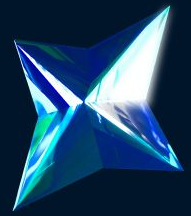
BRO Academy



Working with Version Control Systems

HTML course: Lesson 3

Lesson Plan

1

How the development process works

2

What is a version control system and why is it used

3

Tools for working with Git

4

Basic commands for working

Team Roles

- **Project Manager**: Manages timelines and interacts with other teams
- **Product Owner**: Manages plans and creates requirements
- **UI/UX Designer:** Creates a visual interface
- **Team Lead**: Fully responsible for the project, connects the technical team with project management
- **Markup Developer**: Converts design into HTML/CSS
- **Frontend Developer**: Writes functionality (makes the site interactive)
- **QA**: Finds issues (bugs)

Teamwork

Version Control System

- A version control system is a system for managing versions of source code.

Benefits of Using VCS

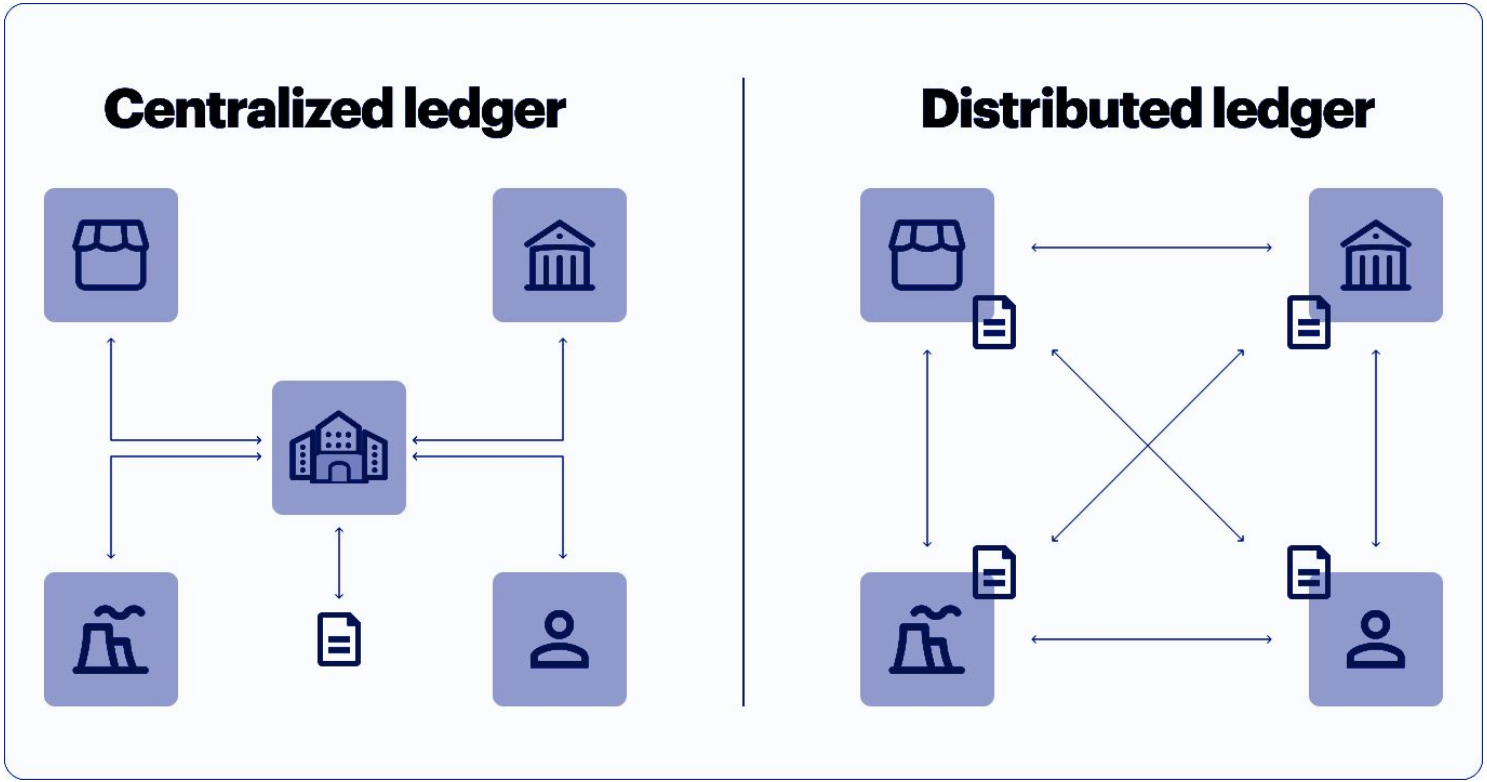
- Stores a complete history of changes
- Describes the reasons for all changes made
- Reverts changes if something goes wrong
- Identifies the cause and the party responsible for errors
- Enables team collaboration on a single project
- Allows code changes without disrupting colleagues' work

Version Control Systems

Git – A distributed (decentralized) version control system for files

- Git
- Mercurial
- Bazaar
- Subversion
- CVS
- etc.

Version Control Systems



Code hosting platforms

GitHub.com

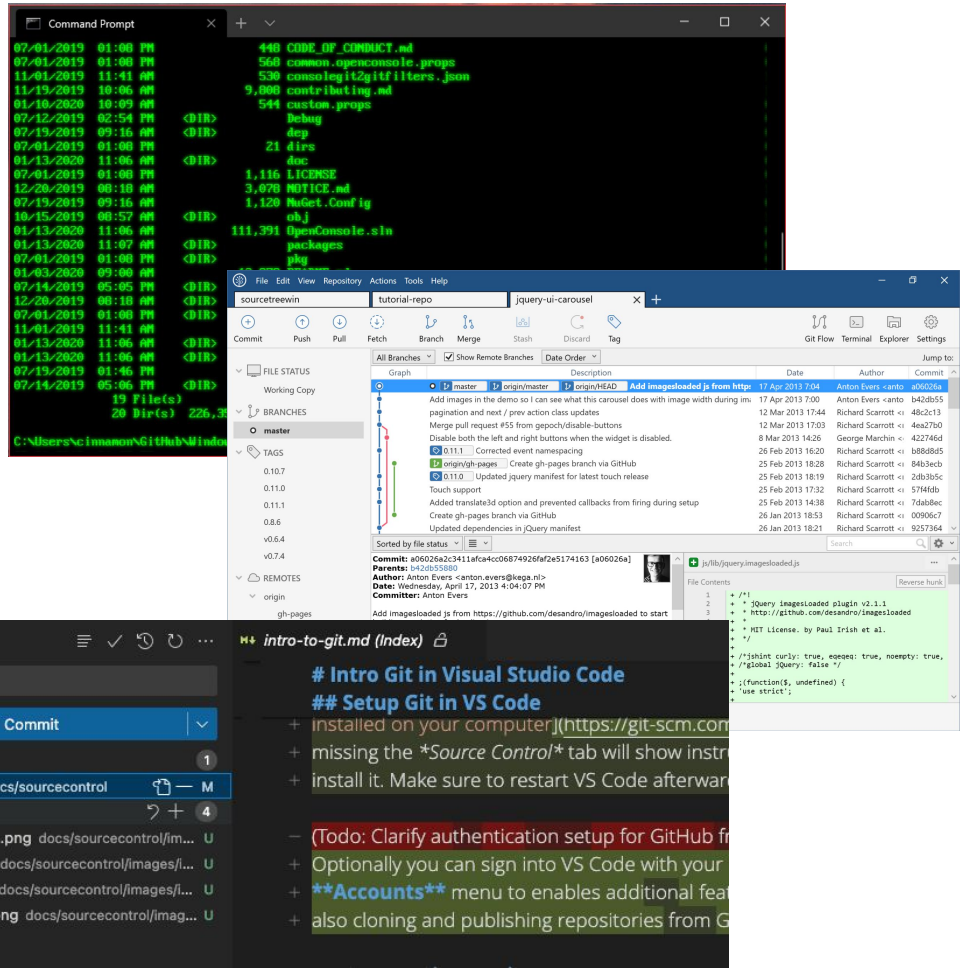
- Git repository storage
- Social network
- Place for open source projects
- Allows you to see how other people's code works
- Use other code in your projects
- Propose your changes
- Issue tracking, code review
- Store your portfolio and publish your projects
- Place for your resume

GitLab.com

BitBucket.org

Git Interfaces

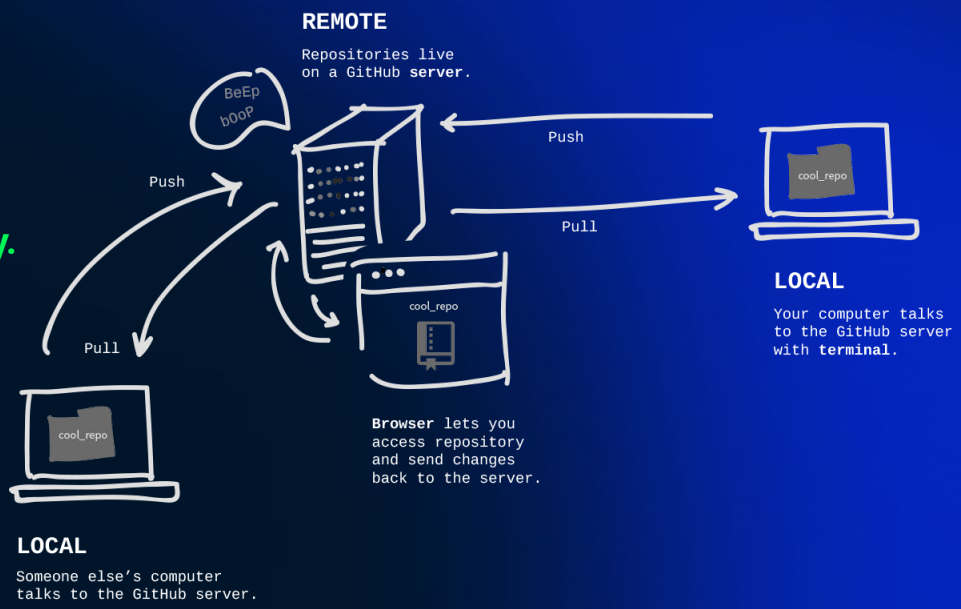
- Console / Terminal
- GUI (graphical user interface)
  - SourceTree
  - GitKraken
  - GitHub Desktop
- Source-code editor
  - **VS Code**
  - IntelliJ IDEA
  - Atom



Repository

A repository is a storage for a project and its history.

- ➔ Configuration files
- ➔ Operation logs (history of changes)
- ➔ File index (their location in project folders)
- ➔ Project files



Local Repository – The repository you work with on your computer.

Remote Repository – The repository hosted on a remote server. This is where all changes made to the project are collected, and where you can retrieve them if needed.

Git Terms

Fork

— A copy of a repository

When you “fork” a project, GitHub will make a copy of the project that is entirely yours.

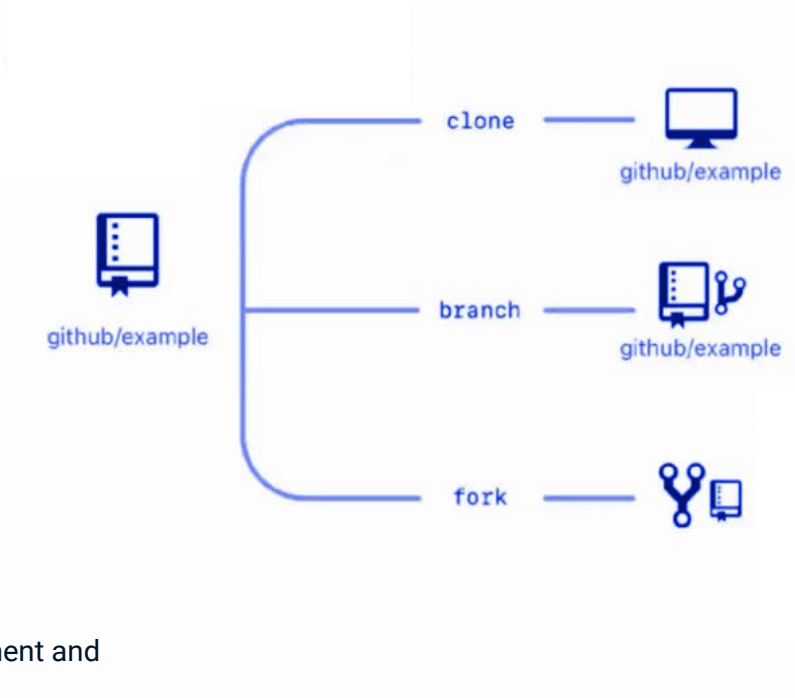
Clone

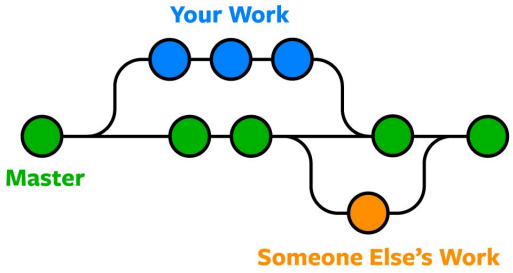
– Copying a repository to your computer

Branch

– A parallel "branch" in the repository

Branching means you diverge from the main line of development and continue to do work without affecting that main line.



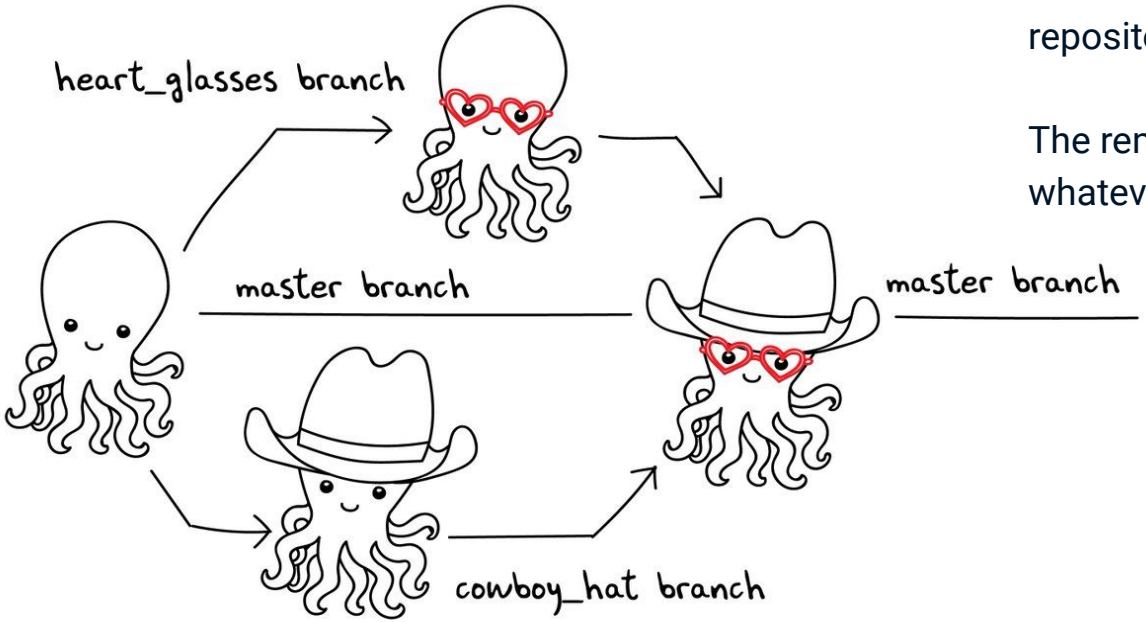


git

branch

Master (main) is the main branch of any repository.

The remaining branches can be called whatever you like.



Branch names

- Descriptive Names
  login, navbar-overflow
- Use Hyphens
- Alphanumeric Lowercase Characters. Avoid punctuation, spaces, underscores, or any special characters whenever possible.

  fix-login-issue
  fixLoginIssue or fix_login_issue
- Avoid Unneccessary Hyphens
  feat/new--login-
- Short and Effective

Git Terms

Commit

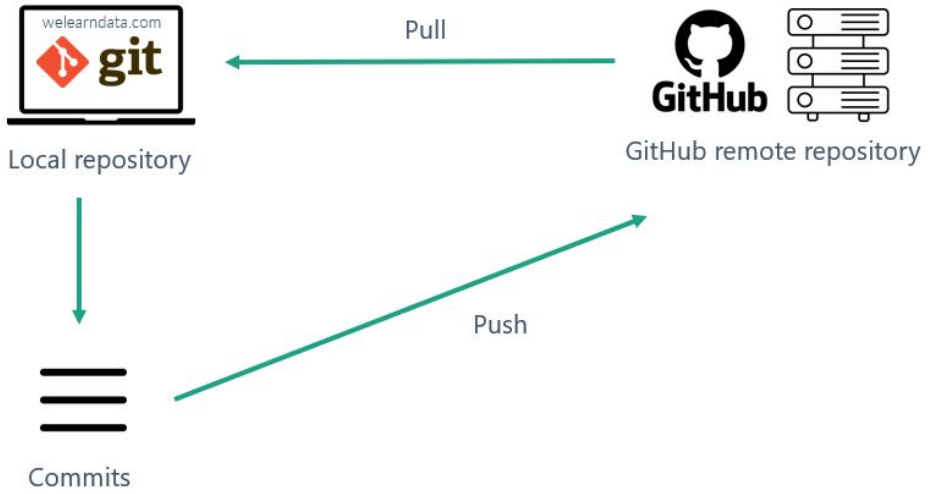
— Saving changes, the current state of the project.

Pull

– Downloading the latest saved changes from the remote repository.

Push

– Sending new commits to the remote repository.

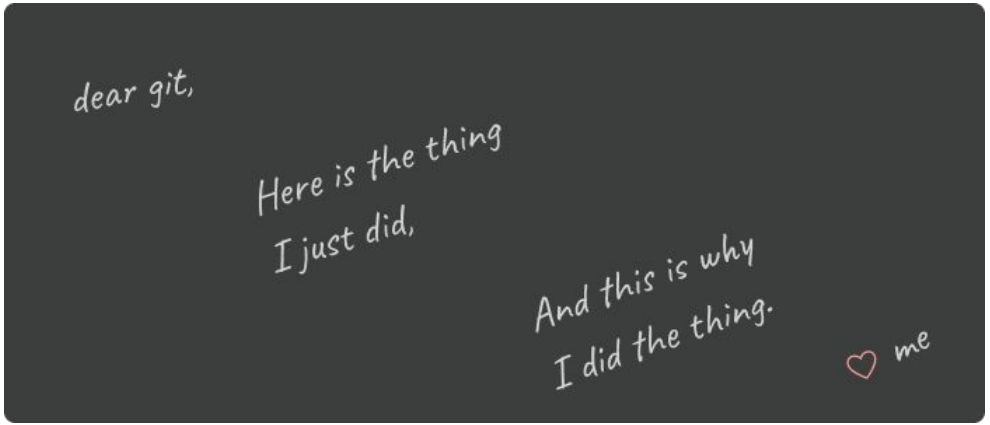


How to name commits

- Imperative Mood
  Fix bug #67
  Fixed bug #67

- Try to fit the subject line within 50 characters.
  Avoid trailing period and unneccessary words/symbols.

- Capitalize the description



Why should I follow the standards?

- Clarity and Understanding
- Collaboration and Teamwork
- Ease of Navigation and Maintenance
- Documentation and Knowledge Transfer
- Project Quality

Git Terms

Pull Request

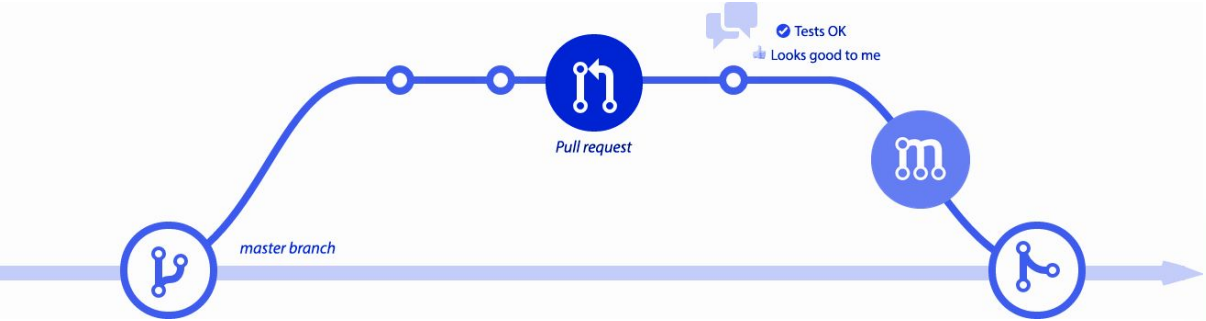
– Request to merge the main repository with its fork (or to merge branches)

Code Review

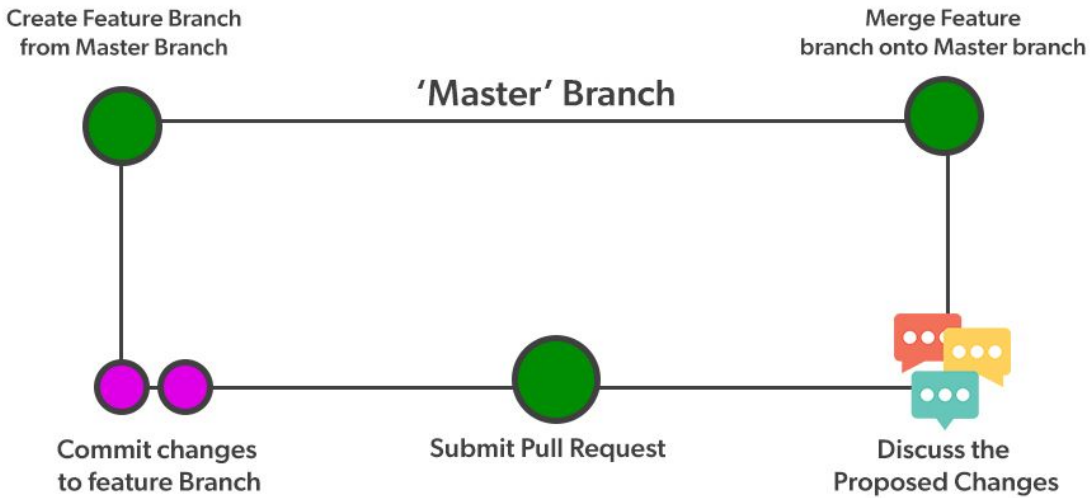
– Checking code (appearance, functionality, and requirement compliance).

Merge

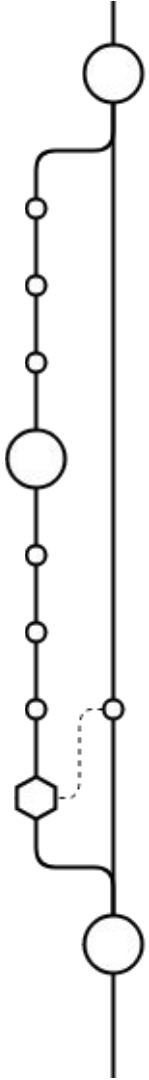
– Combining changes made in a branch or fork



Process of working with a repository



Git workflow for a task:



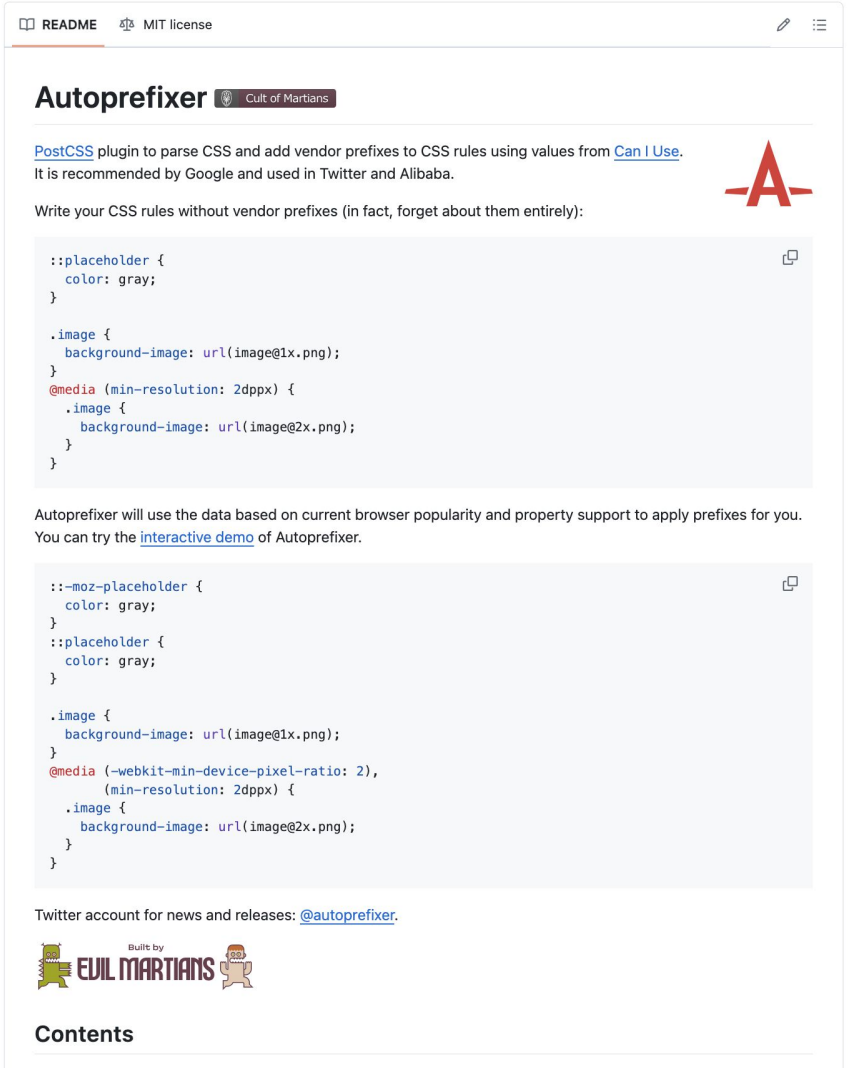
1. Create a new branch for a current lesson.
⇣
2. Write some code.
⇣
3. Commit the changes, specifying what was done in a commit message.
⇣
4. Push the changes to the repository.
⇣
5. Create a pull request.
⇣
6. Change the status in Jira to "In review". Include a link to the open pull request in the task comment.
⇣
7. If the task is reopened, repeat the process from step 2.
A new branch is not needed.
⇣
8. Merge PR after approve
⇣
9. Pull changes from remote master to local master

README.md

is a text file that introduces and explains a project.

It contains information that is commonly required to understand what the project is about.

Can be written in any text file format, the most common one that is used nowadays is **Markdown**.



README MIT license

Autoprefixer Cult of Martians

PostCSS plugin to parse CSS and add vendor prefixes to CSS rules using values from Can I Use.
It is recommended by Google and used in Twitter and Alibaba.

Write your CSS rules without vendor prefixes (in fact, forget about them entirely):

```
::placeholder {
  color: gray;
}

.image {
  background-image: url(image@1x.png);
}
@media (min-resolution: 2dppx) {
  .image {
    background-image: url(image@2x.png);
  }
}
```

Autoprefixer will use the data based on current browser popularity and property support to apply prefixes for you.
You can try the interactive demo of Autoprefixer.

```
::-moz-placeholder {
  color: gray;
}
::placeholder {
  color: gray;
}

.image {
  background-image: url(image@1x.png);
}
@media (-webkit-min-device-pixel-ratio: 2),
       (min-resolution: 2dppx) {
  .image {
    background-image: url(image@2x.png);
  }
}
```

Twitter account for news and releases: @autoprefixer.

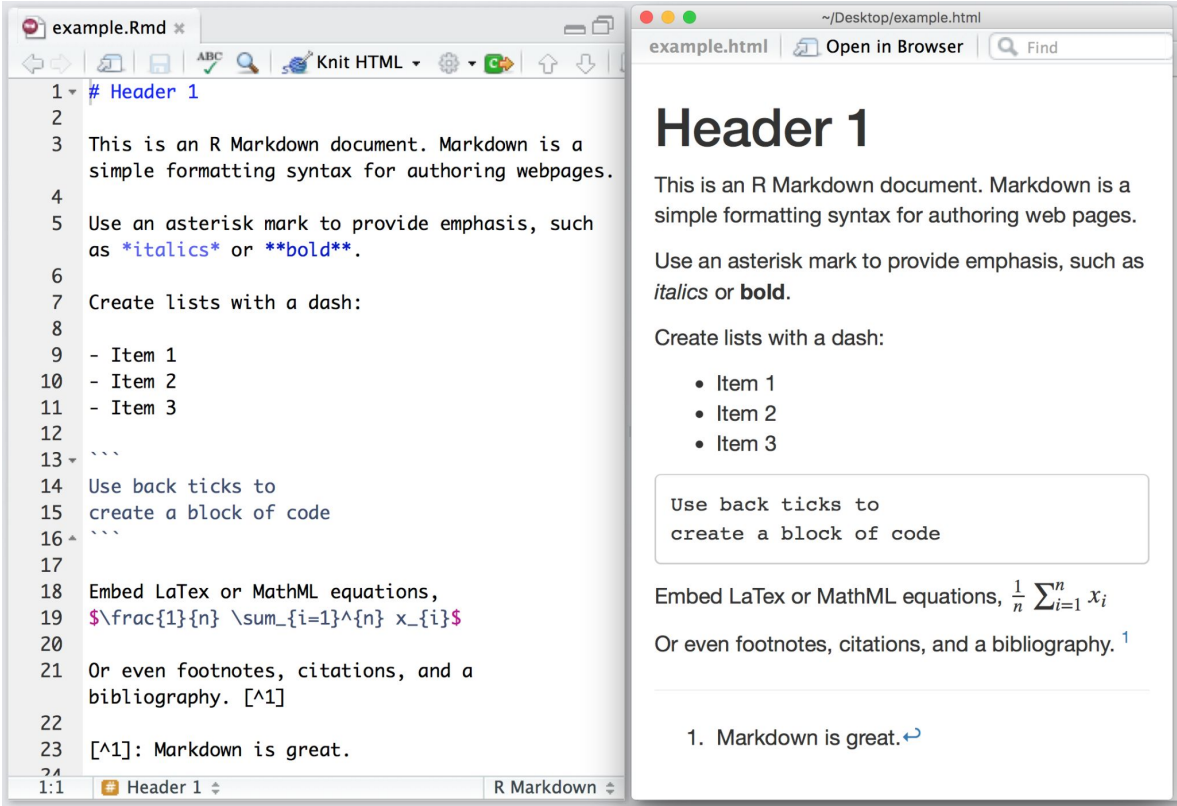
Built by EVIL MARTIANS

Contents

Markdown

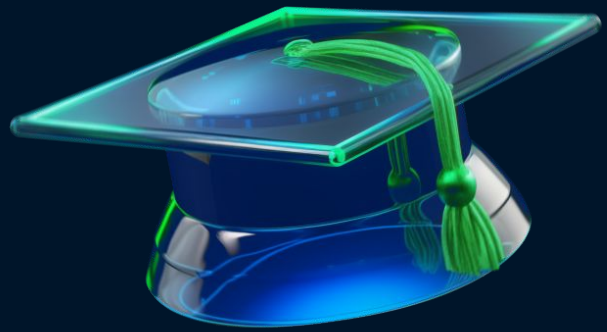
is a simple way to format text that looks great on any device.

It doesn't do anything fancy like change the font size, color, or type — just the essentials, using keyboard symbols you already know.



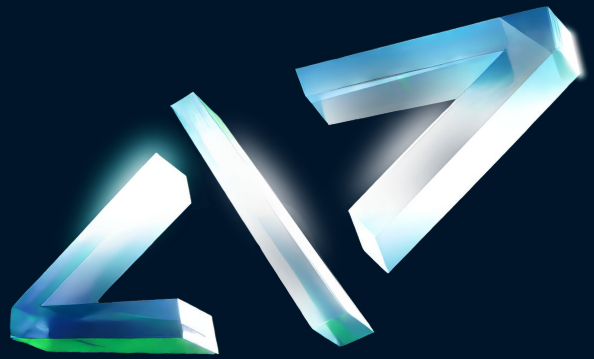
Summary

1. What a version control system is and why it is needed.
2. What Git and GitHub are.
3. How to work with repositories.
4. Basic Git terms.
5. The process of working with Git on the course project.



Homework

1. Fork a repository on GitHub
2. Create your own repository on GitHub
    a. Upload the markup of the homework project from the last lesson
    b. Practice to create branch and pull request
3. Teamwork with one repository



Quality Criteria for HTML Course

❤️Mandatory for passing the course

💛Required for the highest grade

💚Optional

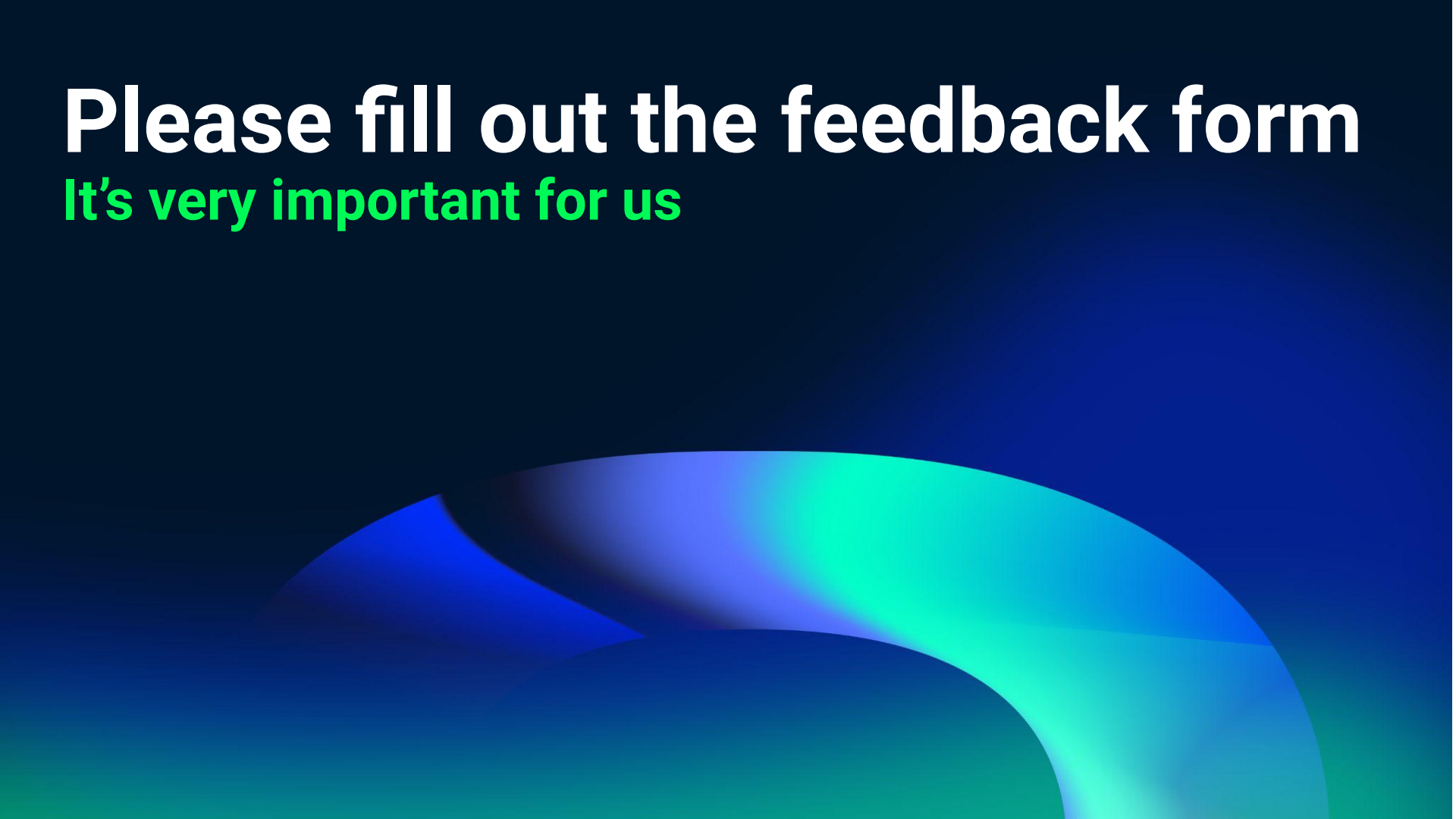
❤️ 4.1 The site should have a GitHub repository.

BRO Academy

QUESTIONS?

Please fill out the feedback form

It’s very important for us




THANK YOU!
Have a good evening!

git

origin

– The alias for the remote repository located on GitHub (this can be changed).