

how to get your set-up to “hack around”

Aman Goel

\$ why contribute?

advance career **feel-good factor** it's fun
improve software the world relies on

make a difference mentor people

get recognition

find mentors **improve skills** build experience

meet new people foster networks

give back to the community

\$ the git(Hub) model

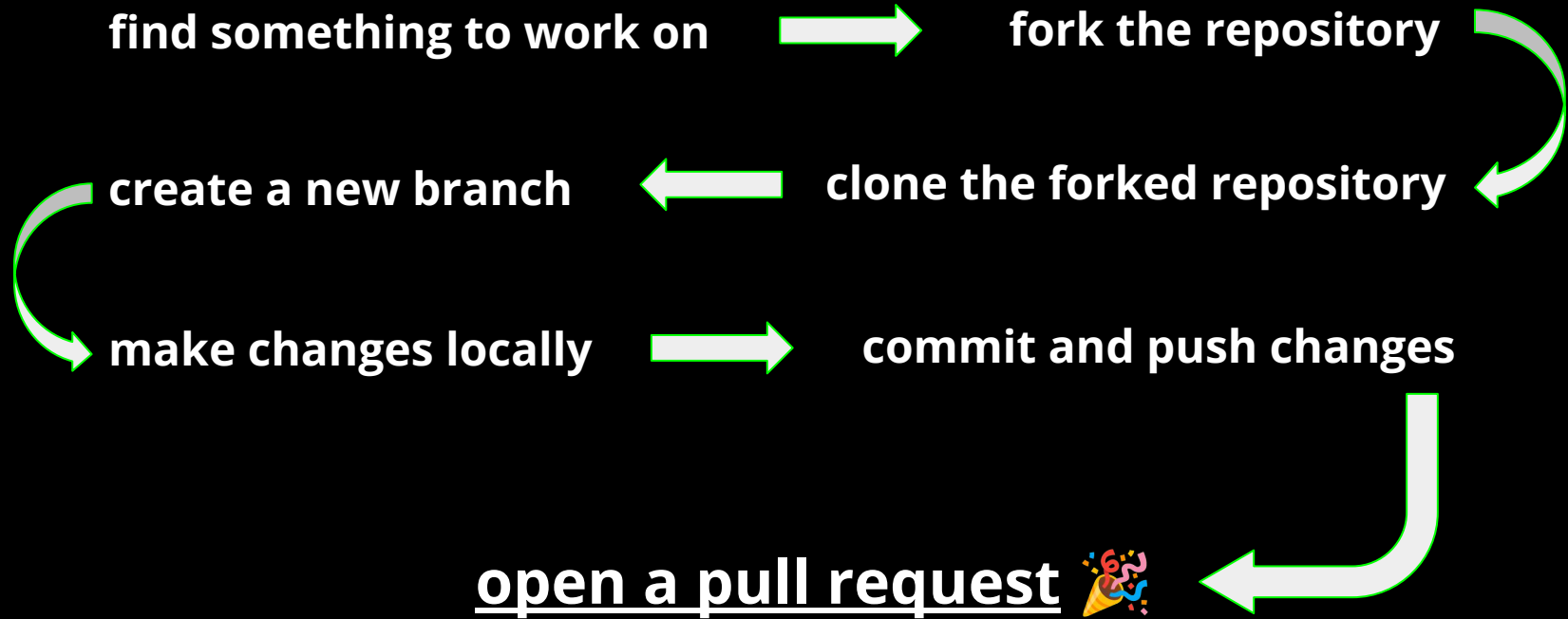
- git-scm.com Git for all platforms
- *fork* copy of a repository on GitHub owned by a different user
- `git clone [url]` retrieve the repository from the hosted location
- `git remote add [alias] [url]` add a git URL as an alias
- `git status` show modified files in working directory
- `git add [file]` add a file to the next commit
- `git reset [file]` unstage a file but retain changes
- `git commit -m "[message]"` commit the staged content as a new commit snapshot
- `git checkout -b [branch-name]` create a new branch and switch to it
- `git fetch [alias]` fetch down all the branches from that Git remote
- `git merge [alias]/[branch]` merge and update a remote branch into current branch
- `git push -u [alias] [branch]` transmit local branch commits to the remote repository branch and track it
- `git pull` fetch and merge any commits from the tracking remote branch

[click for \[cheatsheet\]](#)

\$ the orientation

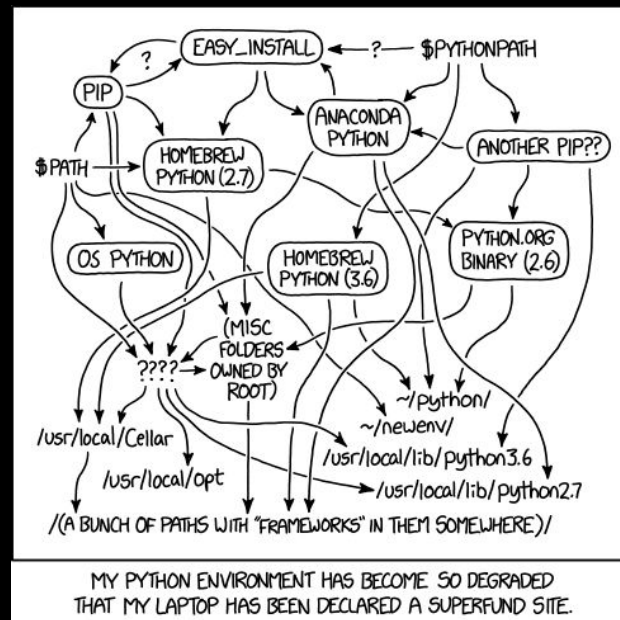
- read the `README.md`
- read the `CONTRIBUTORS.md`
- look for open issues
- look for the documentation
- reach out to the developers
- don't hesitate!

\$ how to start?



\$ the Python environment(s)

- **venv**, available by default from the standard *Python* distribution from *Python 3.3+*
- **virtualenv**, needs to be installed separately but can offer better performance and features
- **pipenv**, combines other features like locking with **venvs**
- **conda** which comes together with the Anaconda *Python* distribution



xkcd.com/1987

[click for \[primer\]](#)

```
$ pip install hackashop
```

```
python3 -m venv pyhep
```

```
source pyhep/bin/activate
```

```
pip install [package]
```

```
deactivate
```

\$ keep in mind

- setup according to the project guidelines
- do not `sudo`
- look out for `pre-commit`
- look out for task runners like `nox`
- write tests and run tests whenever possible
- have fun!

\$ what to hack on?

there are a lot of projects and packages to hack on at the hackashop:

- Common Partial Wave Analysis ([Issues dashboard](#))
- HSF Training Working Group modules ([Issues dashboard](#))
- Scikit-HEP packages ([Issues dashboard](#))
- Scikit-HEP affiliated packages

and more!

find the complete list here: [PyHEP 2022 Workshop - Hackashop](#)

\$ resource pool

- [How to Contribute to Open Source](#) by *Open Source Guides*
- [GIT CHEAT SHEET](#) by *GitHub*
- [Getting started with Git](#) by *GitHub*
- [Version Control with Git](#) by *The Carpentries*
- [Python Virtual Environments: A Primer](#) by *Martin Breuss*
- [Level Up Your Python](#) by *Henry Schreiner*
- [Markdown Cheat Sheet](#) by *The Markdown Guide*

hack away! 🚀

Aman Goel

 @mightaswellcode  @amangoel185

PyHEP 2022 Hackashop