

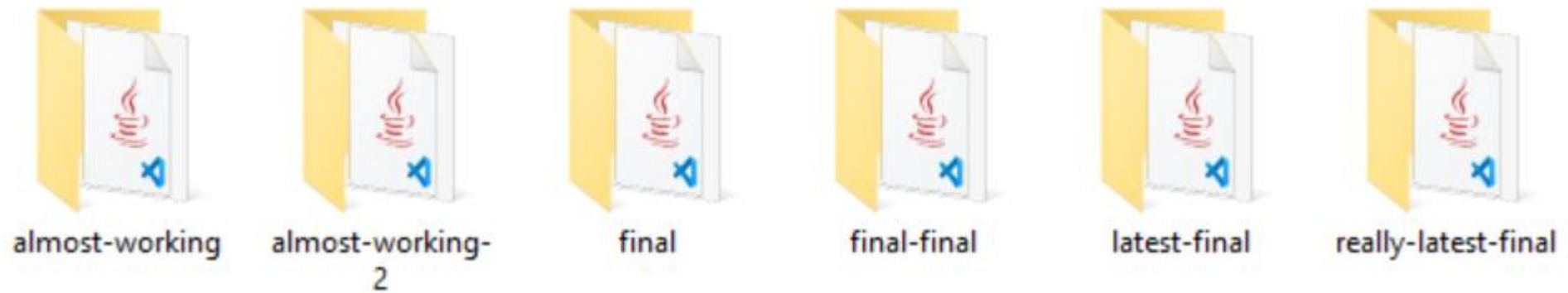


Git

Petar Ivanov, SAP
September, 2019

PUBLIC

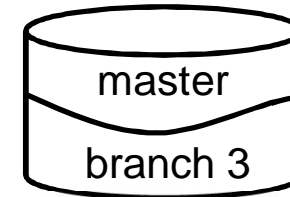
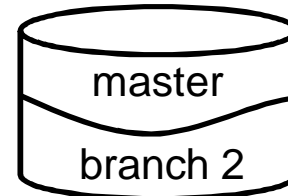
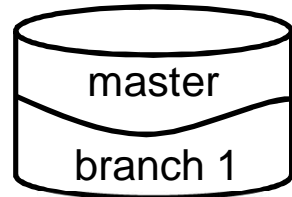
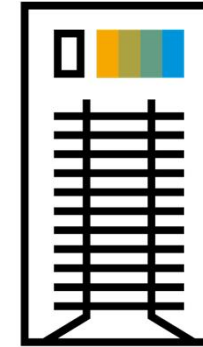
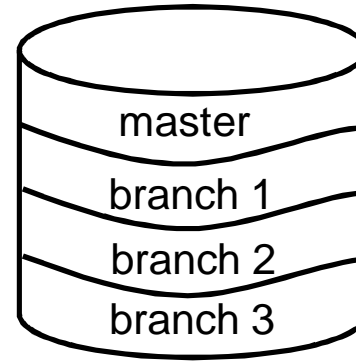
Where to store our code?



Git

Decentralized

remote –
origin



Git

Configurations

§ System (all users)

§ Global (current user)

§ Local (per repository)

```
git config --list (--system | --global | --local)
```

```
git config --global user.name "John Doe"
```

```
git config --global user.email "john.doe@example.com"
```

Git

Credentials

Using HTTPS – git will ask you for your credentials from time to time

- § Cache mode – storing credentials into memory for a certain period of time
- § Store mode – storing credentials as plain text on disk
- § Git Credential Manager for Windows (installed separately, uses Windows Credential Store)
- § osxkeychain mode (also secure)


Using SSH

- § With or without passphrase
- § Requires additional setup

Git

Creating uninitialized repo in GitHub

Repository name

☐ Initialize this repository with a README

This will let you immediately clone the repository to your computer

Add .gitignore: None ▼

Add a license: None ▼



Create repository

git remote -vv

Performing git init locally

...or create a new repository on the

```
echo "# repo-1" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/
git push -u origin master
```

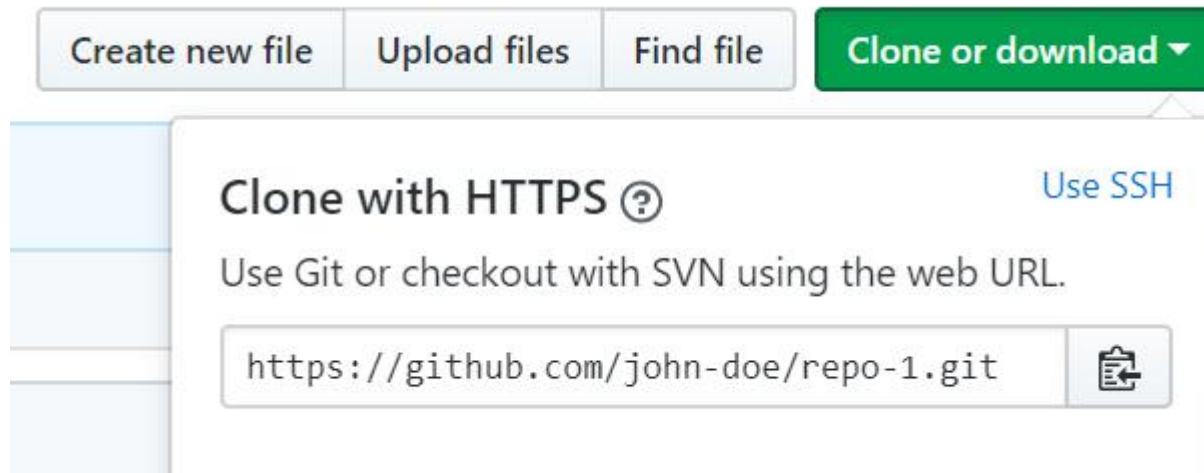
...or push an existing repository from

```
git remote add origin https://github.com/
git push -u origin master
```

Git

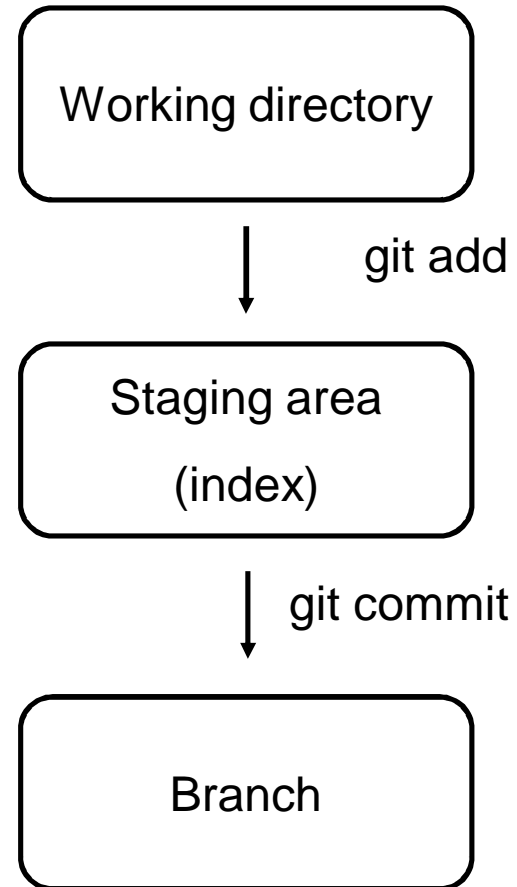
Cloning existing repository

No git init necessary,
Just git clone



Git

File states



Git

Making changes

§ git branch

§ git branch -vv

§ git log

§ git log --online

§ git status

§ git commit -m ""

§ git commit --amend

- Adding to staging area (git add)
- Removing from staging area (git reset HEAD)
- Discarding changes in a file (git checkout --)

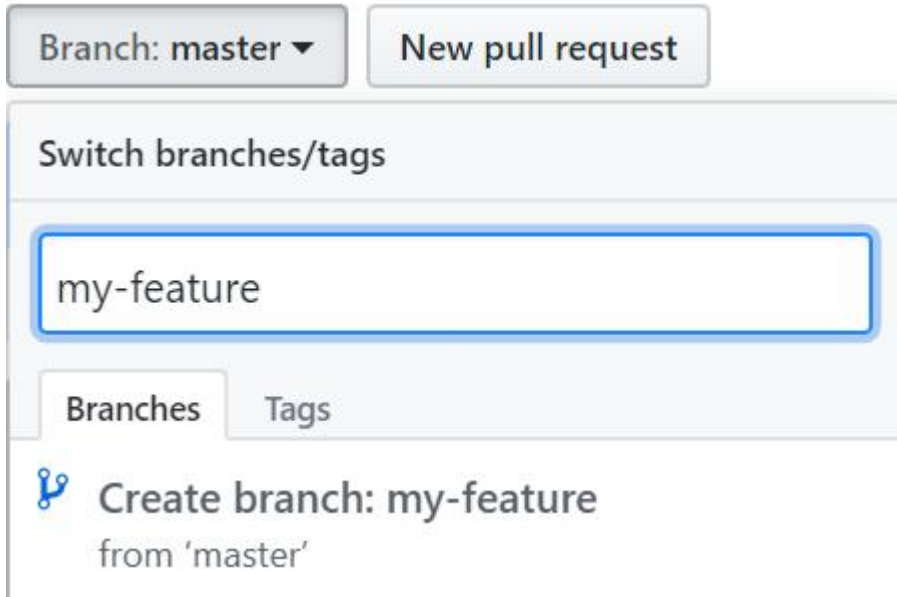
§ git revert

- Creating new files (untracked files)
- Modifying existing files
git diff
- Renaming files
- Deleting files

Git

Branching (multiple users)

Creating branch in GitHub:



Branch: master ▼ New pull request

Switch branches/tags

my-feature

Branches Tags

Create branch: my-feature
from 'master'

§ git branch

§ git pull (git fetch)

§ git checkout

§ git push

Creating branch locally:

§ git checkout -b <branch-name>

§ git push -u origin <branch-name>

Pull requests

git merge

Conflicts

Git

.gitignore