A T AKADEMIA TARNOWSKA	Wydział Politechniczny Katedra Informatyki
Prowadzący	Tomasz Gądek
Kurs	Narzędzia i środowiska programistyczne
Rok / Semestr	1 / Letni
Temat	Lab08: Git, praca zespołowa.

Data ostatniej modyfikacji: 01-10-2024

© Tomasz Gądek | Katedra Informatyki AT

## Zespół

Przed przystąpieniem do ćwiczeń należy dobrać się w dwuosobowe zespoły.

## Student-A

Utwórz projekt w języku Kotlin posługując się zintegrowanym środowiskiem programistycznym **IntelliJ IDEA**.

Utwórz startowy projekt **nisp\_08** zawierający funkcję **main()** w języku Kotlin oraz prawidłowy plik **.gitignore** (plik **README.md** jest niepotrzebny). Nie zapomnij dodać nowego zdalnego repozytorium. Postępuj zgodnie z krokami opisanymi w **laboratorium 7**.

Zanim rozpoczniesz pracę utwórz branch (gałąź) develop. Następnie wykonaj polecenia, aby zsynchronizować dane na platformie **Bitbucket** (zsynchronizuj branch **develop** i **master**). Na zdalnym repozytorium powinny być już dostępne gałęzie develop i master.

III 🗾 Bitbucket Your work Pull requests Repositories Projects More 🗸 C	eate 👻	Q Search		o	K				
nisp_08	studie25 / risp. / risp.08 / Repository settings     Repository permissions     Add users or groups								
Go Back	Repository permissions allow one to extend access beyond that already granted via project permissions. In order to access the workspace, uses must have product access.								
General Repository details	Setto Q Permissions V Access level V								
Repository permissions	Name Permission Access level Actions								
Username aliases	C K) Admin • Repository Remove								
Security Access keys	Admin Workspace								
Access tokens									
Workflow									
Branch restrictions Branching model									

#### Kliknij w link product access.

#### Następnie dodaj zespół.

H Admin studia25 Inf	ormacje ogólne Katalog Produkty Bezpieczeństwo Rozliczenia	Ustawienia	Q. Wyszukaj	' 0	KJ
Użytkownicy Grupy Konta zarządzane	Administrator / studia25 Użytkownicy Zilnedzaj dostępem do produktów wszystkich użytkowników w Two	jej organizacji. <u>Zobacz informacje o ustanieniach dostepu</u>	Zaproś użytkowników		
		Za każdym sukcesem stoi zespół         Dodaj swój sergół i zacznijcie razem tworzyć wspaniale rzeczy         twójemał@domena.com         Adres e-mail         Zodra członków szepolo         To witryna jest chroniona przez reCAPTCHA i mają zastosowanie <u>Polityka prywatrości i Wzeunki świadczenia usług</u> firmy Google.			

Dodaj do repozytorium członka zespołu (kolegę / koleżankę z grupy laboratoryjnej). Następnie wyszukaj osobą po jej publicznej nazwie użytkownika **Bitbucket**, nadaj uprawnienia Administratora. Kliknij przycisk **Add**.

Film - jak utworzyć zespół?

- →	CΔ	Atlassi	an, Inc. [US]	https://bitbuck	et.org/tg	adek/nisp_l	ab_02/admin	access/			T.	ነ 👶 🔿 🚺	• 📀   (	Wstrzyma	ano 👩
Aplik	acje 🛅	prir 🛅 sata	📄 java 🗎	jsp 🛅 pc 🗄	c++	🗎 youtube	🛅 english	🗎 www	altconnect	🗎 javaFX	Polemiki	lk latex 🔅 v	volfram		
-	(°	nisp_lab_02		tomaszgadek	/ nisp	_lab_02									
2	$\langle \rangle$	Source		Settings			llear an	daroup							
F	¢	Commits		Repository	details		Here's w	nere vou ar	access	aroups acces	s to this repo	ository. For a list o	of all users	with acce	ess to an
	្ស	Branches		User and g	roup acc	ess	of your p	rivate repo	sitories, see wh	nich users co	unt towards y	your bill on the Us	sers on pla	.n page. L	earn mor
	ຳວ	Pull requests		Access keys	6		Users	/							
	¢	Pipelines		Username a	liases		tom					Admin		~	Add
	ዋ	Deployments		WORKFLOW			Ото	<b>m</b> Tom				owner			
	Ð	Downloads		Branching n	nodel NE	W	Ko	nstantin To	mashevich Ele	ctronic_KITech					
		Boards		Merge strat	egies		<b>É 🐲</b> Th	omas S tor	nst						
	0	Settings		Default revi	ewers		🙆 То	<b>m</b> Eriksen	polartom			Read		*	Add
				Webhooks Links			📑 То	m Sotil eme	edicus						
				FEATURES											
?				Git LFS											
				Wiki											

Zostanie utworzony dwuosobowy zespół. Od tego momentu zaproszony użytkownik będzie miał dostęp do repozytorium. Repozytorium jest wspólne dla obojga programistów.

Aby rozpocząć pracę wyszukaj repozytorium kolegi / koleżanki z zespołu. Kliknij w logo konta, rozwiń listę i kliknij **All workspaces**. Wybierz udostępnioną przestrzeń pracy zespołu i odszukaj uodstępnione repozytorium.



# Student-B

Klonuje repozytorium oraz otwiera projekt w zintegrowanym środowisku programistycznym.

Klonowanie i uruchomienie projektu składa się z kilku kroków:

- Przejdź do wspólnego repozytorium na platformie **Bitbucket**.
- Przejdź do menu **Source**. Po prawej stronie będzie przycisk **Clone**, kliknij w niego i skopiuj komendę **git clone...**
- Następnie otwórz **terminal systemowy**, przejdź do odpowiedniej lokalizacji i wklej skopiowaną komendę. Następnie kliknij **ENTER**.
- Na dysk twojego komputera zostanie pobrane całe repozytorium, nazwa katalogu będzie identyczna, jak nazwa zdalnego repozytorium.
- Repozytorium zostanie zainicjowane branchem **master**.
- Następnie otwórz pobrany projekt File / Open....

## Praca zespołowa

### Zanim rozpoczniesz jakiekolwiek prace doczytaj konspekt do końca!!!

Przed przystąpieniem do jakiejkolwiek pracy obaj programiści powinni znajdować się na gałęzi **develop**, więc wykonaj polecenie **git checkout develop** (przełącz się na gałąź **develop**). Każdy ze studentów posiada własną klasę do zadań matematycznych:

- Student-A tworzy klasę w oddzielnym pliku w katalogu src: StudentAMath.
- Student-B tworzy klasę w oddzielnym pliku w katalogu src: StudentBMath.

Student-A implementuje:

- Odejmowanie liczb całkowitych funkcja: **sub(int, int)**.
- Mnożenie liczb całkowitych funkcja: mul(int, int).
- Wyszukiwanie minimalnej wartości funkcja min(int, int).

Student-B implementuje:

- Dodawanie liczb całkowitych funkcja: add(int, int).
- Dzielenie liczb całkowitych funkcja: div(int, int).
- Wyszukiwanie maksymalnej wartości funkcja max(int, int).

### Każda funkcjonalność to oddzielny commit oraz branch.

Schemat postępowania dla studenta A i B:

# x = nazwa funkcjonalności



Czy wiesz, że...

Pierwszym komputerem, który został zaprojektowany i zbudowany w całości do obliczeń elektronicznych, był ENIAC (ang. Electronic Numerical Integrator and Computer).



ENIAC został uruchomiony w 1946 roku na Uniwersytecie Pensylwanii (USA) i był ogromny zajmował całe pomieszczenie o powierzchni 167 m². Co ciekawe, ENIAC był zdolny do wykonywania różnych zadań obliczeniowych, w tym obliczeń balistycznych dla celów wojskowych. Liczby były wprowadzane do ENIAC za pomocą przełączników i kabli, co było pracochłonne. ENIAC zapoczątkował erę komputerów elektronicznych i był jednym z pierwszych kroków w kierunku rozwoju nowoczesnych komputerów, jakie znamy dzisiaj.