**GitHub**

GitHub - VCS - version control system

Після встановлення gitHub

В терміналі --git version

Встановлення імені користувача та пароль:

* git config --global user.name "......"
* git config --global user.email " ....."

Команди в терміналі: https://mate-academy.github.io/fe-program/tools/terminal/useful-commands

* **pwd** - подивитись в якій папці ми зараз знаходимось
* **ls**  - подивитись вміст данної папки або вказати шлях **ls d:/** або **ls ~** корінна папка
* **ls ./documents/project** - вміст папки з данної директорії
* **cd /frontEnd/academy** - перехід в данну папку
* **cd..** - вихід на папку вище
* **cd../../** - вихід 2 папками вище
* **mkdir** -створювання папок
* **mkdir -p** c/images - створення папки c а всередині папки images
* **touch** a/style.css - дозволяє створювати файл або декілька
* **rm** style.css -remove видалення файлу або декількох
* **rm -r** b - видалення папки b
* **rm a/\*** - видалення всіх папок всередині папки a
* **rm -r** a/\* - видалення всіх файлів з папки а
* **rm -r ./\***.md - видалення всіх файлів з включеної папки з закінченням md
* **mv** a aaa - перейменувати папку a в ааа
* **mv a c**(якшо така вже створена) - перемістити папку a в папку с
* **cat** ./document/style.css - відкрити файл style.css
* **head** ./document/style.css - покаже перші 10 строк файлу
* **tail -n 5** ./document/style.css - покаже останні 5 строк файлу style.css
* **tail -f** - дозволяє дивитись все шо додається до файлу в реальному часі
* **grep** margin ~ document/myProject/style.css - буде шукати строку margin в файлі style.css
* **less** ./ document/myProject/style.css - редактор дозволяє відкрити файл і запустити help через h
* **man** (назва команди) - допомагає взнати як використовувати команду
* **history** - показує історію введення команд
* **!3** - включає 3 команду з історії
* **ctrl + r** - вводимо початок останньої команду і натискаючи знову ctrl + r перебираємо команди
* **!!** - включає останню команду cd
* **ctrl + с** - відмінити останню введену команду
* **echo 'Hello world'** - текст виведеться на екран
* **echo 'Hello world' > hello.txt** - зявиться файл hello.txt а в ньому текст 'Hello world'
* **cat .zsh\_history | sort | uniq > sorted\_history** - виведе команди відсортовані по алфавіту без повторів в файл sorted\_history
* **ls grep | 'e'** - виведе папки з данної директорії в котрих в назві є 'e'
* **which ls** - допомагає дізнатись звідки взялась якась команда
* **env** - показує змінні оточення
* **х=123** - встановлення значення змінній х локально
* **export x=456** - встановлення значення змінній х глобально
* **х=122 ./document/print\_x** -використання змінної х при запускові скріпта
* **ls -a -l** - покаже всі файли, навіть скриті в вигляді детального списку
* **alias la='ls -la'** - присвоєння команді la - довшої команди ls -la
* **alias** покаже всі псевдоніми
* **unalias** - видалити псевдонім
* **touch ~/.bashrc** - створить файл
* **open -t ~/.bashrc** - відкриє файл в ньому прописуємо псевдонім і зберігаємо, при наступному запускові терміналу псевдоніми будуть працювати. Так само можна зробити зі змінними котрі треба записати на постійній основі.
* в файлі **.bashrc** - можна робити коментарі, перед текстом ставити #

**git status** - статус файла

**git init** - почне стежити за файлом

**git commit -a** - зберегти дані потім в редакторі "i" ввести текст збереження ентер, :wq -записати і вийти

**git commit -a -m** - зміни одразу запишуться

**git log** - покаже всі коміти та коментарі

Над кожним коммітом написаний спеціальний номер ХЕШ

**git checkout номер ХЕШ** - включиться та версія сторінки яка була перед змінами

**git checkout master** - вернеться на останню версію сторінки