

| Komenda                                    | Opis   |
|--|--|
| <code>git status</code>                    | Sprawdzanie aktualnej gałęzi na której pracujemy   |
| <code>git checkout branch_1</code>         | Polecenie umożliwiające przełączenie się na inną gałąź np. <code>branch_1</code>   |
| <code>git checkout -b nowy_branch_1</code> | Polecenie umożliwiające stworzenie nowej gałęzi i automatyczne przełączenie się na nią                                     |
| <code>git branch</code>                    | Komenda wyświetlająca wszystkie lokalne gałęzie z repozytorium (przez * oznaczony jest branch na którym obecnie pracujemy) |
| <code>git branch -r</code>                 | Komenda wyświetlająca listę gałęzi znajdujących się na zdalnym repozytorium  |
| <code>git branch -d branch_1</code>        | Usunięcie dowolnej gałęzi np. <code>branch_1</code>  |
| <code>git push -u origin branch_1</code>   | ↑ - To polecenie uruchamiamy przy "wypchnięciu" naszych danych na zdalne repozytorium do wskazanej gałęzi                  |
| <code>git pull origin branch_1</code>      | ↓ - Pobranie zmian ze zdalnego repozytorium z gałęzi <code>branch_1</code>   |
| <code>git merge branch_1</code>            | Scalenie gałęzi <code>branch_1</code> z gałęzią na której obecnie jesteśmy   |
| <code>git rebase branch_1</code>           | Zaciągnięcie zmian z <code>branch_1</code> do aktywnej gałęzi  |
| <code>git pull --rebase</code>             | Pobranie najnowszych zmian z aktywnej zdalnej gałęzi   |
| <code>git stash</code>                     | Zapisanie zmodyfikowanych plików na stosie i usunięcie ich z aktywnej gałęzi   |
| <code>git stash pop</code>                 | Przywrócenie zmodyfikowanych plików z pamięci stosu  |
| <code>git stash clear</code>               | Czyszczenie pamięci stosu  |