**1. Ręczne przesyłanie plików na GitHuba**

* Wejdź na [GitHub](https://github.com/).
* Stwórz nowe repozytorium.
* Prześlij pliki z Obsidiana za pomocą opcji **"Upload files"**.

**2. Użycie Git do synchronizacji plików**

Jeśli chcesz regularnie synchronizować notatki, najlepiej użyć Git. Możesz to zrobić tak:

**Krok 1: Inicjalizacja repozytorium**

W terminalu (w folderze, gdzie masz notatki Obsidiana):

git init

git add .

git commit -m "Initial commit"

**Krok 2: Połączenie z GitHubem**

Stwórz repozytorium na GitHubie i skopiuj jego adres (np. https://github.com/user/my-obsidian-vault.git).

Potem w terminalu:

git remote add origin https://github.com/user/my-obsidian-vault.git

git branch -M main

git push -u origin main

**Krok 3: Pobieranie na innym komputerze**

Na nowym komputerze:

git clone https://github.com/user/my-obsidian-vault.git

### ****Automatyczna synchronizacja****

Jeśli chcesz regularnie aktualizować pliki, możesz użyć:

git add . - Dodanie zmian

git commit -m "Updated notes" - Zatwierdzenie (commit), "Krótki opis zmian"

git push - Wysłanie na GitHuba

A na drugim komputerze:

git pull

### git remote –v - sprawdzenie, do jakiego repozytorium jesteś podłączony

### git status - jeśli wcześniej wykonałeś git add ., zobaczysz listę plików, które są gotowe do zatwierdzenia (Changes to be committed). (s prawdzenie, jakie pliki zostały dodane do staging area)

**Komentarz**  
git add .

dodaje wszystkie zmienione, nowe i usunięte pliki z bieżącego katalogu i jego podkatalogów do staging area (obszaru indeksowania) w lokalnym repozytorium Git. Polecenie git add . nie wysyła plików do zdalnego repozytorium (np. na GitHub).

**Co dokładnie się stało?**

1. **Dodano wszystkie nowe pliki** (które wcześniej nie były śledzone przez Git).
2. **Zaktualizowano zmodyfikowane pliki** (jeśli były już śledzone).
3. **Oznaczono usunięte pliki** jako przeznaczone do usunięcia w następnym commicie.

**Do jakiego repozytorium?**

* **Lokalnego**, czyli tego, w którym znajduje się katalog .git.
* Możesz sprawdzić, w którym repozytorium jesteś, wpisując:

git rev-parse --show-toplevel

To pokaże ścieżkę do głównego katalogu repozytorium.

git remote –v - sprawdza czy czy repozytorium jest powiązane z GitHubem

 git remote add origin ... – dodaje repozytorium z GitHuba jako „origin”

 git branch -M main – upewnia się, że lokalna gałąź to main

 git push -u origin main – wysyła pliki na GitHuba