

# **Pracownia podstaw programowania**

## **Podstawy git**

## 1. Inicjalizacja repo gita

- a) `git init`
- b) `git clone ADRESURL [folder repo]`

## 2. Info o repozytorium

- a) wyświetlenie repo zdalnego

`git remote -v`

- b) lista wszystkich commitów

`git log`

- c) formatowanie listy wszystkich commitów

`git log --pretty=[oneline, short, medium, full,, email]`

- d) n ostatnich commitów

`git log -n`

- e) wyświetlenie commitów w odwrotnej kolejności

`git log --reverse`

- f) wyświetlenie statystyk dla poszczególnych commitów

`git log --stat`

- g) wyświetlenie commitów określonego autora

`git log --author="Dane autora"`

- h) wyświetlanie listy commitów + skrócony do 4 znaków kod SHA1

`git log --abbrev-commit --abbrev=4`

- i) wyświetlanie różnic wprowadzonych z każdą rewizją.

`git log -p`

- j) wyświetlenie liczby rewizji danego autora

`git log --pretty=oneline --author="Dane autora" | wc -l`

- k) wyświetlenie liczby wszystkich rewizji

`git log --oneline | wc -l`

l) wyświetlenie listy plików w każdym commicie

`git log --name-status`

m) filtrowanie commitów po określonej dacie

`git log --after="DATA"`

Data może być również : yesterday, 1 week ago

n) filtrowanie commitów przed określoną datą

`git log --before="DATA"`

o) filtrowanie commitów po określonych plikach

`git log --` Lista plików

p) filtrowanie commitów po określonej wiadomości commitu

`git log --grep="Szukana fraza"`

q) zmiany wprowadzone w poszczególnych plikach commitów

`git show`

r) wyświetlenie autor + liczby zmian + listy zmian

`git shortlog`

s) wyświetlenie liczby zmian + autora

`git shortlog -s`

t) punkt s) posortowany wg ilości zmian

`git shortlog -sn`

u) wyświetlanie zmian w pliku

`git blame [OPCJA] NAZWAPLIKU`

OPCJE:

-t – zamiana daty na znacznik timestamp

-L x, y – wyświetlanie linii od x do y

v) graf zmienionych plików repo

`git log --name-status --graph --oneline`

w) graf krótkich commitów

```
git log --graph --oneline
```

### 3. Committed

a) konfiguracja danych w repozytorium

```
git config user.name "FIRST_NAME LAST_NAME"
```

```
git config user.email "email"
```

b) sprawdzanie statusów plików

```
git status
```

```
git status -s
```

c) dodawanie plików do śledzenia

```
git add . (wszystkie pliki)
```

```
git add *.txt (wszystkie pliki o rozszerzeniu txt)
```

d) tworzenie commitów

```
git commit
```

lub

```
git commit -m "komentarz"
```

e) usuwanie pliku ze śledzenia

### 4. Tworzenie tagów

a) Lista tagów

```
git tag
```

b) Tworzenie tagu

```
git tag -a "nazwa tagu" -m "komentarz"
```

c) Tworzenie tagu do starszych commitów

```
git tag -a "nazwa tagu" -m "komentarz" SHA1commitu
```

d) Usuwanie tagu

```
Git tag -d "nazwa tagu"
```

## 5. Tworzenie gałęzi

a) Lista gałęzi

`git branch`

b) Tworzenie gałęzi

`git branch NameBranch`

c) Przełączanie na inną gałąź

`git checkout NameBranch`

d) Tworzenie i przełączenie na inną gałąź

`git checkout -b NameBranch`