

# Podstawy Gita

## 1 Wstęp

Git jest rozproszonym systemem kontroli wersji, czyli oprogramowaniem służącym do śledzenia zmian (w kodzie źródłowym). Praca z Gitem opiera się na tworzeniu kolejnych komitów/wersji. Komitowanie w Gicie jest dwuetapowe: najpierw należy przygotować listę zmian, które mają zostać zapisane (dodanie plików do stage'a), następnie należy wkomitować wcześniej przygotowane zmiany (te które znajdują się w stage'u).

W ramach jednego projektu można za pomocą Gita tworzyć wiele gałęzi (branch) a następnie scalać te gałęzie z gałęzią główną (master). Można pracować tylko na lokalnej wersji repozytorium albo przysyłać zmiany w projekcie na repozytorium zdalne.

## 2 Rozpoczęcie pracy z Gitem

Pracę z Gitem możemy rozpocząć na dwa sposoby, za pomocą:

- `git init` - to polecenie zamienia aktualny katalog w repozytorium Gita
- `git clone <link>` - robiąc lokalną kopię repozytorium zdalnego.

1. Stwórz nowy katalog o nazwie "zwierzeta" i przekopij do niego pliki "kot.txt" i "pies.txt" (Pliki są do pobrania na WIKAMPie).
2. Otwórz konsolę (w Windows: Win + R cmd lub konsolę Git Bash) i wejdź do katalogu "zwierzeta".
3. Wykonaj polecenie `git init`, sprawdź czy system dodał do katalogu "zwierzeta" katalog ".git" (plik jest ukryty). Jeżeli tak, to udało się zmienić ten katalog w repozytorium Gita.
4. Sprawdź czy masz plik `.git\config` (znajduje się w katalogu repozytorium). Jeżeli plik nie istnieje należy go stworzyć i dodać do niego wpis:

```
[user]
name = Imię Nazwisko
email = adres@edu.p.lodz.pl
```

Jeżeli plik istnieje to sprawdź czy jest w nim wpis z danymi użytkownika. Jeżeli nie ma danych użytkownika to je dodaj.

Drugim sposobem na ustawienie wartości w pliku `.git\config` jest skorzystanie z poleceń:

```
git config --local user.name "imie i nazwisko"
git config --local user.email "adres@..."
```

## 3 Podstawowe polecenia

- `git status` - sprawdzenie w jakim stanie znajdują się pliki (widać, które zostały zmienione, które zostały dodane do stage'a, które zostały usunięte)
- `git add <nazwaPliku>` - dodanie do stage'a zmian w danym pliku

- `git add .` - dodanie do stage'a wszystkich zmian w danym katalogu i katalogach poniżej
  - `git rm <nazwaPliku>` - dodanie do stage'a usunięcia pliku
  - `git commit -m "<komentarz>"` - wkomitowanie zmian ze stage'a
  - `git diff --staged` - wyświetlenie różnic dodanych do stage'a w stosunku do aktualnego komita.
  - `git diff` - wyświetlenie różnic nie dodanych do stage'a w stosunku do aktualnego komita.
  - `git push` - wypchnięcie zmian do repozytorium zdalnego
  - `git pull` - pobranie zmian z repozytorium zdalnego
1. Wykonaj polecenie `git status` i zobacz w jakim stanie jest twoje repozytorium. Powinny być dwa niesledzone pliki: "kot.txt" i "pies.txt".
  2. Wykonaj polecenie `git add .`, w ten sposób do stage'a zostaną dodane wszystkie zmiany z katalogu "zwierzeta".
  3. Ponownie wykonaj polecenie `git status` i zobacz w jakim stanie teraz jest twoje repozytorium. Powinny być dwa pliki: "kot.txt" i "pies.txt" przygotowane do wkomitowania.
  4. Wkomituj zmiany za pomocą polecenia `git commit -m "dodanie pliku pies.txt i kot.txt"`. Powinna pojawić się informacja o zmienieniu dwóch plików.
  5. Wykonaj polecenie `git log` i zobacz historię zmian. Powinna być widoczna informacja o jednym komicie. Polecenie `git log` pokazuje hash (ciąg liter i cyfr) danego komita, informację na jakiej gałęzi jest HEAD (HEAD → master), autora, datę i jakie pliki zostały dodane.
  6. W przypadku większej ilości komitów i gałęzi polecenie `git log` jest mało czytelne. Można stworzyć własny format wyświetlania, który będzie lepiej pokazywał zmiany na gałęziach.
  7. Wejdź do pliku ".gitconfig" (znajduje się w katalogu użytkownika) i dodaj tam wpis tworzący polecenie `git lg` do "ładniejszego wyświetlania" (<https://stackoverflow.com/questions/1057564/pretty-git-branch-graphs>).
- Warto też dodać do pliku ".gitconfig" informację o kolorach, których git może używać do rysowania grafów. Niektóre standardowo używane kolory są trudno widoczne na czarnym tle konsoli. Oczywiście kolory można ustawić według swoich preferencji.
- ```
[log]
graphColors = white,red,green,magenta,yellow,cyan
```
8. Wykonaj polecenie `git lg`. Teraz zamiast całego hashu widać tylko jego początek, został też dodany węzeł (\*).

## 4 Praca na gałęziach

- `git branch <nazwaGałęzi>` - tworzy gałąź o podanej nazwie
- `git checkout <nazwaGałęzi>` - przełącza na gałąź/komit o podanej nazwie/hashu

- `git merge --no-ff <nazwaGałęzi>` - scala zmiany z gałęzi `nazwaGałęzi`. Najczęściej wykorzystywane do scalenia gałęzi z gałęzią `master`. Wykonywane na gałęzi `master`.
- `git rebase <nazwaGałęzi>` - przenosi daną gałąź na szczyt gałęzi `nazwaGałęzi`. Najczęściej wykorzystywane do przeniesienia gałęzi na szczyt gałęzi `master` (`git rebase master`). Wykonywane na gałęzi `nazwaGałęzi`.

1. Utwórz gałąź o nazwie `kot`, polecenie: `git branch kot`, a następnie gałąź o nazwie `pies`.
2. Przełącz się na gałąź `kot` poleceniem `git checkout kot` i za pomocą polecenia `git status` sprawdź czy udało się to zrobić.
3. Dokonaj zmian w pliku `"kot.txt"`. Zamień małe litery na początku wersu wierszyka na wielkie.
4. Wykonaj polecenie `git diff` i zobacz jaka jest różnica między aktualną wersją projektu a tą w `stag'u`.
5. Dodaj zmiany z pliku `"kot.txt"` do stage'a (polecenie `git add kot.txt`) lub `git add .` - ponieważ zmiany zostały zrobione tylko w pliku `"kot.txt"`
6. Sprawdź, czy teraz polecenie `git diff` pokazuje jakieś zmiany.
7. Sprawdź zmiany stage'a w stosunku do ostatniego komita - polecenie `git diff --staged`.
8. Wkomituj zmiany poleceniem `git commit -m "Zmiana malych liter na wielkie w pliku kot.txt"`
9. Przełącz się na gałąź `pies`, zmień w pliku `"pies.txt"` cyfry 1, 2, 1, 2 na słowa "Jeden, dwa, jeden, dwa", dodaj zmiany do stage'a, zrób komit z odpowiednim komentarzem. Wywołaj polecenie `git lg` i zobacz czy widzisz obie gałęzie `pies` i `kot`.

```
C:\git\zwierzeta>git lg
* 9f2f493 - (3 seconds ago) zmiana cyfr 1, 2 na slowa w pliku pies.txt - Jojczyk (HEAD -> pies)
* 69d6667 - (2 minutes ago) Zmiana malych liter na wielkie w pliku kot.txt - Jojczyk (kot)
/
* 28d533b - (5 minutes ago) dodanie pliku pies.txt i kot.txt - Jojczyk (master)
```

10. Dalszych zmian dokonuj w pliku `"pies.txt"`. Zmień cyfry 3, 4 na słowa, dodaj zmiany do stage'a, zrób komit z odpowiednim komentarzem.
11. Przełącz się na gałąź `master` i za pomocą polecenia `git merge --no-ff pies` scal zmiany z gałęzi `pies`. Wykonaj polecenie `git lg` i sprawdź czy gałąź `pies` została scalona.

```
C:\git\zwierzeta>git lg
* c8ae65b - (5 seconds ago) Merge branch 'pies' - Jojczyk (HEAD -> master)
/
* 784e69d - (46 seconds ago) zmiana cyfr 3 i 4 na slowa w pliku pies.txt - Jojczyk (pies)
* 9f2f493 - (3 minutes ago) zmiana cyfr 1, 2 na slowa w pliku pies.txt - Jojczyk
/
* 69d6667 - (5 minutes ago) Zmiana malych liter na wielkie w pliku kot.txt - Jojczyk (kot)
/
* 28d533b - (8 minutes ago) dodanie pliku pies.txt i kot.txt - Jojczyk
```

12. Przełącz się na gałąź `kot` i przenieś tę gałąź na szczyt gałęzi `master`, polecenie: `git rebase master`. Przenieś się na gałąź `master` i scal zmiany z gałęzi `kot`. Wykonaj polecenie `git lg` i sprawdź czy operacja się udała.

```
C:\git\zwierzeta>git lg
* e00e089 - (1 second ago) Merge branch 'kot' - Jojczyk (HEAD -> master)
* ec70242 - (8 minutes ago) Zmiana małych liter na wielkie w pliku kot.txt - Jojczyk (kot)
* c8ae65b - (3 minutes ago) Merge branch 'pies' - Jojczyk
* 784e69d - (3 minutes ago) zmiana cyfr 3 i 4 na słowa w pliku pies.txt - Jojczyk (pies)
* 9f2f493 - (6 minutes ago) zmiana cyfr 1, 2 na słowa w pliku pies.txt - Jojczyk
* 28d533b - (11 minutes ago) dodanie pliku pies.txt i kot.txt - Jojczyk
```

13. Utwórz nową gałąź pies5 i od razu się na nią przełącz, polecenie: `git checkout -b pies5`. Zmień w pliku "pies.txt" cyfry 5, 6 na słowa, dodaj zmiany do stage'a, zrób komit z odpowiednim komentarzem.
14. Przełącz się na gałąź master, stwórz gałąź pies7 i się na nią przełącz. Zmień w pliku "pies.txt" cyfry 7, 8 na słowa, dodaj zmiany do stage'a, zrób komit z odpowiednim komentarzem. Zauważ, że teraz w pliku "pies.txt" nie widzisz zmian z gałęzi pies5 (czyli 5 i 6 są cyframi a nie słowami).
15. Przełącz się na gałąź master, scal zmiany z gałęzi pies5. Przełącz się na gałąź pies7 i przenieś tę gałąź na szczyt gałęzi master. Zauważ, że mimo tego, że pracowałeś/aś na tym samym pliku, to dzięki temu, że zmiany dotyczyły różnych linii pliku, nie powstały żadne konflikty. Przenieś się na gałąź master i scal zmiany z gałęzi pies7. Wykonaj polecenie `git lg` i sprawdź czy operacja się udała.

```
C:\git\zwierzeta>git lg
* e417f15 - (2 seconds ago) Merge branch 'pies7' - Jojczyk (HEAD -> master)
* e633209 - (4 minutes ago) zmiana cyfr 7, 8 na słowa w pliku pies.txt - Jojczyk (pies7)
* 59a9d43 - (3 minutes ago) Merge branch 'pies5' - Jojczyk
* 53e7549 - (5 minutes ago) zmiana cyfr 5, 6 na słowa w pliku pies.txt - Jojczyk (pies5)
* e00e089 - (8 minutes ago) Merge branch 'kot' - Jojczyk
* ec70242 - (16 minutes ago) Zmiana małych liter na wielkie w pliku kot.txt - Jojczyk (kot)
* c8ae65b - (10 minutes ago) Merge branch 'pies' - Jojczyk
* 784e69d - (11 minutes ago) zmiana cyfr 3 i 4 na słowa w pliku pies.txt - Jojczyk (pies)
* 9f2f493 - (14 minutes ago) zmiana cyfr 1, 2 na słowa w pliku pies.txt - Jojczyk
* 28d533b - (18 minutes ago) dodanie pliku pies.txt i kot.txt - Jojczyk
```

16. Utwórz nową gałąź pies9 i od razu się na nią przełącz, zmień w pliku "pies.txt" cyfry 9 na słowa, dodaj zmiany do stage'a, zrób komit z odpowiednim komentarzem.
17. Przełącz się na gałąź master. Utwórz nową gałąź pies10 i od razu się na nią przełącz, zmień w pliku "pies.txt" liczby 10 na słowa, dodaj zmiany do stage'a, zrób komit z odpowiednim komentarzem.
18. Przełącz się na gałąź master, scal zmiany z gałęzi pies9. Przełącz się na gałąź pies10 i spróbuj przesunąć gałąź pies10 na szczyt gałęzi master. Ponieważ zarówno w gałęzi pies9 i pies10 dokonano zmian w tej samej części pliku, to wystąpił konflikt.

```

Pies.txt - Notepad
File Edit Format View Help
Jeden, dwa, jeden, dwa
Pewna pani miała pasa
Trzy i cztery, trzy i cztery
Pies ten dziwne miał maniere.
Pięć i sześć, pięć i sześć
Wcale lodów nie chciał jeść.
Siedem, osiem, siedem, osiem
Wciąż o kości tylko prosił.
<<<<<< HEAD
Dziewięć, 10, dziewięć, 10,
=====
9, dziesięć, 9, dziesięć,
>>>>>> 3927ed2 (zmiana liczby 10 na słowa w pliku pies.txt)
Kto znas kości mu przyniesie?
Może ja? może ty?
Licz od nowa raz, dwa, trzy ...

```

19. Konflikt należy rozwiązać ręcznie, Git zaznacza które linijki są problematyczne. Popraw plik tak, aby w wierszu zostały same słowa (bez cyfr i liczb), zapisz zmiany w pliku i kontynuuj przesunięcie gałęzi pies10 na szczyt gałęzi master. W tym celu musisz dodać plik "pies.txt" do stage'a i wykonać polecenie `git rebase --continue`
20. przełącz się na gałąź master i scal zmiany z gałęzi pies10.
21. Wykonaj polecenie `git lg` i sprawdź czy operacja się udała. Sprawdź czy plik wygląda tak jak oczekiwałeś.

```

C:\git\zwierzeta>git lg
* 2797d07 - (2 seconds ago) Merge branch 'pies10' - Jojczyk (HEAD -> master)
/ \
* 7df1f64 - (5 minutes ago) zmiana liczby 10 na słowa w pliku pies.txt - Jojczyk (pies10)
/ \
* 056a7df - (4 minutes ago) Merge branch 'pies9' - Jojczyk
/ \
* 2d4a130 - (6 minutes ago) zmiana cyfry 9 na słowa w pliku pies.txt - Jojczyk (pies9)
/ \
* e417f15 - (9 minutes ago) Merge branch 'pies7' - Jojczyk
/ \
* e633209 - (13 minutes ago) zmiana cyfr 7, 8 na słowa w pliku pies.txt - Jojczyk (pies7)
/ \
* 59a9d43 - (12 minutes ago) Merge branch 'pies5' - Jojczyk
/ \
* 53e7549 - (14 minutes ago) zmiana cyfr 5, 6 na słowa w pliku pies.txt - Jojczyk (pies5)
/ \
* e00e089 - (17 minutes ago) Merge branch 'kot' - Jojczyk
/ \
* ec70242 - (25 minutes ago) Zmiana małych liter na wielkie w pliku kot.txt - Jojczyk (kot)
/ \
* c8ae65b - (19 minutes ago) Merge branch 'pies' - Jojczyk
/ \
* 784e69d - (20 minutes ago) zmiana cyfr 3 i 4 na słowa w pliku pies.txt - Jojczyk (pies)
* 9f2f493 - (23 minutes ago) zmiana cyfr 1, 2 na słowa w pliku pies.txt - Jojczyk
/ \
* 28d533b - (27 minutes ago) dodanie pliku pies.txt i kot.txt - Jojczyk

```

22. Utwórz nową gałąź poprawka i od razu się na nią przełącz, dodaj na końcu plików "pies.txt" i "kot.txt" w nowej linii autora wiersza "Julian Tuwim", dodaj zmiany do stage'a, zrób komit z komentarzem "dodanie autora wierszy Juliana Tuwima".

```
C:\git\zwierzeta>git lg
* 65d16b7 - (2 minutes ago) dodanie autora wierszy Juliana Tuwima - Jojczyk (HEAD -> poprawka)
* 2797d07 - (5 months ago) Merge branch 'pies10' - Jojczyk (master)
/ \
* 7df1f64 - (5 months ago) zmiana liczby 10 na słowa w pliku pies.txt - Jojczyk (pies10)
/ \
* 056a7df - (5 months ago) Merge branch 'pies9' - Jojczyk
/ \
* 2d4a130 - (5 months ago) zmiana cyfry 9 na słowa w pliku pies.txt - Jojczyk (pies9)
/ \
* e417f15 - (5 months ago) Merge branch 'pies7' - Jojczyk
/ \
* e633209 - (5 months ago) zmiana cyfr 7, 8 na słowa w pliku pies.txt - Jojczyk (pies7)
/ \
* 59a9d43 - (5 months ago) Merge branch 'pies5' - Jojczyk
/ \
* 53e7549 - (5 months ago) zmiana cyfr 5, 6 na słowa w pliku pies.txt - Jojczyk (pies5)
/ \
* e00e089 - (5 months ago) Merge branch 'kot' - Jojczyk
/ \
* ec70242 - (5 months ago) Zmiana małych liter na wielkie w pliku kot.txt - Jojczyk (kot)
/ \
* c8ae65b - (5 months ago) Merge branch 'pies' - Jojczyk
/ \
* 784e69d - (5 months ago) zmiana cyfr 3 i 4 na słowa w pliku pies.txt - Jojczyk (pies)
* 9f2f493 - (5 months ago) zmiana cyfr 1, 2 na słowa w pliku pies.txt - Jojczyk
/ \
* 28d533b - (5 months ago) dodanie pliku pies.txt i kot.txt - Jojczyk
```

23. Julian Tuwim nie jest autorem wiersza o psie. Trzeba to poprawić. Można dodać nowy komit z poprawionym autorem. Tutaj jednak zmienimy ostatni komit, tak żeby nie mnożyć niepotrzebnie komitów. Metoda ta działa tylko dla ostatniego komita. Zmień ostatnią linijkę w pliku "pies.txt" na "Danuta Wawiłow". Dodaj zmiany do stage'a. Następnie zmień ostatni komit używając polecenia `git commit --amend` i w oknie notatnika, które wyskoczy, zmień komentarz na "dodanie autorów wierszy Juliana Tuwima i Danuty Wawiłow", zapisz plik i zamknij okno notatnika.
24. przełącz się na gałąź master i scal zmiany z gałęzi poprawka.

```
C:\git\zwierzeta>git lg
* f489b23 - (2 seconds ago) Merge branch 'poprawka' - Jojczyk (HEAD -> master)
/ \
* 673a02a - (23 minutes ago) dodanie autorów wierszy Juliana Tuwima i Danuty Wawiłow - Jojczyk (poprawka)
/ \
* 2797d07 - (5 months ago) Merge branch 'pies10' - Jojczyk
/ \
* 7df1f64 - (5 months ago) zmiana liczby 10 na słowa w pliku pies.txt - Jojczyk (pies10)
/ \
* 056a7df - (5 months ago) Merge branch 'pies9' - Jojczyk
/ \
* 2d4a130 - (5 months ago) zmiana cyfry 9 na słowa w pliku pies.txt - Jojczyk (pies9)
/ \
* e417f15 - (5 months ago) Merge branch 'pies7' - Jojczyk
/ \
* e633209 - (5 months ago) zmiana cyfr 7, 8 na słowa w pliku pies.txt - Jojczyk (pies7)
/ \
* 59a9d43 - (5 months ago) Merge branch 'pies5' - Jojczyk
/ \
* 53e7549 - (5 months ago) zmiana cyfr 5, 6 na słowa w pliku pies.txt - Jojczyk (pies5)
/ \
* e00e089 - (5 months ago) Merge branch 'kot' - Jojczyk
/ \
* ec70242 - (5 months ago) Zmiana małych liter na wielkie w pliku kot.txt - Jojczyk (kot)
/ \
* c8ae65b - (5 months ago) Merge branch 'pies' - Jojczyk
/ \
* 784e69d - (5 months ago) zmiana cyfr 3 i 4 na słowa w pliku pies.txt - Jojczyk (pies)
* 9f2f493 - (5 months ago) zmiana cyfr 1, 2 na słowa w pliku pies.txt - Jojczyk
/ \
* 28d533b - (5 months ago) dodanie pliku pies.txt i kot.txt - Jojczyk
```

25. zmiana starszego komita jest dużo trudniejsza. Polecam zacząć od wątku na stackoverflow <https://stackoverflow.com/a/1186549>  
Przy czym parametr `--no-edit` dodajemy, kiedy nie chcemy zmieniać komentarza.