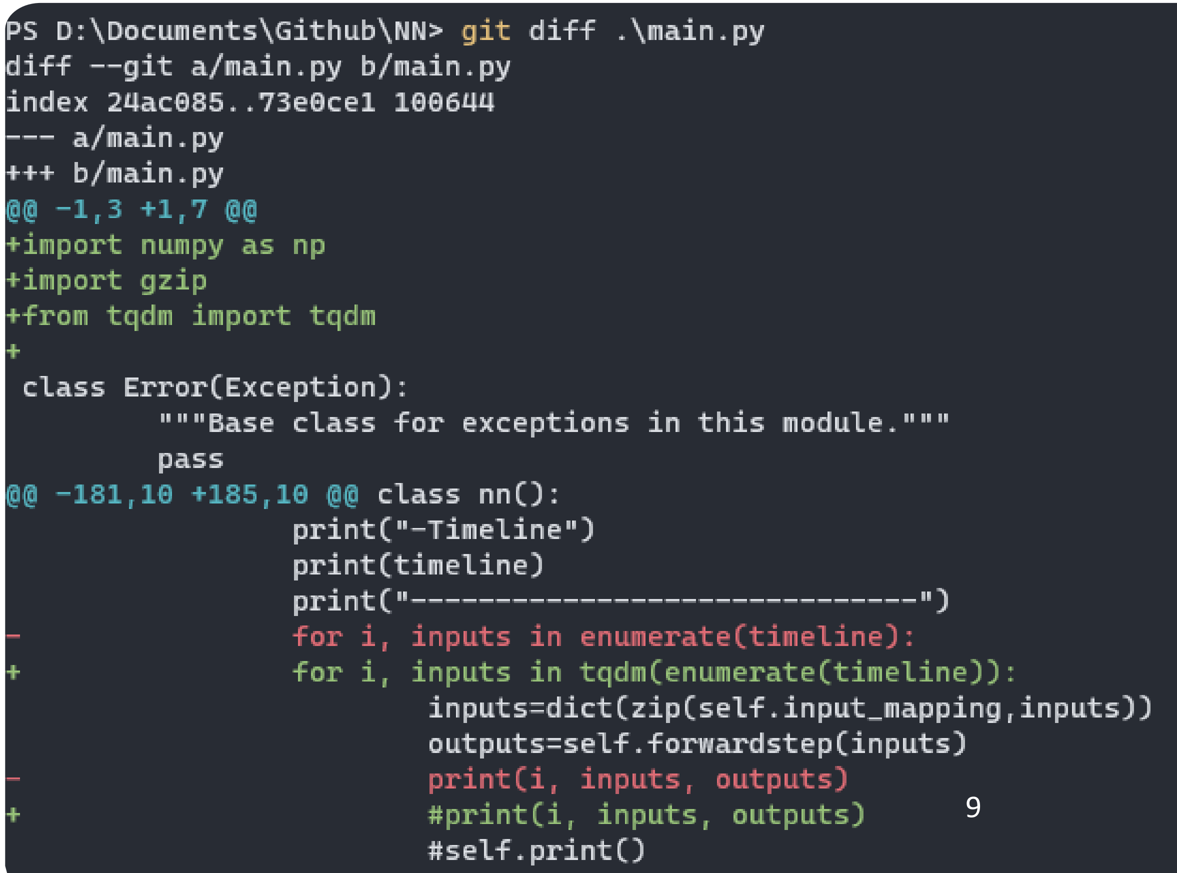
Saving Changes

git add

- Stages an edited file
  - Saying I want to upload this on next commit
- Allows for directories or files
- git add <file or dir>
- git status
  - Shows changes waiting to be comitted

git diff

- Shows what has changed between an edited file
- git diff <file>



```
PS D:\Documents\Github\NN> git diff .\main.py
diff --git a/main.py b/main.py
index 24ac085..73e0ce1 100644
--- a/main.py
+++ b/main.py
@@ -1,3 +1,7 @@
+import numpy as np
+import gzip
+from tqdm import tqdm
+
 class Error(Exception):
         """Base class for exceptions in this module."""
         pass
@@ -181,10 +185,10 @@ class nn():
                 print("-Timeline")
                 print(timeline)
                 print("-------------------------------")
-                for i, inputs in enumerate(timeline):
+                for i, inputs in tqdm(enumerate(timeline)):
                         inputs=dict(zip(self.input_mapping,inputs))
                         outputs=self.forwardstep(inputs)
-                        print(i, inputs, outputs)
+                        #print(i, inputs, outputs)
                         #self.print()
```

Commit & Syncing

Commit

- Saves a snapshot of project
- Adds it to the local repository
- Usage:
  - git commit
  - git commit -m "commit message"

Push

- Upload local repo to GitHub
- Usage
  - git push

Pull

- Download content from GitHub and update local repo
- Usage
  - git pull

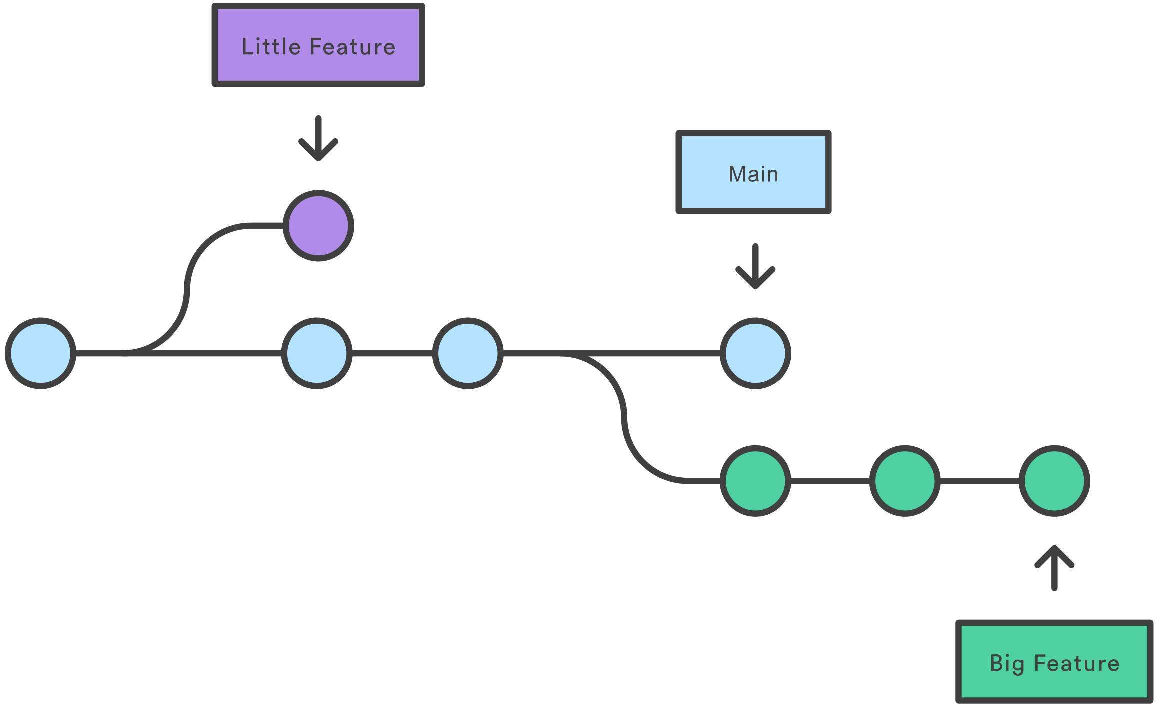
Branches

Branching

✓ Used when we want to add a new feature or fix bugs

Helps keep main branch clean

Can clean up code in branch before merging to main.



Commands

git branch

- lists all branches in repo

git branch <branch>

- create a new branch with name <branch>

git branch -d <branch>

- delete branch after merge

git checkout <branch>

- swich local repo to commit to <branch>

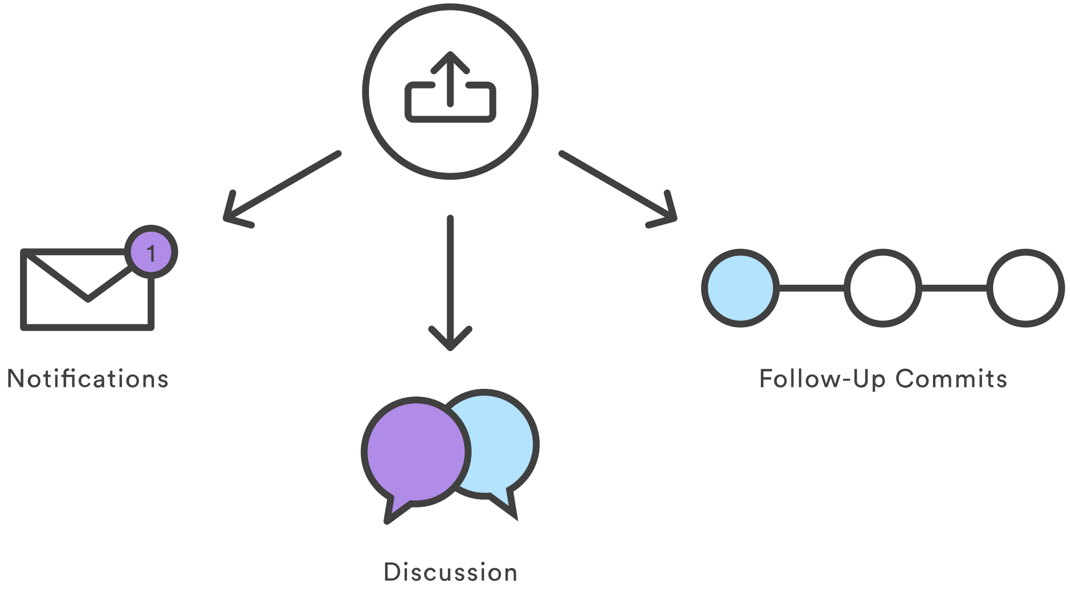
git checkout -b <new-branch>

- combines checkout and create branch

Pull Request
&
Merge

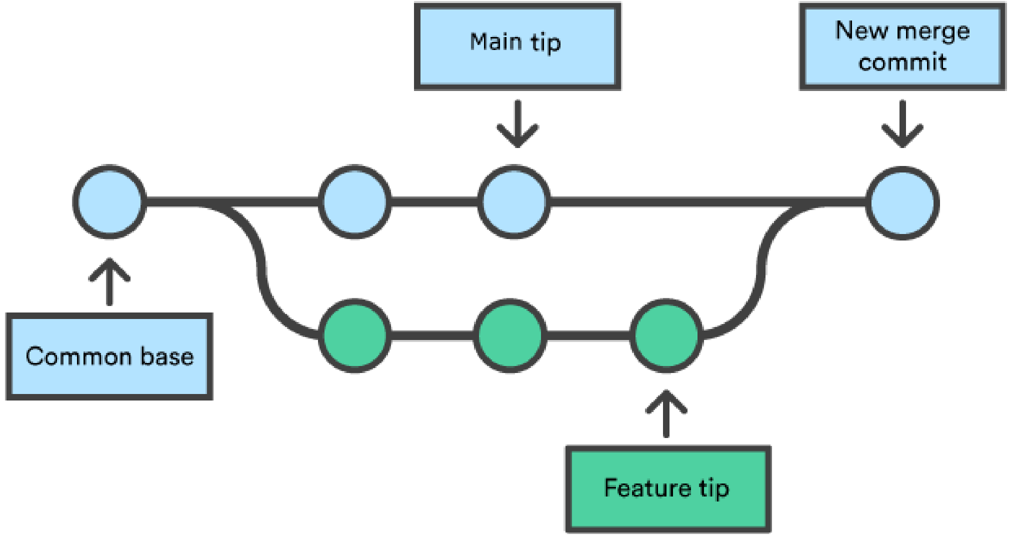
Pull Request

- Used to discuss changes before moving them into main branch
- Request a branch or fork to get merged into another



Merge

- Combine multiple commits
- commonly used to combine branches together
- Steps



Merge Steps

Make sure the receiving branch is checked out.

- git status

If not on right branch

- git checkout <branch>

Fetch latest remote commits

- git fetch
- git pull

Run the merge command

- git merge <branch>

Thank you

Christian Vaughn

https://www.christianvaughn.net