Seth Galluzzi
Vzw6yk
Bonus Points Lab 2

And for those of you in the middle, where you sort of got things to work right out of the gate, but can't get the versioning working as just described above. You're done too if you want, but for an extra 1 point on the quizes you can google or look up in a book the following questions and turn in a short answer.

Where does pip install package contents when you execute `pip install` ?

The contents of packages installed from pip install go to the packages directory of the running python.

In that line we add to requirements.txt, the `ssh+` line, you can replace the `v0.0.0` with a git hash, that is to say, the long hash you see when you execute a `git log`. In fact, you could install any revision of your repo using a git hash. Why does that work? In other words, what are we doing when we create a github release that results in our being able to use a 'version' like `v0.0.0` ?

A github release is how you make specific versions of your content available to users. Different release versions can contain different content. Thus, understanding what release you are using will determine what version of the content you access.

What is the difference between a package and module in python? In other words, what is the purpose of the __init__.py file in your repo?

A python package is a directory with a collection of modules, a module is a python file itself.

What is the purpose of the 'setup.py' file in your repo, and more importantly, the version field in it.

The purpose of the setup.py is to include information about the repo. This includes versions, authors, licenses, and packages.
Since we all named our package 'shared', what do you think will happen if I run a requirements.txt file containing TWO of our `git+...` lines right after another?

In my estimation the computer would collect the information from both git repositories.