

# Warsztaty przed kurse

v3.1

# Plan

- Przed przystąpieniem do preworku
- Co i kiedy? Czyli w jakiej kolejności robić prework?
- IDE - czyli twoje przyszłe narzędzie pracy
- Dobrze jeżeli zajrzysz - czyli co robić, jeżeli zostało trochę czasu?
- Miejsca które mogą Cię zainteresować - czyli lista przydatnych linków

**Przed  
przystąpieniem  
do preworku**

# Szybkie pisanie

- Z naszego kursu wyniesiesz o wiele więcej, jeżeli nie będziesz miał problemów z szybkim pisanem na klawiaturze. W pracy też będzie od Ciebie wymagane żeby szybko pisać na klawiaturze.
  - Sprawdź się zatem w tym teście: <http://www.szybkiepisanienaklawiaturze.pl/test.htm>
  - Jeżeli Twój wynik to ponad 200 znaków na minutę, nie musisz się o nic martwić – bez problemu dasz sobie radę z szybkim pisanem podczas kursu.
- Jeżeli jednak twój wynik jest gorszy niż 200 znaków na minutę, to poćwicz szybkie pisanie.
  - Polecamy skorzystanie z następujących serwisów:
    - ➔ <http://www.kurspisania.pl>,
    - ➔ <http://www.typingstudy.com/pl-polish-3>,
    - ➔ <http://www.keybr.com>.

# Jak działają przeglądarki internetowe?

- Przed kursem poznaj dobrze działanie współczesnych przeglądarek internetowych.
  - Bardzo klarownie tłumaczy to artykuł, który znajdziesz na poniższej stronie:
    - ➔ <http://sekurak.pl/jak-dzialaja-wspolczesne-przegladarki-internetowe>.
- Chcesz dowiedzieć się więcej? Nie masz problemów z czytaniem po angielsku, przeczytaj ten artykuł:
    - ➔ <http://www.html5rocks.com/en/tutorials/internals/howbrowserswork>.

**Co i kiedy?  
Czyli w jakiej  
kolejności  
robić prework?**

# Krok po kroku jak przerobić prework

**Dokładne przerobienie preworku jest bardzo ważne.** Wiedza którą tutaj poznacie będzie wam potrzebna już od pierwszego dnia kursu, więc nie można niczego zaniedbać.

**Pamiętaj, że warsztaty muszą być oddane na 2 tygodnie przed rozpoczęciem kursu i ich zaliczenie jest warunkiem dopuszczenia do kursu.**

Najlepiej jeżeli prework będziecie przerabiać w takiej kolejności:

1. Przygotowanie komputera do pracy,
2. Nauka GIT (narzędzie do kontroli wersji),
3. Nauka HTML i CSS,
4. Przerobienie warsztatów z HTML i CSS,
5. Nauka obsługi Linuxa,
6. Nauka Javy,
7. Przerobienie warsztatów z Javy.

# Przygotowanie komputera do pracy

**Ta część preworku powinna Ci zająć około godziny.**



# **Przygotowanie komputera do pracy**

# Przygotowanie komputera do pracy

**Do pracy na kursie back-end będziecie potrzebować komputera z systemem Linux oraz oprogramowaniem potrzebnym do rozwijania aplikacji back-end w Java/Spring. Na kolejnych slajdach dowiesz się:**

- Jak zainstalować Linuksa na swoim komputerze?
- Jak zweryfikować poprawność instalacji?

# Instalacja Linuxa UWAGA

Pamiętaj, instalacja drugiego systemu operacyjnego jest skomplikowana. Jeśli będziesz postępował zgodnie z naszymi instrukcjami i czytał ze zrozumieniem wszystkie komunikaty to instalacja powinna przebiec pomyślnie. Jednak istnieje możliwość utraty Twoich danych, więc radzimy byś przed instalacją zrobił kopię zapasową wszystkich ważnych plików z komputera.

Jeżeli z jakiegoś powodu nie chcesz instalować systemu Linux na swoim komputerze zawsze możesz skorzystać z Maszyny Wirtualnej **VirtualBox** (opisane [tutaj](#)). Jest to rozwiązanie które może działać dużo wolniej, i bardzo utrudnić Ci pracę na kursie.

# INSTALACJA UBUNTU

# Instalacja Linuxa

Jeżeli posiadasz już zainstalowany system Linux to możesz od razu przejść do instalacji potrzebnego oprogramowania ( [tutaj](#)).

**Twój system musi spełniać następujące warunki żeby nasz instalator zadziałał:**

- Posiadać manager pakietów apt-get,

# Instalacja Linuxa

Instalacja Linuksa jest bardzo prosta. Najlepiej jeżeli skorzystacie z **Ubuntu 16.04** – jest to jeden z prostszych w obsłudze systemów.

**System działa zarówno na komputerach PC jak i na Macach.**

Żeby go zainstalować musicie przejść przez poniższe kroki:

- **Przygotowanie dysku instalacyjnego USB.**
- **Instalacja obok istniejącego systemu.**
- **Weryfikacja instalacji.**

# Przygotowanie dysku instalacyjnego

Przed instalacją ściągnij plik ISO z systemem:

<http://www.ubuntu.com/download/desktop>

Jeżeli nie wiesz jak sprawdzić czy twój komputer jest 32-bitowy czy 64-bitowy instrukcję możesz znaleźć tutaj:

<http://www.computerhope.com/issues/ch001121.htm>

Następnie musisz nagrać system na nośnik USB. Jak to zrobić masz opisane tutaj:

- Na systemie Windows:

<http://www.ubuntu.com/download/desktop/create-a-usb-stick-on-windows>

- Na systemie MacOS:

<http://www.ubuntu.com/download/desktop/create-a-usb-stick-on-mac-osx>

# Bootowanie z USB

## Na komputerach typu PC:

- Jeśli Twój komputer jest włączony, rozpocznij podłączając Twój nośnik instalacyjny i zrestartuj komputer.
- Większość komputerów pozwoli Ci tymczasowo zmienić kolejność nośników, z jakich podejmie próbę rozruchu. Zrobisz to naciskając określony klawisz - najczęściej F12, ale czasem też Esc lub inny klawisz funkcyjny. Sprawdź informacje wyświetlane na ekranie dla pewności, lub sprawdź na stronie producenta swojego komputera.





# Bootowanie z USB

- Naciśnij F12 (lub inny, odpowiedni klawisz) i wybierz nośnik instalacyjny - najczęściej "USB-HDD" lub coś, co zawiera słowo "USB", lecz użyte słowa mogą się różnić. Jeśli wybierzesz niewłaściwie, Twój komputer prawdopodobnie uruchomi się jak zwykle. W takim przypadku, zrestartuj komputer i spróbuj wybrać inną opcję.
- Tuż po wybraniu prawidłowego nośnika instalacyjnego, powinieneś zobaczyć na ekranie listę opcji do wybrania – wybierz „Install Ubuntu”. Jeżeli jednak nie będziesz miał wyboru opcji to poczekaj – Ubuntu powinno się samoistnie uruchomić – wtedy na pulpicie będziesz miał ikonkę instalatora – uruchom ją.

# Bootowanie z USB

- Tuż po wybraniu prawidłowego nośnika instalacyjnego, powinieneś zobaczyć na ekranie listę opcji do wybrania – wybierz „Install Ubuntu”. Jeżeli jednak nie będziesz miał wyboru opcji to poczekaj – Ubuntu powinno się samoistnie uruchomić – wtedy na pulpicie będziesz miał ikonkę instalatora – uruchom ją.

# Podczas instalacji

Wszystkie kroki instalacji są opisane tutaj:

<http://www.ubuntu.com/download/desktop/install-ubuntu-desktop>

**Podczas instalacji wybierzcie opcję instalacji obok istniejącego już systemu.**

**Dzięki temu podczas uruchomienia komputera będziecie mogli wybrać który system chcecie uruchomić. Po kursie będziecie mogli odzyskać miejsce zajęte przez ten system.**

**Jeżeli nie masz takiej opcji to przerwij instalację systemu i skontaktuj się z swoim mentorem!**

# Po instalacji Systemu

Po instalacji podczas uruchamiania komputera powinno być widoczne menu wyboru systemu (zwane też GRUB). Po wybraniu opcji Ubuntu powinien Ci się uruchomić nowy system.

Jeżeli wszystko działa i możesz zalogować do obu systemów możesz przejść do instalacji Javy i innych potrzebnych do kursu rzeczy (opisane [tutaj](#)).

# Instalacja macOS

Jeżeli posiadasz już zainstalowany system macOS na kolejnych slajdach dowiesz się jak użyć naszego skryptu instalacyjnego aby przygotować komputer do pracy.

**Twój system musi spełniać następujące warunki żeby nasz instalator zadziałał:**

- Posiadać terminal,
- Musisz znać swoje hasło do konta aby móc instalować pakiety jako root,
- Podczas instalacji musisz mieć dostęp do internetu.

# Alternatywa - VirtualBox

# VirtualBox

Alternatywą do instalowania systemu Linux obok swojego normalnego systemu jest użycie programu **VirtualBox**.

**VirtualBox** jest specjalnym programem który potrafi stworzyć maszynę wirtualną na naszym komputerze. Maszyna wirtualna to symulacja prawdziwego komputera która posiada wydzielony dostęp do zasobów naszego komputera.

Maszyny wirtualne cechują się dużym bezpieczeństwem (nie mogą dokonać żadnych zmian na naszym prawdziwym komputerze), ale też gorszą wydajnością (są symulacją i posiadają dostęp tylko do części zasobów naszego komputera).

# VirtualBox - instalacja

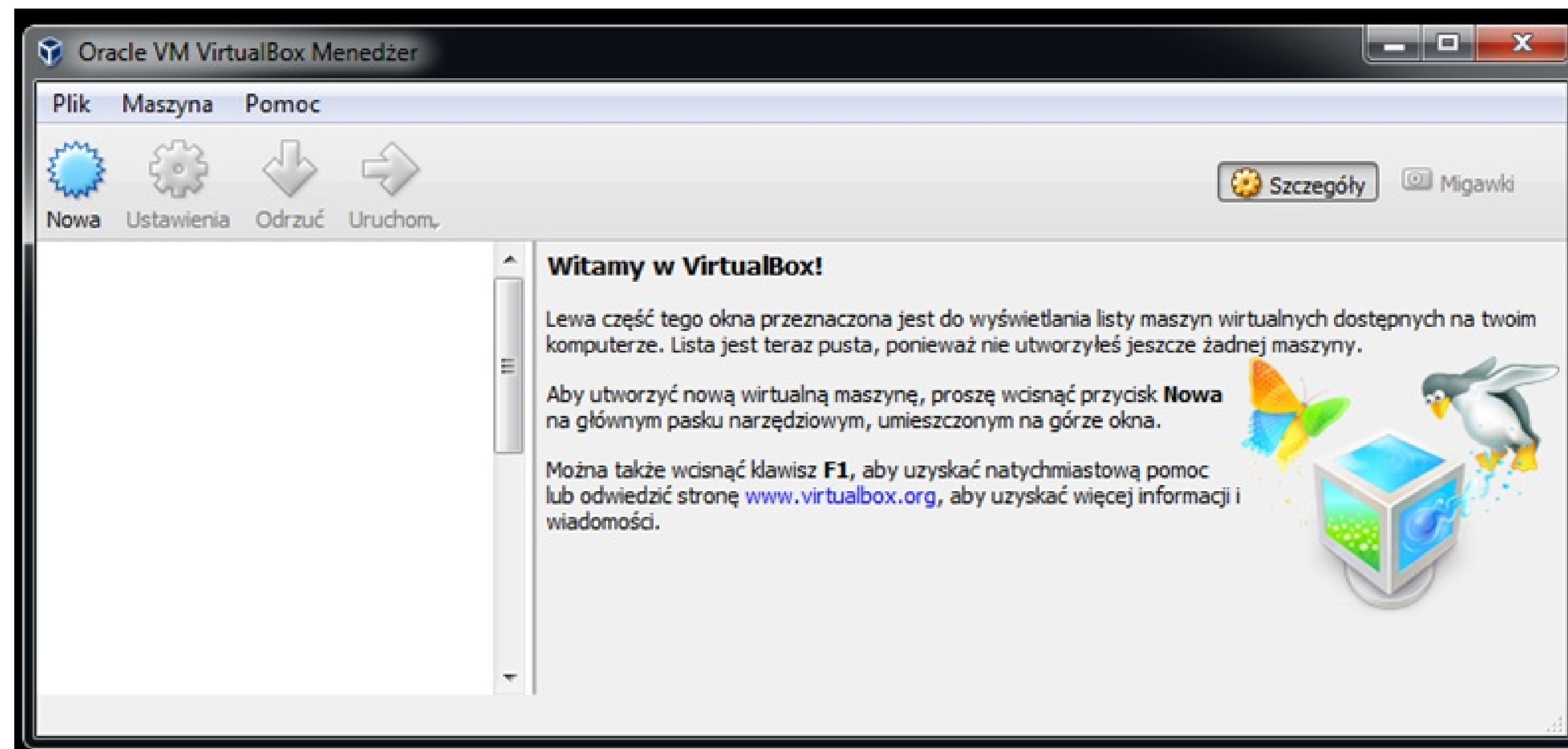
Instalacja programu VirtualBox jest niezwykle łatwa. Wystarczy ściągnąć plik instalacyjny z strony <https://www.virtualbox.org/wiki/Downloads> i go uruchomić.

Instalator przeprowadzi nas przez kolejne kroki.



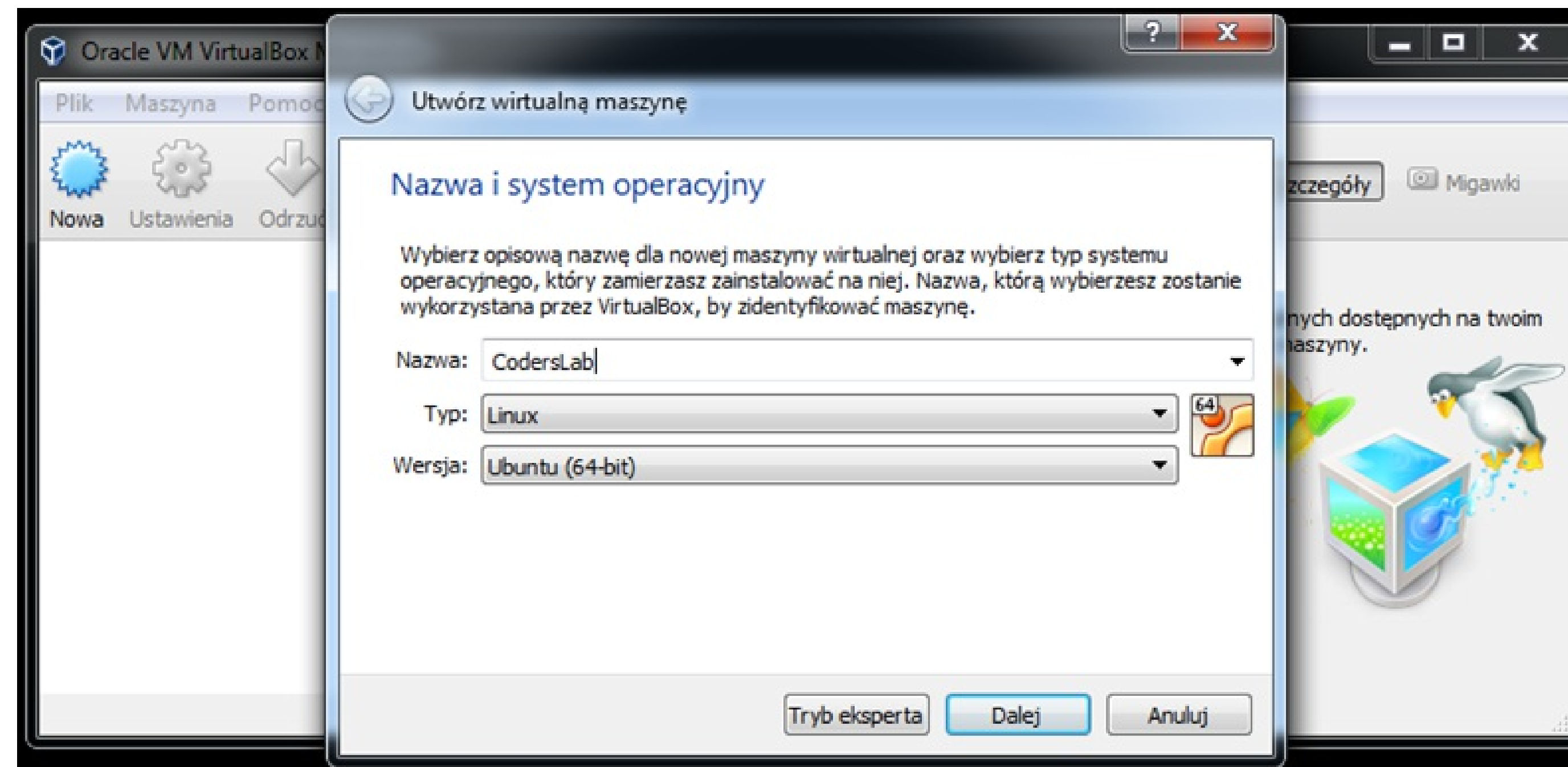
# VirtualBox – tworzenie nowej maszyny

Po zainstalowaniu programu virtual box będziemy musieli stworzyć nową maszynę wirtualną. Żeby to zrobić po uruchomieniu programy należy wybrać opcję „Nowa” widoczną na górnym pasku aplikacji.



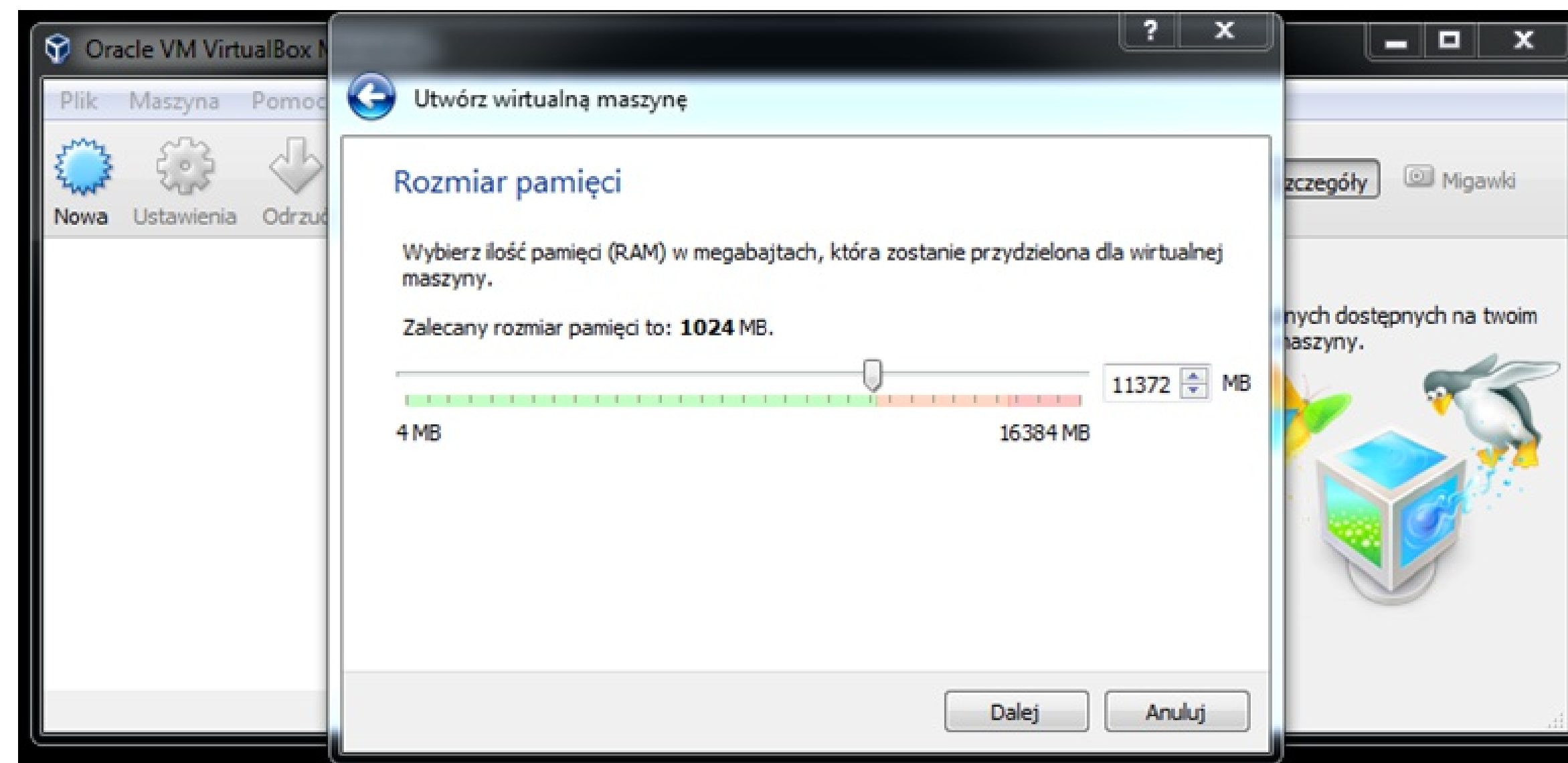
# VirtualBox – tworzenie nowej maszyny

Następnie należy wpisać nazwę naszej maszyny (może być dowolna, ale nie możecie posiadać dwóch maszyn o tej samej nazwie), wybrać typ: Linux, i Wersja: Ubuntu.



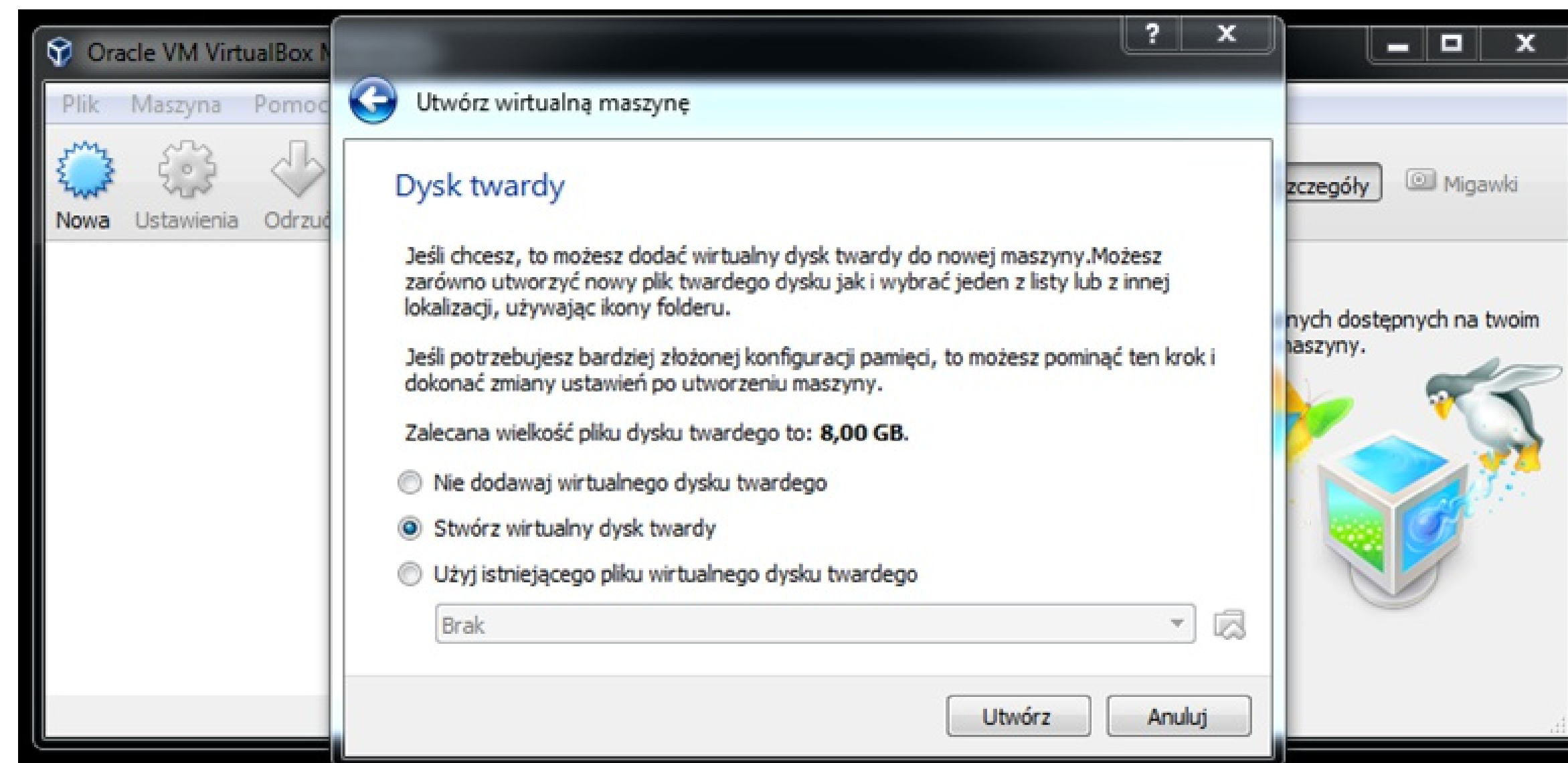
# VirtualBox – tworzenie nowej maszyny

Następnym krokiem jest przydzielenie ilości pamięci RAM do której będzie miała dostęp nasza maszyna. Im więcej tym maszyna będzie działać szybciej. Optymalna ilość to  $\frac{3}{4}$  całkowitej ilości pamięci RAM



# VirtualBox – tworzenie nowej maszyny

Następnym krokiem jest przydzielenie miejsca na dysku twardym do którego będzie miała dostęp maszyna. Powinno być to co najmniej 30 GB. Robimy to poprzez wybranie opcji „Stwórz wirtualny dysk twardy”



# VirtualBox – tworzenie nowej maszyny

Następnie program przeprowadzi nasz przez kilka pytań dotyczących tworzenia wirtualnego dysku twardego dla naszej maszyny. Należy wybrać opcje:

- Typ pliku z wirtualnym dyskiem: VDI (VirtualBox Disk Image)
- Pamięć na fizycznym dysku twardym: Dynamicznie przydzielany
- Lokalizacja pliku i rozmiar:
  - Nazwa pliku najlepiej jak będzie taka sama jak nazwa maszyny,
  - Rozmiar co najmniej 30 GB

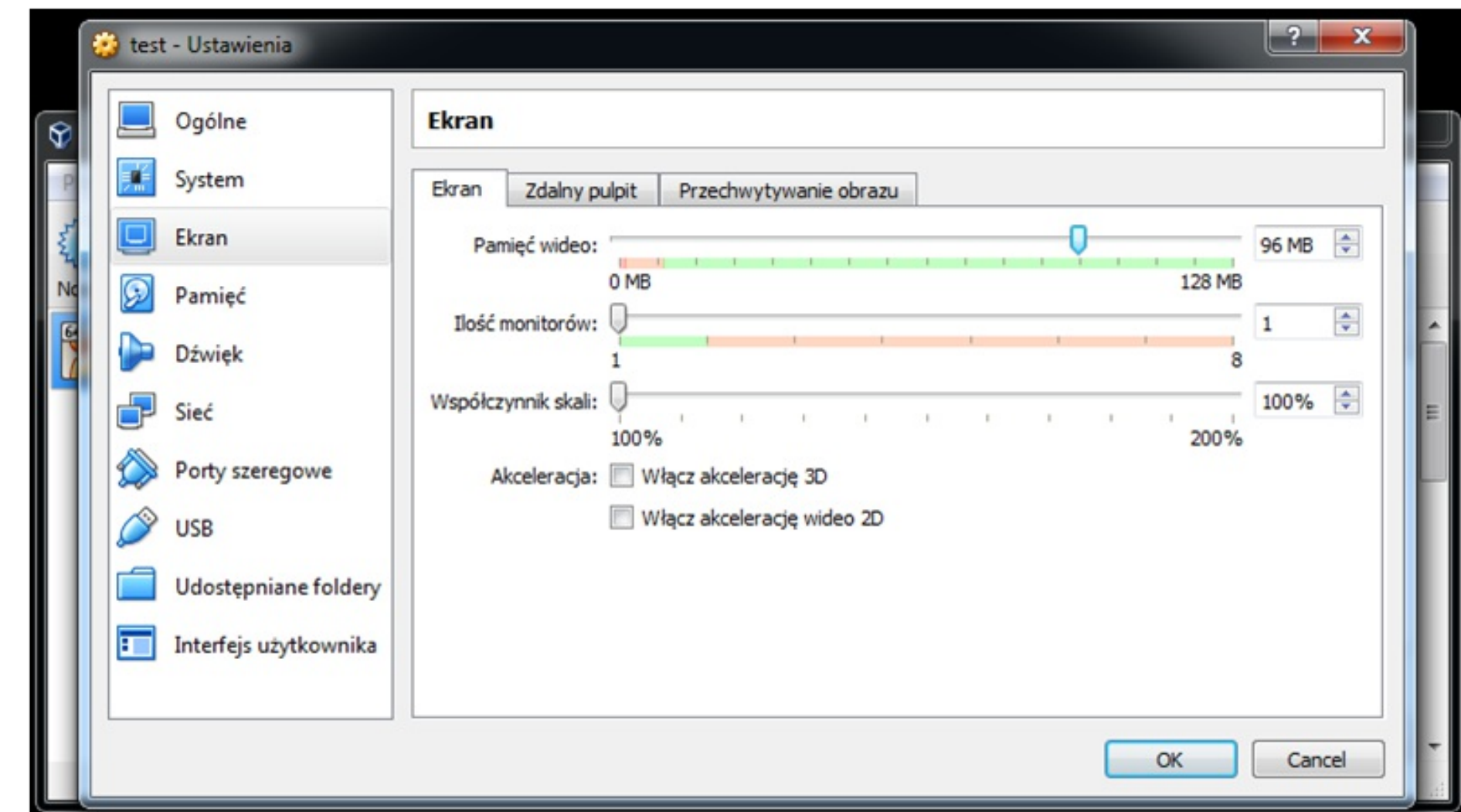
Po przejściu tych kroków Twoja maszyna wirtualna powinna być gotowa do działania.



# Zwiększenie wydajności maszyny wirtualnej

W celu zwiększenia wydajności maszyny wirtualnej możemy przydzielić jej więcej pamięci graficznej. Żeby to zrobić wykonaj następujące kroki:

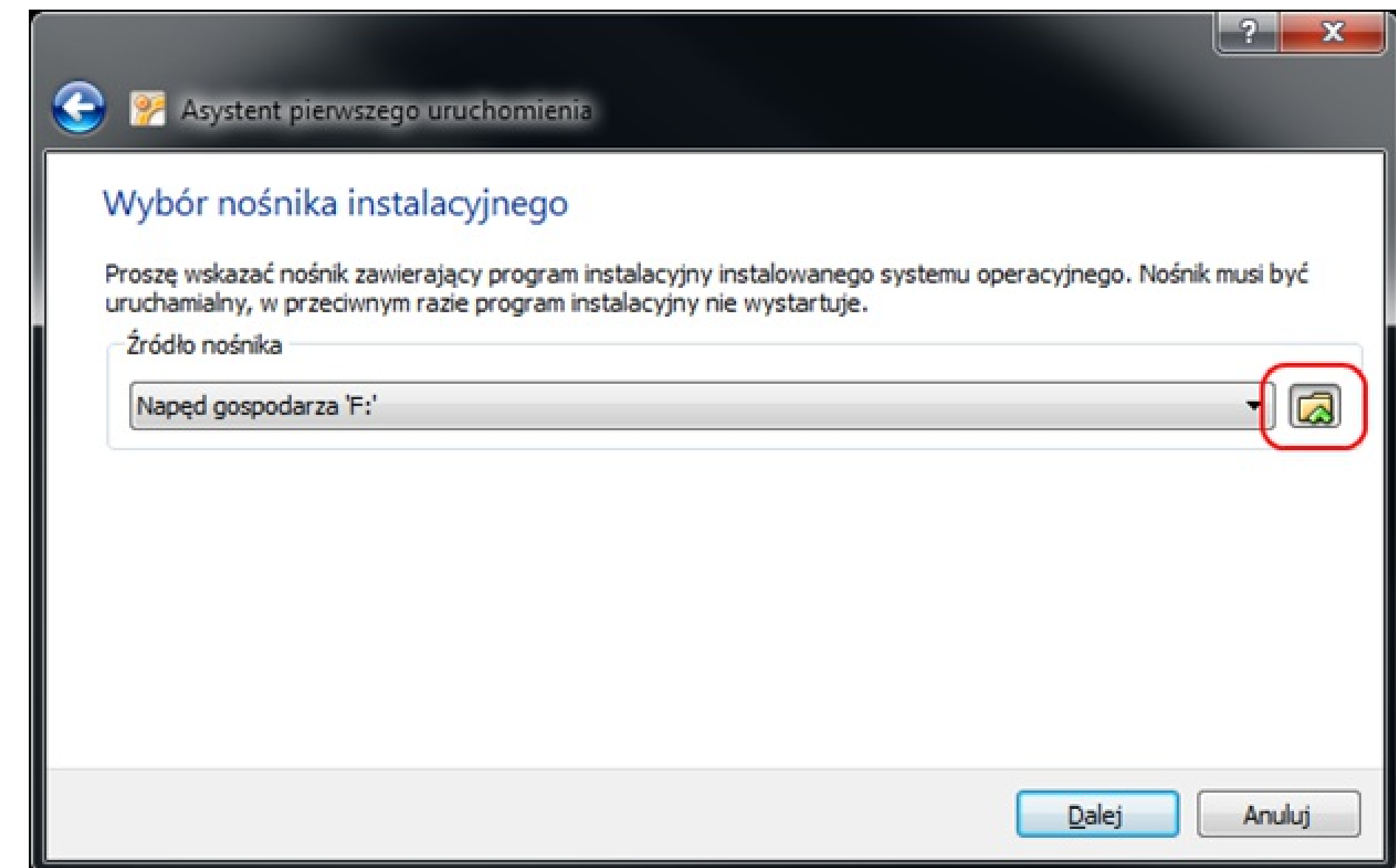
1. Uruchom ustawienia maszyny wirtualnej (kliknij prawym przyciskiem myszy na nazwie swojej maszyny i wybierz Ustawienia),
2. Przejdź do zakładki „Ekran”,
3. Zwiększ suwak przy wyborze „Pamięć Wideo” żeby maszyna mogła korzystać z około  $\frac{3}{4}$  pamięci dostępnej dla Twojego komputera.



# Instalacja Linuxa na maszynie wirtualnej

Podczas pierwszego uruchomienia maszyny wirtualnej VirtualBox poprosi Cię o podanie mu nośnika, z którego będzie mógł zainstalować system Linux. Należy wybrać wcześniej ściągniętą płytę ISO z obrazem system Ubuntu.

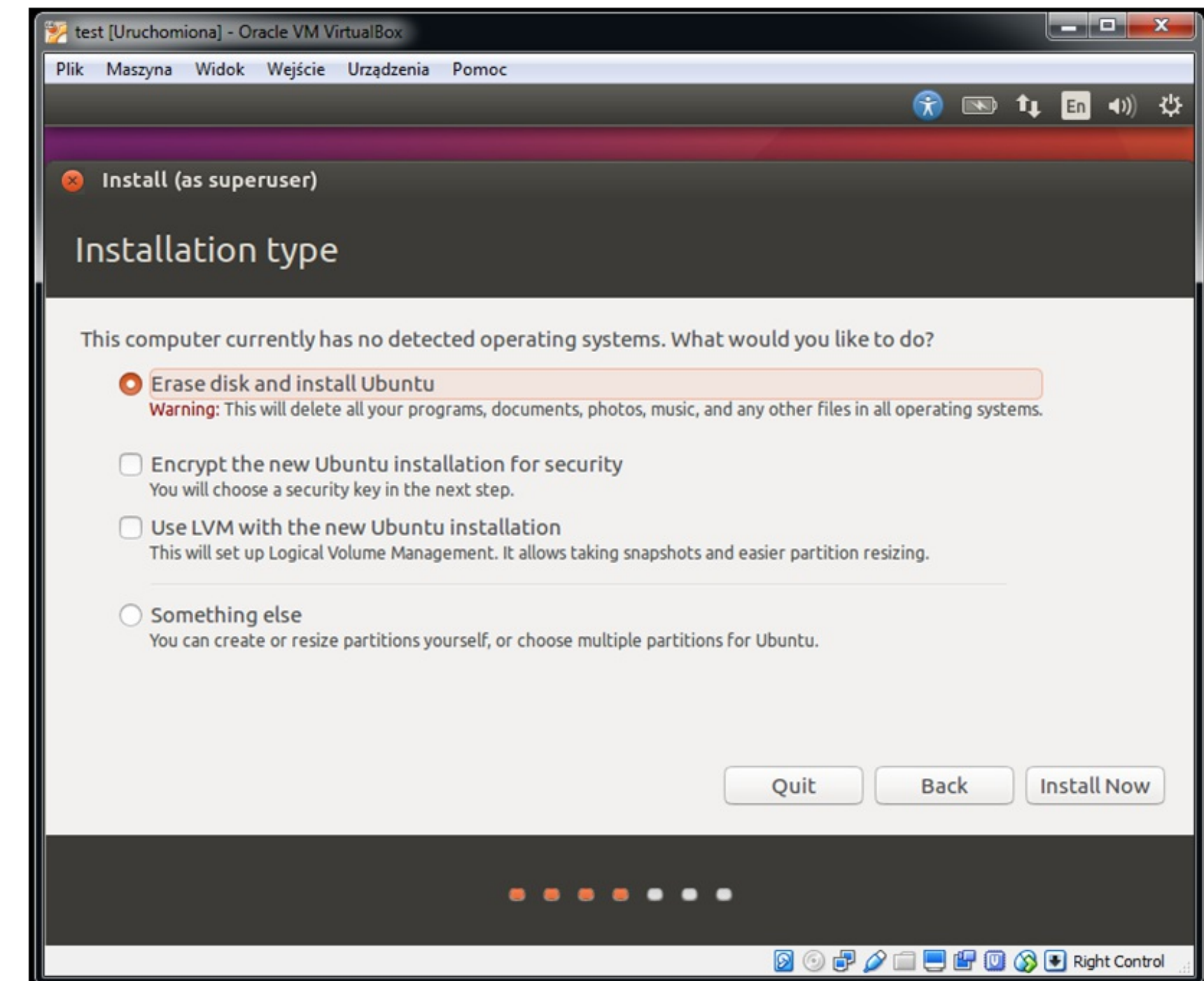
Automatycznie uruchomi się instalator systemu Ubuntu, który przeprowadzi nas przez wszystkie kroki.



# Instalacja Linuxa na maszynie wirtualnej

Podczas instalacji należy wybrać opcję Wyczyszczenia dysku twardego i zainstalowania Ubuntu.

Spowoduje to sformatowanie Wirtualnego Dysku maszyny, a nie rzeczywistego dysku w komputerze.






# Po instalacji

Po instalacji systemu operacyjnego można doinstalować jeszcze dodatkowe oprogramowanie na Wirtualnym komputerze. Umożliwi to lepsze działanie maszyny, możliwość współdzielonego schowka i automatyczną zmianę wielkości ekranu.

Żeby doinstalować Dodatkowe oprogramowanie musisz:

1. Uruchomić maszynę wirtualną,
2. Kliknąć "Devices" na górnym menu uruchomionej maszyny wirtualnej,
3. Kliknąć "Install Guest Additions CD Image...",
4. Zrestartować maszynę wirtualną.

Jeżeli wszystko działa i możesz uruchomić maszynę wirtualną, możesz przejść do instalacji Java i innych potrzebnych do kursu rzeczy (opisane [tutaj](#)).



Instalacja potrzebnego  
oprogramowania

# Instalacja oprogramowania

Na razie masz zainstalowanego nowego Linuxa. Musimy jeszcze zainstalować inne potrzebne nam rzeczy:

- JDK – czyli oprogramowanie udostępniające środowisko niezbędne do programowania w języku Java,
- **IntelliJ IDEA**, czyli IDE, na którym będziecie pracować
- MySQL i panel administracyjny MySQL Workbench,
- Inne przydatne programy: git, curl, maven, tomcat, vim.

# Instalacja oprogramowania

Instalacja jest bardzo prosta. Najpierw musisz ściągnąć skrypt instalacyjny z tego miejsca:  
<https://goo.gl/rFwvzw>

Żeby użyć skryptu musisz:

1. Otworzyć terminal a następnie wpisać poniższe komendy:
2. Przejść do katalogu z ściągniętymi plikami wpisując:
  - **cd Downloads** - na systemie anglojęzycznym
  - **cd Pobrane** - na systemie polskojęzycznym

**Po każdej wpisanej komendzie musisz wcisnąć enter.**

3. Nadać plikowi prawa do wykonywania, wpisując komendę (po wpisaniu tej komendy system poprosi Cię o wpisanie hasła – **uwaga:** przy wpisywaniu hasła, znaki się nie wyświetlają):

```
sudo chmod 777 ./install.sh
```

4. Uruchomić plik, wpisując komendę (twój komputer musi być podłączony do Internetu):

```
./install.sh
```

# Instalacja oprogramowania

Następnym krokiem jest zainstalowanie IDE – w tym celu pobierz i zainstaluj testową wersję (na 30 dni) IntelliJ Ultimate, znajdziesz ją pod tym adresem:

<https://www.jetbrains.com/idea/download/#section=linux>

Skorzystaj z niej w czasie przygotowania do kursu. Po przyjeździe na zajęcia otrzymasz kod aktywacyjny na wersję Ultimate, z której będziesz mógł korzystać przez kolejne pół roku.

Pozostała nam jeszcze instalacja narzędzia potrzebnego w kolejnych modułach.

Pobierz kontener aplikacji z adresu: <https://tomcat.apache.org/download-90.cgi>

Plik **zip** z paragrafu: **Binary Distributions**, punkt **Core**.

Rozpakuj w dowolnej znanej Ci lokalizacji - na zajęciach wykładowca pokaże jak z niego korzystać.

# Weryfikacja instalacji

Po instalacji sprawdź czy wszystko poprawnie się zainstalowało. Żeby to zrobić uruchom nowo zainstalowany system i sprawdź:

- Czy masz zainstalowany program **IntelliJ**?
- Czy w twoim katalogu domowym pojawił się katalog **workspace**?
- Czy masz zainstalowany program „MySQL Workbench”?
- Otwórz terminal i wpisz słowo **java -version** (java, spacja, minus, version). Jeśli instalacja przebiegła poprawnie, powinieneś dostać odpowiedź **java version <numer aktualnej wersji>**.
- Otwórz terminal i wpisz słowo **git --version** (git, spacja, minus, minus, version). Jeśli instalacja przebiegła poprawnie, powinieneś dostać odpowiedź **git version <numer aktualnej wersji>**.
- Otwórz terminal i wpisz słowo **mvn -v** (mvn, spacja, minus, v). Jeśli instalacja przebiegła poprawnie, powinieneś dostać odpowiedź **Apache Maven <numer aktualnej wersji>**.



# Instalacja – informacje macOS

Instalacja jest bardzo prosta. Najpierw musisz ściągnąć skrypt instalacyjny z tego miejsca: <https://goo.gl/k3lyD5>

Aby to zrobić po wejściu na podany adres w Safari wybierz z menu opcję zapisania strony i zapisz plik na **pulpicie** pod nazwą **install\_macOS.sh**, na kolejnych slajdach znajdziesz screeny z kolejnych etapów instalacji.

**Plik instalacyjny został przygotowany dla wersji macOS Sierra**

# Instalacja – informacje macOS

Żeby użyć skryptu musisz:

1. Otworzyć terminal a następnie wpisać poniższe komendy:
2. Przejść do katalogu z ściągniętymi plikami wpisując:
  - **cd Desktop** - na systemie anglojęzycznym
  - **cd Pulpit** - na systemie polskojęzycznym

**Po każdej wpisanej komendzie musisz wcisnąć enter.**

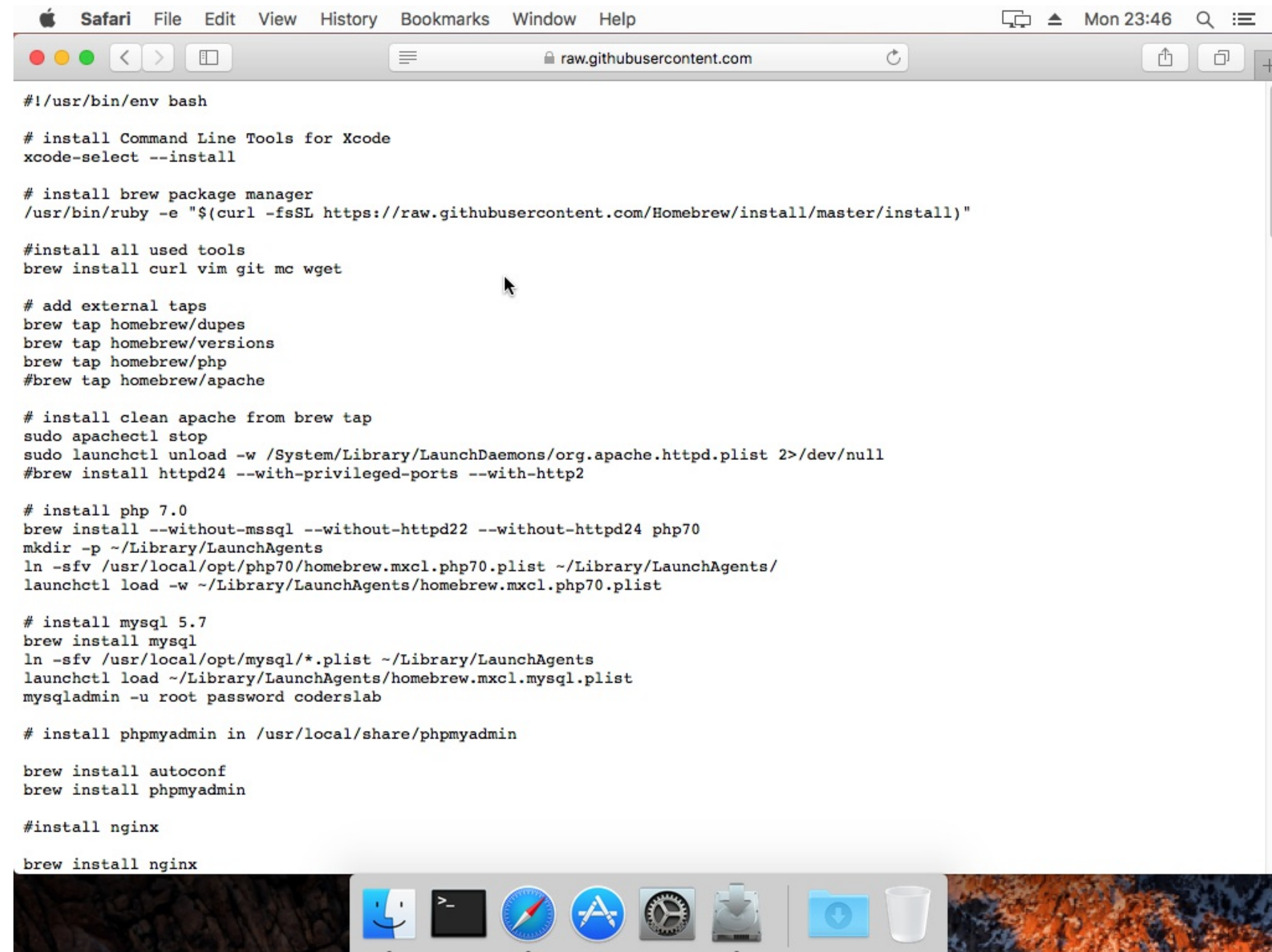
3. Nadać plikowi prawa do wykonywanie, wpisując komendę (po wpisaniu tej komendy system poprosi Cię o wpisanie hasła – normalne jest, że wpisywane znaki się nie wyświetlają):

```
chmod 777 ./install_macOS.sh
```

4. Uruchomić plik, wpisując komendę (twój komputer musi być podłączony do Internetu): **./install\_macOS.sh**



# Instalacja – pobranie pliku



```
#!/usr/bin/env bash

# install Command Line Tools for Xcode
xcode-select --install

# install brew package manager
/usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"

#install all used tools
brew install curl vim git mc wget

# add external taps
brew tap homebrew/dupes
brew tap homebrew/versions
brew tap homebrew/php
#brew tap homebrew/apache

# install clean apache from brew tap
sudo apachectl stop
sudo launchctl unload -w /System/Library/LaunchDaemons/org.apache.httpd.plist 2>/dev/null
#brew install httpd24 --with-privileged-ports --with-http2

# install php 7.0
brew install --without-mysql --without-httpd22 --without-httpd24 php70
mkdir -p ~/Library/LaunchAgents
ln -sfv /usr/local/opt/php70/homebrew.mxcl.php70.plist ~/Library/LaunchAgents/
launchctl load -w ~/Library/LaunchAgents/homebrew.mxcl.php70.plist

# install mysql 5.7
brew install mysql
ln -sfv /usr/local/opt/mysql/*.plist ~/Library/LaunchAgents
launchctl load ~/Library/LaunchAgents/homebrew.mxcl.mysql.plist
mysqladmin -u root password coderslab

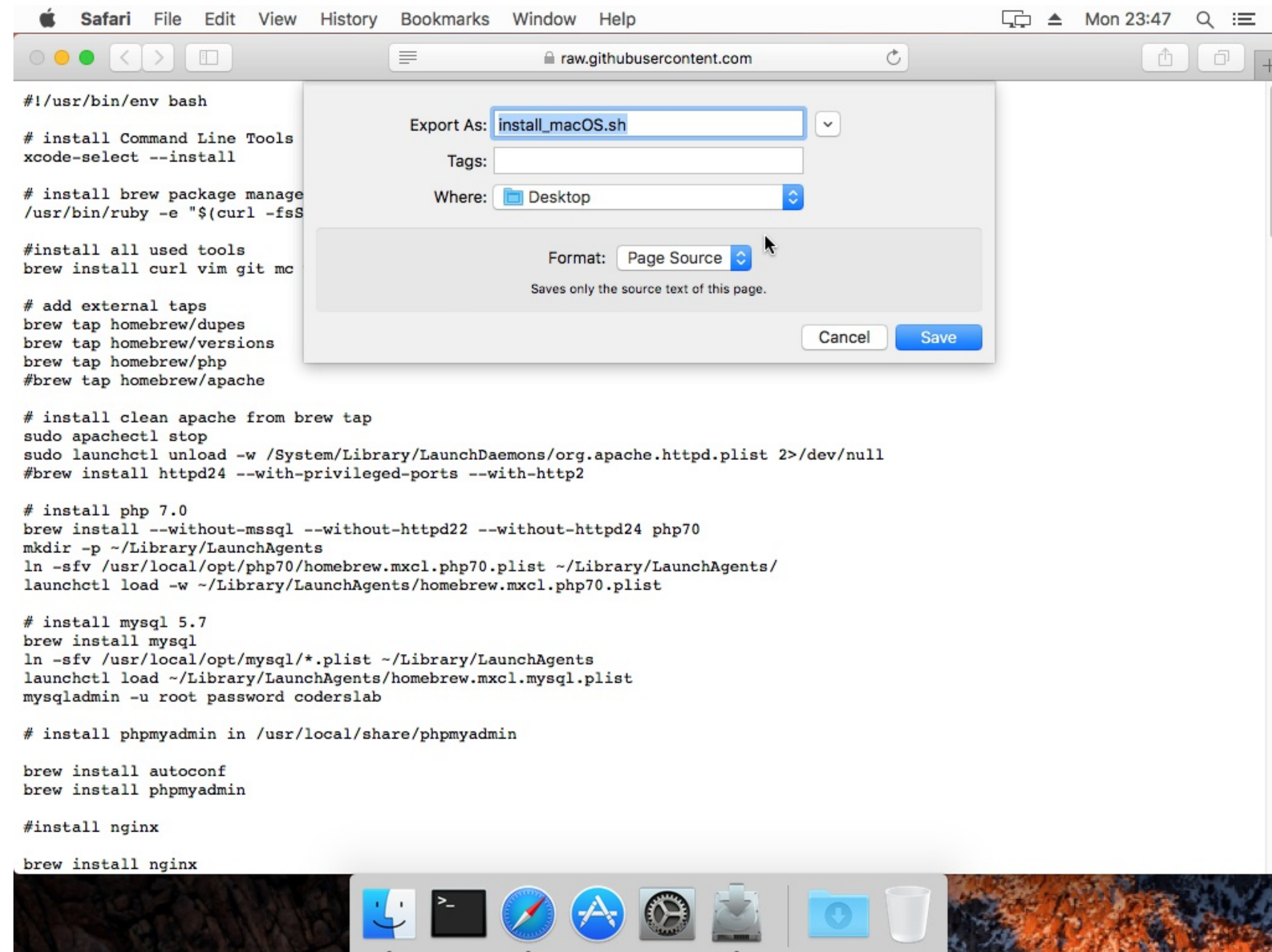
# install phpmyadmin in /usr/local/share/phpmyadmin

brew install autoconf
brew install phpmyadmin

#install nginx
brew install nginx
```

Po wejściu na podany adres zostaniemy przekierowani na odpowiedni adres.

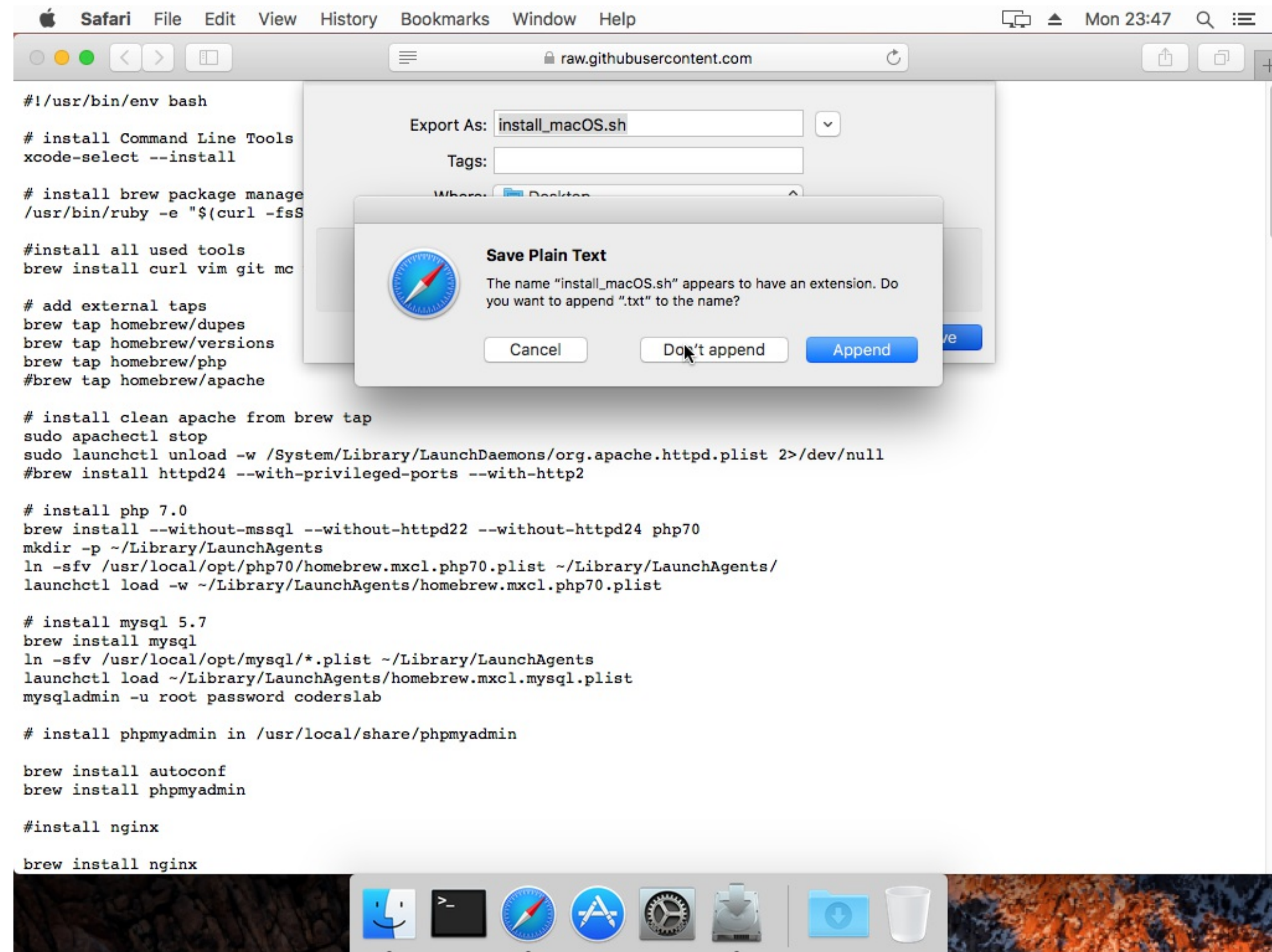
# Instalacja – pobranie pliku



Wybieramy opcję zapisu pliku, ustawiając dane jak na screenie.



# Instalacja – pobranie pliku

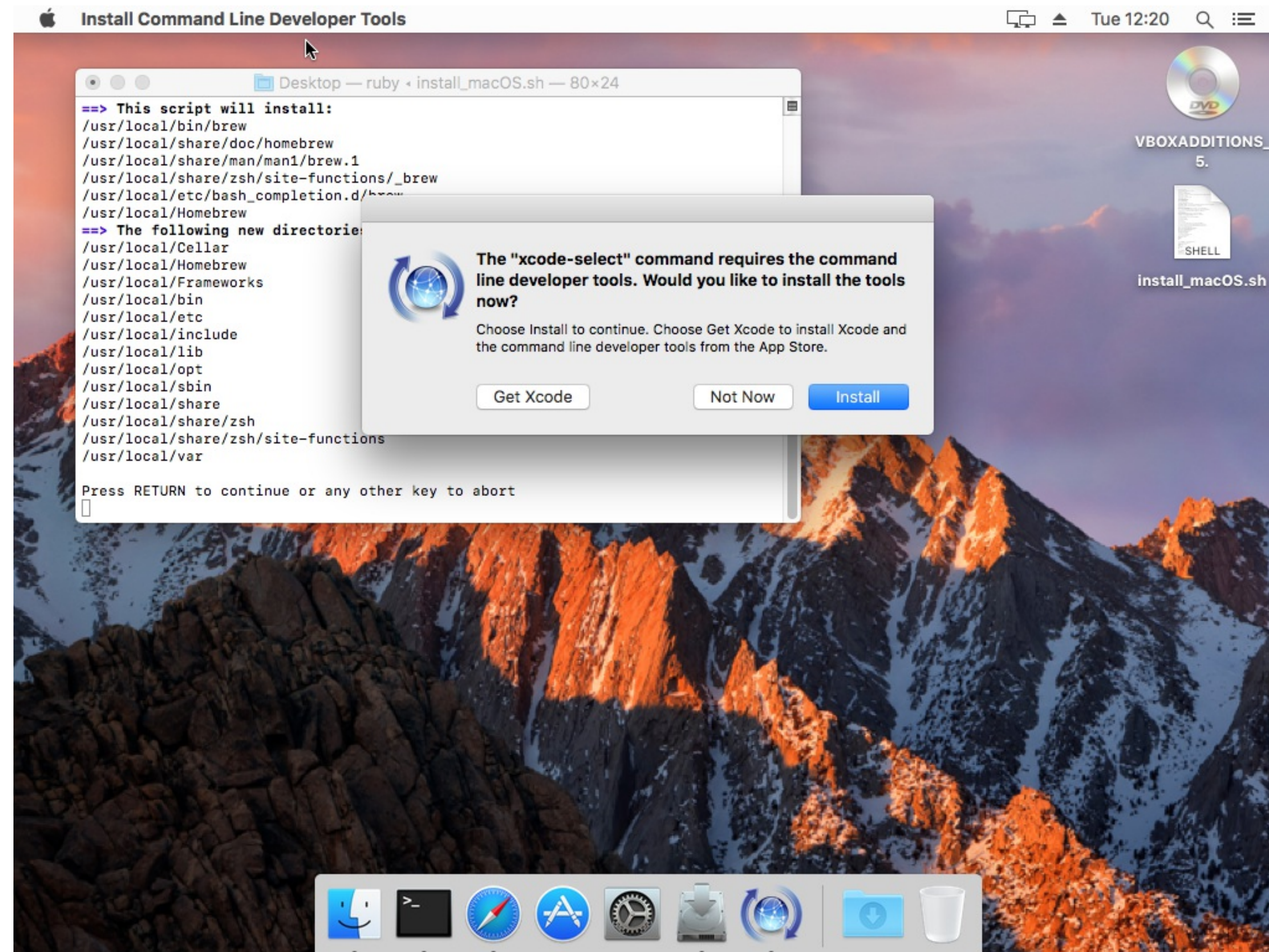


W okienku, które się pojawi wybieramy  
**"Don't append"**

Następnie przechodzimy w konsoli do  
pulpitu oraz uruchamiamy skrypt



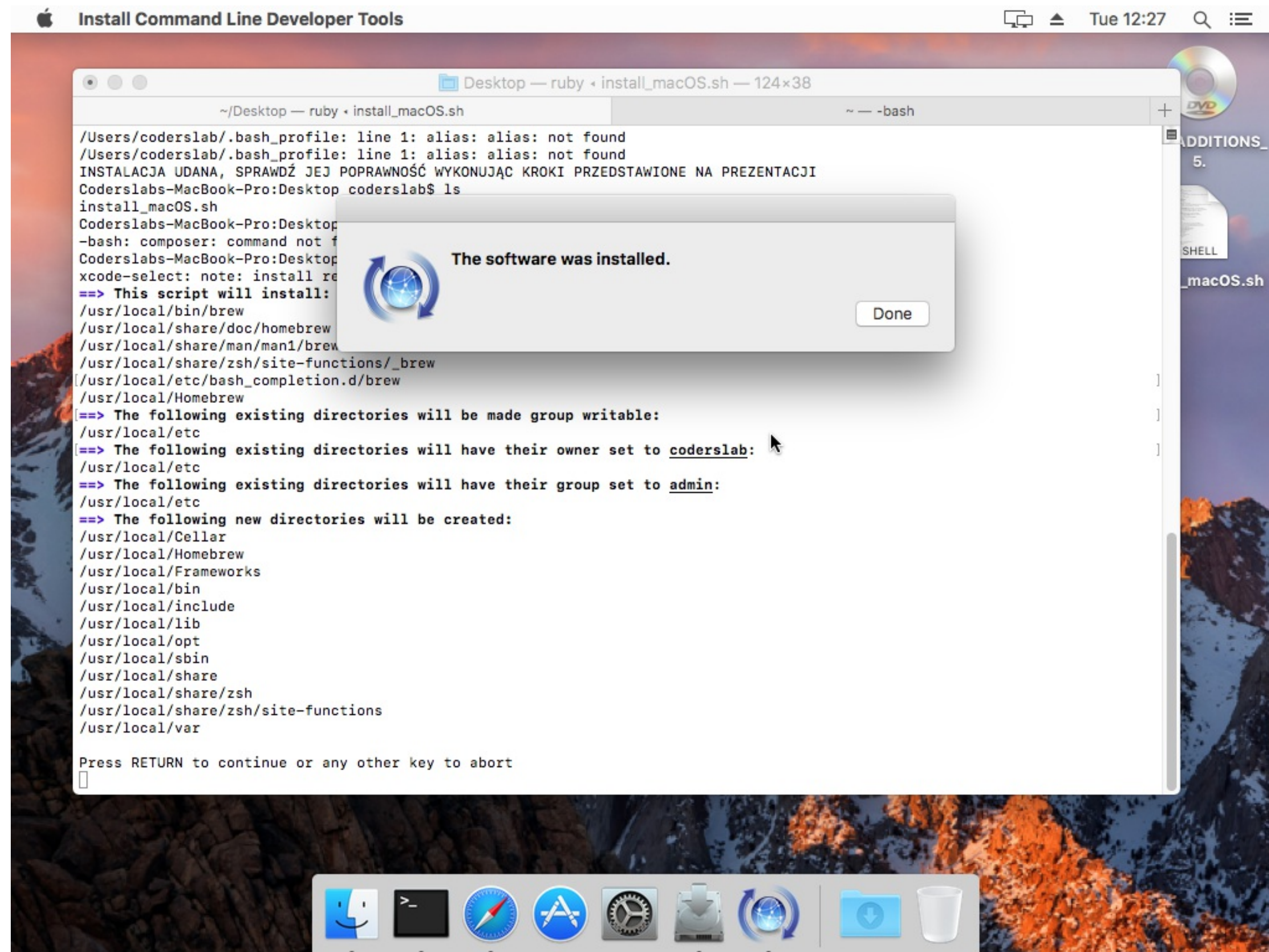
# Instalacja – potwierdzenia



Podczas instalacji zostaniemy na początku poproszeni o potwierdzenie instalacji specjalnych narzędzi konsolowych. Potwierdzamy klikając na **install** a następnie akceptujemy warunki licencyjne i czekamy na zakończenie instalacji.



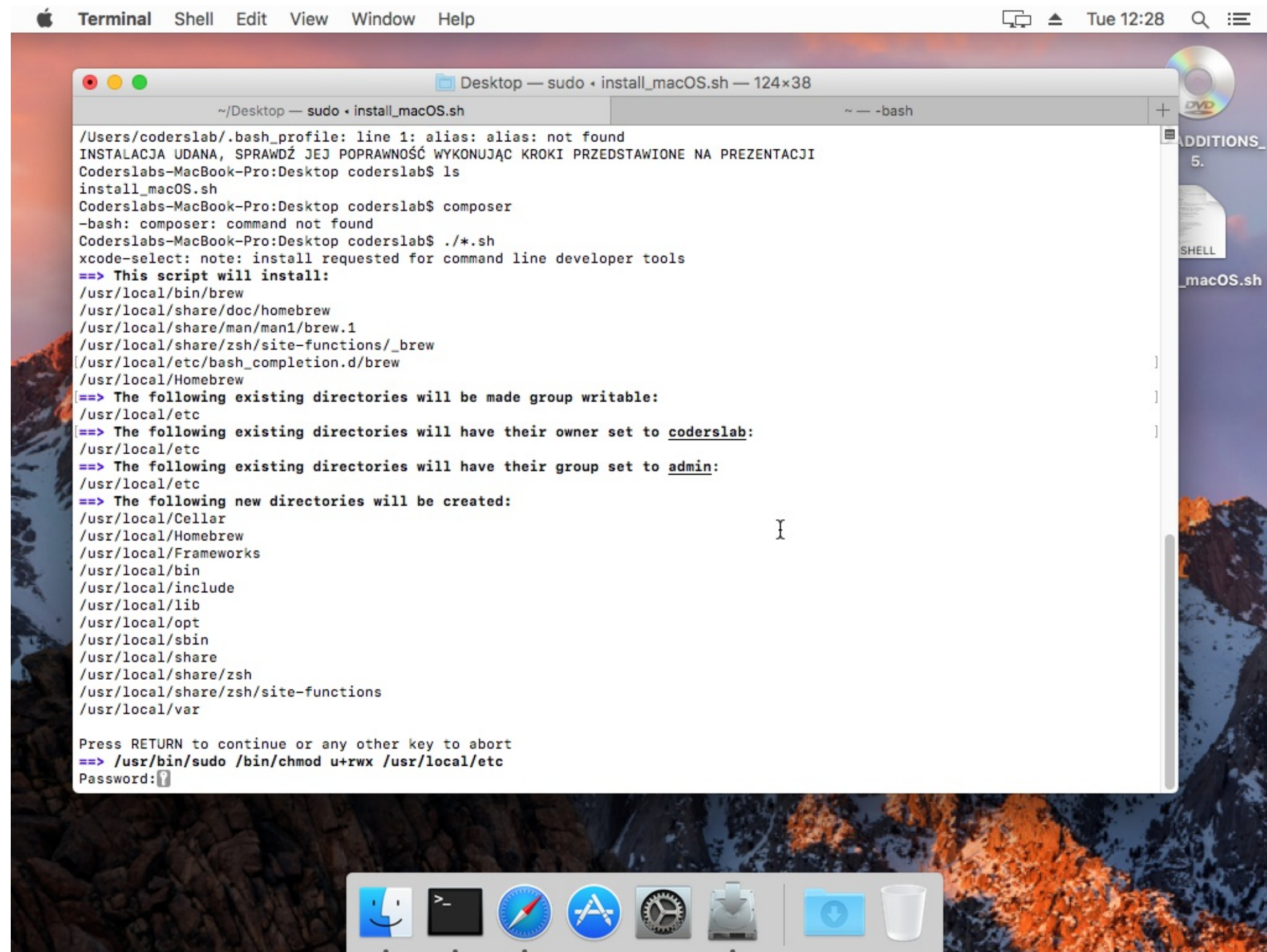
# Instalacja – dalsza instalacja



Po zakończeniu instalacji xcode klikamy na **Done** a następnie przechodzimy do konsoli i wciskamy **enter**.



# Instalacja – dalsza instalacja

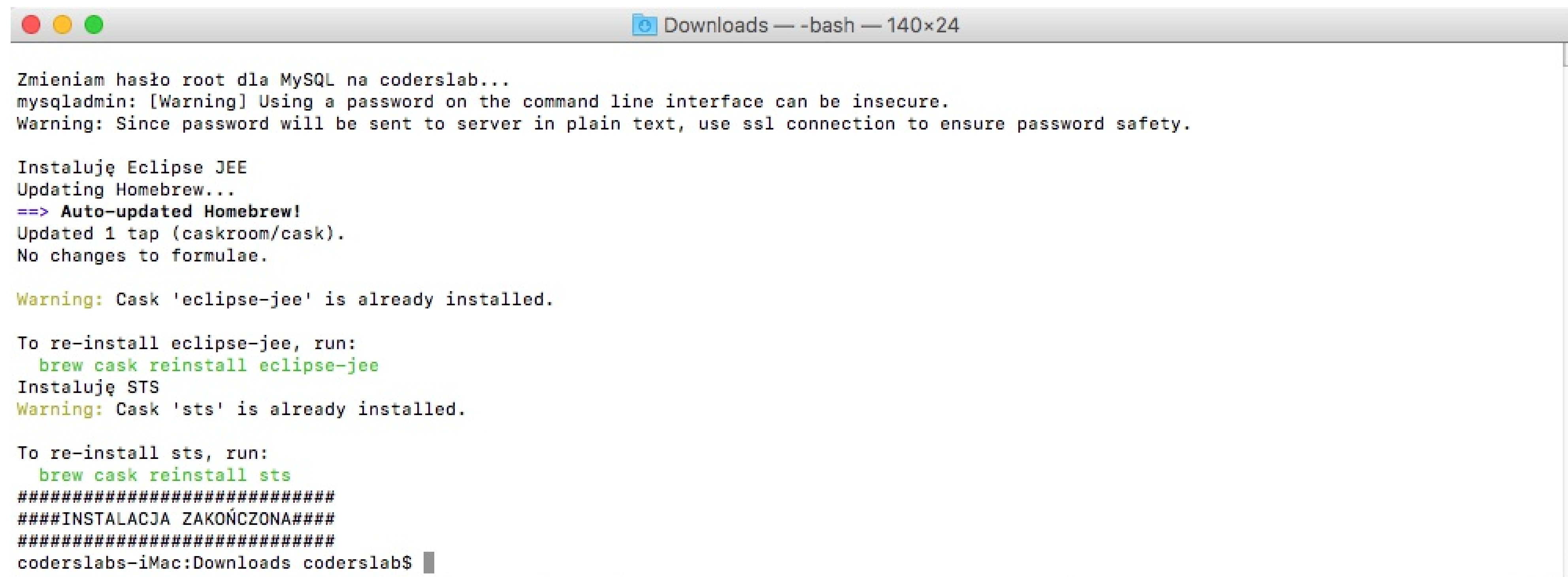


```
Terminal Shell Edit View Window Help
Desktop — sudo • install_macOS.sh — 124x38
~/Desktop — sudo • install_macOS.sh
/Users/coderslab/.bash_profile: line 1: alias: alias: not found
INSTALACJA UDANA, SPRAWDŹ JEJ POPRAWNOŚĆ WYKONUJĄC KROKI PRZEDSTAWIONE NA PREZENTACJI
Coderslabs-MacBook-Pro:Desktop coderslab$ ls
install_macOS.sh
Coderslabs-MacBook-Pro:Desktop coderslab$ composer
-bash: composer: command not found
Coderslabs-MacBook-Pro:Desktop coderslab$ ./*.sh
xcode-select: note: install requested for command line developer tools
==> This script will install:
/usr/local/bin/brew
/usr/local/share/doc/homebrew
/usr/local/share/man/man1/brew.1
/usr/local/share/zsh/site-functions/_brew
/usr/local/etc/bash_completion.d/brew
/usr/local/Homebrew
==> The following existing directories will be made group writable:
/usr/local/etc
==> The following existing directories will have their owner set to coderslab:
/usr/local/etc
==> The following existing directories will have their group set to admin:
/usr/local/etc
==> The following new directories will be created:
/usr/local/Cellar
/usr/local/Homebrew
/usr/local/Frameworks
/usr/local/bin
/usr/local/include
/usr/local/lib
/usr/local/opt
/usr/local/sbin
/usr/local/share
/usr/local/share/zsh
/usr/local/share/zsh/site-functions
/usr/local/var
Press RETURN to continue or any other key to abort
==> /usr/bin/sudo /bin/chmod u+rw /usr/local/etc
Password:
```

Musimy monitorować co dzieje się w konsoli, gdyż może ona nas poprosić o wpisanie hasła, wówczas wpisujemy hasło i wciskamy **enter**.

Prośba o hasło może pojawić się kilkukrotnie.

# Instalacja – zakończenie instalacji



```
Downloads — -bash — 140x24

Zmieniam hasło root dla MySQL na coderslab...
mysqladmin: [Warning] Using a password on the command line interface can be insecure.
Warning: Since password will be sent to server in plain text, use ssl connection to ensure password safety.

Instaluję Eclipse JEE
Updating Homebrew...
==> Auto-updated Homebrew!
Updated 1 tap (caskroom/cask).
No changes to formulae.

Warning: Cask 'eclipse-jee' is already installed.

To re-install eclipse-jee, run:
  brew cask reinstall eclipse-jee
Instaluję STS
Warning: Cask 'sts' is already installed.

To re-install sts, run:
  brew cask reinstall sts
#####
####INSTALACJA ZAKOŃCZONA####
#####
coderslabs-iMac:Downloads coderslab$
```

Po zakończeniu instalacji w konsoli pojawi się stosowny komunikat.

Zamknij konsolę, instalacja została zakończona.



# Instalacja IntelliJ

Następnym krokiem jest zainstalowanie IDE – w tym celu pobierz i zainstaluj testową wersję (na 30 dni) IntelliJ Ultimate, znajdziesz ją pod tym adresem:

<https://www.jetbrains.com/idea/download/#section=mac>

Skorzystaj z niej w czasie przygotowania do kursu. Po przyjeździe na zajęcia otrzymasz kod aktywacyjny na wersję Ultimate, i będziesz mógł z niej korzystać przez kolejne pół roku.

# Weryfikacja instalacji – informacje macOS

Po instalacji sprawdź czy wszystko poprawnie się zainstalowało. Żeby to zrobić uruchom ponownie komputer i sprawdź:

1. Czy masz zainstalowany program **IntelliJ**?
2. Czy w twoim katalogu domowym pojawił się katalog **workspace**?
3. Czy masz zainstalowany program „MySQL Workbench”?
4. Otwórz terminal i wpisz słowo **java -version** (java, spacja, minus, version). Jeśli instalacja przebiegła poprawnie, powinieneś dostać odpowiedź **java version <numer aktualnej wersji>**.
5. Otwórz terminal i wpisz słowo **git --version** (git, spacja, minus, minus, version). Jeśli instalacja przebiegła poprawnie, powinieneś dostać odpowiedź **git version <numer aktualnej wersji>**.
6. Otwórz terminal i wpisz słowo **mvn -v** (mvn, spacja, minus, v). Jeśli instalacja przebiegła poprawnie, powinieneś dostać odpowiedź **Apache Maven <numer aktualnej wersji>**.

# Problemy

Jeżeli podczas instalacji coś poszło nie tak to napisz do swojego mentora

Nie zostawiaj instalacji na ostatnią chwilę!

# Jeżeli nie chcesz instalować Linuxa

Jeżeli z jakiegoś powodu nie chcesz instalować Ubuntu to możesz przygotować samodzielnie swój system do pracy na kursie. Musisz wtedy zainstalować i skonfigurować następujące rzeczy:

- **Baza danych MySQL,**
- **MySQL Workbench,**
- **JDK Java 8,**
- **Git**
- **Maven**
- **IntelliJ IDEA**

Pamiętaj też że jeżeli samodzielnie skonfigurujesz swój komputer to wykładowca nie zawsze będzie mógł Ci pomóc jeżeli napotkasz niestandardowe błędy w czasie kursu.

# Nauka GIT

# Nauka GIT

**Ta część preworku powinna Ci zająć pomiędzy 3-6 godzin.**

# Nauka GIT

Przed kursem **musicie dobrze poznać system kontroli wersji GIT**. Będziecie go używać do ściągania zadań na zajęciach jak i do wysyłania swoich odpowiedzi żeby mentor mógł je sprawdzić.

## 1. Przerób kursy online:

Najlepiej jeżeli przerobisz następujący kurs GIT online:

<https://try.github.io/levels/1/challenges/1>

Pamiętaj – notuj podczas niego wszystkie komendy i zapisz sobie co robią – przyda Ci się to!

## 2. Przeczytaj naszą prezentację "System kontroli wersji - GIT":

Wprowadzi Cię ona w podstawowe sposoby używania GITa – zrozumiesz dlaczego go stosujemy.



# Nauka GIT

## 3. **Jeżeli masz czas:**

Jeżeli nadal masz czas i chcesz więcej dowiedzieć się o systemie GIT możesz zajrzeć do jednej z tych książek:

<https://git-scm.com/book/pl/v1> - książka twórców GITa. Tłumaczy działanie całego systemu.

<https://www.git-tower.com/learn/git/ebook> - wprowadza w używanie GITa od podstaw.

# Nauka HTML i CSS

# Nauka HTML i CSS

**Ta część preworku powinna Ci zająć pomiędzy 10-15 godzin.**

# Nauka HTML i CSS

**Pomimo tego że idziesz na kurs back-end to znajomość HTML i CSS będzie od Ciebie wymagana. Dlatego:**

1. **Przeczytaj naszą prezentację 'HTML i CSS':** Wprowadzi Cię ona w podstawy HTML i CSS. Pamiętaj żeby każdy przykład, który jest w prezentacji przepisać do pliku HTML i uruchomić, dzięki temu więcej zapamiętasz.
2. **Sprawdź swoją znajomość selektorów CSS:** Jeżeli przerobiłeś naszą prezentację to swoją znajomość selektorów CSS możesz łatwo sprawdzić tutaj: <http://flukeout.github.io/>
3. **Jeżeli masz czas:** Jeżeli nadal masz czas i chcesz tworzyć ładne strony HTML to dobrze jeżeli zainteresujesz się jakimś frameworkiem. Najpopularniejsze to:  
Bootstrap - <http://getbootstrap.com/getting-started/>  
Foundation - <http://foundation.zurb.com/sites/getting-started.html>

# Warsztaty z HTML i CSS

# Warsztaty z HTML i CSS

**Ta część preworku powinna Ci zająć około 10 godzin.**

# Warsztaty z HTML i CSS

**Żeby sprawdzić swoją znajomość z HTML i CSS zrób ćwiczenia umieszczone na GitHubie:**

1. **Dostań dostęp do ćwiczeń:** Żeby dostać dostęp do ćwiczeń musisz wysłać do swojego mentora następujące informacje:
  1. Nazwę swojego konta na GitHubie,
  2. Termin kursu na który idziesz
2. **Przeczytaj prezentację „Jak robić zadania”:** Prezentacje wytłumaczy Ci jak robić zadania i jak je wysłać do swojego mentora

**Pamiętaj że zaliczenie tej części preworku jest obowiązkowe. Zadania musisz wysłać do mentora najpóźniej na 2 tygodnie przed rozpoczęciem twojego kursu!**



# Warsztaty z Javy

# Nauka podstaw Javy

**Przed kursem prosimy Cię o przerobienie prezentacji zawierającej podstawy programowania w Javie.**

**Przeczytaj prezentację "Java podstawy":** Wprowadzi Cię ona w podstawy języka.

Pamiętaj żeby każdy przykład który jest w prezentacji przepisać do pliku .java i uruchomić, dzięki temu więcej zapamiętasz.

Jeżeli masz jakieś problemy lub coś nie jest do końca jasne, napisz do swojego mentora.

**Jeżeli masz czas:** Przed kursem przerób podstawowy kurs z języka Java, dostępny pod adresem:

<https://www.codecademy.com/learn/learn-java>

**Dzięki temu zajęcia pójdą Ci o wiele sprawniej, szybciej i więcej z nich wyniesiesz.**

# Warsztaty z Javy

**Żeby sprawdzić swoją znajomość podstaw Javy zrób ćwiczenia umieszczone na GitHubie:**

## 1. **Dostań dostęp do ćwiczeń:**

Żeby dostać dostęp do ćwiczeń musisz wysłać do swojego mentora następujące informacje:

- Nazwę swojego konta na GitHubie,
- Termin kursu na który idziesz.

## 2. **Przeczytaj prezentację "Jak robić zadania":**

Prezentacja wytłumaczy Ci jak robić zadania i jak je wysłać do swojego mentora.

**Pamiętaj że zaliczenie tej części preworku jest obowiązkowe. Zadania musisz wysłać do mentora najpóźniej na 2 tygodnie przed rozpoczęciem twojego kursu!**

# **Nauka obsługi systemu Linux/ terminala Mac**

# Nauka obsługi systemu Linux

**Ta część preworku powinna Ci zająć około 15 godzin.**

# Nauka obsługi systemu Linux / terminala Mac

**Prawie cały Internet jest oparty o systemy typu Linux. Tak samo jak wasza maszyna wirtualna. Dlatego musisz nauczyć się chociaż podstaw korzystania z tego systemu (i to używając tylko konsoli):**

- 1. Obejrzyj wideotutoriale:** Najłatwiej Ci będzie nauczyć się pracy w konsoli Linuxowej patrząc jak ktoś z niej korzysta. Dlatego obejrzyj <https://goo.gl/C0z6KD>
- 2. Przeczytaj prezentację "Podstawy Linuxa" i zrób ćwiczenia do niej:** Prezentacje jeszcze raz przeprowadzi Cię przez podstawy korzystania z Linuxa i wprowadzi też z bardziej zaawansowane części które mogą Ci się kiedyś przydać.

**IDE - czyli twoje  
przyszłe narzędzie  
pracy**



# IDE

## Czym jest IDE?

IDE (ang. Integrated Development Environment) jest aplikacją, która wspomaga proces pisanie kodu.

Dobre IDE ma wiele funkcji, najważniejsze z nich to:

- **Zarządzanie projektami** – IDE pomoże nam utrzymywać porządek w naszych projektach, odseparowując je od siebie.
- **Autouzupełnianie kodu** – IDE będzie umiało podpowiedzieć nam kod, który chcemy napisać. Musimy jednak pamiętać, że IDE nie jest bezbłędne, czasami może źle podpowiadać.
- **Debugger** – IDE powinno wspomagać proces wyszukiwania błędów

# IDE

## Najpopularniejsze IDE

Dla języka Java istnieje wiele IDE. Najpopularniejsze z nich to:

- **IntelliJ IDEA** – uznawane za najpotężniejsze IDE, inteligentne i efektywne, płatne do zastosowań komercyjnych. Z niego będziemy korzystali na zajęciach.
- **NetBeans** – darmowe IDE ze wsparciem dla Swing GUI Builder (wizualny edytor do biblioteki graficznej SWING).
- **Eclipse** – darmowe IDE zarówno do użytku prywatnego jak i komercyjnego.
- **STS** – darmowe IDE oparte na Eclipse z wtyczką wspierającą programowanie z użyciem frameworka Spring.

# IntelliJ IDEA

- Podczas kursu będziemy korzystać z IDE **IntelliJ IDEA** w wersji Ultimate, która ma wsparcie dla **Javy EE**, **Springa** oraz tworzenia aplikacji webowych (**HTML**, **JS**).  
Wszystkie te funkcjonalności są wbudowane w platformę **IntelliJ**, dzięki czemu nie musimy już doinstalowywać żadnych dodatkowych pluginów.
- Środowisko to ma bardzo dużo przydatnych funkcji i jego dobre opanowanie bardzo pomaga i przyspiesza tworzenie oprogramowania.
- W **IntelliJ** wiele czynności można wykonywać przy pomocy skrótów klawiszowych. Poniżej znajdziesz link do ich listy:

[https://resources.jetbrains.com/storage/products/intellij-idea/docs/IntelliJIDEA\\_ReferenceCard.pdf](https://resources.jetbrains.com/storage/products/intellij-idea/docs/IntelliJIDEA_ReferenceCard.pdf)

# IntelliJ IDEA

- Podczas przygotowywania komputera do pracy, instalowaliśmy **IntelliJ** w wersji testowej. Pozwala ona na korzystanie z IDE przez 30 dni. Po tym czasie program będzie wyświetlał informację o konieczności wprowadzenia kodu aktywacyjnego, który umożliwi dalsze używanie wersji Ultimate.
- Indywidualny kod aktywacyjny zostanie przesłany drogą mailową – po jego zarejestrowaniu otrzymacie półroczną licencję na korzystanie z **IntelliJ**.
- Jeżeli jednak wolisz używać innego środowiska, to nic nie stoi na przeszkodzie. Pamiętaj jednak że wykładowca może nie być w stanie pomóc Ci w szczególnych przypadkach (takich jak np. debugowanie czy instalacja potrzebnych wtyczek).

# IntelliJ IDEA

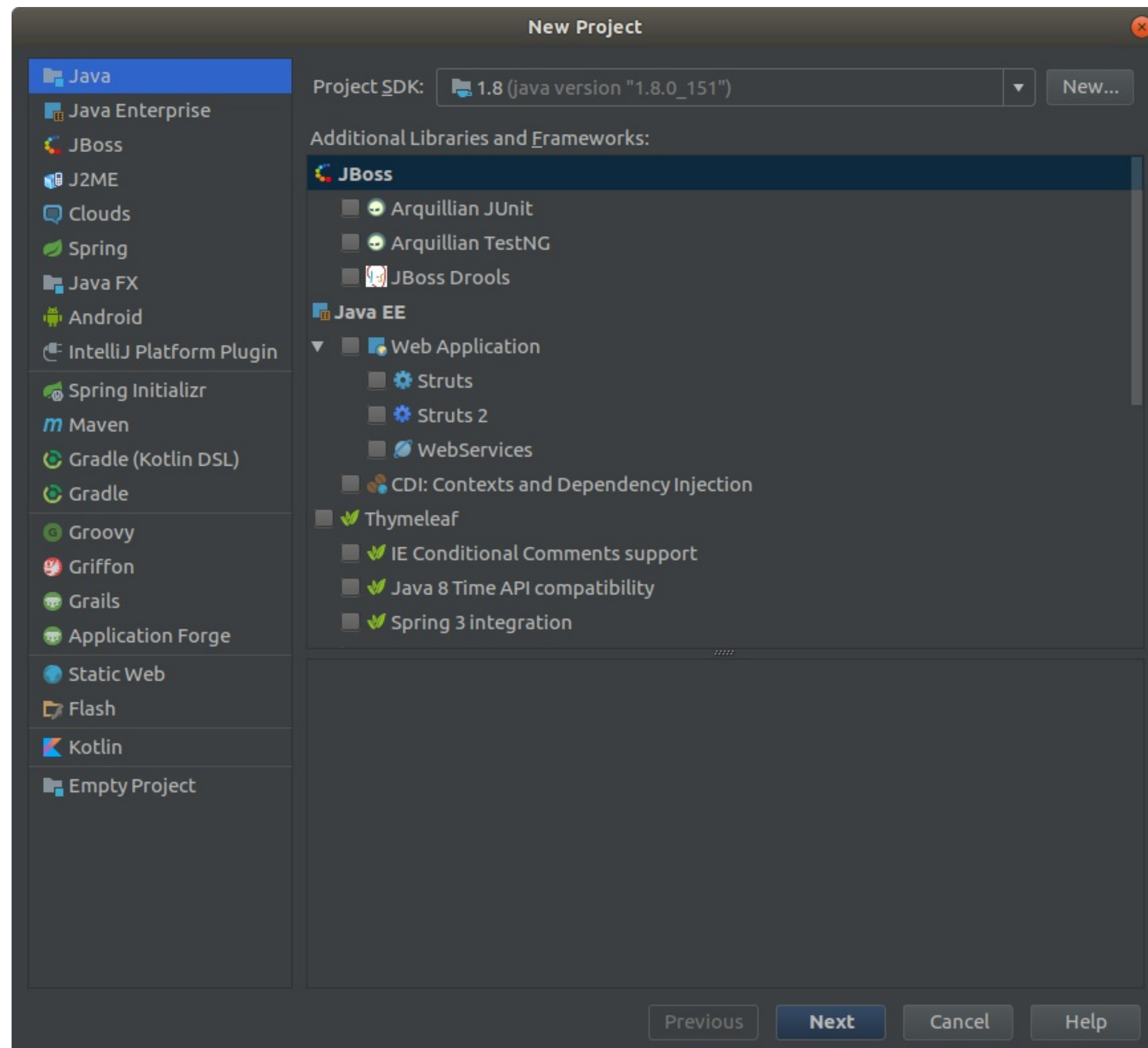
- **IntelliJ** to IDE, które zostało w bardzo wysokim stopniu dostosowane do pisania kodu w języku Java.
- Jest popularne wśród programistów i bardzo często stosowane przez firmy do użytku komercyjnego.
- Wydawca **IntelliJ** posiada też wiele środowisk, przystosowanych do pisania kodu w innych językach (także z bardzo dobrym wsparciem).
- Ogromną zaletą są inteligentne podpowiedzi, udoskonalona statyczna kontrola kodu, wsparcie dla wielu frameworków, współpraca z Gitem i wiele innych.
- Dokładne informacje na temat **IntelliJ** – instalacji, tworzenia nowego projektu, importowania już stworzonych itd., znajdziesz na stronie producenta.  
Dowiesz się tam także o wielu przydatnych sposobach na efektywne wykorzystanie narzędzia, które obecnie jest uznawane za najlepsze do rozwijania projektów pisanych w **Java**:

<https://www.jetbrains.com/idea/documentation/>

# **Podstawy korzystania z IntelliJ**



# IntelliJ – nowy projekt

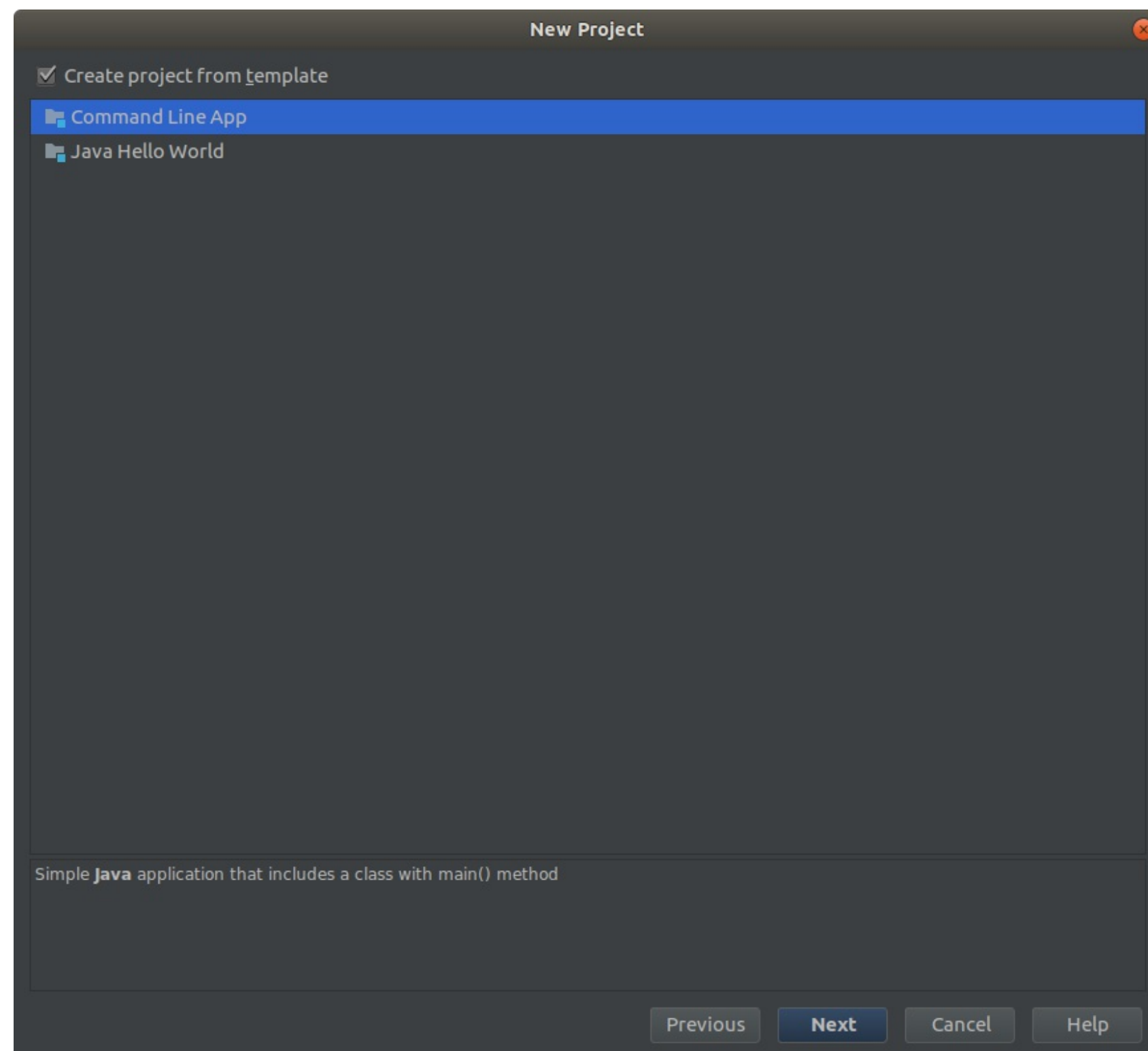


## Utworzenie nowego projektu

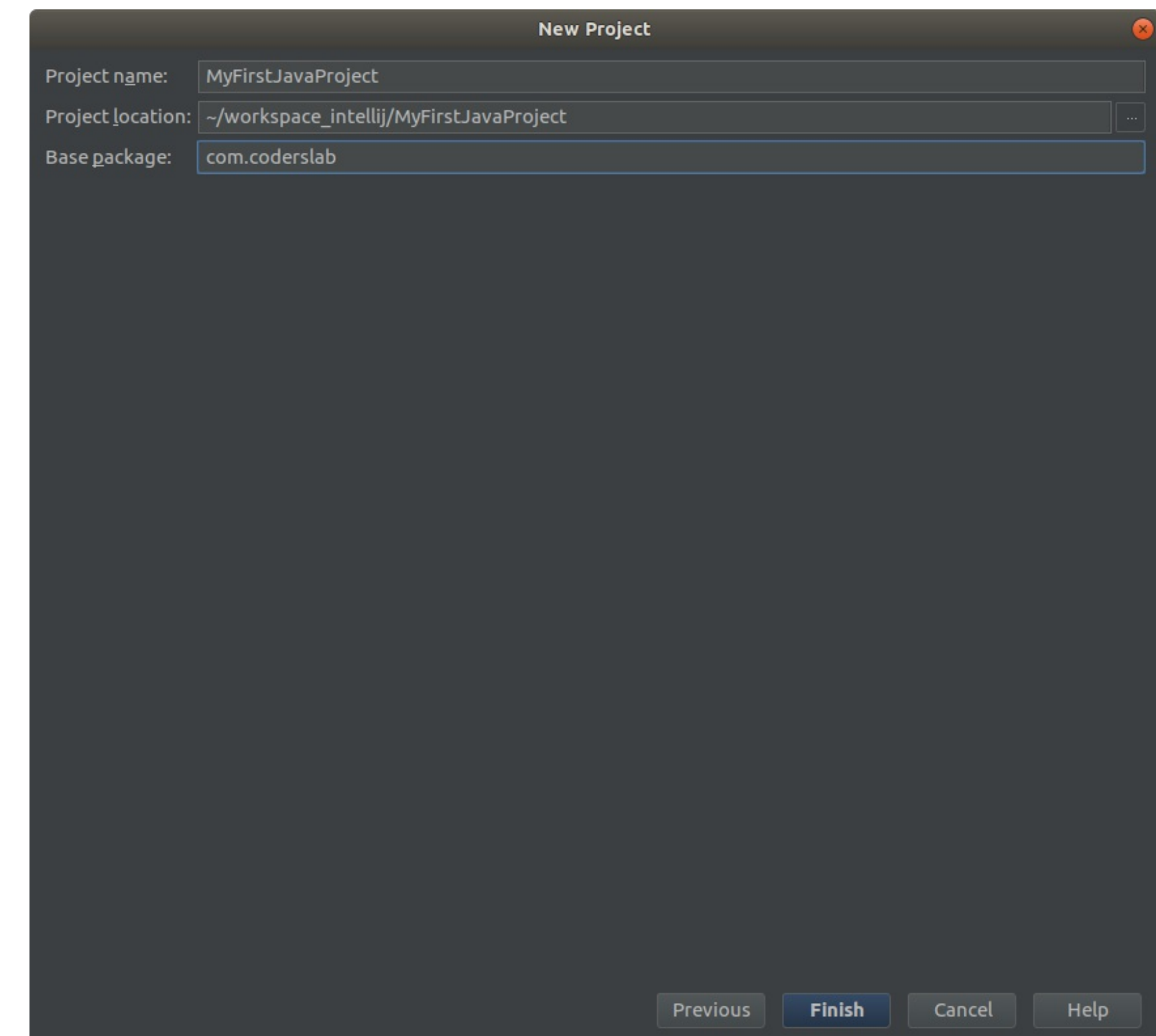
- Z paska narzędzi otwórz menu **File**
- Wybierz **New -> Project...**
- Pojawi się okno jak po lewej stronie, przejdź dalej, klikając przycisk **Next**.



# IntelliJ – nowy projekt

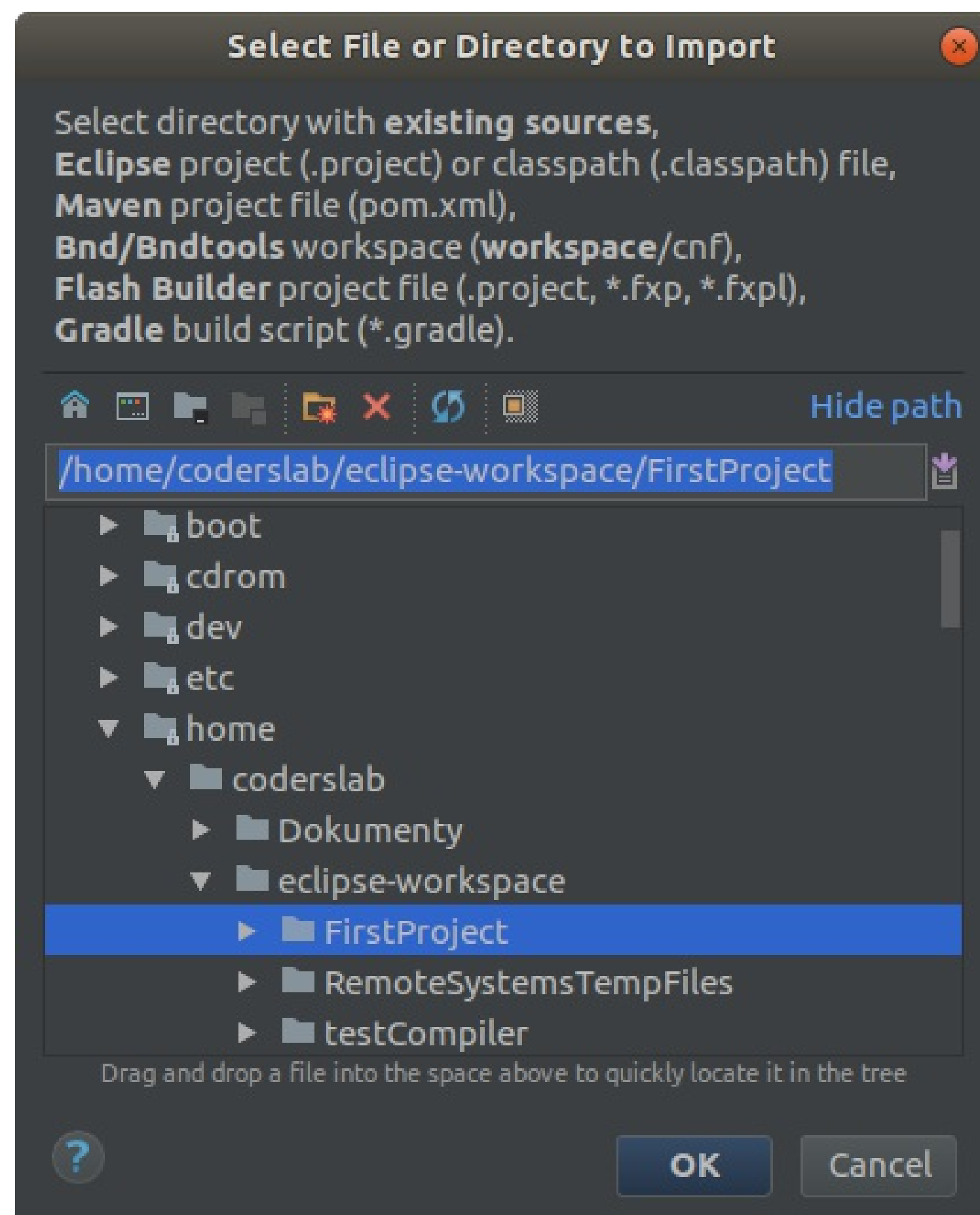


Wybierz opcję **Create project from template** i wciśnij **Next**.



Wpisz nazwę projektu i wybierz jego lokalizację, a następnie zatwierdź utworzenie nowego projektu klikając **Finish**.

# IntelliJ – nowy projekt z istniejących źródeł

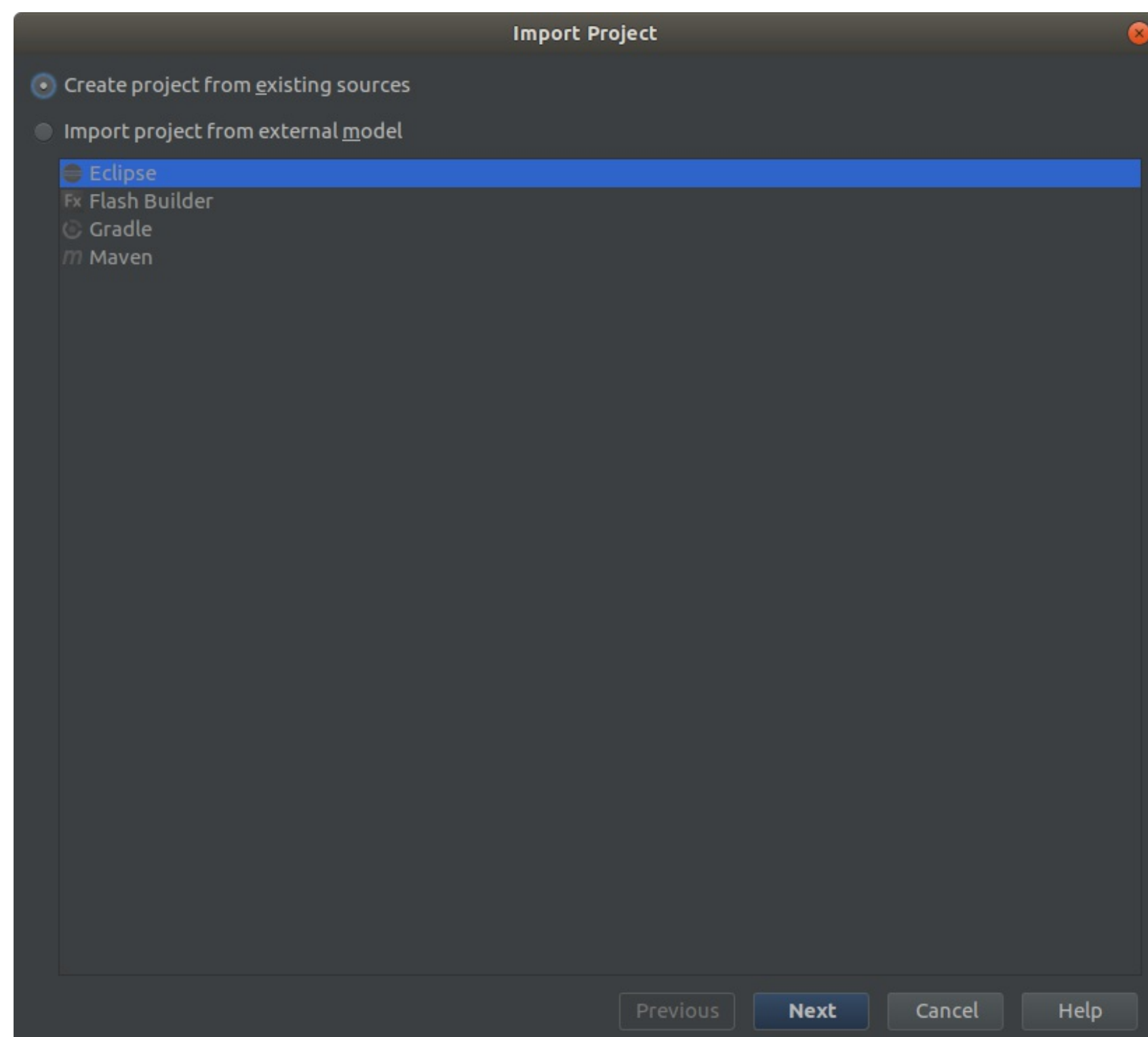


## Utworzenie nowego projektu z istniejących źródeł

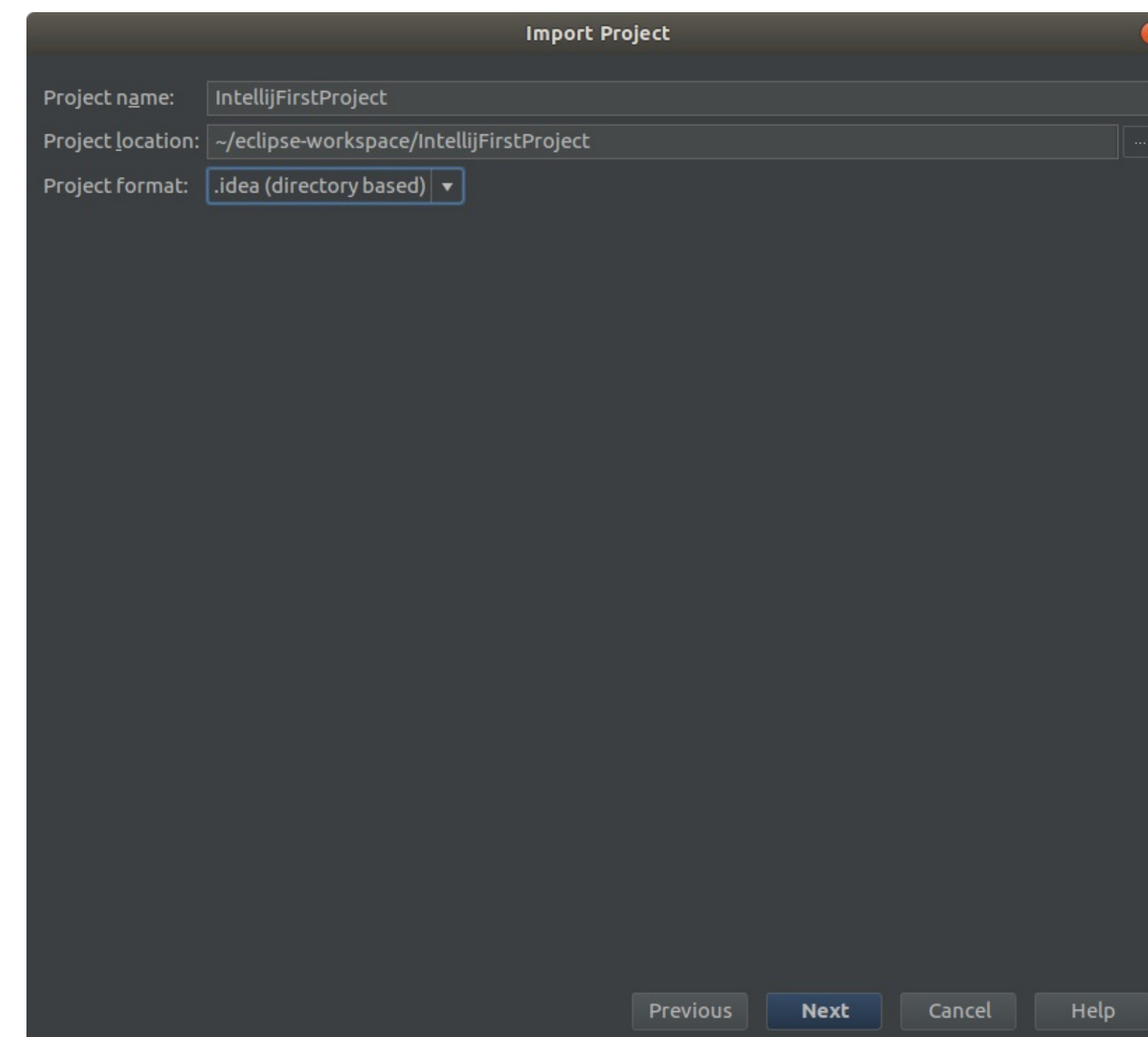
Z menu **Files** wybierz opcję **New -> Project from Existing Sources...**

Pojawi się okno jak po lewej stronie, należy wybrać folder źródłowy, z którego chcemy utworzyć nowy projekt.

# IntelliJ – nowy projekt z istniejących źródeł

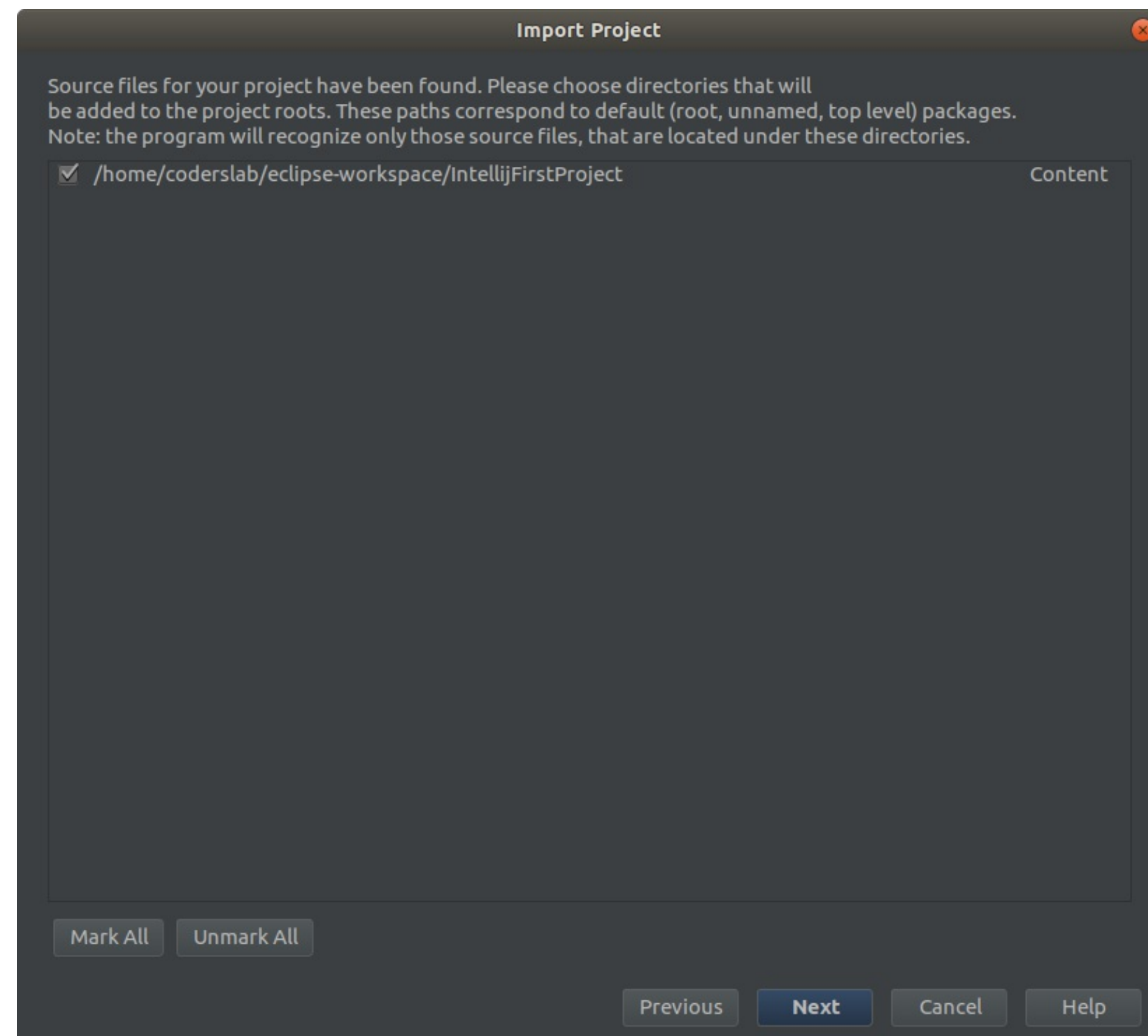


W kolejnym oknie potwierdzamy tworzenie projektu, wciskając **Next**.

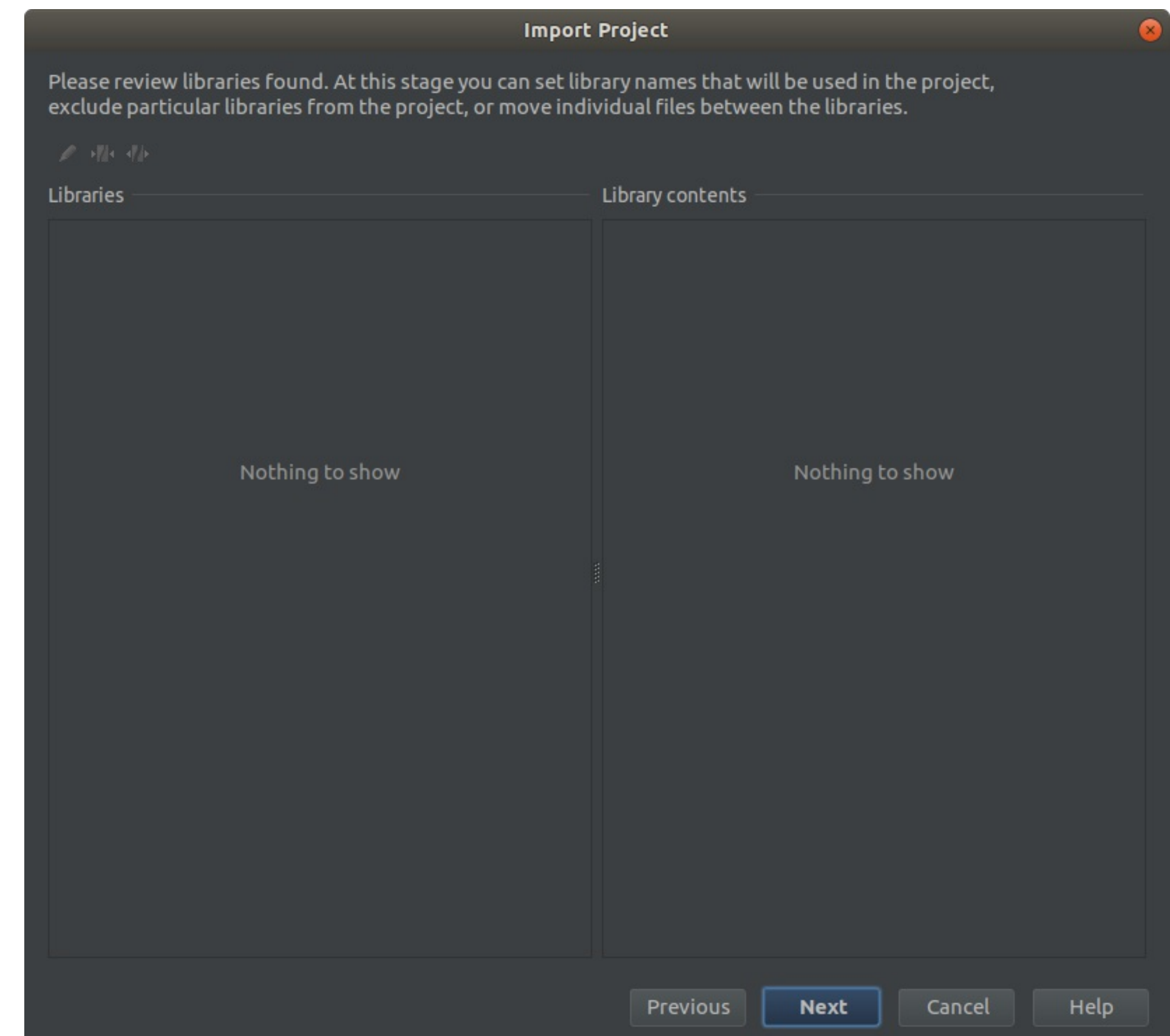


Określamy nazwę projektu, jego lokalizację i klikamy **Next**.

# IntelliJ – nowy projekt z istniejących źródeł



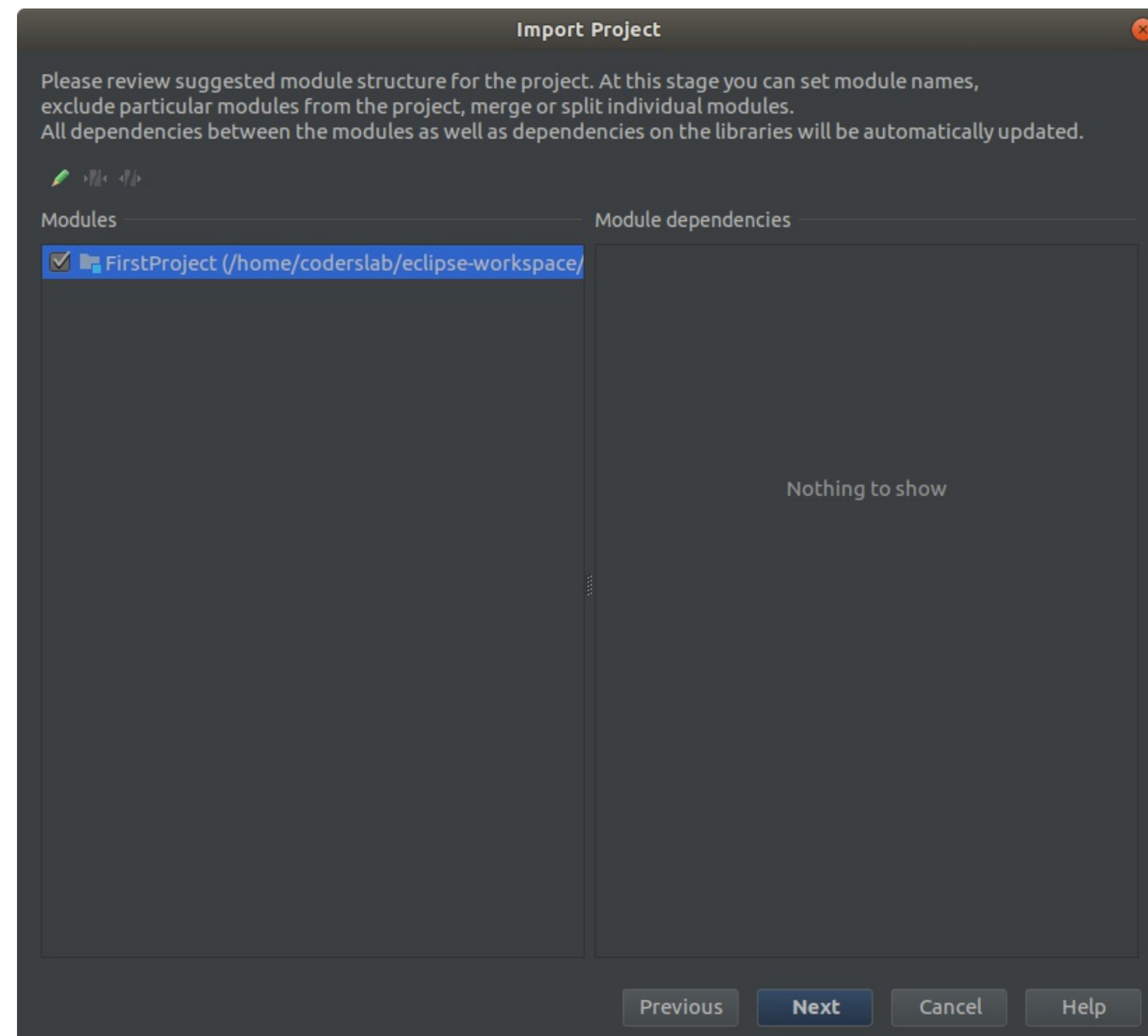
-> Next



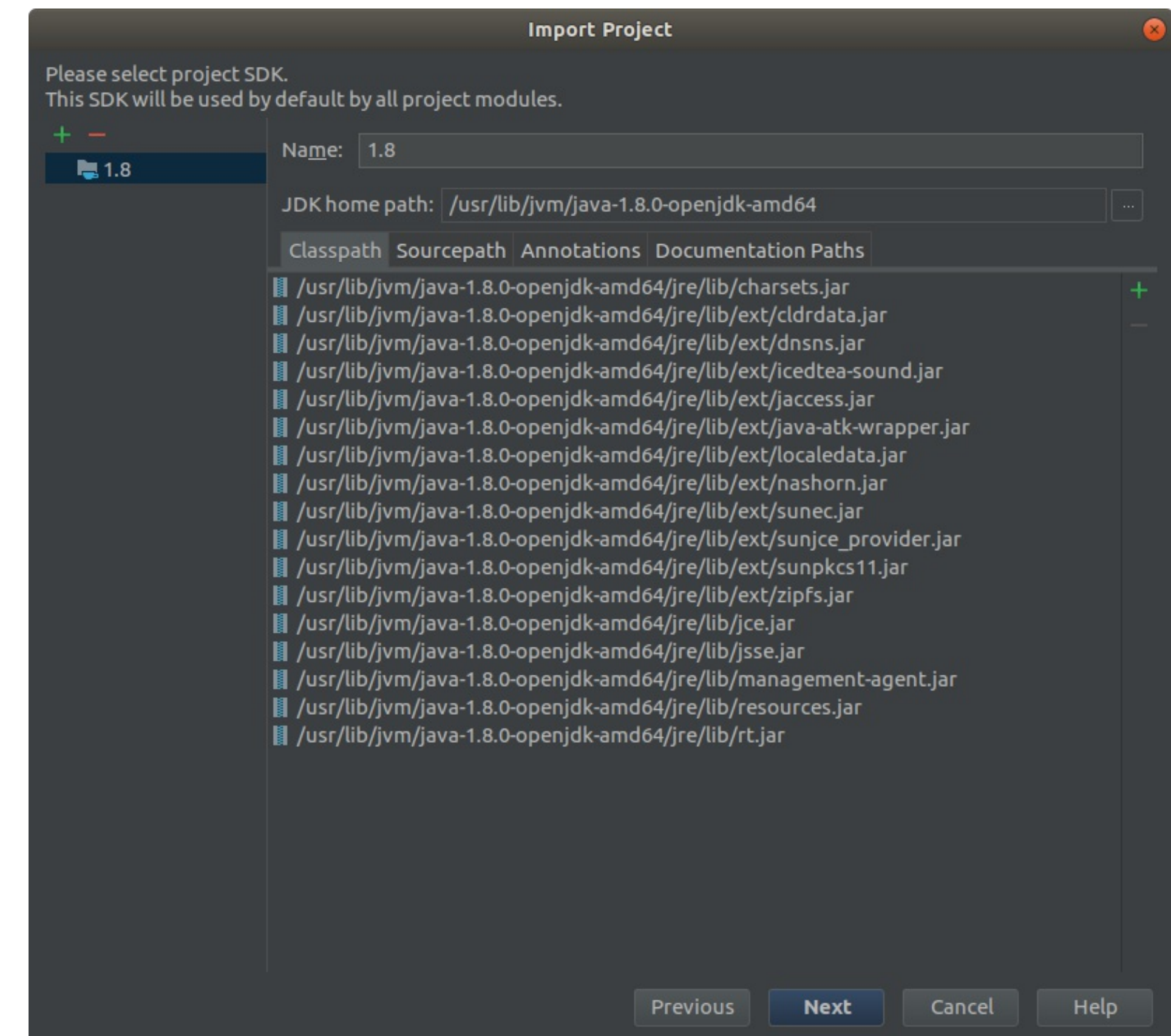
-> Next



# IntelliJ – nowy projekt z istniejących źródeł

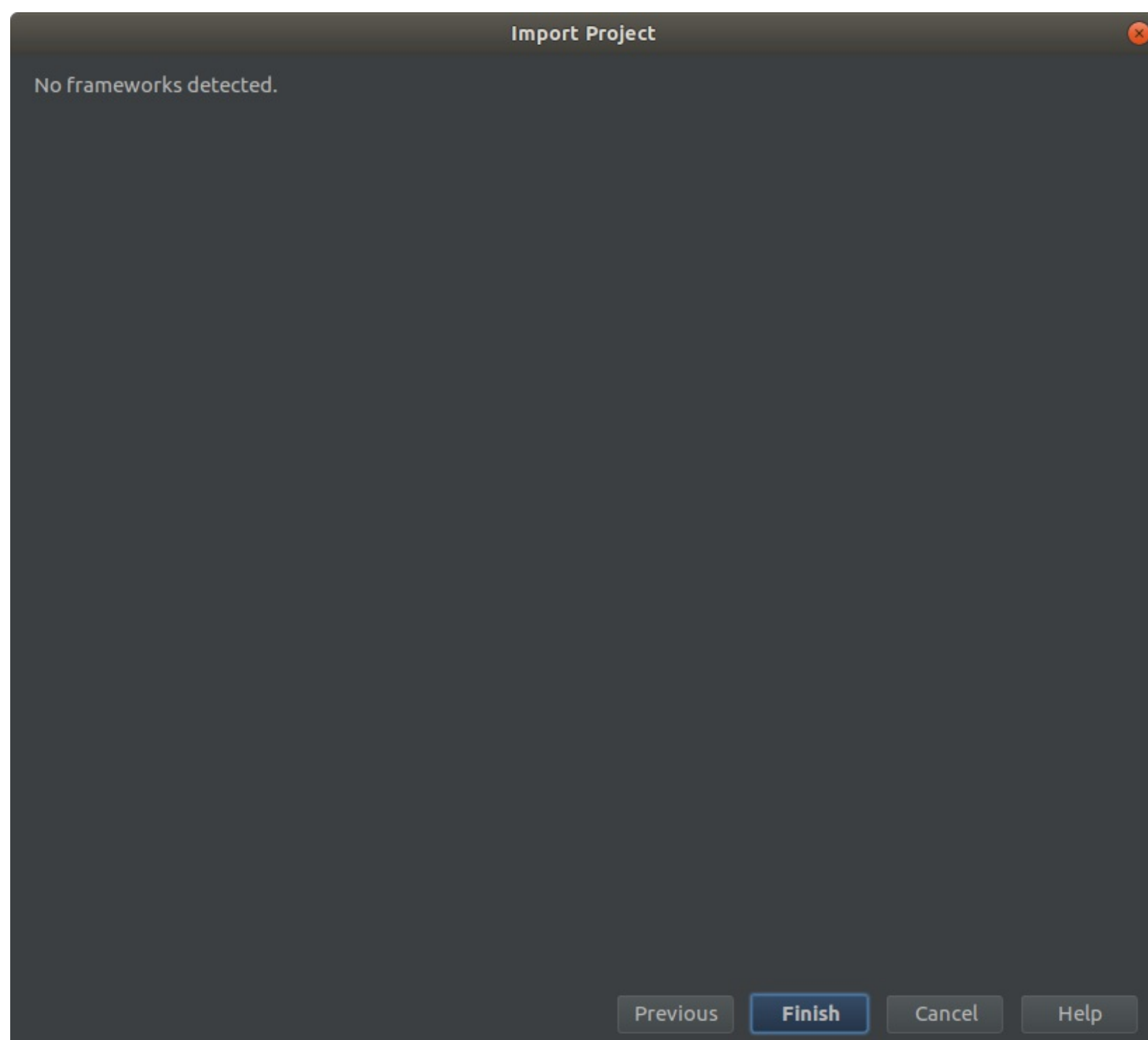


-> Next

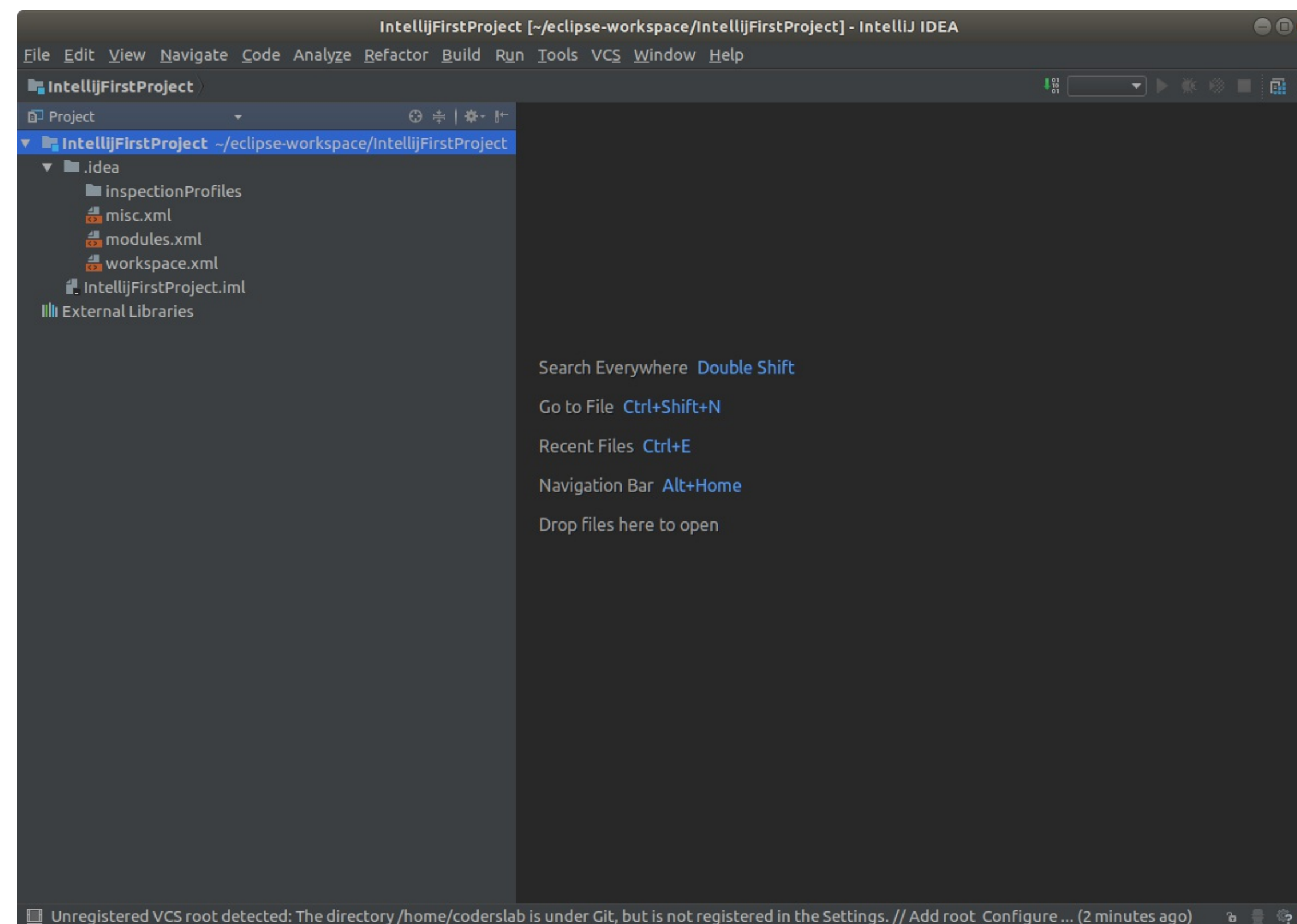


-> Next

# IntelliJ – nowy projekt z istniejących źródeł



Zatwierdzamy utworzenie nowego projektu klikając **Finish** i wybierając czy ma się otworzyć w nowym czy w istniejącym oknie.



Powinno pojawić się nam nowe okno z utworzonym projektem.



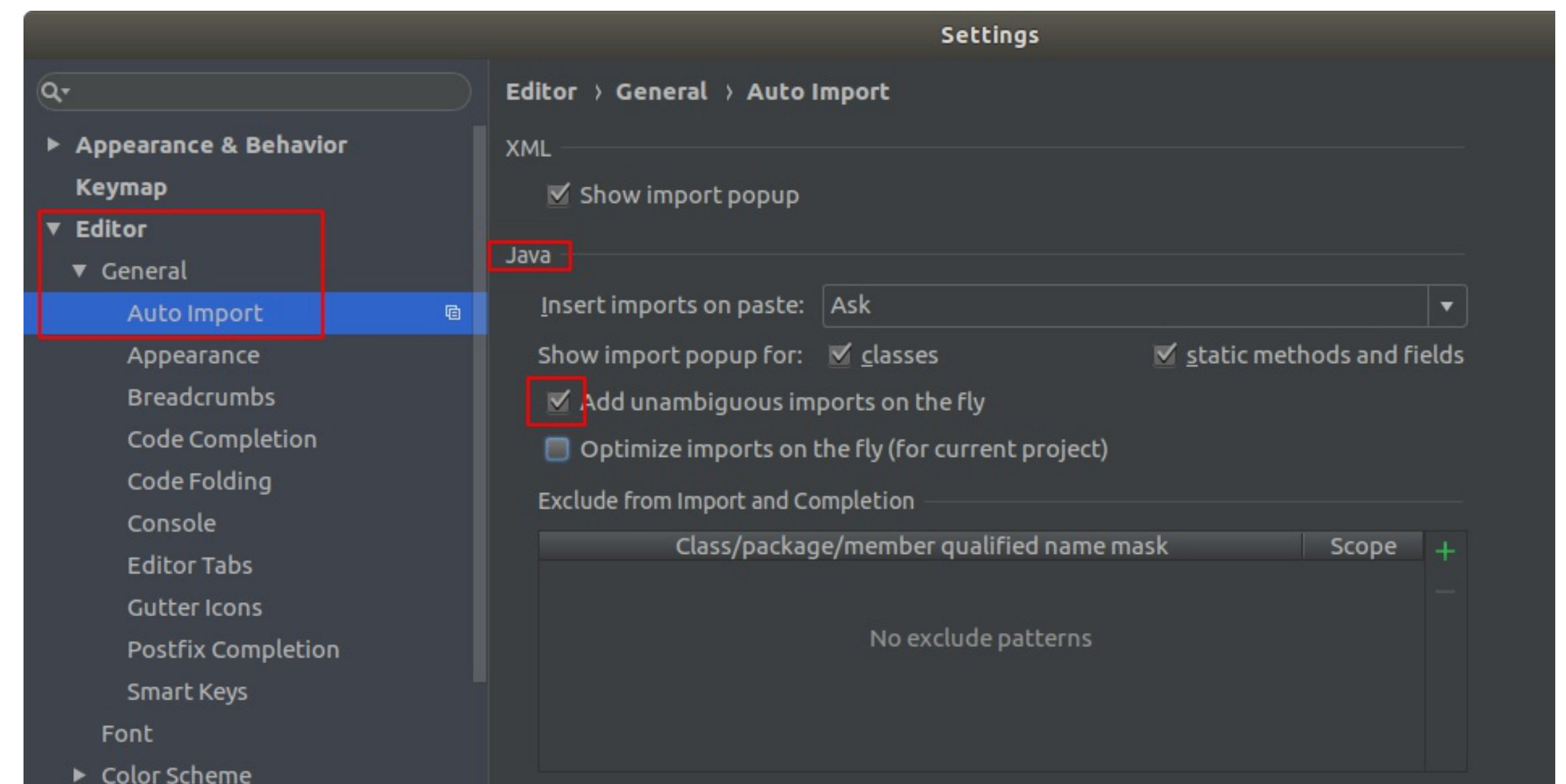
# IntelliJ – ustawienia projektu

Dużym usprawnieniem pracy będzie ustawienie w IDE automatycznego dodawania importów – wtedy nie musimy ich ręcznie dodawać do klasy. Import nie zostanie dodany, jeśli IntelliJ nie będzie w stanie jednoznacznie stwierdzić, o który chodzi. Wtedy podpowiada listę możliwych bibliotek do zaimportowania, z których wybieramy właściwą. Automatyzację dodawania importów włączamy wybierając:

**File → Settings... → Editor → General → Auto Import**

Tam zaznaczamy opcję:

**Add unambiguous imports on the fly**

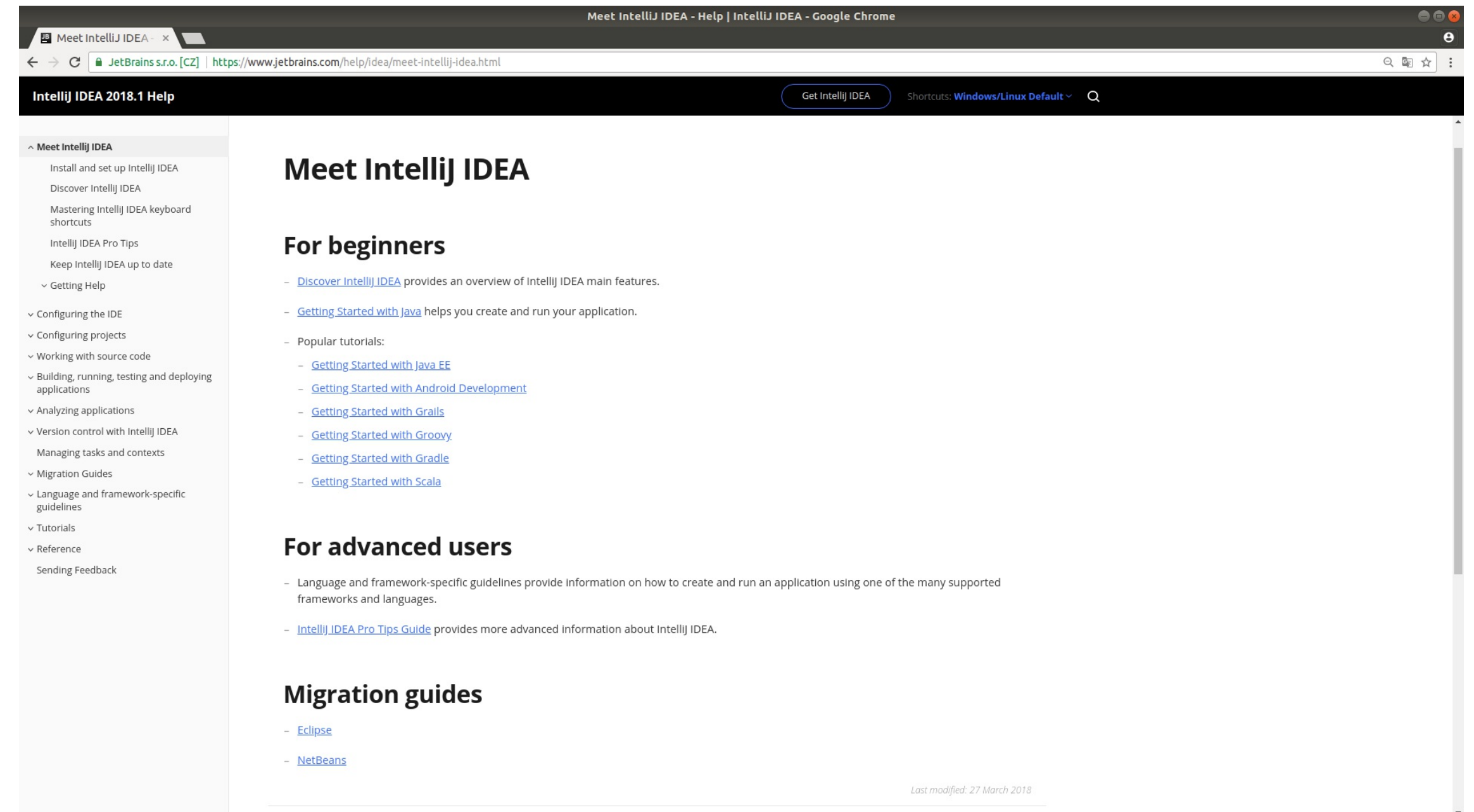


# IntelliJ – dokumentacja

## Przewodnik po IntelliJ

Jeśli chcesz zapoznać się dokładniej z programem **IntelliJ**, to pod tym linkiem znajdziesz tutorial:

<https://www.jetbrains.com/help/idea/meet-intellij-idea.html>





**Co możesz  
zrobić jeżeli  
zostało Ci  
trochę czasu.**

# Powtórka z HTML i CSS

Co prawda na kursie uczymy się programowania serwerowego, ale dobrze jeżeli programista backendowy zna też technologie frontendowe. Dlatego przyswój następującą wiedzę:

- <http://flukeout.github.io/> - tutorial, który w łatwy sposób wprowadzi Cię w język znaczników CSS. Ta wiedza będzie bardzo potrzebna w 3cim tygodniu kursu.
  - <https://css-tricks.com/snippets/css/a-guide-to-flexbox/> - tutorial który wytłumaczy na czym polega Flexbox (technologia wprowadzona w CSS3)
  - <http://getbootstrap.com/getting-started/> - bootstrap to framework CSS dzięki któremu łatwiej zrobisz wygodną i responsywną stronę
  - <http://foundation.zurb.com/sites/getting-started.html> - Foundation to konkurencja dla Bootstrapa.
- Wybierz ten framework który bardziej Ci się podoba

**Miejsca które  
mogą Cię  
zainteresować**

# Strony związane z HTML i CSS

Tutaj znajdziesz strony z informacjami na temat HTML i CSS:

- <http://flukeout.github.io/> - tutorial który w łatwy sposób wprowadzi Cię w język znaczników CSS.
- <https://css-tricks.com/snippets/css/a-guide-to-flexbox/> - tutorial który wytłumaczy na czym polega Flexbox (technologia wprowadzona w CSS3)

- <http://getbootstrap.com/getting-started/> - bootstrap to framework CSS dzięki któremu łatwiej zrobisz wygodną i responsywną stronę
- <http://foundation.zurb.com/sites/getting-started.html> - Foundation to konkurencja dla Bootstrapa.

Wybierz ten framework który bardziej Ci się podoba

# Strony związane z Java

Strony na których znajdziesz wiele fajnych informacji na temat Java. Warto żebyś zaglądał na nie często, także podczas kursu.

- <http://docs.oracle.com/javase/8/docs/api/index.html> - specyfikację API dla platformy Java Standard Edition.
  - <https://docs.oracle.com/javase/tutorial/index.html> - praktyczne przewodniki dla programistów, którzy chcą korzystać z języka programowania Java do tworzenia aplikacji.
- <http://spring.io/> - oficjalna strona frameworka Spring, którego będziecie uczyć się podczas kursu.
  - <http://spring.io/guides> - przewodniki i tutoriale związane ze Spring
  - <http://docs.oracle.com/en/java/> - dokumentacja

# Strony o tematyce ogólnej

Tutaj znajdziesz strony wartę odwiedzenia z wieloma przydatnymi informacjami o tematyce nie związanej z kursem:

- <http://sekurak.pl/> - magazyn o bezpieczeństwie w sieci.
- <http://wazniak.mimuw.edu.pl/> - strona prowadzone przez wykładowców UW. Można znaleźć na niej materiały z wielu wykładów przez nich prowadzonych.
- <https://blog.codinghorror.com/> - blog z przymrużeniem oka. Nie tylko o programowaniu, ale poświęcony tematyce IT.