A T AKADEMIA TARNOWSKA	Wydział Politechniczny Katedra Informatyki
Prowadzący	Tomasz Gądek
Kurs	Narzędzia i środowiska programistyczne
Rok / Semestr	1 / Letni
Temat	Lab06: Git, Bitbucket.

Data ostatniej modyfikacji: 01-10-2024

© Tomasz Gądek | Katedra Informatyki AT



Git to wolnodostępny, rozproszony system kontroli wersji. Stworzony przez **Linusa Torvaldsa** podczas prac nad jądrem systemu operacyjnego **Linux**.

Do przechowywania kodów źródłowych projektów i kontroli wersji wykorzystywane jest repozytorium Git. Repozytoria Git hostowane są w serwisie **Bitbucket**.

Bitbucket



Bitbucket to platforma online, która umożliwia korzystanie z **Git** w najprostszy możliwy sposób. Interfejs użytkownika jest łatwy i przystępny. Poniższy screen przedstawia listę **commitów**.

• • • Transfeusz / MySchool / Commit x +									
$\leftarrow \ \rightarrow $	CÓ	Atlassian, Inc. [US]	https://bitbucket.or	rg/morfeusz/myschoo	ol/commits/all 🖄 🖧 🖉 🕻	🕑 🛛 🛛 🛛 🕑	ymano 侧		
Aplik	acje 🗎	prir 🗎 sata 🗎 java 🗎	jsp 📄 pc 📄 c4	++ 🗎 youtube 🗎	english 🗎 www 🛅 altconnect 🗎 javaFX 🗋 Polemiki 🗽 latex 🌞	wolfram			
v	Java	MySchool	Tomasz Gadek /	MySchool					
Q	\diamond	Source	Committe						
+	¢	Commits	All branches	•		Q Find comm	its		
	ţ	Branches	Author	Commit	Message	Date	Builds		
	î٦	Pull requests	• 🧳 Tomasz G	adek 5abe5d9	usuniecie klasy	2017-02-27			
	0	Bipolines	• 🧳 Tomasz G	adek 2fbb190	lista klas poprawki, rozszerzenie encji user	2017-02-26			
	0	ripennes	• 🧳 Tomasz G	adek 7388104	lista klas + poprawki w uslugach i projekcie	2017-02-25			
	C _f 2	Deployments	• 🥞 Tomasz G	adek 9ca96b9	przygotowanie podstrony dla dziennika, dopracowanie wylogowania	2017-02-22			
	-	Wiki	• 🧳 Tomasz G	adek 48f986d	wystawienie przykladowych usług API (dodawanie, tworzenie i aktual	2017-02-17			
	F	Downloads	• 🥞 Tomasz G	adek a408ab0	dodanie mapperow, przebudowa strutury projektu, przygotowanie us	2017-02-12			
		Boards	• 🥝 Tomasz G	adek a3d6751	podpiecie angulara, dodanie zabezpieczen, przebudowa struktury pr	2017-02-12			
	ö	Settings	• 🥝 Tomasz G	adek 44d55be	poprawa strony logowania, implementacja strony glownej	2017-02-01			
			• 🥞 Tomasz G	adek c05b9b0	przygotowanie hibernate, porzadkiw projekcie, dostosowanie bootstr	2017-01-29			
~			🔹 🤪 Tomasz G	adek 02f3372	test komentarza	2016-11-13			
?			🔹 🎱 Tomasz G	adek 802d976	dodanie bazy danych	2016-10-08			
ø			🔹 🤪 Tomasz G	adek 5f218e1	dodanie encji uzytkownika	2016-09-20			
			Tamaaa C	adaly 021-the	nanzawa akzanu langunania amiana watawian ianukawuah	2016 00 07			

Git

System kontroli wersji - komu to potrzebne?

Systemy kontroli wersji wspierają:

- komunikację między programistami,
- zarządzanie wydaniami projektu,
- śledzenie błędów,
- sprawne łączenie tworzonego kodu,
- tworzenie niezależnych wersji,
- korzystanie z wcześniejszych wersji danego pliku.

Instalacja

Na zajęciach nic nie musisz instalować. Git jest gotowy do działania.

https://git-scm.com/downloads



Dostęp z poziomu terminala

- Windows: win + r, komenda cmd,
- Linux: ctrl + alt + t,
- Mac OSX: cmd + spacja, wyszukaj: terminal.



Konfiguracja



Repozytorium

Jest to miejsce przechowywania zmian.



Przepływ pracy

Twoje lokalne repozytorium będzie składało się z trzech **stanów** zarządzanych przez Git. Pierwszy to **Katalog Roboczy**, który przechowuje bieżące pliki. Drugi to **Index**, który działa jak poczekalnia i ostatni z nich to **HEAD**, który wskazuje na ostatni utworzony **commit**.



Przygotowanie środowiska pracy

- Otwórz IntelliJ IDEA oraz utwórz projekt nisp_06.
- Utwórz funkcję main(), który wyprowadzi na wyjście tekst Hello World.
- Załóż konto na platformie Bitbucket.
- Następnie otwórz terminal w IntelliJ IDEA. Uzyskasz do niego dostęp z poziomu menu: View / Tool Windows / Terminal.
- Przejdź do zadań i postępuj zgodnie z krokami.

Tutorial

Krok 1. Tworzenie lokalnego repozytorium.

Wpisz następujące polecenie.

git init

W odpowiedzi otrzymasz informację, że puste repozytorium zostało zainicjowane:

```
(base) tomaszgadek@MacBook-Pro-Tomasz-2 nisp_06 % git init
Zainicjowano puste repozytorium Gita w /Users/tomaszgadek/Desktop/lecture_tijo/nisp_06/.git/
(base) tomaszgadek@MacBook-Pro-Tomasz-2 nisp_06 %
```

W tym kroku wykonaj jeszcze jedną komendę, która informuje o stanie plików w repozytorium:

```
git status
```

Oto rezultat działania komendy.

```
(base) tomaszgadek@MacBook-Pro-Tomasz-2 nisp_06 %
(base) tomaszgadek@MacBook-Pro-Tomasz-2 nisp_06 % git status
Na gałęzi master
Jeszcze nie ma zapisów
Nieśledzone pliki:
  (użyj "git add <plik>...", żeby uwzględnić, co zostanie złożone)
        .idea/
        nisp_06.iml
        out/
        src/
```

nie dodano nic do złożenia, ale są nieśledzone pliki (użyj "git add", aby śledzić)

Pliki jeszcze nie są śledzone. Znajdujesz się w stanie **working dir** (rozdział **Git - przepływ pracy**). Kolor plików sugeruje w jakim stanie się znajdują.

Krok 2. Dodawanie plików do poczekalni.

Za dodanie plików do poczekalni (index) odpowiadają poniższe komendy.

Dodanie konkretnego pliku (przykład):

git add nisp_06.iml

Ponownie zweryfikuj stan plików w repozytorium:



Rezultat działania komendy jest następujący. Zwróć uwagę, że tylko jeden plik został dodany do poczekalni.

Dobrze. W takim razie dodajmy wszystkie pliki i katalogi do poczekalni. Skorzystaj z następującej komendy.

git add .

Jeszcze raz sprawdź status:

Oto rezultat.

(base) tomaszgadek@MacBook-Pro-Tomasz-2 nisp_06 % git status Na gałęzi master

Jeszcze nie ma zapisów

Zmiany do złożenia:

(użyj	"git	rmcached	<plik>", aby wycofać)</plik>
	nowy	plik:	.idea/.gitignore
	nowy	plik:	.idea/inspectionProfiles/Project_Default.xml
	nowy	plik:	.idea/kotlinc.xml
	nowy	plik:	.idea/libraries/KotlinJavaRuntime.xml
	nowy	plik:	.idea/misc.xml
	nowy	plik:	.idea/modules.xml
	nowy	plik:	nisp_06.iml
	nowy	plik:	<pre>out/production/nisp_06/META-INF/nisp_06.kotlin_module</pre>
	nowy	plik:	out/production/nisp_06/MainKt.class
	nowy	plik:	src/main/kotlin/Main.kt

Nazwy plików powinny zmienić kolor na zielony (rozdział Git - przepływ pracy).

Krok 3. Zapisywanie zmian w lokalnym repozytorium.

Aby zatwierdzić zmiany wykonaj poniższe polecenie.

git commit -m "utworzono nowy projekt"

Rezultat jest następujący.

```
(base) tomaszgadek@MacBook-Pro-Tomasz-2 nisp_06 % git commit -m "utworzono nowy projekt"
[master (zapis-korzeń) 60fb023] utworzono nowy projekt
10 files changed, 82 insertions(+)
create mode 100644 .idea/.gitignore
create mode 100644 .idea/inspectionProfiles/Project_Default.xml
create mode 100644 .idea/kotlinc.xml
create mode 100644 .idea/libraries/KotlinJavaRuntime.xml
create mode 100644 .idea/misc.xml
create mode 100644 .idea/modules.xml
create mode 100644 .idea/modules.xml
create mode 100644 nisp_06.iml
create mode 100644 out/production/nisp_06/META-INF/nisp_06.kotlin_module
create mode 100644 out/production/nisp_06/MainKt.class
create mode 100644 src/main/kotlin/Main.kt
(base) tomaszgadek@MacBook-Pro-Tomasz-2 nisp_06 %
```

Zmiany zostaną zatwierdzone i zostanie dodany stosowny komentarz: **utworzono nowy projekt**.

Ważne! Komunikat powinien opisywać dokładnie to co robisz. Jeżeli dodajesz nową funkcjonalność to zmień komentarz przy kolejnym zatwierdzaniu zmian!

Wykonaj ponownie komendę git status:

git status

Wydaje się, że już wszystko jest gotowe. Aby nasza praca miała sens to musimy połączyć repozytorium lokalne z repozytorium zdalnym i wysłać zmiany. Przejdź do kolejnego kroku.

Krok 4. Połączenie repozytorium lokalnego ze zdalnym.

Na platformie **Bitbucket** proszę utworzyć nowe repozytorium **nisp_06** (przejdź do **Repositories** i kliknij w button **Create repository**).

		import repositor
Workspace	🗃 tgadek	
Project name*	nisp	0
Repository name*	nisp_06	
Access level	 Private repository Uncheck to make this repository putypically contain open-source code anyone. 	ublic. Public repositories and can be viewed by
Include a README?	No	~
Default branch	master	
name	No	~
Include .gitignore?		
Hame	No	~

Bardzo proszę wypełnić formularz zgodnie z poniższym screenem.

Po utworzeniu zdalnego repozytorium skopiuj komendę i wklej ją w takiej postaci do terminala (prawdopodobnie Twoja ścieżka będzie się różniła od mojej - to jest tylko screen poglądowy):

				Let's put	some	bits in	your b	ucke	t		
		HTTPS	•	git clone https:	//tgadek@bit	tbucket.org/t	gadek/nisp_0	6.git		ſ	
Get	star	ted qu	uick	ly							
Creatin	ig a RE	ADME or	r a .git	ignore is a quick	and easy wa	y to get some	ething into yo	ur reposi	tory.		
Creat	te a RE	EADME	Cre	ate a .gitignore				_			
Get y Step 1:	YOUI Switcl	f local h to your	Git repos	Zazna repository	on Bitbu	isko ucket	piuj (/	do '	iern	ninala	
1	cd /p	ath/to/y	your/ı	геро							
Step 2:	Conn	ect your	existir	g repository to B	itbucket						
1	git r	emote ad	dd or:	igin https://tg	adek@bitbuc	cket.org/tga	adek/nisp_06	.git			
2	git p	ush —u d	origin	n master							
Need m	nore in	formatio	n? <mark>Le</mark> a	rn more							

Twoje lokalne repozytorium zostało właśnie połączone ze zdalnym repozytorium. Nie pozostało Nam nic innego jak wypchnięcie (wysłanie) zmian na serwer.

Krok 5. Wysłanie zmian na repozytorium zdalne.

Aby wypchnąć zmiany na repozytorium zdalne wykonaj polecenie oraz podaj potrzebne dane do logowania:

```
git push origin master
```

Po wykonaniu polecenia możesz zostać poproszony (poproszona) o wygenerowanie tymczasowego hasła na potrzeby wykonywania zmian na repozytorium. Skorzystaj z tutoriala.

Przejdź na aplatformę **Bitbucket**. Twoje pliki, kod oraz historia commitów znajdują się w repozytorium zdalnym. Przejdź do menu: **Source** i **Commits**. Przeanalizuj dane.

Widok Source

e tgade	ek / nisp 06								Invite	Clone	••••
Here's w	posit	tory's source	files. To g	ive you	ır users an io	dea of what they'll find here, add a description to your repository.					
្ងៃ ma	aster 🗸	Files	~	Filter files		Q					
Name					Size	Last	commit	Message			
.id	lea					28 m	inutes ago	utworzono nowy projekt			
our	ıt					28 m	inutes ago	utworzono nowy projekt			
src	с					28 m	inutes ago	utworzono nowy projekt			
🗗 nis	sp_06.iml				778 B	28 m	inutes ago	utworzono nowy projekt			

Widok Commits

∎ Co	tgadek / nisp / nisp_06				Compare	Clone
Se	arch commits Q	کی All branches م				
	Author	Commit	Message	Date		
•	Tomasz Gądek	60fb023	utworzono nowy projekt	28 minutes ago		

Zadania

Twoje zadanie będzie polegało na implementacji prostego kalkulatora (Utwórz plik oraz klasę **Calculator**). Klasa kalkulator powinna zawierać **konstruktor** (przyjmuje dwie liczby typu **Double**) oraz 4 metody, które będę wykonywały podstawowe operacje na danych (liczbach zmiennoprzecinkowych):

- konstruktor (po implementacji wykonaj polecenia z kroku 2, 3 i 5),
- dodawanie funkcja sum() (po implementacji wykonaj polecenia z kroku 2, 3 i 5),
- odejmowanie funkcja sub() (po implementacji wykonaj polecenia z kroku 2, 3 i 5),
- mnożenie funkcja mul() (po implementacji wykonaj polecenia z kroku 2, 3 i 5),
- dzielenie funkcja **div()** (po implementacji wykonaj polecenia z kroku 2, 3 i 5).

Po implementacji każdej funkcjonalności wykonuj polecenia z kroku 2, 3 i 5 (w kroku 5 dostosuj **message**, nie kopiuj za każdym razem **utworzono nowy projekt**).

W rezultacie repozytorium powinno zawierać 6 commitów!

Tak wyglądają moje commity. Historia commitów Twojego repozytorium powinna wyglądać podobnie.

Г Сс	tgadek SMM	: / nisp / nisp_06 its				Compare	Clone
Se	arch co	ommits Q	່ໃງ All branches 🗸				
	Autho	or	Commit	Message	Date		
1		Tomasz Gądek	80e58d2	dodano metode div()	2 minutes ago		
+		Tomasz Gądek	cd54e3e	dodano metode mul()	3 minutes ago		
+		Tomasz Gądek	9f61907	dodano metode sub()	3 minutes ago		
+		Tomasz Gądek	981a271	dodano metode sum()	4 minutes ago		
+		Tomasz Gądek	ebbd351	dodano konstruktor	5 minutes ago		
•		Tomasz Gądek	60fb023	utworzono nowy projekt	49 minutes ago		

Dodatkowo możesz zweryfikować historię commitów posługując się następującymi poleceniami.

```
git log
```

git log --oneline

- Aby przeglądać kolejne linie wciśnij ENTER.
- Aby wyjść naciśnij q.

Czy wiesz, że...



Język Python został stworzony przez Guido van Rossuma i po raz pierwszy wydany w 1991 roku. Nazwa języka nie pochodzi od węża, ale od popularnego brytyjskiego serialu komediowego **Monty Python's Flying Circus**.