Versionning - Git - Exercises

I - Installation & configuration

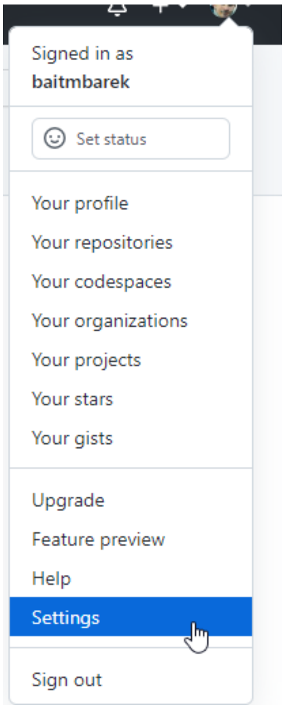
- Download and install a Git CLI client (https://git-scm.com/downloads)
- Configure Git:

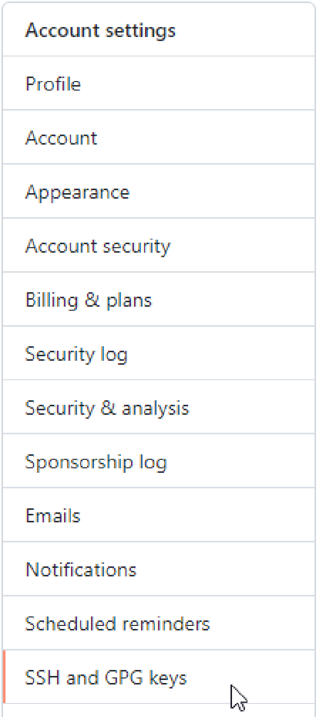
```
> git config --global user.name “Firstname Lastname”

> git config --global user.email “your.email@domain.com”
```

- Create a Github account
- Create a SSH key (RSA) from your terminal:

```
ssh-keygen -t rsa -b 4096 -C your@email.com
```

- Add your RSA public key to your account:
    - Go to your account’s Settings
    - Click the SSH and GPG keys submenu
    - Add your public key (id_rsa.pub content)





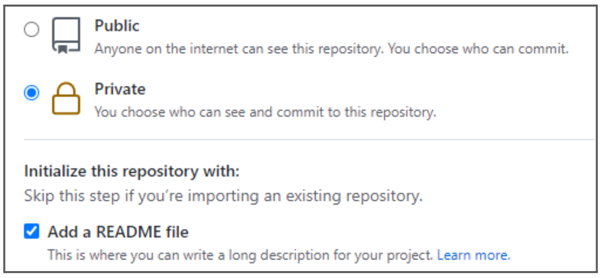
- Test your authentication from the terminal

```
ssh -T git@github.com
```

II - Creating repositories

A - Cloning an existing repository

- Go to https://github.com/new and create a **Private** repository with a Readme file.



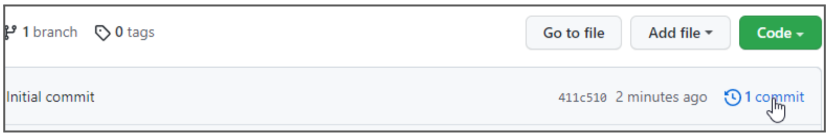
○ Public
Anyone on the internet can see this repository. You choose who can commit.

● Private
You choose who can see and commit to this repository.

Initialize this repository with:
Skip this step if you're importing an existing repository.

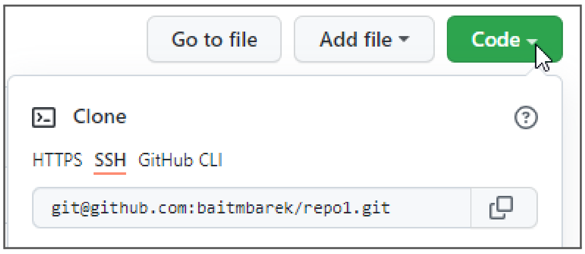
☑ Add a README file
This is where you can write a long description for your project. Learn more.

- A new repository will be created, containing a markdown file.
- Click on the commit(s) link and note that a first commit was done on your behalf. Feel free to explore the UI.



- Clone your repository on your computer using your CLI. Example:

```
mkdir ~/devops-data/ && cd ~/devops-data/

git clone git@github.com:baitmbarek/repo1.git
```



```
git@github.com:baitmbarek/repo1.git
```

- Make changes to the README.md file like below:

```
echo “### Work in progress” > README.md && echo “- Learning to use
```

```
Git” >> README.md
```

- Stage your changes while viewing the difference. Press ‘y’ if you agree to stage each listed change:

```
git add -p .
```

- Commit and push your changes then refresh your browser’s page.

```
git commit -m “Updated README.md”

git push origin master
```

B - Initializing a new repository

- Go to https://github.com/new and create a **Private** repository. Do not add files while creating the Repo.
- A new repository will be created, and no commit exists yet.
- Create a new directory on your computer:

```
mkdir ~/devops-data/repo2 && cd ~/devops-data/repo2
```

- Ask git to initialize a repository at this location:

```
git init
```

- Show the directory’s structure and note a new hidden directory was created. This directory will hold the repository’s metadata.

```
ls -al
```

- Add a remote (named origin by default). This will update the .git/config file by adding an entry.

```
git remote add origin git@github.com:baitmbarek/repo2.git
```

- Create a .gitignore file. This file will contain the directories / files patterns which should not be tracked by git.

```
echo “tmp/**” > .gitignore
```

- Create another text file containing anything you want.
- Create a tmp directory with other files in it.
- Show your workspace’s status to view your files’ state:

```
git status
```

- Note that your files in tmp/ are not tracked by git.
- Stage your new files and commit them.

```
git add .

git commit -m “Some free commit message”
```

- Show your workspace’s status again and refresh your browser’s page (in your repository).
- Send your changes to Github and refresh your page again.

```
git push origin master
```

III - Hands On

https://github.com/eficode-academy/git-katas

- Clone the repository
- Follow the Suggested Learning Path
- For each exercise, you’ll need to open the specific directory in your terminal and type `source setup.sh`. This will prepare a local git repository for your work.

IV - Test your knowledge

https://www.w3schools.com/git/exercise.asp

V - Fixing History

In this scenario, we will suppose that you or one of your teammates have pushed some secret in your repository by mistake. Unfortunately you cannot change the secret yet (even if you have to !) because it would impact a lot of other applications.

- Create a new repository and a development branch.
- Commit and push a file containing a secret.
- Merge your development branch into the master branch and push it.
- Add some changes to the master branch
- Create a new branch from your master branch and add some data (new files)
- Fix your repository:
    - Make sure you keep all your features
    - Make sure you cannot access the secret’s value from your history

VI - Quizz

1. Git is an alias for GitHub
    a. True
    b. False
2. What is the command to add all files and changes of the current folder to the staging environment of the Git repository?
    a. git add --all
    b. git add
    c. git add --files
3. What is the command to initialize Git on the current repository?
    a. start git
    b. git start
    c. initialize git
    d. git init
4. Git automatically adds new files to the repository and starts tracking them.
    a. True
    b. False
5. What is the command to commit with the message "New email":
    a. git commit -m "New email"
    b. git commit -log "New email"
    c. git commit message "New email"
    d. git commit -mess "New email"
6. What is the command to view the history of commits for the repository?
    a. git log
    b. git commits
    c. git history
    d. git --full-log
7. What is the command to create a new branch named "new-email"?
    a. git branch new "new-email"
    b. git add branch "new-email"
    c. git newBranch "new-email"
    d. git branch new-email

8. What is the command to move to the branch named "new-email"?
    a. git branch new-email
    b. git checkout branch new-email
    c. git checkout new-email
    d. git branch -move new-email
9. What is the option, when moving to a branch, to create the branch it if it does not exist?
    a. -newbranch
    b. -all
    c. -b
    d. -new
10. What is the command to push the current repository to the remote origin?
    a. git merge remote
    b. git remote push
    c. git remote commit
    d. git push origin
11. What is the command to show the differences between the current branch and the branch "new-email"?
    a. git changes new-email
    b. git diff new-email
    c. git log new-email
    d. git status new-email
12. Git Pull is a combination of:
    a. branch and checkout
    b. fetch and merge
    c. add and commit
13. Which two configuration properties does the tool expect to be configured after installing Git and prior to issuing the first commit?
    a. email address and password
    b. username and password
    c. username and IP address
    d. username and email address
14. After initializing a new Git repository and creating a file named git_file.html, which of the following commands will not work if issued?
    a. git add git_file.html
    b. git commit -m "Your commit message"
    c. git status
    d. git add .

15. What file is used to instruct Git to ignore certain files?
    a. ignore.git
    b. gitignore.txt
    c. .gitignore
    d. git.ignore
16. How should you save the current state of your code into Git?
    a. By adding all changes and staging them with git add
    b. By adding all changes and staging them with git stage
    c. By committing the staged changes with git commit
    d. By creating a new commit with git init
17. What should you run to modify your last commit?
    a. git commit --amend
    b. git change
    c. git commit amend
    d. git commit --change
18. Which option is used to stash your untracked files?
    a. git stash -a
    b. git stash -u
    c. git stash -q
    d. git stash -k