GIT – Exercises

**GitHub** is Built for developers - GitHub is a development platform inspired by the way software developers work. From open source to business, GitHub can be used to host and review code, manage projects, and build software alongside millions of other developers.

# Create a GitHub account

To be able to collaborate on GitHub, you need first to **create an account** for yourself. Should you already have an account on GitHub, you can proceed to the next step.

1. Open a web browser and visit the GitHub website: [*https://github.com*](https://github.com).
2. Choose **“Sign up”** in order to create an account for you.
3. Create an account using either your private or HEVs email address.
4. Choose **“Unlimited public repositories for free”** as your plan.

# Install GIT on your personal computer

If you have installed GIT already on your personal computer or you do not plan to use it for these exercises, you can proceed to the next step.

A good and free to test client for Windows and macOS is Fork: <https://git-fork.com>.

# Fork the “git-introduction” project on GitHub

A **fork** is a copy of a GIT repository. Forking a repository allows you to freely experiment with changes without affecting the original project - it is still possible to send pull request to the original developer.

Most commonly, forks are used to either propose changes to someone else's project or to use someone else's project as a starting point for your own idea.

A great example of using forks to propose changes is for bug fixes. Rather than logging an issue for a bug you've found, you can:

* **Fork** the repository.
* Make the **fix**.
* Submit a **pull request** to the project owner.

1. Find the project **“git-introduction”** of the user **“cm0x4D”** on GitHub.
2. Fork the project (The fork button is on top right).

# Fix the spelling error

Unfortunately, the file **readme.md** has a typo. Can you help us and fix the typo?

1. Clone your fork of the git-example repository on your machine.
2. Create a new branch **bugfixes** from the **main** branch.
3. Open the file readme.md in a text editor.
4. Fix the typo in **readme.md**.
5. Commit your changes. Always explain in the commit message what you change.
6. Merge your changes from the **bugfixes** branch back into your **main** branch. Ensure that a commit is created for the merge operation (no fast forward).
7. Push your commits to your fork of the git-introduction repository on GitHub.

# Create a pull request

Pull requests let you tell others about changes you've pushed to a repository on GitHub. Once a pull request is opened, you can discuss and review the potential changes with collaborators and add follow-up commits before the changes are merged into the origin repository by the maintainer of the project.

1. Go to the project site of your fork on GitHub.
2. Create a pull request to the origin repository.  
   Explain what you have changed.
3. If we request you for changes, make them and push the changes to your repository again – your pull request will be updated automatically.
4. Once we approve your pull request, you have successfully completed this exercise.