



EXERCISES — Capture The Flag

version #1.0.0



**The way is lit. The path is clear.
We require only the strength to follow it.**

Copyright

This document is for internal use at EPITA ([website](#)) only.

Copyright © 2022-2023 Assistants <assistants@tickets.assistants.epita.fr>

The use of this document must abide by the following rules:

- ▷ You downloaded it from the assistants' intranet.*
- ▷ This document is strictly personal and must **not** be passed onto someone else.
- ▷ Non-compliance with these rules can lead to severe sanctions.

Contents

1	Capture The Flag	3
1.1	GOAL	3
1.2	Examples	3
1.3	Flags	4

*<https://intra.assistants.epita.fr>

1 Capture The Flag

Files to submit:

- capture_the_flag/flags.txt

Provided files:

- capture_the_flag/ctf.tar.gz

1.1 GOAL

The purpose of this exercise is to make you discover the basic commands of git. You will have to explore a git repository to find different flags. To find those different flags, you will have to browse through files and the repository history.

You may also have to perform different manipulations to obtain the flags.

To validate the exercise, you will have to add flags in a `flags.txt` file. The expected format is `flag{<flag>}`. Make sure to strip the different flags before submitting them.

The repository ctf you will use is available in the given `capture_the_flag-provided_files.tar`. Pay attention, there are two layers of tar. Feel free to check *info tar* for more information about the command. You can extract the files like that:

```
42sh$ tar -xvf capture_the_flag-provided_files.tar
42sh$ cd capture_the_flag
42sh$ tar -xvf ctf.tar.gz
42sh$ cd ctf
```

Be careful!

Work outside the repository you will use to submit the exercises.

1.2 Examples

The first flag is `This is an example`. Try to submit this flag. The file you have to submit should look like that.

```
42sh$ cat -e flags.txt
flag{This is an example}$
```

1.3 Flags

Name	Description	Diffi- culty
Example	Follow the example in the subject.	Trivial
Author's name	Find the author all of those commits.	Easy
Author's mail	Find the email used by the author.	Easy
Initial Commit Type	Find the type of the initial commit (refer to conventional commits).	Easy
Initial Commit Description	Find the description of the initial commit.	Easy
Initial Commit README	Find the initial content of the README.md.	Medium
Hello World Tag	Find the tag used for the first submission of the exercise hello_world.	Easy
Hello World Char	Find which character has been added in the second submission of the exercise hello_world.	Medium
Gitignore Branch	Find the branch used to write the gitignore.	Easy
Gitignore Flag	Find the flag contained in the first version of the gitignore.	Medium
Gitignore Latest Version	Find the flag contained in the last version of the gitignore.	Medium
Gitignore Date	Find the date when the last version of the gitignore was written.	Easy
Merged Flag	Find the flag by running the script shell obtained after merging branch_to_merge in master.	Hard

The way is lit. The path is clear. We require only the strength to follow it.