

Informacja o git


Instalacja

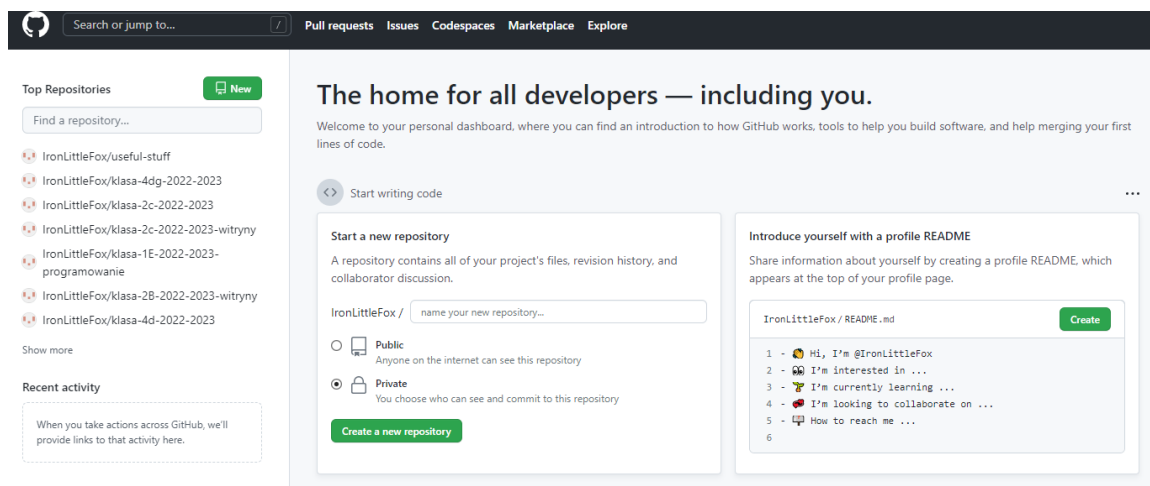
- 1) Należy zainstalować następujące programy
 - a) <https://git-scm.com>
 - b) <https://tortoisegit.org>
- 2) Dla programu "Tortoise git" po instalacji przechodzimy przez konfigurator akceptując domyślne opcje. Należy tylko uzupełnić "Name" i "e-mail"

Zakładanie konta:

- 1) Należy założyć konto na stronie <https://github.com>

Zakładanie repozytorium:

- 1) Po zalogowaniu przechodzimy na stronę główną. Do tego można wykorzystać symbol kotka  znajdującego się w lewym górnym rogu strony.
- 2) W zależności czy jest już repozytorium czy nie strona będzie wyglądać różnie.

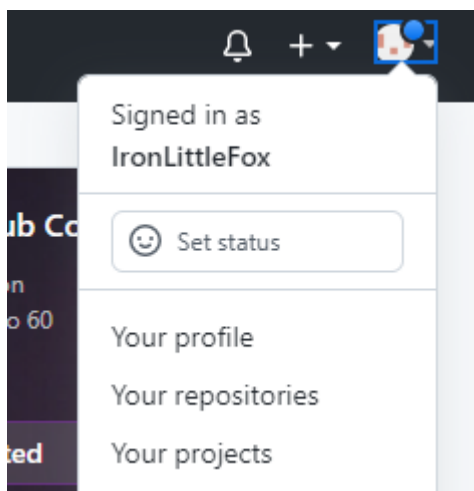


- 3) Klikamy w zielony przycisk "New" lub "Create a new repository" (w zależności czy są już jakieś repozytoria)
- 4) Uzupełniamy pole "Repository name" o nazwę. Zamiast spacji dajemy znak minus np. "Programowanie-1C-grupa-1-2022-2023"
- 5) Ustalamy czy repozytorium ma być publiczne czy prywatne. Aby przeglądać repozytoria prywatne potrzebne są login i hasło do konta.
 1. Repozytoria udostępniane uczniom przez nauczycieli robimy jako publiczne.
 2. Uczniowie mogą zrobić repozytorium prywatne

3. Repozytoria udostępniane nauczycielom przez uczniów do sprawdzenia programu powinny być publiczne.
- 6) Zaznaczamy dodanie pliku README (przyda się)
- 7) Czasami nie wszystkie pliki powinny być trzymane na repozytorium (np. pliki tymczasowe, wyniki kompilacji itp.) można wybrać plik szablon dla pliku .gitignore. Pisząc projekty w Visual Studio można wybrać szablon VisualStudio.
- 8) Licencję nie musimy ustawiać.
- 9) Klikamy w "Create repository"

Wyświetlenie listy repozytoriów

1. Na stronie głównej (po kliknięciu kotka) lista jest z lewej strony
2. Po rozwinięciu listy w prawym górnym rogu przy ikonce profilu jest opcja "Your repositories"



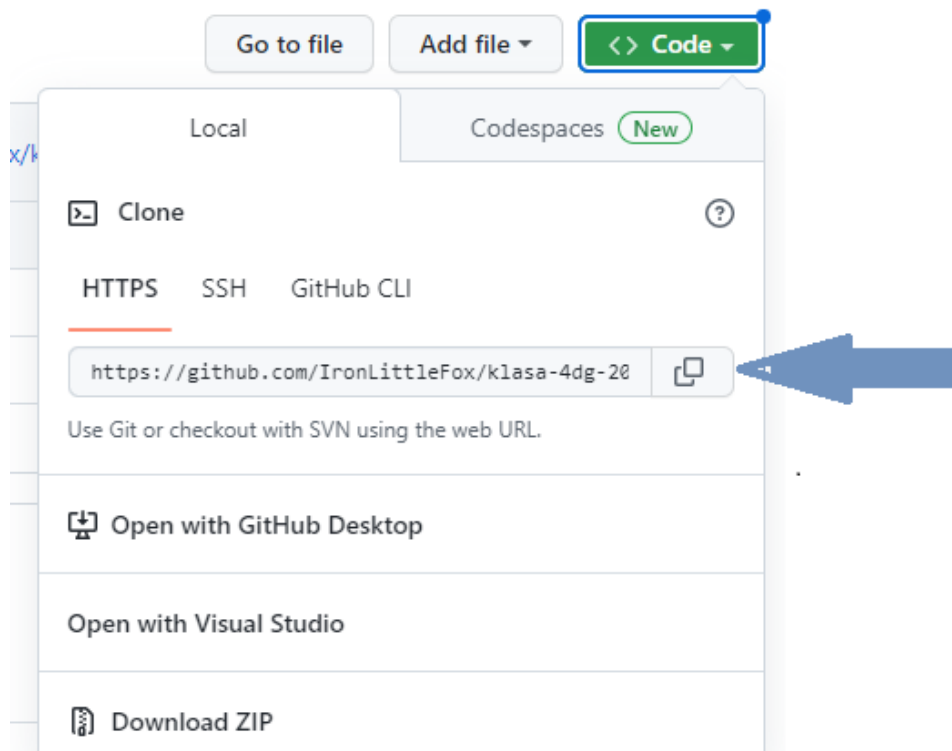
3. Po kliknięciu w link danego repozytorium przechodzimy do strony jego przeglądania/edytowania.

Rozwiązywanie problemów z gitem związanym z zapisanym kontem na github:

- 1) Otwieramy "git bash" i wklejamy i wykonujemy polecenie **git config --system --unset credential.helper**. To polecenie robimy tylko raz.
- 2) W innym przypadku za każdym razem jak zmieniamy konto na inne należy wykonać:
 - a. Należy udać się do menadżera poświadczeń i usunąć zapamiętane konto do serwisu
- 3) Przelogować się lub wylogować w przeglądarce

Klonowanie repozytorium na dysk lokalny

1. Na stronie głównej przeglądania danego repozytorium jest zielony przycisk "Code". Po jego kliknięciu rozwija się menu.



2. Repozytorium można ściągnąć jako spakowany plik ZIP (nie polecam tego rozwiązania ponieważ traci się dużo z możliwości gita.)
3. Klonowanie za pomocą gita (patrz problemy z gitem dla konta):
 - a. Strzałką zaznaczono przycisk do skopiowania linku
 - b. Na dysku tworzymy folder do przetrzymywania repozytoriów
 - c. Będąc w tym katalogu klikamy prawym przyciskiem myszy i wybieramy opcję "Git clone".
 - d. W oknie które się pokaże powinien już być wcześniej skopiowany link. Nie trzeba nic więcej ustawiać i klikamy na "OK".
 - e. W tym katalogu zostanie utworzony katalog z domyślną nazwą repozytorium (chyba że ją zmienimy we wcześniejszym oknie).
 - f. Wszelkie pliki które mają należeć do repozytorium dodajemy je do tego katalogu.

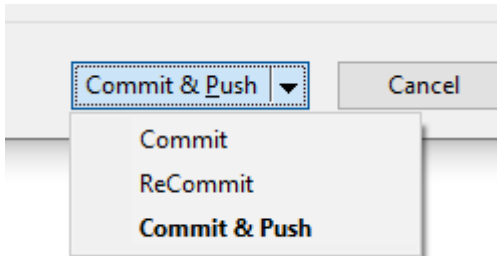
Słownik:

1. Commit – lokalny zapis zmian które zostały wykonane na repozytorium od ostatniego commit'a
2. Push - wypchnięcie commitów na zdalny serwer
3. Pull - zaciągnięcie commitów których nie mamy na lokalnym repo ze zdalnego serwera

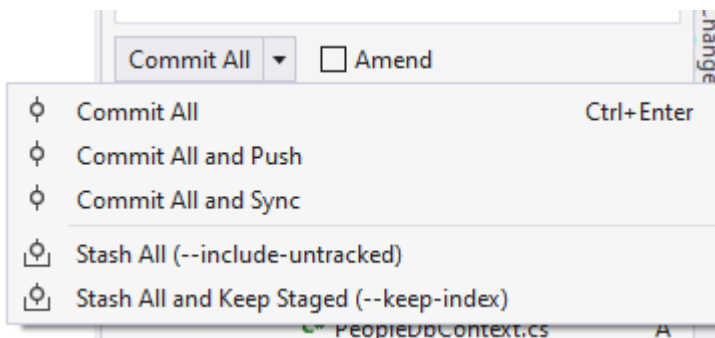
Commit and push:

Jeśli commit się powiedzie ale wypchnięcie się nie uda to patrz "Samodzielne wypychanie commitów"

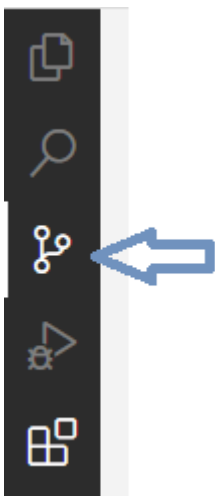
1. W folderze dla konkretnego repozytorium klikamy prawym przyciskiem myszy i wybieramy "Git commit -> main...". W górnej części wpisujemy opis commita, czyli co zostało zrobione (opis ten będzie widoczny w historii). W dolnej części znajduje się lista zmodyfikowanych plików od ostatniego commit'a. Zaznaczamy te które chemy dodać do commita. Jeśli chcemy wszystkie trochę wyżej jest przycisk "All". Na dole znajduje się przycisk z rozwijaną listą. Wybranie "Commit" spowoduje zapis commita lokalnie a "Commit & Push" jednocześnie zapisanie i wypchnięcie wszystkich niewypchniętych commit'ów na zdalny serwer.



2. Visual Studio Community: Z menu wybieramy "View->Git changes". W tym oknie wpisujemy komentarz do commita. Znajduje się tam też przycisk z rozwijaną listą. "Commit all" zapisze zmiany lokalnie, "Commit all and Push" zapisze zmiany i wyśle wszystkie niewysłane commity na zdalny serwer.

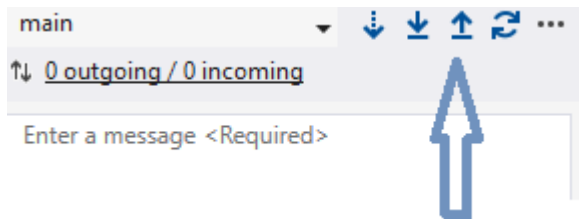


3. Visual Studio Code: Po lewej stronie jest przycisk z grafem/drzewkiem. W tym postępujemy podobnie jak w Visual Studio Community.

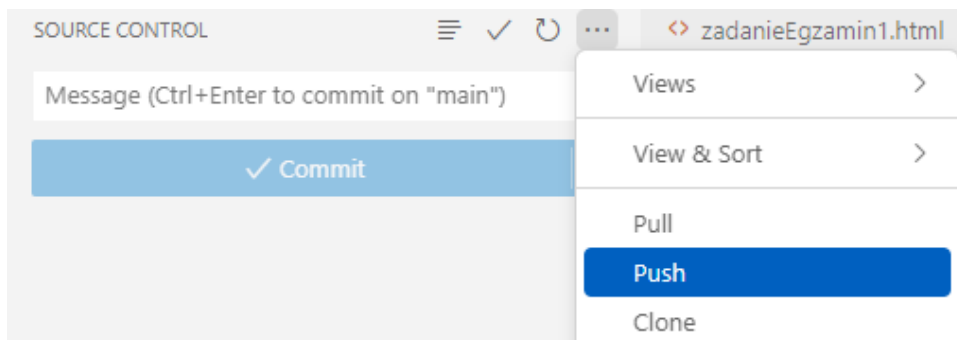


Samodzielne wypychanie commitów:

1. W folderze z repo klikamy prawym przyciskiem myszy i wybieramy "Git sync...". W nowym oknie klikamy na przycisk "Push" aby wypchnąć commit'y.
2. Visual Studio Community: W oknie "Git Changes" klikamy na przycisk ze strzałką do góry

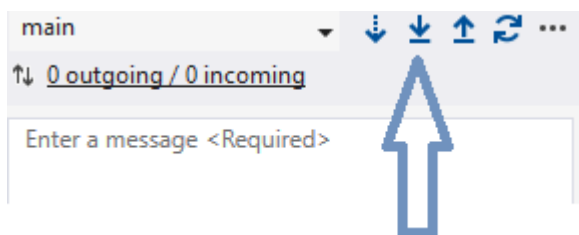


3. Visual Studio Code: W oknie gita wybieramy przycisk z trzema kropkami i "Push"



Pobieranie commitów ze zdalnego serwera:

1. W folderze z repo klikamy prawym przyciskiem myszy i wybieramy "Git sync...". W nowym oknie klikamy na przycisk "Pull" aby wypchnąć commit'y.
2. Visual Studio Community: W oknie "Git Changes" klikamy na przycisk ze strzałką do dołu



3. Visual Studio Code: W oknie gita wybieramy przycisk z trzema kropkami i "Pull"

