Komendy GIT

**Wersja GITa**

git --version

**Dokumentacja**

Dokumentacja komend Gita:

git add --help

git branch --help

git config --help

git init --help

**Konfiguracja**

Lista wszystkich opcji konfiguracyjnych:

git config -l

Ustalenie wartości wybranej opcji konfiguracyjnej:

git config --global user.name „Imie Nazwisko”

git config --global user.mail [you@example.com](mailto:you@example.com)

Globalny plik konfiguracyjny użytkownika:

C:\Users\nazwa-użytkownika\.gitconfig

Plik konfiguracyjny dotyczący konkretnego repozytorium:

.git\config

**Inicjalizacja repozytorium**

Inicjalizacja nowego repozytorium zwykłego w bieżącym folderze:

git init

Inicjalizacja nowego repozytorium zwykłego w podanym folderze:

git init scierzka/do/folderu

Inicjalizacja nowego repozytorium surowego w bieżącym folderze:

git init --bare

Inicjalizacja nowego repozytorium surowego w podanym folderze:

git init --bare scierzka/do/folderu

**Klonowanie repozytorium**

Klonowanie repozytorium do folderu o nazwie idenytcznej jak repozytorium:

git clone adres-repozytorium

Klonowanie repozytorium do bieżącego folderu:

git clone adres-repozytorium .

Klonowanie repozytorium do dowolnego folderu:

git clone adres-repozytorium ścieżka/do/folderu

**Informacje o repozytorium**

Lista uczestników projektu:

git shortlog -s -n

Lista uczestników projektu:

git shortlog -s -n | wc -l

Liczba plików w repozytorium

find . -type f -print | grep -v -E ‘/\.git/’ | wc -l

Liczba rewizji zawartych w repozytroium:

git log --pretty=oneline | wc -l

**Historia projektu**

Sprawdzanie histroii projektu:

git log

Skrócona historia projektu:

git log --pretty=oneline

git log --oneline

Skrócona historia zawierająca zminimalizowane identyfikatory SHA-1

git log --abbrev-commit --abbrev=4 --pretty=oneline

Historia zawierająca trzy ostatnie rewizje:

git log -3

Historia zawierająca rewizje podanego użytkownika:

git log --author=użytkownik

Historia zawierająca rewizję z podanego okresu:

git log --since=”2012-01-01” --until=”2012-01-15”

Formatowanie wydruku historii według dowolnych reguł:

git log --pretty=format:”%h %cd”

Lista ostatnich 10 rewizji w skróconym formacie prezentowana graficznie:

git log --pretty=oneline --abbrev-commit --abbrev=4 -10 --graph

**Interfejs graficzny**

Uruchomienie programu Git GUI:

git gui

Uruchomienie programu do wizualnej prezentacji zawartości repozytorium:

gitk

**Przywracanie obszaru roboczego do wybranej rewizji**

Przywrócenie stanu plików w obszarze roboczym do postaci z ostatniej rewizji (polecenia nie usuwają nowych plików)

git reset --hard

git checkout -f

Przywrócenie stanu plików w obszarze roboczym do postaci z podanej rewizji (rewizje późniejsze są usuwane z historii projektu):

git reset --hard [SHA-1]

Przywrócenie stanu plików w obszarze roboczym do postaci z podanej rewizji(po tej operacji repozytorium znajduję się w stanie detached head):

git checkout -f [SHA-1]

**Stan repozytorium**

Informacje o stanie repozytorium:

git status

skrócona informacja o stanie repozytorium:

git status -s