git repositories

Another SELFSCRUM Story

What do I have?

Pile of stuff
(aka workspace)

What do I want?

a proper archive
(local)

What do I do?

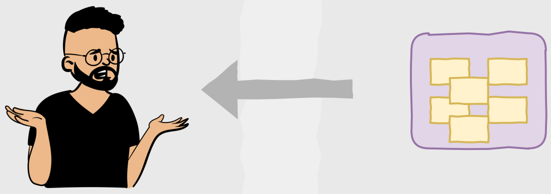
git init

call that command *in* your local working dir.

What happened?

Your local repo is created. It resides in the (invisible) **.git** folder. Git eyes your pile of stuff suspiciously ("untracked files", it mutters), but doesn't do anything else.

What do I have?



no ideas

What do I want?

steal some ideas from a remote repo

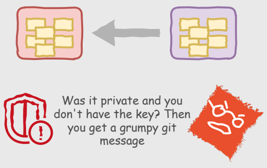
What do I do?

git clone remote-git-address

call that command where your local working dir shall reside. You will get the remote git address at the repo's web site (or from a friend). Normally, it starts with **https://**... or **git@**...

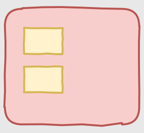
What happened?

Depends. Was it a public repo or do you have allowance? Then you have now a working dir with a local repo that contains exact the same stuff as the remote one.



Was it private and you don't have the key? Then you get a grumpy git message

What do I have?



my private local archive

What do I want?

share my project with friends so that we can work together

What do I do?

Now you have two things to do:

git remote add myproj remote-git-address

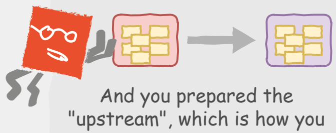
1. Tell your local repo where to connect to. You must have access to the remote git.

git push --set-upstream myproj master

2. Tell your local repo how your content is to be brought into the remote repo, which on your machine is now known as "myproj"

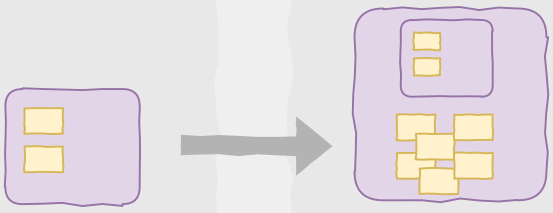
What happened?

Congrats, you just created your first repo in the net and connected your own repo.



And you prepared the "upstream", which is how you push new content from you into the remote repo in the future. It is now connected to your and the remote master branch. Of course, there is a downstream too. It was created automatically

What do I have?



my private remote repo

What do I want?

merge my repo into an existing group repo

What do I do?

You have 3 options:

1. Forget it.

2. If you don't really need the history, just checkout the both repos and copy the personal files into the group repo.

3. If you are a real git geek meanwhile, do the following:

git clone remote-groupproj-address

This clones the group project to your disk. Do all further operations within that project.

git remote add -f myproj remote-git-address

Ask git to fetch your own project from the remote repo and add it to the group project to a "special branch"

git checkout myproj/main

Change to that special branch. It is special because it is not yet connected to your group repo.

git switch -c newbranch

Now it is! After that, you can do whatever cleanup you need. E.g. you could move all content into an myproj folder to make sure it doesn't interfere with the group project. In Git Bash, type:

```
mkdir myproj
ls -I myproj | xargs -I '{}' mv '{}' myproj
```

which will move all content into a new myproj folder. When you are finished, stage and commit all changes, with

```
git add .
git commit -m "cleaned up my mess"
```

Now return to the group project's original state.

git checkout main

All that is left to do is to bring the new branch back home!

git merge newbranch --allow-unrelated-histories

Which we will do with this command. The --allow... option is to enable git to forget that these were two projects once.

git push

What happened?

This is a trap. You were lured into this course because someone told you learning git is an easy undertaking!

But relax - this use case here is rare and shows you a tiny glimpse of the power that git has... We are going back to normal!

What do I normally have?

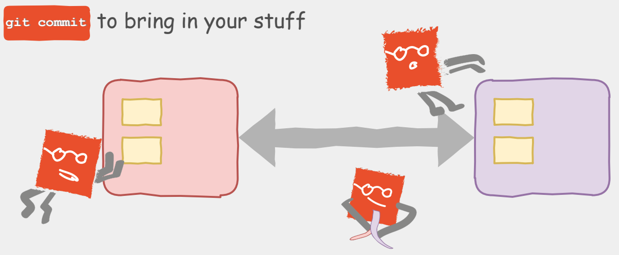
a local and a remote repo, securely linked and mostly synchronized

What do I normally want?

`git init` to start your repo

`git commit` to bring in your stuff

`git pull` to get changes that others made.



`git push` to deploy your own changes

`git merge` to combine your and the others' changes

What do I do next?

Install a Git Bash and start experimenting.

Go get a Github account.

Check out what's going on there, then checkout a project of your choice.

Find others to work on a common project. Anyone can do!

Enjoy Building!

What do you think?

Liked our poster? Found a bug?
New Ideas?
Let's have a chat!