Creating Repositories and Collaborating with Git

Git branches

Creating a new feature:
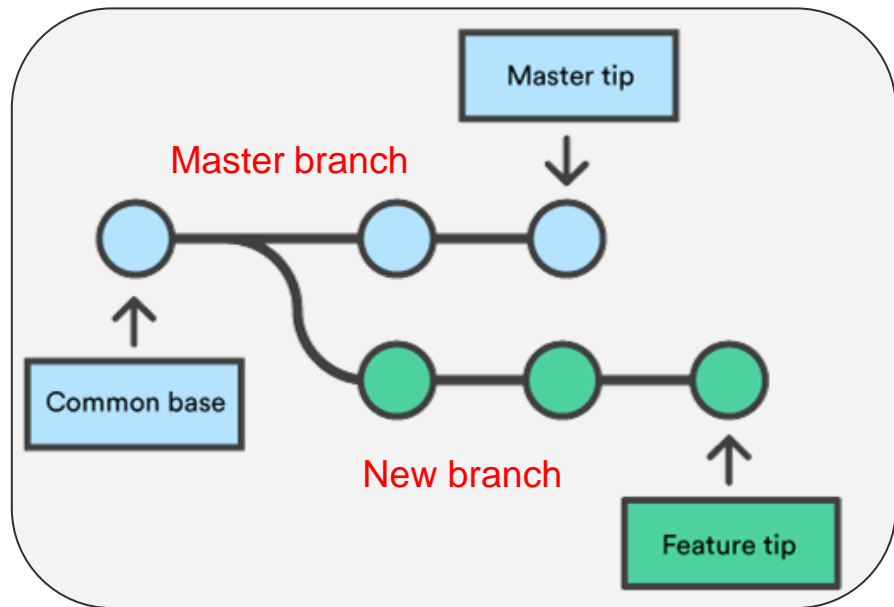
- Want to try out some new stuff.

Don't want to affect main project yet.

Create a branch!

Two ways:

- git branch <new_branch_name>

- git checkout –b <branch_name>

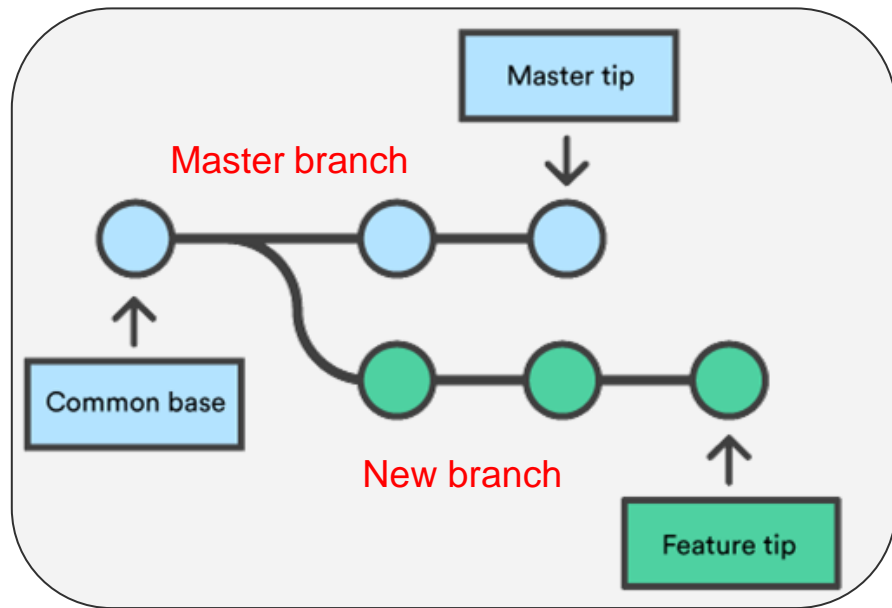# Creating branches

Creating a new feature:

- Want to try out some new stuff.

Don't want to affect main project yet.

Create a branch!

Two ways:

- git branch <new_branch_name> ————— Creates new branch

- git checkout –b <branch_name>

Creating branches
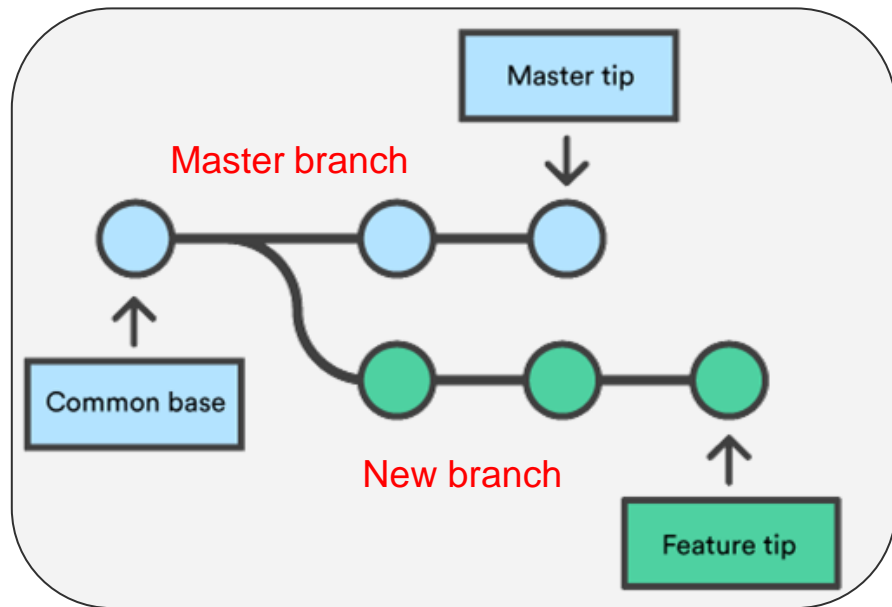
Creating a new feature:

- Want to try out some new stuff.

Don't want to affect main project yet.

Create a branch!

Two ways:

- git branch <new_branch_name>

- git checkout –b <branch_name>

Master branch

New branch

Creates new branch and switches to branch

Git branches

Branch has been created.

- git branch <new_branch_name>

Now let's see all the branches in repo:

- git branch

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git branch mess_with_stuff
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git branch
* master
  mess_with_stuff
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ |
```
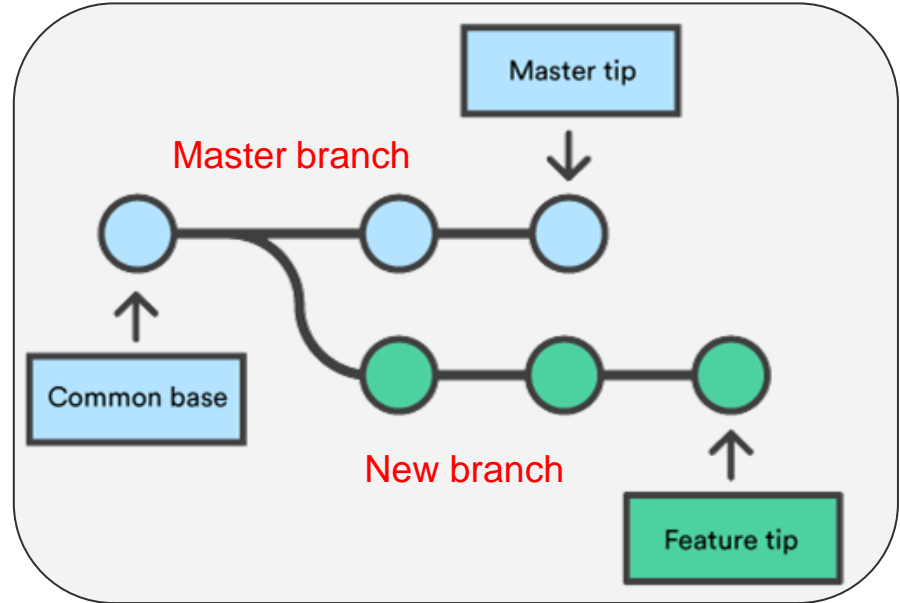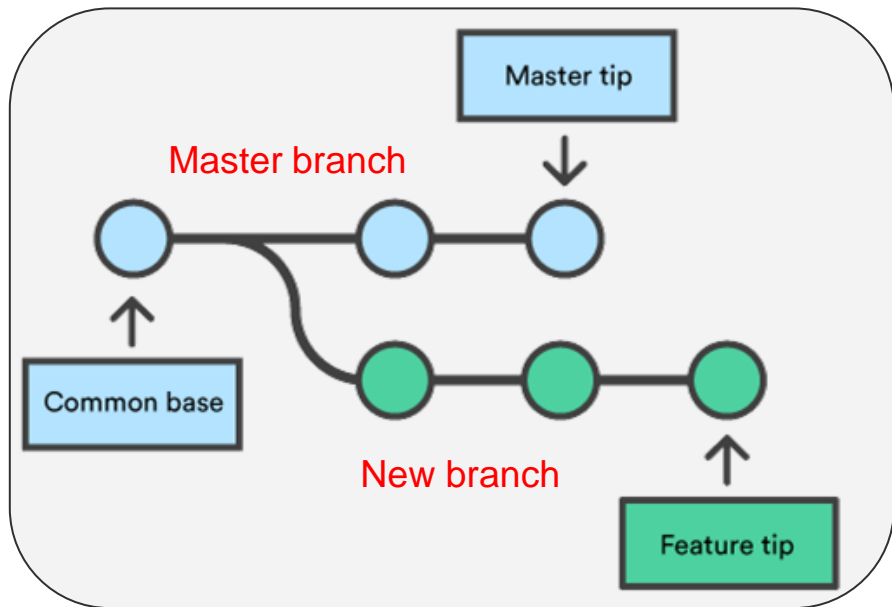
Switching branches (git checkout)

Branch has been created.

- git branch <new_branch_name>

Now switch to the new branch:

- git checkout <branch_name>

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git branch mess_with_stuff
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git branch
* master
  mess_with_stuff
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git checkout mess_with_stuff
Switched to branch 'mess_with_stuff'
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (mess_with_stuff)
$
```
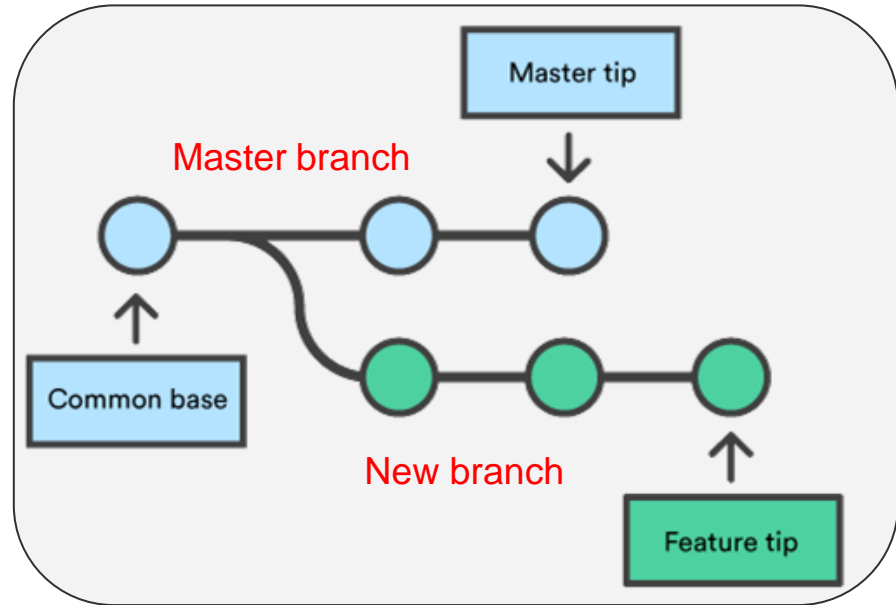
Git branches

Or can do this:

- git checkout -b <branch_name>

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git status
On branch master
nothing to commit, working tree clean
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git branch
* master
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git checkout -b mess_with_stuff
Switched to a new branch 'mess_with_stuff'
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (mess_with_stuff)
$ |
```

Git branches

- Make changes to goodfibonacci notebook
- Turn into a class object (will see objects later)

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (mess_with_stuff)
$ git status
On branch mess_with_stuff
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   good_fibonacci.ipynb
```

Fibonacci with Dynamic Programming

(a.k.a don't repeat work)

Fibonacci class

```python
class Fibonacci:

    def __init__(self, n):

        self.n = n
        self.result_dict = {0: 1, 1: 1}

    def compute_sequence(self):
        self.__fib(self.n)

    def __fib(self, n):
        if n == 0:
            return 0
        elif n == 1:
            return 1
        else:

            # if already computed, use value in lookup table
            if (n - 1) in self.result_dict.keys():
                fibn1 = self.result_dict[n-1]
            else:
                # otherwise, compute
                fibn1 = self.__fib(n-1)
            # if already computed, use value in lookup table
            if (n - 2) in self.result_dict.keys():
                fibn2 = self.result_dict[n-2]
            else:
                # otherwise, compute
                fibn2 = self.__fib(n-2)
```

Git branches

On new branch:
- Add change to staging area

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (mess_with_stuff)
$ git status
On branch mess_with_stuff
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   good_fibonacci.ipynb
```
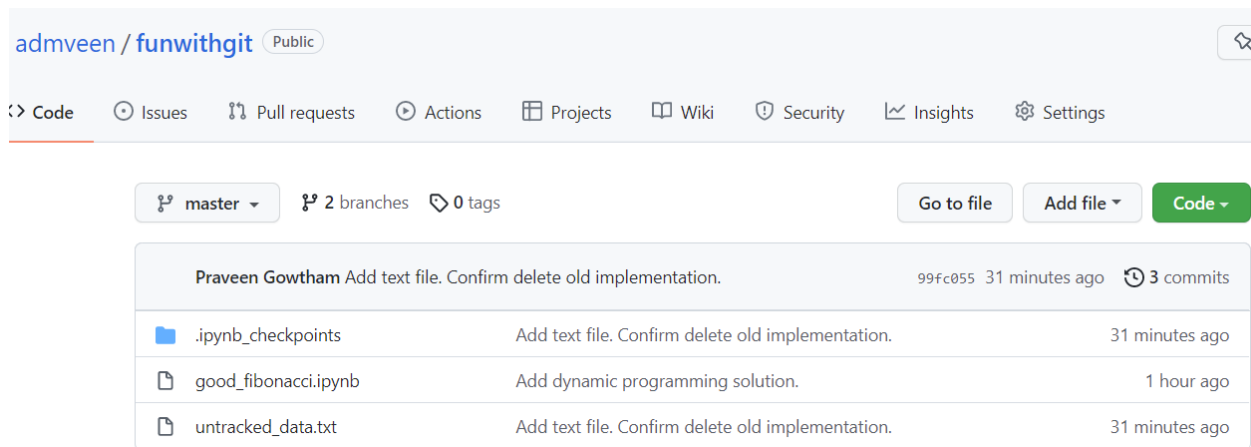
And commit.

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (mess_with_stuff)
$ git commit -m "Put dynamic programming solution into class object"
[mess_with_stuff c34afdb] Put dynamic programming solution into class object
 1 file changed, 75 insertions(+), 45 deletions(-)
```

Pushing from branch

On new branch:
- Add change to staging area

git push origin new_branch

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (mess_with_stuff)
$ git status
On branch mess_with_stuff
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   good_fibonacci.ipynb
```

And commit.

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (mess_with_stuff)
$ git commit -m "Put dynamic programming solution into class object"
[mess_with_stuff c34afdb] Put dynamic programming solution into class object
1 file changed, 75 insertions(+), 45 deletions(-)
```

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (mess_with_stuff)
$ git push origin mess_with_stuff
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 840 bytes | 420.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
remote:
remote: Create a pull request for 'mess_with_stuff' on GitHub by visiting:
remote:      https://github.com/admveen/funwithgit/pull/new/mess_with_stuff
remote:
To https://github.com/admveen/funwithgit.git
 * [new branch]      mess_with_stuff -> mess_with_stuff
```

Github before new branch

Before changes/commit on new branch

After pushing new branch

Github repo now looks like this!

On the master branch:

There's a new branch!

Will talk about compare/pull requests
later.

git merge

Say that:

- Didn't mess up anything else.
- Like new changes:

Want to combine changes in new branch into master workflow.

Our case: simple.
- Merge two local branches (mess_with_stuff into master)

Step 1: go into destination branch

git checkout <destination_branch>

Step 2: merge the source branch into destination

git merge <source_branch>

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (mess_with_stuff)
$ git checkout master
Switched to branch 'master'
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git merge mess_with_stuff
Updating 99fc055..c34afdb
Fast-forward
 good_fibonacci.ipynb | 120 +++++++++++++++++++++++++++++++++--------------------
 1 file changed, 75 insertions(+), 45 deletions(-)
```

Git branches

Bam! Our class object based solution is
now in master.

git branch –d (deleting branches)

Merge complete.

Don't want/need new branch
(mess_with_stuff)

git branch –d <branch_name>

Deletes branch if safe to do so (i.e obsolete branch).

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git branch -d mess_with_stuff
Deleted branch mess_with_stuff (was c34afdb).
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git branch
* master
(base)
```

git branch –d (deleting branches)

Merge complete.

Don't want/need new branch
(mess_with_stuff)

git branch –d <branch_name>

Deletes branch if safe to do so (i.e obsolete branch).

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git branch -d mess_with_stuff
Deleted branch mess_with_stuff (was c34afdb).
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git branch
* master
(base)
```

Pushes branch deletion onto remote github repo

git push remote_name :deleted_branch

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git push origin :mess_with_stuff
To https://github.com/admveen/funwithgit.git
 - [deleted]        mess_with_stuff
```

Git branches

Creating a new feature:

- Want to try out some new stuff.

Don't want to affect main project yet.

Create a branch!

Two ways:

- git branch <new_branch_name>

- git checkout –b <branch_name>

Master branch

New branch

Creates new branch and switches to branch

Restoring a previous state

Boss hates everything except first, old (crappy) implementation

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git log --graph
* commit c34afdb910aa96c7668253fd7525eb6796094d75 (HEAD -> master, origin/master)
| Author: Praveen Gowtham <praveen.gowtham@flatironschool.com>
| Date:   Wed Feb 23 20:08:38 2022 -0500
|
|     Put dynamic programming solution into class object
|
* commit 99fc0555ff0a98a221ca22f2fad45ca01c2cbb48
| Author: Praveen Gowtham <praveen.gowtham@flatironschool.com>
| Date:   Mon Feb 21 14:30:58 2022 -0500
|
|     Add text file. Confirm delete old implementation.
|
* commit 847e7582b0d7deb7bf0e3258256c2e88beb822cb
| Author: Praveen Gowtham <praveen.gowtham@flatironschool.com>
| Date:   Mon Feb 21 14:13:13 2022 -0500
|
|     Add dynamic programming solution.
|
* commit 3bc106469af06625930ff9fa6ac0762f1170fee3
  Author: Praveen Gowtham <praveen.gowtham@flatironschool.com>
  Date:   Mon Feb 21 12:53:04 2022 -0500

      Add naive implementation of Fibonacci
```

Git hash identifier for commit:
First 7 or so characters enough.

Restoring via new branch

Boss hates everything except first, old (crappy) implementation

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git log --graph
* commit c34afdb910aa96c7668253fd7525eb6796094d75 (HEAD -> master, origin/master)
| Author: Praveen Gowtham <praveen.gowtham@flatironschool.com>
| Date:   Wed Feb 23 20:08:38 2022 -0500
|
|     Put dynamic programming solution into class object
|
* commit 99fc0555ff0a98a221ca22f2fad45ca01c2cbb48
| Author: Praveen Gowtham <praveen.gowtham@flatironschool.com>
| Date:   Mon Feb 21 14:30:58 2022 -0500
|
|     Add text file. Confirm delete old implementation.
|
* commit 847e7582b0d7deb7bf0e3258256c2e88beb822cb
| Author: Praveen Gowtham <praveen.gowtham@flatironschool.com>
| Date:   Mon Feb 21 14:13:13 2022 -0500
|
|     Add dynamic programming solution.
|
* commit 3bc106469af06625930ff9fa6ac0762f1170fee3
  Author: Praveen Gowtham <praveen.gowtham@flatironschool.com>
  Date:   Mon Feb 21 12:53:04 2022 -0500
|
|     Add naive implementation of Fibonacci
```

Git hash identifier for commit:
First 7 or so characters enough.

Create new branch with contents of branch as referenced commit state

git checkout –b branchname commit_hash

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git checkout -b naive_old_state 3bc1064
Switched to a new branch 'naive_old_state'
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (naive_old_state)
$ ls
naivefibonacci.ipynb
```

Restoring via new branch on Github

Create new branch with contents of branch as referenced commit state

git checkout –b branchname commit_hash

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git checkout -b naive_old_state 3bc1064
Switched to a new branch 'naive_old_state'
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (naive_old_state)
$ ls
naivefibonacci.ipynb
```

After push:

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (naive_old_state)
$ git push origin naive_old_state
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'naive_old_state' on GitHub by visiting:
remote:          https://github.com/admveen/funwithgit/pull/new/naive_old_state
remote:
To https://github.com/admveen/funwithgit.git
 * [new branch]      naive_old_state -> naive_old_state
```

| ⑂ naive_old_state ▾ | ⑂ 3 branches | ⊘ 0 tags | | Go to file | Add file ▾ | Code ▾ |

This branch is 3 commits behind master.   ⇅ Contribute ▾

	Praveen Gowtham Add naive implementation of Fibonacci	3bc1064 2 days ago	⟳ 1 commit
🗋 naivefibonacci.ipynb	Add naive implementation of Fibonacci		2 days ago

git revert

This reverts a certain commit

Really nice: it logs the undoing as a commit.

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git log --graph
* commit c34afdb910aa96c7668253fd7525eb6796094d75 (HEAD -> master, origin/master)
| Author: Praveen Gowtham <praveen.gowtham@flatironschool.com>
| Date:   Wed Feb 23 20:08:38 2022 -0500
|
|     Put dynamic programming solution into class object
|
* commit 99fc0555ff0a98a221ca22f2fad45ca01c2cbb48
| Author: Praveen Gowtham <praveen.gowtham@flatironschool.com>
| Date:   Mon Feb 21 14:30:58 2022 -0500
|
|     Add text file. Confirm delete old implementation.
|
* commit 847e7582b0d7deb7bf0e3258256c2e88beb822cb
| Author: Praveen Gowtham <praveen.gowtham@flatironschool.com>
| Date:   Mon Feb 21 14:13:13 2022 -0500
|
|     Add dynamic programming solution.
|
* commit 3bc106469af06625930ff9fa6ac0762f1170fee3
| Author: Praveen Gowtham <praveen.gowtham@flatironschool.com>
| Date:   Mon Feb 21 12:53:04 2022 -0500
|
|     Add naive implementation of Fibonacci
```

Undo all changes made by this commit.

git revert commit_hash

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git revert 99fc0555
[master 810ca80] This reverts commit 99fc0555ff0a98a221ca22f2fad45ca01c2cbb48.
 3 files changed, 100 deletions(-)
 delete mode 100644 .ipynb_checkpoints/good_fibonacci-checkpoint.ipynb
 rename .ipynb_checkpoints/naivefibonacci-checkpoint.ipynb => naivefibonacci.ipynb (100%)
 delete mode 100644 untracked_data.txt
```

git revert

This reverts a certain commit

Really nice: it logs the undoing as a commit.

Text editor pops up with revert details for editing. :wq to exit when done.

```
Revert "Add text file. Confirm delete old implementation."

This reverts commit 99fc0555ff0a98a221ca22f2fad45ca01c2cbb48.

# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch master
# Changes to be committed:
#       deleted:    .ipynb_checkpoints/good_fibonacci-checkpoint.ipynb
#       renamed:    .ipynb_checkpoints/naivefibonacci-checkpoint.ipynb -> naivefibonacci.ipynb
#       deleted:    untracked_data.txt
#
```

git revert

This reverts a certain commit

Really nice: it logs the undoing as a commit.

- Text file gone.
- Naïve implementation recovered.
- Dynamic programming solution: class implementation still there.

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git log
commit 810ca80ef81d12b41621c1bdadb90fab1088fed8 (HEAD -> master)
Author: Praveen Gowtham <praveen.gowtham@flatironschool.com>
Date:   Thu Feb 24 06:17:04 2022 -0500

    This reverts commit 99fc0555ff0a98a221ca22f2fad45ca01c2cbb48.

commit c34afdb910aa96c7668253fd7525eb6796094d75 (origin/master)
Author: Praveen Gowtham <praveen.gowtham@flatironschool.com>
Date:   Wed Feb 23 20:08:38 2022 -0500

    Put dynamic programming solution into class object

commit 99fc0555ff0a98a221ca22f2fad45ca01c2cbb48
Author: Praveen Gowtham <praveen.gowtham@flatironschool.com>
Date:   Mon Feb 21 14:30:58 2022 -0500

    Add text file. Confirm delete old implementation.

commit 847e7582b0d7deb7bf0e3258256c2e88beb822cb
Author: Praveen Gowtham <praveen.gowtham@flatironschool.com>
Date:   Mon Feb 21 14:13:13 2022 -0500

    Add dynamic programming solution.

commit 3bc106469af06625930ff9fa6ac0762f1170fee3 (origin/naive_old_state, naive_old_state)
Author: Praveen Gowtham <praveen.gowtham@flatironschool.com>
Date:   Mon Feb 21 12:53:04 2022 -0500

    Add naive implementation of Fibonacci
```

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ ls
good_fibonacci.ipynb   naivefibonacci.ipynb
```

Pushing reversion

Before:

After:

One last way: git reset

git reset –hard <commit_hash>
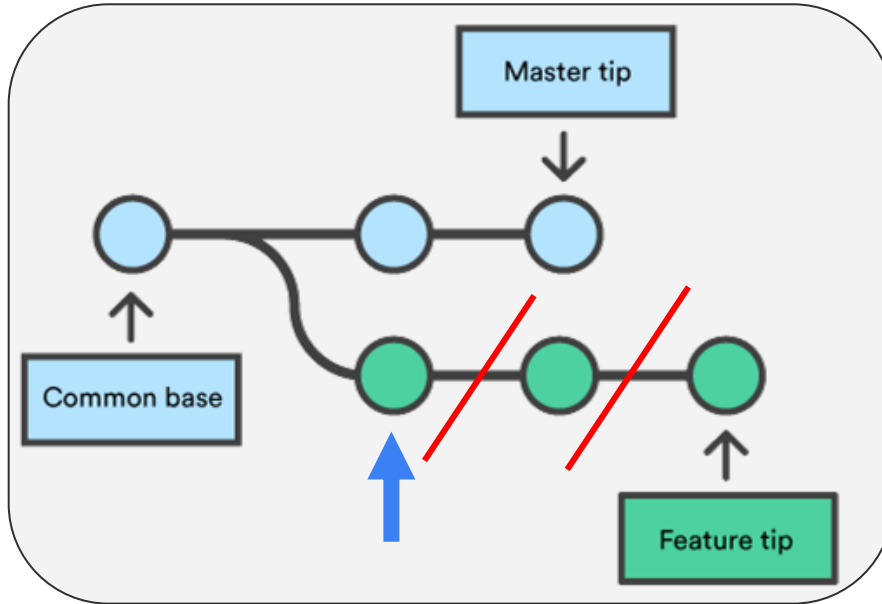
Destructive!!! Deletes files and rewrites history. Could be very bad news.

git fetch

So far: creating, merging and pushing local branches.

Sometimes want to create a branch that points to a branch in remote repository.

Case: Alfie has been working on one of the branches: naïve_old_state

- Says he's made some fun changes.

- I want to check it out (but don't want to pull directly)

git fetch

So far: creating, merging and pushing local branches.

Sometimes want to create a branch that points to a branch in remote repository.

Case: Alfie has been working on one of the branches: naïve_old_state

-   Says he's made some fun changes.

-   I want to check it out (but don't want to pull directly)

git fetch <remote_name> <remote_branch_name>

git fetch

So far: creating, merging and pushing local branches.

Sometimes want to create a branch that points to a branch in remote repository.

Case: Alfie has been working on one of the branches: naïve_old_state

- Says he's made some fun changes.

- I want to check it out (but don't want to pull directly)

git fetch <remote_name> <remote_branch_name>

git fetch origin naïve_old_state

git fetch

So far: creating, merging and pushing local branches.

Sometimes want to create a branch that points to a branch in remote repository.

Case: Alfie has been working on one of the branches: naïve_old_state

- Says he's made some fun changes.

- I want to check it out (but don't want to pull directly)

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git fetch origin naive_old_state
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 756 bytes | 63.00 KiB/s, done.
From https://github.com/admveen/funwithgit
 * branch            naive_old_state -> FETCH_HEAD
   3bc1064..5b6e3e2  naive_old_state -> origin/naive_old_state
```

Updated information we have on origin/naïve_old_state .

Wanna check out what Alfie did.

git fetch

So far: creating, merging and pushing local branches.

Sometimes want to create a branch that points to a branch in remote repository.

- Create local branch.
- Want local branch to contain current history of naïve_old_state

Case: Alfie has been working on one of the branches: naïve_old_state

- Says he's made some fun changes.

- I want to check it out (but don't want to pull directly)

```
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git checkout -b alfie_naive origin/naive_old_state
Switched to a new branch 'alfie_naive'
branch 'alfie_naive' set up to track 'origin/naive_old_state'.
```

Updated information we have on origin/naïve_old_state .

git fetch

So far: creating, merging and pushing local branches.

Sometimes want to create a branch that points to a branch in remote repository.

Case: Alfie has been working on one of the branches: naïve_old_state

- Says he's made some fun changes.

- I want to check it out (but don't want to pull directly)

- Create local branch.
- Want local branch to contain current history of naïve_old_state

```
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git checkout -b alfie_naive origin/naive_old_state
Switched to a new branch 'alfie_naive'
branch 'alfie_naive' set up to track 'origin/naive_old_state'.
```

Updated information we have on origin/naïve_old_state .

git fetch

So far: creating, merging and pushing local branches.

Sometimes want to create a branch that points to a branch in remote repository.

Case: Alfie has been working on one of the branches: naïve_old_state

- Says he's made some fun changes.

- I want to check it out (but don't want to pull directly)

Commit history? What did Alfie do?

git diff

Can check the files and actual changes
that were made

git diff old_branch new_branch

git diff naïve_old_state alfie_naive

What did Alfie do?

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (alfie_naive)
$ git diff naive_old_state alfie_naive
diff --git a/naivefibonacci.ipynb b/naivefibonacci.ipynb
index 9abd20c..d85adc8 100644
--- a/naivefibonacci.ipynb
+++ b/naivefibonacci.ipynb
@@ -9,7 +9,9 @@
     "\n",
     "Implement the Fibonacci sequence with recursion naively. Element sum of previous two elements.\n",
     "\n",
-    "0,1,1,2,3,5,8,13,21,34..."
+    "0,1,1,2,3,5,8,13,21,34...",
+    "\n",
+    "I'm Alfie and I make fun changes to things.\n"
     ]
    },
    {
```

git diff

Can check the files and actual changes that were made

git diff old_branch new_branch

git diff naïve_old_state alfie_naive

What did Alfie do?

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (alfie_naive)
$ git diff naive_old_state alfie_naive
diff --git a/naivefibonacci.ipynb b/naivefibonacci.ipynb
index 9abd20c..d85adc8 100644
--- a/naivefibonacci.ipynb
+++ b/naivefibonacci.ipynb
@@ -9,7 +9,9 @@
     "\n",
     "Implement the Fibonacci sequence with recursion naively. Element sum of previous two elements.\n",
     "\n",
-    "0,1,1,2,3,5,8,13,21,34..."
+    "0,1,1,2,3,5,8,13,21,34...",
+    "\n",
+    "I'm Alfie and I make fun changes to things.\n"
     ]
   },
   {
```

… Alfie … dropping silly markdown comments again.

Take it or leave it

We can accept the change by merging: Drop it by deleting alfie's branch!

naïve_old_state and alfie_naive

git merge naïve-old_state alfie_naive   git branch –d alfie_naive

 I decided to not incorporate his changes in my repo.

```
(base)
prave@LAPTOP-JA891BKF MINGW64 ~/Funwithgit (master)
$ git branch -d alfie_naive
warning: deleting branch 'alfie_naive' that has been merged to
        'refs/remotes/origin/naive_old_state', but not yet merged to HEAD.
Deleted branch alfie_naive (was 5b6e3e2).
(base)
```

Any Questions?