

Informacja o git


Instalacja

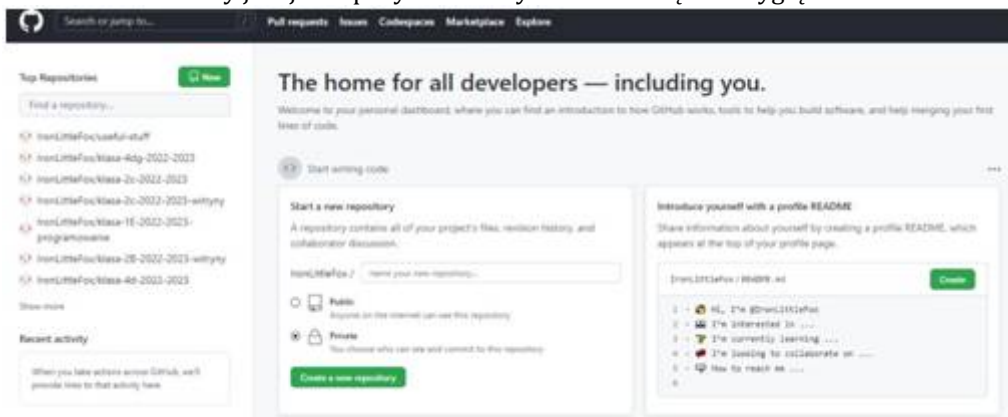
- 1) Należy zainstalować następujące programy
 - a) <https://git-scm.com>
 - b) <https://tortoisegit.org>
- 2) Dla programu "Tortoise git" po instalacji przechodzimy przez konfigurator akceptując domyślne opcje. Należy tylko uzupełnić "Name" i "e-mail" dowolnymi danymi

Zakładanie konta:

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

Zakładanie repozytorium:

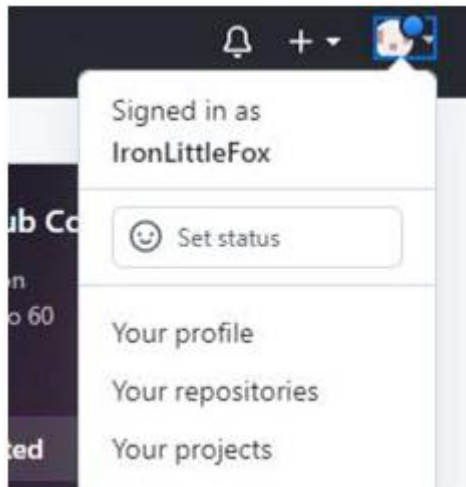
- 1) Po zalogowaniu przechodzimy na stronę główną. Do tego można wykorzystać symbol kotka  znajdującego się w lewym górnym rogu strony.
- 2) W zależności czy jest już repozytorium czy nie strona będzie wyglądać różnie.



- 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.
- 5) "Programowanie-1C-grupa-1-2022-2023". Znak kropka jest też niedozwolony.
- 6) Ustalamy czy repozytorium ma być publiczne czy prywatne. Aby przeglądać repozytoria prywatne potrzebne są login i hasło do konta.
 - a) Repozytoria udostępniane uczniom przez nauczycieli robimy jako publiczne.
 - b) Uczniowie mogą zrobić repozytorium prywatne
 - c) Repozytoria udostępniane nauczycielom przez uczniów do sprawdzenia programu powinny być publiczne.
- 7) Zaznaczamy dodanie pliku README (przyda się)
- 8) 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.
- 9) Licencję nie musimy ustawiać.
- 10) Klikamy w "Create repository"

Wyświetlenie listy repozytoriów

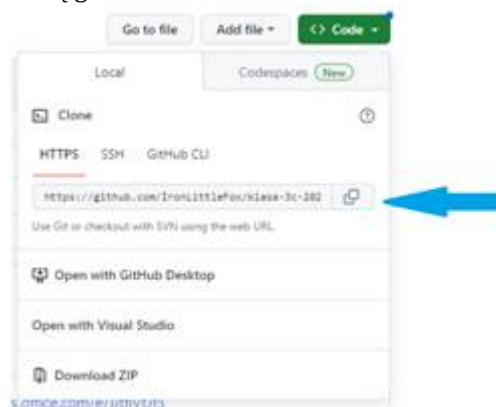
1. Na stronie głównej (po kliknięciu kotka) lista jest z lewej strony
2. Po rozwinięciu listy w prawym górnym rogu przy ikonce profilu jest opcja "Your repositories"



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

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:



- 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 przez git'a, 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 zdanego serwera

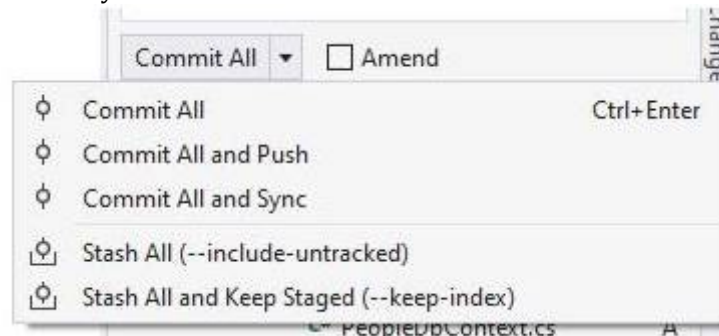
Commit and push:

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

1. **Z folderu:** 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 chcemy 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.

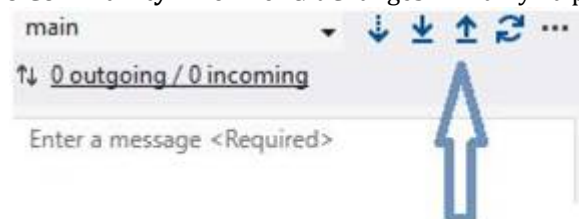


3. **Visual Studio Code:** Po lewej stronie jest przycisk z grafem/drzewkiem. W nim postępujemy podobnie jak w Visual Studio Community.

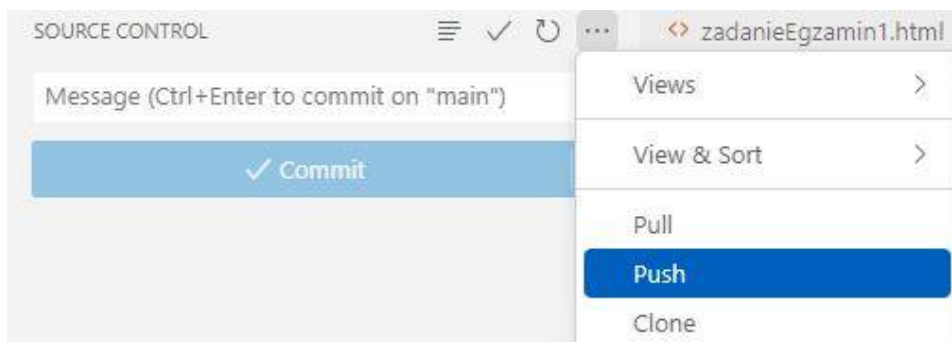
Samodzielne wypychanie commitów:

Commity możemy wypchnąć tylko wtedy jeśli na repo nie ma żadnych niezapisanych zmian.

1. **Z folderu:** 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

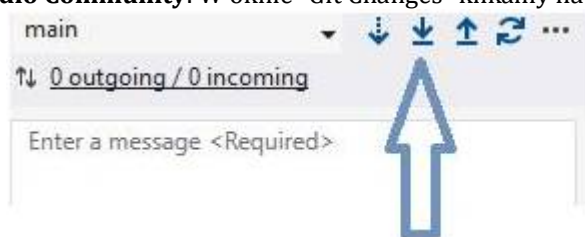


3. **Visual Studio Code:** W oknie gita wybieramy przycisk z trzema kropkami i "Push"

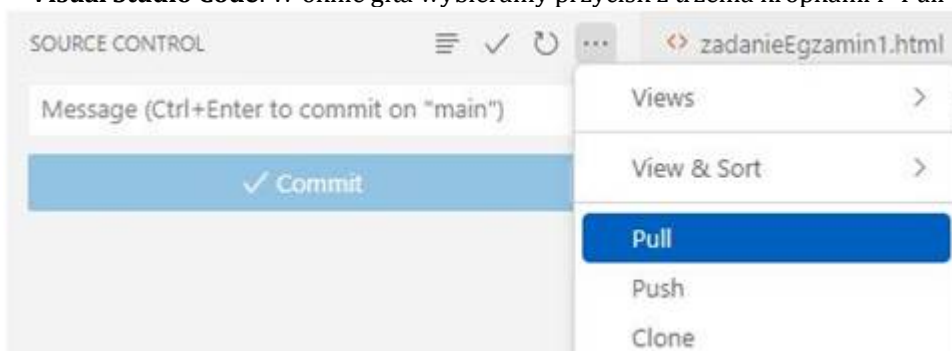


Pobieranie commitów ze zdalnego serwera:

1. **Z folderu :** 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"



Czynności na lekcji

1. Otwarcie przeglądark Microsoft Edge, Google Chrome i wejście na stronę github.com i wylogowanie się lub można od razu zalogować się ucznia na swoje konto.
2. Otwarcie z menu start „Menadżer poświadczeń” (wystarczy wpisać „poś”). Z poświadczeń systemu Windows należy usunąć wpisy do strony github.com
3. Wykonanie operacji „pull” – patrz „pobieranie commitów ze zdalnego serwera”
4. Praca na zajęciach z repo
5. Pod koniec lekcji zapisanie commita i wypchnięcie na zdalny serwer – patrz „samodzielne wypychanie commitów”
6. Powtórzenie punktów 1 i 2.

Uwaga:

W domu nie trzeba wykonywać czynności z punktów 1, 2 i 6.

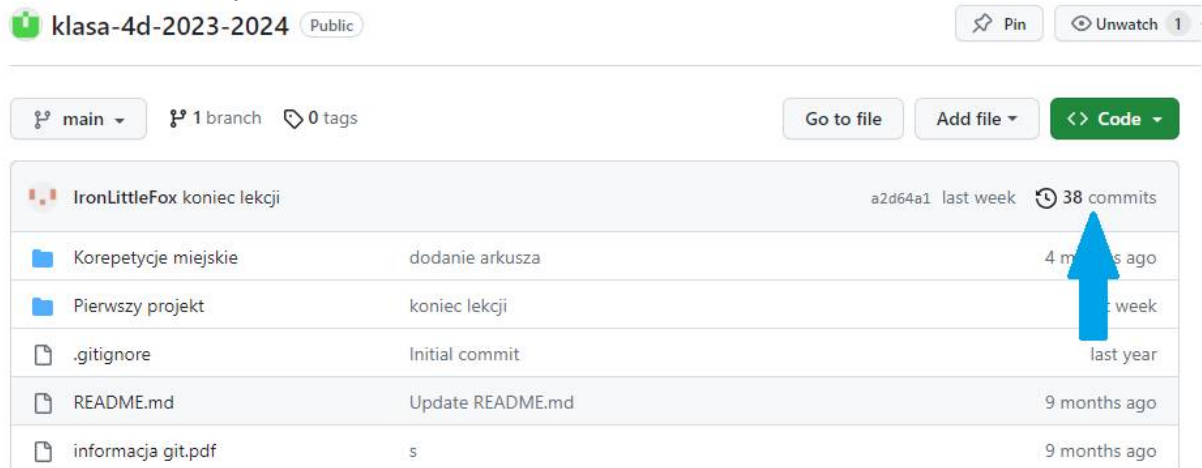
Uwaga:

Wykonanie powyższych czynności za każdym razem uchroni uczniów z problemami kolizji commitów.

Uzupełnianie braków przez ucznia w jego repozytorium

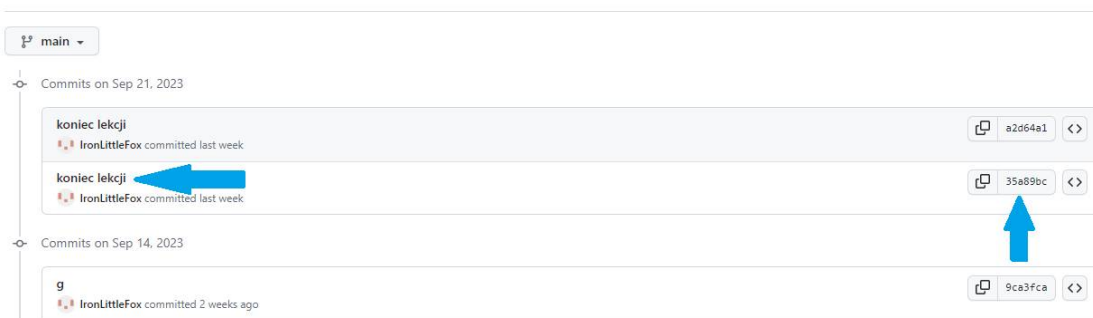
Podany poniżej opis dotyczy sytuacji gdzie nauczyciel ma utworzone repozytorium dla klasy i po zakończeniu zajęć danego dnia do niego wysyła wszystkie materiały które zrobił na zajęciach.

1. Pamiętaj o wykonaniu „Czynności na lekcji” aby repo lokalne było zsynchronizowane z github’em.
2. Należy wejść na repozytorium z którego będziemy pobierać braki.
3. Przechodzimy do historii commitów.



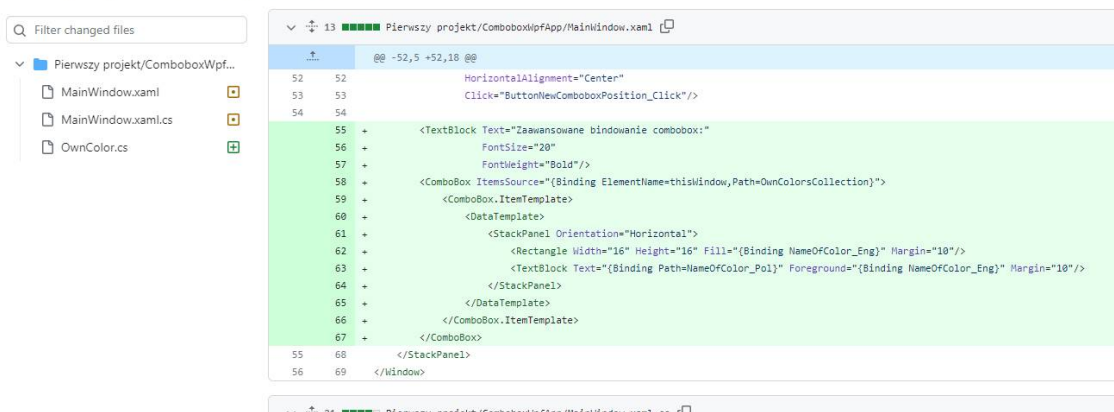
4. W historii sprawdzamy po datach kiedy ucznia nie było w szkole na zajęciach i których commitów brakuje. Jeśli braki musimy uzupełnić z dwóch lub więcej commitów zaczynamy od tego który jest niżej na liście.

5. Wchodzimy w commit klikając na opis lub przycisk z numerem.



6. Pojawi się strona ze zmianami zapisanymi w commicie.

Showing 3 changed files with 47 additions and 1 deletion.



7. Z lewej strony jest drzewko z plikami które zostały zmodyfikowane/dodane/usunięte. Symbol plusa w kwadracie oznacza że plik został dodany, symbol kropki w kwadracie, że został zmodyfikowany.

8. Po prawej stronie znajduje się podgląd pliku lub wszystkich plików jeśli dopiero co została załadowana strona. Oznaczenie kolorów linijek w plikach:
 - a. Zielony – linijka dodana
 - b. Czerwony – linijka usunięta
9. Należy już samodzielnie za pomocą kopiowania przenieść różnice do swoich plików w lokalnym repozytorium. W Visual Studio Community nowe pliki należy dodawać poprzez prawy przycisk myszy na nazwie projektu lub folderu projekcie i wybranie odpowiedniej pozycji z Add.
10. Należy powtórzyć powyższe punkty w celu przeniesienia różnic z kolejnych commitów.

UWAGA!!:

Plików z rozszerzeniem

- a) *.sln
- b) *.vcxproj,
- c) *.vcxproj.filters
- d) *.vcxproj.user
- e) *.csproj

nie należy modyfikować samodzielnie.

Uwaga:

Jeśli jest dużo plików które są nowe, to można przyspieszyć pracę klonując repozytorium bazowe i samodzielnie kopiować całe pliki/katalogi do lokalnego repozytorium (należy pamiętać o odtworzeniu również ścieżek podfolderów przy kopiowaniu samych).

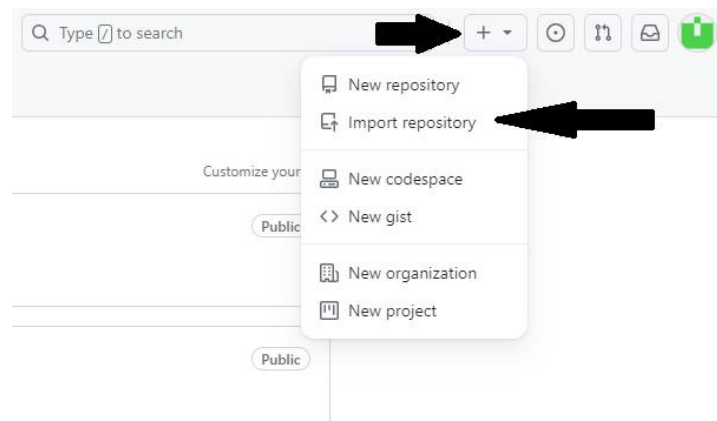
Uwaga (dotyczy Visual Studio Community):

1. Jeśli do przeniesienia jest cały nowy projekt to należy jego folder skopiować z repozytorium bazowego do docelowego. Następnie po załadowaniu solucji w VS klikamy prawym przyciskiem myszy albo na nazwę solucji albo folderu solucji (w zależności gdzie ten projekt ma się znaleźć) i wybieramy Add->Existing Project. W otwartym oknie przechodzimy w repozytorium docelowym do folderu z projektem i wskazujemy plik z nazwą projektu. Filtrowanie powinno pokazać tylko jeden plik.
2. Pojedyncze pliki w projekcie: kopiujemy pliki z zachowaniem podfolderów z repozytorium bazowego do docelowego.
 - a. Projekt C++ : w VS klikamy prawym przyciskiem myszy na filtrze do którego ma być przypisany plik i wybieramy Add->Existing Item. Wskazujemy pliki do dodania.
 - b. Projekt C#: nic dodatkowego nie musimy robić. Pliki lub foldery z plikami pojawią się w projekcie.

Przeniesienie repozytorium z konta na konto

Przenoszenie można zrobić na dwa sposoby:

1. Należy założyć nowe repozytorium lub skorzystać z istniejącego. Klonujemy repozytorium źródłowe i docelowe. Z repozytorium lokalnego źródłowego kopiujemy interesującą zawartość i wykonujemy operację commit i push.
2. Jeśli jednak interesuje nas kopia całego repozytorium (oraz dodatkowo z wszystkimi commitami, które były wykonane na repozytorium źródłowym) to należy wybrać pozycję zgodnie z poniższym zdjęciem



3. Na otwartej stronie wklejamy link do repozytorium źródłowego, nadajemy nazwę nowemu repozytorium, wybieramy widoczność i klikamy na przycisk „Begin import”.