

 **ai2-education-fiep-turma-4 / git-katas** Publicforked from [eficode-academy/git-katas](#)**Code**


Pull requests

Actions

Wiki

Security

Insights

 master ▾

...

**git-katas** / **ignore** /

This branch is 1 commit ahead of eficode-academy:master.

 **Contribute** ▾**JKrag** ...on 26 Mar 

..



README.md

17 months ago



setup.ps1

7 months ago



setup.sh

7 months ago

# Git Kata: Basic Ignore

We'll work a bit with the `.gitignore` file in this kata. In this file you can specify both file extensions and folder structures that you do not want Git to track. You can still `git add` files and folder that are ignored in the `.gitignore` file.

 **README.md**

issuing a `git add filename` on the file that was just deleted. Sometimes you add a file by accident that was not meant for Git e.g. binary files, class files etc.

If you want to signal to Git that a file needs to be removed from git, but still want it in your working directory, then use `git rm --cached` to issue a remove command on the staging area, but not in your working directory.

## Setup:

1. Run `source setup.sh` (or `.\setup.ps1` in PowerShell)

## The task

---

1. Create a file with the name `foo.s`
2. What is the output of `git status` ?
3. Create a `.gitignore` file in your working directory containing `*.s`
4. What is the output of `git status` ?
5. Commit the `.gitignore` file
6. Commit `file1.txt`
7. Add `txt` files to `.gitignore` by adding a line in the file containing `*.txt`
8. What does `git status` tell us?
9. Change `file1.txt`
10. What does `git status` tell us? Why was the file tracked even though the `txt` extension is in the ignore file?
11. Make another text file in the repository, what does `git status` look like now? Why is it not tracked?
12. Stage the removal of `file1.txt` with the command `git rm --cached`
13. What does `git status` say?
14. Add a new file `file2.txt` and add `!file2.txt` to `.gitignore`.
15. What does `git status` say? Can you think of a use-case for keeping track of a file although the extension is ignored?

## Useful commands

---

- `git rm`
- `git add`
- `git commit`
- `git commit -m`
- `git rm --cached`

## Aliases

---

You can set up aliases as such: `git config --global alias.lol 'log --oneline --graph --all'` This might be useful to you.