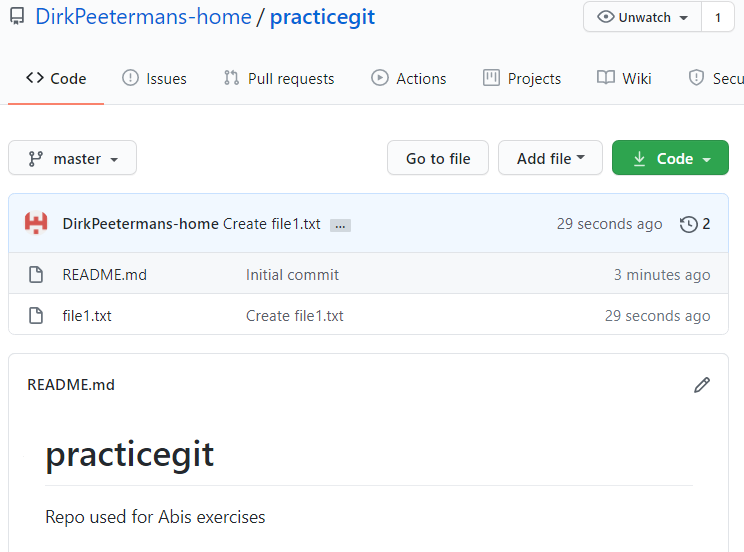
***Terminal and Git – Exercises***

**General Setup**

Create a file called solutionsGit<Yourname>.txt (e.g. solutionsGitSandy.txt). Copy all entered statements together with the number of the exercise in it. Make sure to push it to GitHub at the end of part A, and/or mail it to your instructor.

**A) Basic Git Commands**

1. Create a new repo called “practicegit” in your public GitHub account. Initialize it with a readme file. Add a new file “file1.txt” to it, with content “hello from file 1.”.



2. Open GitBash. Check out which directory you are in.

Create a new directory structure JavaReskilling -> GitProjects on your computer. Use “ls” to check whether the

directories were created correctly. Move to the GitProjects directory, and initialize git there.

dpcvaptp@PC1 MINGW64 ~/Desktop

$ mkdir JavaReskilling

dpcvaptp@PC1 MINGW64 ~/Desktop

$ cd JavaReskilling/

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling

$ mkdir GitProjects

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling

$ cd GitProjects/

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects

$ git config --global user.name "Dirk-Peetermans-home"

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects

$ git config --global user.email [peetermans.dirk@telenet.be](mailto:peetermans.dirk@telenet.be)

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects

$ git init

Initialized empty Git repository in C:/Users/dpcvaptp/Desktop/JavaReskilling/GitProjects/.git/

3. Clone the repository created in exercise 1 into this directory.

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects (master)

$ git clone https://github.com/DirkPeetermans-home/practicegit

Cloning into 'practicegit'...

remote: Enumerating objects: 6, done.

remote: Counting objects: 100% (6/6), done.

remote: Compressing objects: 100% (3/3), done.

remote: Total 6 (delta 0), reused 0 (delta 0), pack-reused 0

Unpacking objects: 100% (6/6), 1.27 KiB | 4.00 KiB/s, done.

4. Create a new branch called “firstbranch” and switch to it. Try to do this in 1 command! (git checkout -b firstbranch) (‘git switch’ is an alternative for ‘git checkout’)

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects (master)

$ cd practicegit/

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (master)

$ git branch firstbranch

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (master)

$ git checkout firstbranch

Switched to branch 'firstbranch'

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ ls -l

total 2

-rw-r--r-- 1 dpcvaptp 197121 20 sep 18 11:26 file1.txt

-rw-r--r-- 1 dpcvaptp 197121 45 sep 18 11:26 README.md

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git status

On branch firstbranch

nothing to commit, working tree clean

5. Add a file called “file2.txt” to the branch (you can make up some content) and stage it. Try to commit it without adding a commit message. Use “git status” regularly to see what is happening as from now!

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ echo "Content for file2" > ./file2.txt

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ ls -l

total 3

-rw-r--r-- 1 dpcvaptp 197121 20 sep 18 11:26 file1.txt

-rw-r--r-- 1 dpcvaptp 197121 18 sep 18 11:32 file2.txt

-rw-r--r-- 1 dpcvaptp 197121 45 sep 18 11:26 README.md

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git status

On branch firstbranch

Untracked files:

(use "git add <file>..." to include in what will be committed)

file2.txt

nothing added to commit but untracked files present (use "git add" to track)

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git add file2.txt

warning: LF will be replaced by CRLF in file2.txt.

The file will have its original line endings in your working directory

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git status

On branch firstbranch

Changes to be committed:

(use "git restore --staged <file>..." to unstage)

new file: file2.txt

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git commit

Aborting commit due to empty commit message.

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git commit -m "Committing file2.txt"

[firstbranch e643bc3] Committing file2.txt

1 file changed, 1 insertion(+)

create mode 100644 file2.txt

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git status

On branch firstbranch

nothing to commit, working tree clean

6. Push the result to your remote repository, and check via GitHub whether everything is there.

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git push origin firstbranch

Enumerating objects: 4, done.

Counting objects: 100% (4/4), done.

Delta compression using up to 2 threads

Compressing objects: 100% (2/2), done.

Writing objects: 100% (3/3), 340 bytes | 340.00 KiB/s, done.

Total 3 (delta 0), reused 0 (delta 0), pack-reused 0

remote:

remote: Create a pull request for 'firstbranch' on GitHub by visiting:

remote: https://github.com/DirkPeetermans-home/practicegit/pull/new/firstbranch

remote:

To https://github.com/DirkPeetermans-home/practicegit

\* [new branch] firstbranch -> firstbranch

**B) Undoing things**

1. Add two more files file3.txt and file4.txt. Stage both of them. Unstage file4.txt again. (touch file1, file2)

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git status

On branch firstbranch

Untracked files:

(use "git add <file>..." to include in what will be committed)

file3.txt

file4.txt

nothing added to commit but untracked files present (use "git add" to track)

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git add file3.txt (git add –all)

warning: LF will be replaced by CRLF in file3.txt.

The file will have its original line endings in your working directory

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git add file4.txt

warning: LF will be replaced by CRLF in file4.txt.

The file will have its original line endings in your working directory

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git status

On branch firstbranch

Changes to be committed:

(use "git restore --staged <file>..." to unstage)

new file: file3.txt

new file: file4.txt

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git rm --cached file4.txt

rm 'file4.txt'

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git status

On branch firstbranch

Changes to be committed:

(use "git restore --staged <file>..." to unstage)

new file: file3.txt

Untracked files:

(use "git add <file>..." to include in what will be committed)

file4.txt

1. Commit and check git log. Change the commit message. Revert the last commit such that file3.txt is deleted from your directory.

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git commit -m "Committing file3.txt"

[firstbranch 5a8c634] Committing file3.txt

1 file changed, 1 insertion(+)

create mode 100644 file3.txt

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git status

On branch firstbranch

Changes not staged for commit:

(use "git add <file>..." to update what will be committed)

(use "git restore <file>..." to discard changes in working directory)

modified: Result exercices.docx

Untracked files:

(use "git add <file>..." to include in what will be committed)

file4.txt

~WRL0003.tmp

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

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git commit –amend -m “Committing file3.txt again with new comment”

[firstbranch 45619c3] Committing file3.txt

Date: Fri Sep 18 12:06:51 2020 +0200

1 file changed, 1 insertion(+)

create mode 100644 file3.txt

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git log

commit 45619c3214bf83d59e11b18f2824b55f9cb7f3e1 (HEAD -> firstbranch)

Author: Dirk-Peetermans-home <peetermans.dirk@telenet.be>

Date: Fri Sep 18 12:06:51 2020 +0200

Committing file3.txt

Committing file3.txt again with new comment

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git revert 45619c3214bf83d59e11b18f2824b55f9cb7f3e1 --no-edit

Removing file3.txt

[firstbranch 12e785b] Revert "Committing file3.txt"

Date: Fri Sep 18 15:36:30 2020 +0200

1 file changed, 1 deletion(-)

delete mode 100644 file3.txt

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ ls -l

total 88

-rw-r--r-- 1 dpcvaptp 197121 20 sep 18 11:26 file1.txt

-rw-r--r-- 1 dpcvaptp 197121 47 sep 18 15:01 file2.txt

-rw-r--r-- 1 dpcvaptp 197121 22 sep 18 13:24 file4.txt

-rw-r--r-- 1 dpcvaptp 197121 45 sep 18 11:26 README.md

-rw-r--r-- 1 dpcvaptp 197121 83316 sep 18 15:01 'Result exercices.docx'

1. Try to undo your revert.

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git reset --hard HEAD^ (git revert --no-edit HEAD^)

HEAD is now at feb8162 Revert "Committing again file3.txt"

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ ls -l

total 89

-rw-r--r-- 1 dpcvaptp 197121 20 sep 18 11:26 file1.txt

-rw-r--r-- 1 dpcvaptp 197121 47 sep 18 15:01 file2.txt

-rw-r--r-- 1 dpcvaptp 197121 19 sep 18 15:42 file3.txt

-rw-r--r-- 1 dpcvaptp 197121 22 sep 18 13:24 file4.txt

-rw-r--r-- 1 dpcvaptp 197121 45 sep 18 11:26 README.md

-rw-r--r-- 1 dpcvaptp 197121 83316 sep 18 15:01 'Result exercices.docx'

1. Stage file4.txt again and commit. Undo the commit, but this time, file4.txt should still stay in your directory.

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git add file4.txt

warning: LF will be replaced by CRLF in file4.txt.

The file will have its original line endings in your working directory

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git status

On branch firstbranch

Changes to be committed:

(use "git restore --staged <file>..." to unstage)

new file: file4.txt

Changes not staged for commit:

(use "git add <file>..." to update what will be committed)

(use "git restore <file>..." to discard changes in working directory)

modified: Result exercices.docx

Untracked files:

(use "git add <file>..." to include in what will be committed)

~WRL0003.tmp

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git commit -m "Commit file4.txt !"

[firstbranch 0734a5b] Commit file4.txt !

1 file changed, 1 insertion(+)

create mode 100644 file4.txt

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git status

On branch firstbranch

Changes not staged for commit:

(use "git add <file>..." to update what will be committed)

(use "git restore <file>..." to discard changes in working directory)

modified: Result exercices.docx

Untracked files:

(use "git add <file>..." to include in what will be committed)

~WRL0003.tmp

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

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ ls -l

total 161

-rw-r--r-- 1 dpcvaptp 197121 77161 sep 18 11:47 '~WRL0003.tmp'

-rw-r--r-- 1 dpcvaptp 197121 20 sep 18 11:26 file1.txt

-rw-r--r-- 1 dpcvaptp 197121 19 sep 18 11:50 file2.txt

-rw-r--r-- 1 dpcvaptp 197121 19 sep 18 13:13 file3.txt

-rw-r--r-- 1 dpcvaptp 197121 18 sep 18 11:57 file4.txt

-rw-r--r-- 1 dpcvaptp 197121 45 sep 18 11:26 README.md

-rw-r--r-- 1 dpcvaptp 197121 79892 sep 18 12:08 'Result exercices.docx'

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git reset HEAD^

Unstaged changes after reset:

M Result exercices.docx

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git status

On branch firstbranch

Changes not staged for commit:

(use "git add <file>..." to update what will be committed)

(use "git restore <file>..." to discard changes in working directory)

modified: Result exercices.docx

Untracked files:

(use "git add <file>..." to include in what will be committed)

file4.txt

~WRL0003.tmp

no changes added to commit (use "git add" andExerexDone with /or "git commit -a")

1. Push to your remote repository again. Check that file4.txt is not present there.

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git push origin firstbranch

Enumerating objects: 17, done.

Counting objects: 100% (17/17), done.

Delta compression using up to 2 threads

Compressing objects: 100% (13/13), done.

Writing objects: 100% (15/15), 47.91 KiB | 4.79 MiB/s, done.

Total 15 (delta 7), reused 0 (delta 0), pack-reused 0

remote: Resolving deltas: 100% (7/7), completed with 2 local objects.

To https://github.com/DirkPeetermans-home/practicegit

f827ef2..0588d96 firstbranch -> firstbranch

1. Add file7.txt and file6.txt. Commit and push in one go. Undo the push of file7.txt to github. File7.txt should stay in your local directory. (

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git commit -m "Committing file6.txt & file7.txt" && git push origin firstbranch

[firstbranch a7e5027] Committing file6.txt & file7.txt

2 files changed, 2 insertions(+)

create mode 100644 file6.txt

create mode 100644 file7.txt

Enumerating objects: 10, done.

Counting objects: 100% (10/10), done.

Delta compression using up to 2 threads

Compressing objects: 100% (5/5), done.

Writing objects: 100% (8/8), 852 bytes | 426.00 KiB/s, done.

Total 8 (delta 2), reused 0 (delta 0), pack-reused 0

remote: Resolving deltas: 100% (2/2), completed with 1 local object.

To https://github.com/DirkPeetermans-home/practicegit

e648eda..a7e5027 firstbranch -> firstbranch

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

git rm – cached file7.txt

git commit -m “removing 7”

git push origin firstbranch

$ git reset --soft feb8162287c09d3616e4fb6a9db09af4a9d70e82

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git status

On branch firstbranch

Changes to be committed:

(use "git restore --staged <file>..." to unstage)

new file: file6.txt

new file: file7.txt

Untracked files:

(use "git add <file>..." to include in what will be committed)

file4.txt

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git rm --cached file7.txt

rm 'file7.txt'

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ ls -l

total 91

-rw-r--r-- 1 dpcvaptp 197121 20 sep 18 11:26 file1.txt

-rw-r--r-- 1 dpcvaptp 197121 47 sep 18 15:01 file2.txt

-rw-r--r-- 1 dpcvaptp 197121 19 sep 18 15:42 file3.txt

-rw-r--r-- 1 dpcvaptp 197121 22 sep 18 13:24 file4.txt

-rw-r--r-- 1 dpcvaptp 197121 22 sep 18 15:45 file6.txt

-rw-r--r-- 1 dpcvaptp 197121 22 sep 18 15:46 file7.txt

-rw-r--r-- 1 dpcvaptp 197121 45 sep 18 11:26 README.md

-rw-r--r-- 1 dpcvaptp 197121 83316 sep 18 15:01 'Result exercices.docx'

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git status

On branch firstbranch

Changes to be committed:

(use "git restore --staged <file>..." to unstage)

new file: file6.txt

Untracked files:

(use "git add <file>..." to include in what will be committed)

file4.txt

file7.txt

Stuck here ????

**C) Some more Git**

1. Make some changes to file2.txt directly in the remote repository. Pull the changes into your local repository. Check the contents of file2.txt

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git pull origin firstbranch

From https://github.com/DirkPeetermans-home/practicegit

\* branch firstbranch -> FETCH\_HEAD

Updating 0588d96..e648eda

Fast-forward

file2.txt | 1 +

1 file changed, 1 insertion(+)

$ cat file2.txt

Content for file2

Content modified on guthub

1. Create a new branch “secondbranch” in your local repository, and make sure it contains everything from firstbranch.

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git branch -c firstbranch secondbranch

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (firstbranch)

$ git checkout secondbranch

Switched to branch 'secondbranch'

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (secondbranch)

$ ls -l

total 91

-rw-r--r-- 1 dpcvaptp 197121 20 sep 18 11:26 file1.txt

-rw-r--r-- 1 dpcvaptp 197121 47 sep 21 08:34 file2.txt

-rw-r--r-- 1 dpcvaptp 197121 19 sep 21 08:34 file3.txt

-rw-r--r-- 1 dpcvaptp 197121 22 sep 18 13:24 file4.txt

-rw-r--r-- 1 dpcvaptp 197121 23 sep 21 08:34 file6.txt

-rw-r--r-- 1 dpcvaptp 197121 23 sep 18 17:10 file7.txt

-rw-r--r-- 1 dpcvaptp 197121 45 sep 18 11:26 README.md

-rw-r--r-- 1 dpcvaptp 197121 83316 sep 21 08:34 'Result exercices.docx'

3. Delete firstbranch both in your local and remote repository.

$ git checkout master

Switched to branch 'master'

Your branch is up to date with 'origin/master'.

$ git branch -d firstbranch

error: The branch 'firstbranch' is not fully merged.

If you are sure you want to delete it, run 'git branch -D firstbranch'.

$ git branch -D firstbranch

Deleted branch firstbranch (was 334c991).

$ git branch -a

\* master

secondbranch

remotes/origin/HEAD -> origin/master

remotes/origin/firstbranch

remotes/origin/master

$ git remote prune origin => does not work

$ git remote prune firstbranch => does not work

$ git branch -d -r firstbranch

error: remote-tracking branch 'firstbranch' not found.

$ git push origin --delete firstbranch

To https://github.com/DirkPeetermans-home/practicegit

- [deleted] firstbranch

1. Switch to master. Add a file called file2.txt there too. Add in some text which is different than the content of the file in secondbranch.

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (master)

$ echo "Content for file2.txt in master repo" > file2.txt

5. Stage and commit these changes. Merge the master branch with secondbranch. What happened? Try to solve this.

Hint:

https://openclassrooms.com/en/courses/3321726-manage-your-code-with-git-and-github/3323386-resolving-conflicts

<https://opensource.com/article/20/4/git-merge-conflict>

$ git add file2.txt

$ git commit -m "Committing file2.txt"

$ git merge secondbranch

CONFLICT (add/add): Merge conflict in file2.txt

Auto-merging file2.txt

Automatic merge failed; fix conflicts and then commit the result.

$ cat file2.txt

<<<<<<< HEAD

Content for file2.txt in master repo

=======

Content for file2

Content modified on guthub

>>>>>>> secondbranch

$ git status

On branch master

Your branch is ahead of 'origin/master' by 1 commit.

(use "git push" to publish your local commits)

You have unmerged paths.

(fix conflicts and run "git commit")

(use "git merge --abort" to abort the merge)

Changes to be committed:

new file: Result exercices.docx

new file: file3.txt

new file: file6.txt

Unmerged paths:

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

both added: file2.txt

Untracked files:

(use "git add <file>..." to include in what will be committed)

file4.txt

file7.txt

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (master|MERGING)

$ vim file2.txt

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (master|MERGING)

$ cat file2.txt

Content for file2.txt in master repo

$ git add file2.txt

$ git commit

[master 8face94] Merge branch 'secondbranch' into master

$ git status

On branch master

Your branch is ahead of 'origin/master' by 20 commits.

(use "git push" to publish your local commits)

Untracked files:

(use "git add <file>..." to include in what will be committed)

file4.txt

file7.txt

nothing added to commit but untracked files present (use "git add" to track)

6. Checkout the following tutorial:

https://openclassrooms.com/en/courses/3321726-manage-your-code-with-git-and-github/3323406-ignoring-files

Create files file5.txt and secret.txt in secondbranch. Stage them and commit. Push the changes to your remote repository, but make sure that secret.txt is ignored.

$ git checkout secondbranch

$ echo "secret.txt" > .gitignore

$ git add .gitignore

$ git commit -m "Commit .gitignore"

$ touch file5.txt secret.txt

$ git add file5.txt secret.txt

$ git commit -m "Commit file5.txt & secret.txt"

[secondbranch 6161bce] Commit file5.txt & secret.txt

2 files changed, 0 insertions(+), 0 deletions(-)

create mode 100644 file5.txt

create mode 100644 secret.txt

$ git add file5.txt secret.txt

The following paths are ignored by one of your .gitignore files:

secret.txt

hint: Use -f if you really want to add them.

hint: Turn this message off by running

hint: "git config advice.addIgnoredFile false"

$ git commit -m "Commit file5.txt"

[secondbranch 3347c1c] Commit file5.txt

1 file changed, 0 insertions(+), 0 deletions(-)

create mode 100644 file5.txt

$ git push origin secondbranch

Enumerating objects: 50, done.

Counting objects: 100% (50/50), done.

Delta compression using up to 2 threads

Compressing objects: 100% (39/39), done.

Writing objects: 100% (49/49), 118.90 KiB | 4.76 MiB/s, done.

Total 49 (delta 19), reused 0 (delta 0), pack-reused 0

remote: Resolving deltas: 100% (19/19), done.

remote:

remote: Create a pull request for 'secondbranch' on GitHub by visiting:

remote: https://github.com/DirkPeetermans-home/practicegit/pull/new/secondbranch

remote:

To https://github.com/DirkPeetermans-home/practicegit

\* [new branch] secondbranch -> secondbranch

$ git merge secondbranch

Merge made by the 'recursive' strategy.

.gitignore | 1 +

file5.txt | 0

2 files changed, 1 insertion(+)

create mode 100644 .gitignore

create mode 100644 file5.txt

$ git push origin master

1. Create a subdirectory “subdir” of the practicegit directory and put some files in there. Make sure that everything is pushed to your remote repository.

dpcvaptp@PC1 MINGW64 ~/Desktop/JavaReskilling/GitProjects/practicegit (master)

$ ls -Rl

.:

total 91

-rw-r--r-- 1 dpcvaptp 197121 20 sep 18 11:26 file1.txt

-rw-r--r-- 1 dpcvaptp 197121 38 sep 21 10:19 file2.txt

-rw-r--r-- 1 dpcvaptp 197121 19 sep 21 09:40 file3.txt

-rw-r--r-- 1 dpcvaptp 197121 22 sep 18 13:24 file4.txt

-rw-r--r-- 1 dpcvaptp 197121 0 sep 21 10:23 file5.txt

-rw-r--r-- 1 dpcvaptp 197121 23 sep 21 09:40 file6.txt

-rw-r--r-- 1 dpcvaptp 197121 23 sep 18 17:10 file7.txt

-rw-r--r-- 1 dpcvaptp 197121 45 sep 18 11:26 README.md

-rw-r--r-- 1 dpcvaptp 197121 83316 sep 21 09:40 'Result exercices.docx'

-rw-r--r-- 1 dpcvaptp 197121 0 sep 21 10:06 secret.txt

drwxr-xr-x 1 dpcvaptp 197121 0 sep 21 10:20 subdir/

./subdir:

total 0

-rw-r--r-- 1 dpcvaptp 197121 0 sep 21 10:20 file1.txt

-rw-r--r-- 1 dpcvaptp 197121 0 sep 21 10:20 file2.txt

$ git add subdir

$ git commit -m "commit subdir"

$ git push origin master

**D) Working in a team**

1. Navigate back to your gitprojects directory.

Make a clone of https://github.com/sschillebeeckx/gitex .

Navigate to it, you can keep using the master branch.

2. Edit the hello.txt file, where you add a line containing “Hello from <your name>”.

3. Commit, and try to push your file to the remote repository? Why does (n’t) work? Solve this.

4. Again try to push your solution. At the end we should have a file with a hello message from everybody in the

repository.