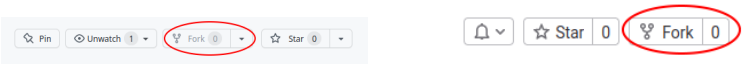
1 Exercise 1: collaborate on a project

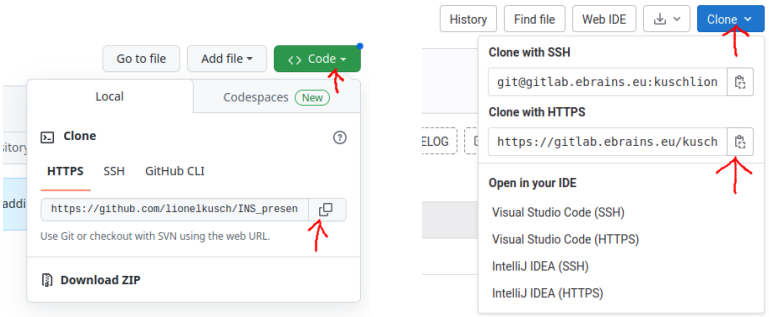
1.1 Create a fork

Click on the right button:



1.2 Initialize a repository

Get the url of your project:



Clone your repository with **git clone ...url...**

1.3 Create a branch

git checkout -b ...name...branch....
List branches : **git branch**

1.4 Create a commit

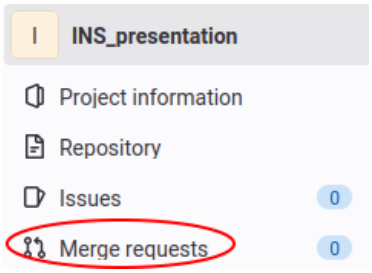
Add your name to the file 'partipants.txt'.
Index your modification : **git add participants.txt**
Commit your modification and add the description: **git commit**

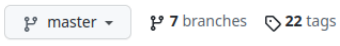
1.5 Push your commit

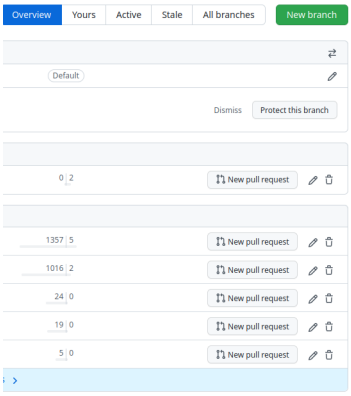
git push -u origin ...name...branch....

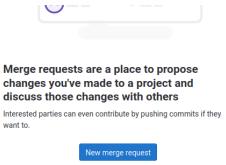
1.6 Create a pull request

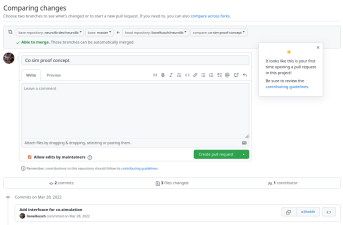
Create your pull request:











2 Exercise 2: Practice

2.1 Prepare your environment

Create 2 folders: *history* and *src*.
Create 3 text files: *README.txt*, *Mycode.txt* and *test.txt*

2.2 Create and Initialize a repository from existing files

1. Initialize a git repository
   git
2. Add the 3 text files
   git
3. Commit the 3 text files
   git
4. Write in the *README.txt* : 'I am on master branch'
5. Take a screenshot of the status and save it in *history*: *status_1.png*
   git
6. Index the modification of *README.txt*
   git
7. Take a screenshot of the status and save it in *history*: *status_2.png*
   git
8. Commit the modification of *README.txt*
   git
9. Take a screenshot of the status and save it in *history*: *status_3.png*
   git
10. Take a screenshot of git history and save it in *history* : *master_1.png*
    git

2.3 Work with Branches

1. Create a Branch : **branch_test**
   git
2. Take a screenshot of the list of branches and save it in *history*: *master_2.png*
   git
3. Modify *README.txt* : 'I am on the branch branch_test'
4. Index and commit the modification of *README.txt*
   git
   git
5. Take a screenshot of git history and save it in *history*: *master_3.png*
   git
6. Merge **master** branch with **branch_test**
   git
   git
7. Take a screenshot of git history and save it in *history*: *master_4.png*
   git

2.4 Managing conflicts

1. Modify *README.txt* on the branch **branch_test**: 'I am on the branch branch_test merge with master'
   git
2. Index and commit the modification of the *README.txt*
   git
   git
3. Go on the branch **master**
   git
4. Check if the text in *README.txt* is : 'I am on the branch branch_test'
5. Modify *README.txt* on the branch **master**: 'I am on the branch master'
6. Index and commit the modification of *README.txt*
   git
   git
7. Go on the branch **branch_test**
   git
8. Check if the text in *README.txt* is : 'I am on the branch branch_test merge with master'

9. Take a screenshot of git history and save it in *history*: *merge_1.png*
   git
10. Merge the **branch_test** with **master**
    git
11. Take a screenshot of the status and save it in *history*: *merge_2.png*
    git
12. Solve the conflict
    git
    git
13. Take a screenshot of git history and save it in *history*: *merge_3.png*
    git

2.5 Management of files

1. Go on the branch **branch_test**
   git
2. Move the files *Mycode.txt* and *test.txt* to *src*
   git
3. Commit these modifications
   git
4. Remove the file *src/test.txt*
   git
5. Commit this modification
   git
6. Take a screenshot of the folders: *folders_1.png*
7. Go on the branch **master**
   git
8. Take a screenshot of the folders: *folders_2.png*
9. Check if the files are at their previous location and that *test.txt* exists
10. Merge the branch **master** with **branch_test**
    git
    git

3 Exercise 3: Create a git repository for your project

1. Create a new project on github or gitlab
2. Clone your repository
3. Commit the files of your project
4. Push your first commit
5. Ready to work

4 Solution

4.1 Exercise 2.2: Create and Initialize a repository from existing filess

1. Initialize a git repository
   git init

2. Add the 3 text files
   git add *README.txt*, *Mycode.txt* *test.txt*

3. Commit the 3 text files git commit *README.txt* *Mycode.txt* *test.txt*

4. Write in the *README.txt* : 'I am on master branch'

5. Take a screenshot of the status and save it in *history*: *status_1.png*
   git status

```
kusch@INS-Precision-7540:~/Documents/project/github/test$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   README.txt

no changes added to commit (use "git add" and/or "git commit -a")
```

6. Index the modification of *README.txt*
   git add README.txt

7. Take a screenshot of the status and save it in *history*: *status_2.png*
   git status

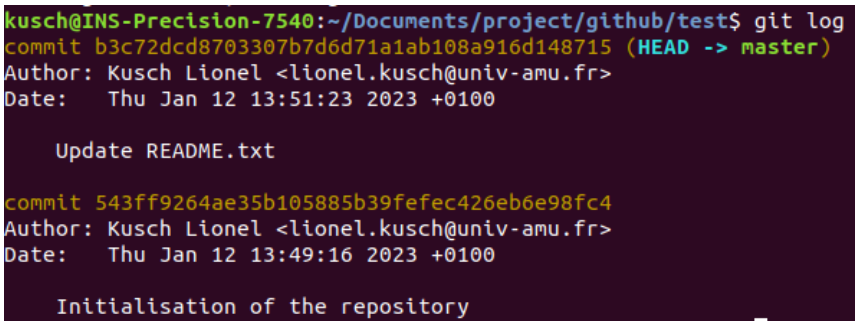
```
kusch@INS-Precision-7540:~/Documents/project/github/test$ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        modified:   README.txt
```

8. Commit the modification of *README.txt*
   git commit *README.txt*

9. Take a screenshot of the status and save it in *history*: *status_3.png*
   git status

```
kusch@INS-Precision-7540:~/Documents/project/github/test$ git status
On branch master
nothing to commit, working tree clean
```

10. Take a screenshot of git history and save it in *history*: *master_1.png*
    git log


kusch@INS-Precision-7540:~/Documents/project/github/test$ git log
commit b3c72dcd8703307b7d6d71a1ab108a916d148715 (HEAD -> master)
Author: Kusch Lionel <lionel.kusch@univ-amu.fr>
Date: Thu Jan 12 13:51:23 2023 +0100
Update README.txt
commit 543ff9264ae35b105885b39fefec426eb6e98fc4
Author: Kusch Lionel <lionel.kusch@univ-amu.fr>
Date: Thu Jan 12 13:49:16 2023 +0100
Initialisation of the repository

4.2 Exercise 2.3: Work on Branches

1. Create a Branch : **branch_test**
   git checkout -b branch_test

2. Take a screenshot of the list of branches and save it in *history*: *master_2.png*
   git branch

```
kusch@INS-Precision-7540:~/Documents/project/github/test$ git branch
* branch_test
  master
```

3. Modify the *README.txt* : 'I am on the branch branch_test'

4. Index and commit the modification of *README.txt*
   git add *README.txt*
   git commit *README.txt*

5. Take a screenshot of git history and save it in *history*: *master_3.png*
   git log

```
commit 94ab0694146fb122b432f7d42b8914aa779958af (HEAD -> branch_test)
Author: Kusch Lionel <lionel.kusch@univ-amu.fr>
Date:   Thu Jan 12 13:55:09 2023 +0100

    Update README on branche test

commit b3c72dcd8703307b7d6d71a1ab108a916d148715 (master)
Author: Kusch Lionel <lionel.kusch@univ-amu.fr>
Date:   Thu Jan 12 13:51:23 2023 +0100

    Update README.txt

commit 543ff9264ae35b105885b39fefec426eb6e98fc4
Author: Kusch Lionel <lionel.kusch@univ-amu.fr>
Date:   Thu Jan 12 13:49:16 2023 +0100

    Initialisation of the repository
```

6. Merge **master** branch with **branch_test**
   git checkout master git merge branch_test

7. Take a screenshot of git history and save it in *history*: *master_4.png*
   git log

```
kusch@INS-Precision-7540:~/Documents/project/github/test$ git log
commit 94ab0694146fb122b432f7d42b8914aa779958af (HEAD -> master, branch_test)
Author: Kusch Lionel <lionel.kusch@univ-amu.fr>
Date:   Thu Jan 12 13:55:09 2023 +0100

    Update README on branche test

commit b3c72dcd8703307b7d6d71a1ab108a916d148715
Author: Kusch Lionel <lionel.kusch@univ-amu.fr>
Date:   Thu Jan 12 13:51:23 2023 +0100

    Update README.txt

commit 543ff9264ae35b105885b39fefec426eb6e98fc4
Author: Kusch Lionel <lionel.kusch@univ-amu.fr>
Date:   Thu Jan 12 13:49:16 2023 +0100

    Initialisation of the repository
```

4.3 Exercise 2.4: Managing conflicts

1. Modify *README.txt* on the branch **branch_test**: 'I am on the branch branch_test merge with master'
   git checkout branch_test

2. Index and commit the modification of the *README.txt*
   git add *README.txt*
   git commit *README.txt*

3. Go on the branch **master**
   git checkout master

4. Check if the text in README.txt is: 'I am on the branch branch_test'

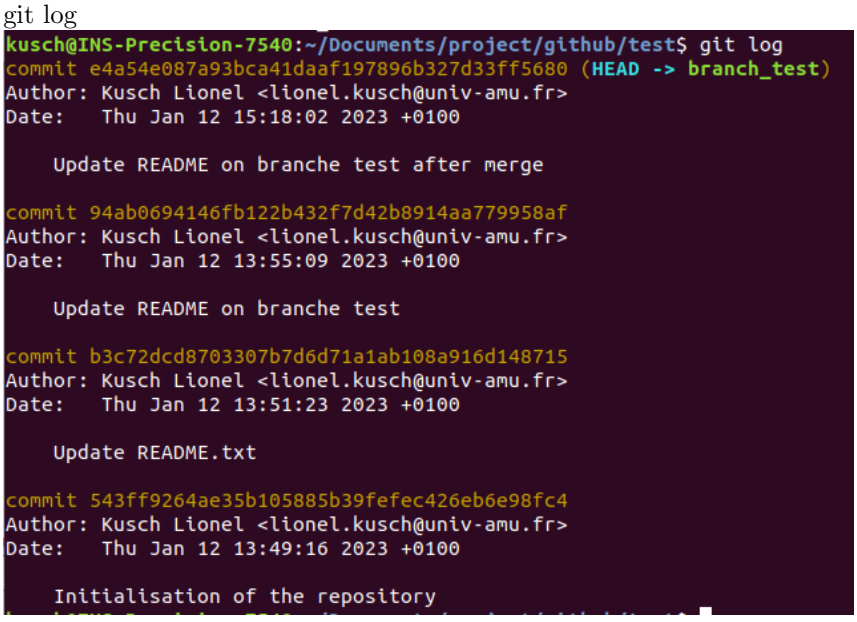
5. Modify the *README.txt* on the branch **master**: 'I am on the branch master'

6. Index and commit the modification of the *README.txt*
   git add *README.txt*
   git commit *README.txt*

7. Go on the branch **branch_test**
   git checkout branch_test

8. Check if the text in README.txt is: 'I am on the branch branch_test merge with master'

9. Take a screenshot of git history and save it in the *history*: *merge_1.png*
   git log



10. Merge the **branch_test** with **master**
    git merge master

11. Take a screenshot of the status and save it in *history*: *merge_2.png*
    git status

```
kusch@INS-Precision-7540:~/Documents/project/github/test$ git merge master
Auto-merging README.txt
CONFLICT (content): Merge conflict in README.txt
Automatic merge failed; fix conflicts and then commit the result.
kusch@INS-Precision-7540:~/Documents/project/github/test$ git status
On branch branch_test
You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)

Unmerged paths:
  (use "git add <file>..." to mark resolution)

        both modified:   README.txt

no changes added to commit (use "git add" and/or "git commit -a")
```

12. Solve the conflict
    git add README.txt
    git commit

13. Take a screenshot of git history and save it in *history*: *merge_3.png*
    git log

```
kusch@INS-Precision-7540:~/Documents/project/github/test$ git log
commit e9f3c9fd08ca7e3c92ac93189e40fb5cc6d805b6 (HEAD -> branch_test)
Merge: e4a54e0 8269219
Author: Kusch Lionel <lionel.kusch@univ-amu.fr>
Date:   Thu Jan 12 15:22:25 2023 +0100

    Merge branch 'master' into branch_test

commit 8269219a2f053261dd1cb59f9017507caa7f243b (master)
Author: Kusch Lionel <lionel.kusch@univ-amu.fr>
Date:   Thu Jan 12 15:18:34 2023 +0100

    Update README after merge

commit e4a54e087a93bca41daaf197896b327d33ff5680
Author: Kusch Lionel <lionel.kusch@univ-amu.fr>
Date:   Thu Jan 12 15:18:02 2023 +0100

    Update README on branche test after merge

commit 94ab0694146fb122b432f7d42b8914aa779958af
Author: Kusch Lionel <lionel.kusch@univ-amu.fr>
Date:   Thu Jan 12 13:55:09 2023 +0100

    Update README on branche test

commit b3c72dcd8703307b7d6d71a1ab108a916d148715
Author: Kusch Lionel <lionel.kusch@univ-amu.fr>
Date:   Thu Jan 12 13:51:23 2023 +0100

    Update README.txt

commit 543ff9264ae35b105885b39fefec426eb6e98fc4
Author: Kusch Lionel <lionel.kusch@univ-amu.fr>
Date:   Thu Jan 12 13:49:16 2023 +0100

    Initialisation of the repository
```

4.4 Management of files

1. Go on the branch **branch_test**
   git checkout branch_test

2. Move the files *Mycode.txt* and *test.txt* to *src*
   git mv *Mycode.txt test.txt src/*

3. Commit these modifications
   git commit

4. Remove the file *src/test.txt*
   git rm *src/test.txt*

5. Commit this modification
   git commit

6. Take a screenshot of the folders: *folders_1.png*

```
kusch@INS-Precision-7540:~/Documents/project/github/test$ ls -lR
.:
total 12
drwxrwxr-x 2 kusch kusch 4096 janv. 12 13:48 history
-rw-rw-r-- 1 kusch kusch   49 janv. 12 14:01 README.txt
drwxrwxr-x 2 kusch kusch 4096 janv. 12 14:04 src

./history:
total 0

./src:
total 0
-rw-rw-r-- 1 kusch kusch 0 janv. 12 13:48 Mycode.txt
```

7. Go on the branch **master**
   git checkout master

8. Take a screenshot of the folders: *folders_2.png*

```
kusch@INS-Precision-7540:~/Documents/project/github/test$ ls -lR
.:
total 8
drwxrwxr-x 2 kusch kusch 4096 janv. 12 13:48 history
-rw-rw-r-- 1 kusch kusch    0 janv. 12 14:05 Mycode.txt
-rw-rw-r-- 1 kusch kusch   26 janv. 12 14:05 README.txt
-rw-rw-r-- 1 kusch kusch    0 janv. 12 14:05 test.txt

./history:
total 0
```

9. Check if the file are at their previous location and *test.txt* exist

10. Merge the branch **branch_test** with master
    git checkout branch_test
    git merge master