

Hakersi:

Wstęp do git-a

Czym jest git?

Czy kiedykolwiek przypadkowo usunąłeś plik lub wprowadziłeś zmiany do pliku, których później żałowałeś?

Czy kiedykolwiek musiałeś pracować nad projektem z innymi ludźmi i śledzić, kto wprowadzał jakie zmiany i kiedy?

Czym jest git?

Git to system kontroli wersji, który pomaga programistom śledzić zmiany w kodzie i pracować w grupie.

Dzięki Gitowi programiści mogą pracować jednocześnie nad tym samym kodem i łatwo integrować swoje zmiany.

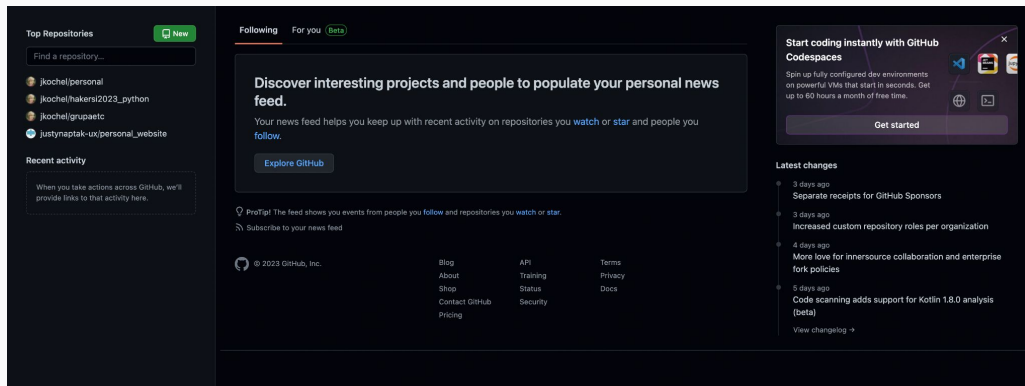
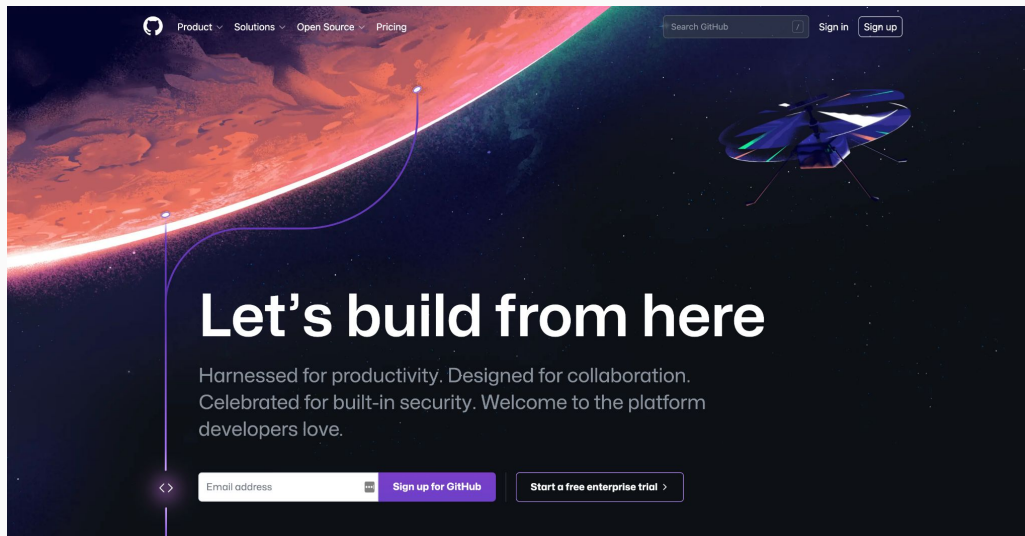
Git jest również bardzo popularnym narzędziem do udostępniania kodu open-source.

Git - podstawowe pojęcia

1. **Repozytorium (repository)** - miejsce, w którym przechowywany jest kod oraz historia jego zmian.
2. **Commit** - zapisanie zmian w repozytorium. Commit zawiera opis zmian, datę oraz autora zmian.
3. **Branch** - gałąź w repozytorium, która umożliwia niezależną pracę nad kodem i łączenie zmian.
4. **Merge** - połączenie dwóch gałęzi w repozytorium.
5. **Pull request / Merge request** - prośba o zaakceptowanie zmian dokonanych w gałęzi przez innego programistę.

Git: repozytorium

1. Załóż konto na GitHubie, jeśli jeszcze go nie masz.
2. Utwórz nowe repozytorium na GitHubie (swoje prywatne).



Git: instalacja

1. Zainstaluj Git na swoim komputerze.

Instalacja Git na **Linuxie**

1. **Debian/Ubuntu:**

```
sudo apt-get update && sudo apt-get install git
```

2. **Fedora:**

```
sudo dnf install git
```

3. **CentOS/RHEL:**

```
sudo yum install git
```

Instalacja Git na Windowsie

1. Przejdź do <https://git-scm.com/download/win>
2. Pobierz wersję 32-bitową lub 64-bitową, w zależności od systemu
3. Uruchom instalator i postępuj zgodnie z instrukcjami. Domyślne opcje powinny być odpowiednie dla większości użytkowników.

Git: podstawowe komendy

1. Zapoznaj się z podstawowymi komendami.

`git clone` # Ta komenda klonuje repozytorium z GitHuba.

`git init` # Ta komenda inicjuje nowe repozytorium Git.

`git add` # Ta komenda dodaje pliki do obszaru staging, co oznacza, że są gotowe do zatwierdzenia.

`git commit` # Ta komenda tworzy nowy commit ze zmianami dodanymi do obszaru staging.

`git push` # Ta komenda uploaduje zmiany do repozytorium online

`git pull` # Ta komenda downloaduje zmiany z repozytorium online

`git status` # Ta komenda pokazuje bieżący stan twojego repozytorium, w tym jakie pliki zostały zmienione i które są gotowe do zatwierdzenia.

`git branch` # Ta komenda pokazuje gałęzie (branch) w twoim repozytorium i umożliwia tworzenie nowych gałęzi.

`git switch` # Ta komenda zmienia twoją lokalną branch

`git merge` # Ta komenda łączy zmiany z jednej gałęzi z inną.

Git: klonowanie repozytorium

1. Zostań dodany do repozytorium Hakersów.
2. Pobierz repozytorium na swój komputer.
3. Stwórz nową lokalną branch
4. Push lokalną branch do repozytorium
5. Przełącz się na nową branch

```
git clone git@github.com:jkochel/hakersi2023_python.git  
  
git branch twoja_nazwa  
  
git push origin twoja_nazwa  
  
git switch twoja_nazwa
```


Code review - co to i do czego służy

Przegląd kodu jest ważnym procesem, który może pomóc poprawić jakość kodu i wyłapać potencjalne problemy przed wprowadzeniem ich do produkcji.

Code review - ważne elementy

1. Ustal **jasne wytyczne**: **kto** powinien być zaangażowany, jakie są **kryteria** przeglądu i jak powinna być udzielana opinia.
2. Ustal **oczekiwania**: wszyscy rozumieją, **jak szybko** powinni odpowiadać, **ile czasu** powinni poświęcić i **jakiego rodzaju opinie** są najbardziej wartościowe.
3. Bądź **szanujący i konstruktywny**: przeglądy kodu mają na **celu poprawę jakości kodu, a nie krytykowanie programisty**, który go napisał. Opinie powinny skupiać się na sposobach poprawy kodu, a nie wskazywaniu błędów.
4. Skup się na **szerszym kontekście**: wyłapywanie małych błędów i literówek jest ważne, ale trzymanie się **szerszego kontekstu** jest ważniejsze. Przeglądy kodu powinny skupiać się na ogólnym projekcie i architekturze kodu, a także na potencjalnych problemach z wydajnością i bezpieczeństwem.

Code review - praktyka

1. Używaj do tego GitHuba.
2. Dobrze jest mieć kanał komunikacji do proszenia o code review.
3. Osoba która otwiera wątki powinna je zamknąć.

