- `git pull origin main`
- `git pull

- git pull = git fetch + git merge

- git fetch - fetches code from remote
- git merge - merge remote repository to local repository

### how to create new branch?

```
git checkout -b <new-branch-here>
```

### how to switch between branches?

```
git checkout <branch-name>
```

### collaborative development

https://github.com/prashant5nov/starwarsAPI

```
bug-1111 :: fix exception handling  (Prashant)

feature-2222: add multi-threading support  (Rahul)
```

In [ ]:

```
# switch to main branch
git checkout main


# I will ensure which branch I am on
git branch


# ensure changes from remote are available in local copy
git pull origin main


# create a new branch from main branch
git checkout -b bug-1111
```

##### git branching strategy

```
master-branch / main-branch :: production ready code
```

##### environments in software development lifecycle (SDLC)

```
DEV (linux1) -> TEST (linux2)  -> UAT (linux3) -> PROD (linux4)
```

### merge conflict

```
This is a situation when 2 or more developers are trying to
change same portion of code.
```

developer-1

feature-1111

```
git checkout main
git pull origin main
git checkout -b feature-1111

# make changes to a certain file (lets call it file1.py)

git add file1.py
git commit -m "developer-1 changes are done."
git push origin feature-1111

# create pull request on github
# main <--- feature-1111
```

developer-2

feature-2222

```
git checkout main
git pull origin main
git checkout -b feature-2222

# make changes to a certain file (lets call it `file1.py`)

git add file1.py
git commit -m "developer-2 changes are done."
git push origin feature-2222

# create pull request on github
# main <--- feature-2222
```

how to produce merge conflict?

developer-1 will merge his changes in main branch (by clicking on merge PR button) developer-2 will go to his own pull request and he should see "merge conflicts".

how to resolve merge conflict?

```
git checkout feature-2222
git pull --no-ff origin main




# above step will add merge conflicts to conflicting file

# manually edit the file and remove merge conflict artifacts


git add <conflicting-file>
git commit -m "resolving conflicts"

git push origin feature-2222



# go to github and you should see conflicts are resolved now
# finally click on merge pull request
```