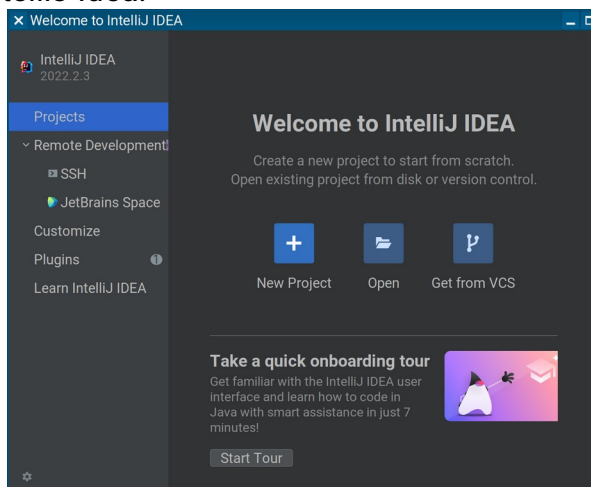
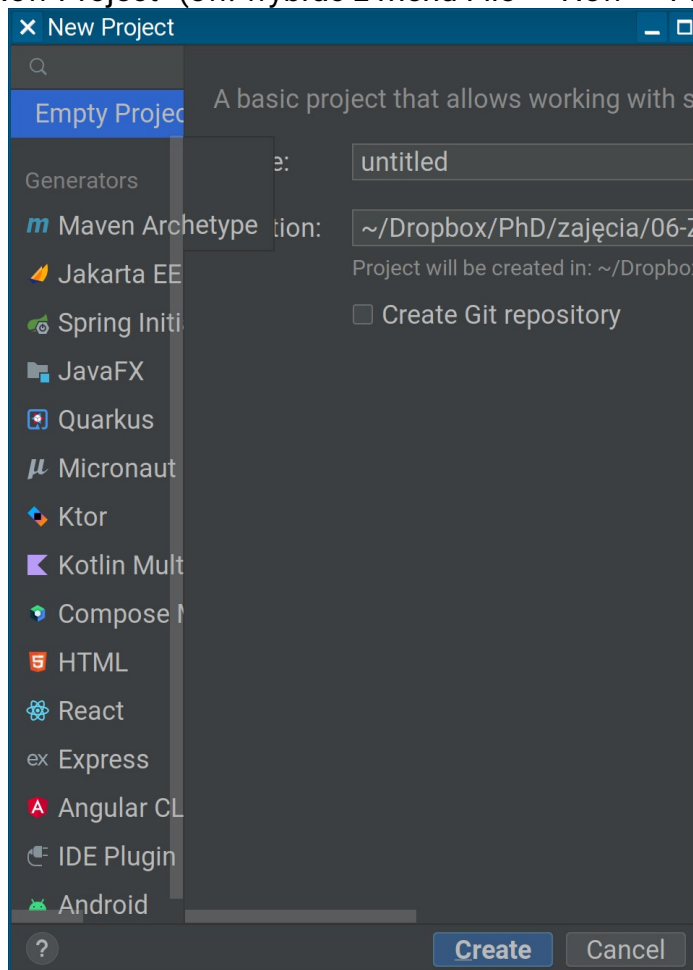


Ćwiczenie 1. Tworzenie projektu z archetypu Maven.

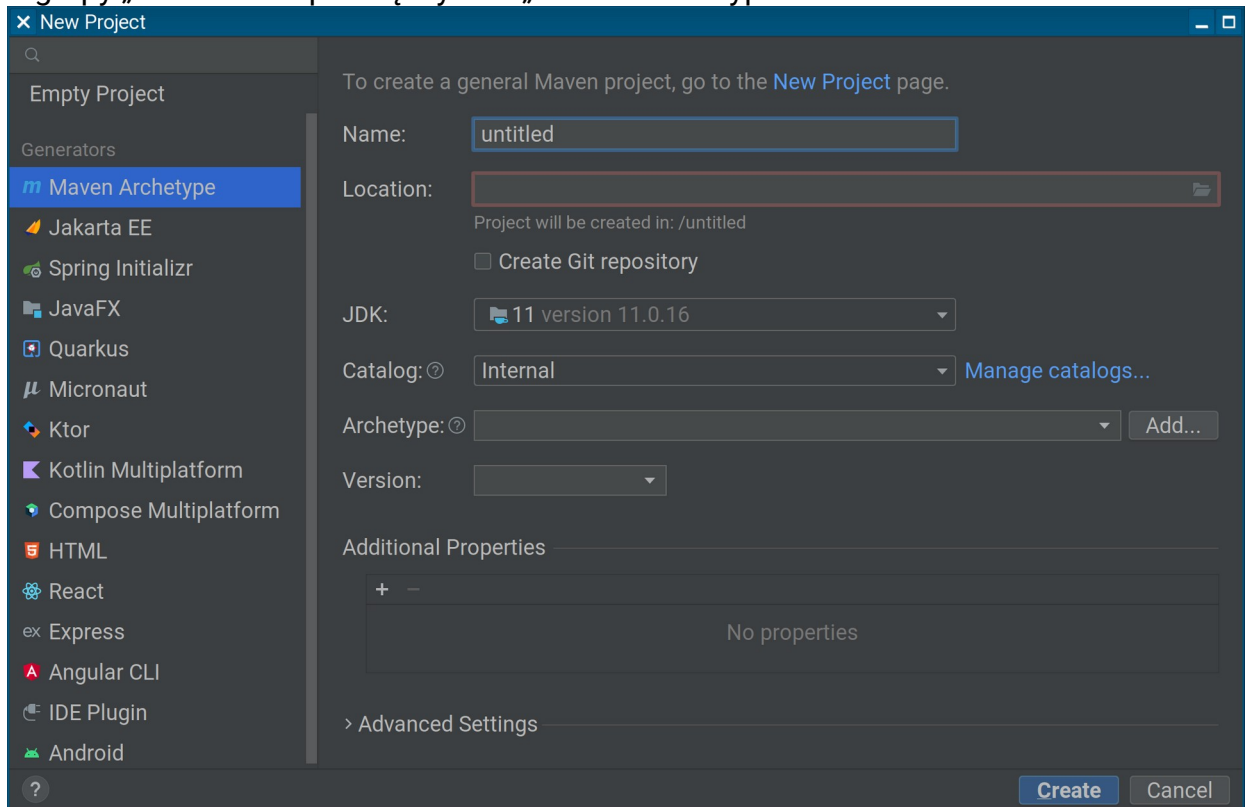
1. Proszę uruchomić IntelliJ Idea.



2. Proszę kliknąć „New Project” (ew. wybrać z menu File → New → Project)

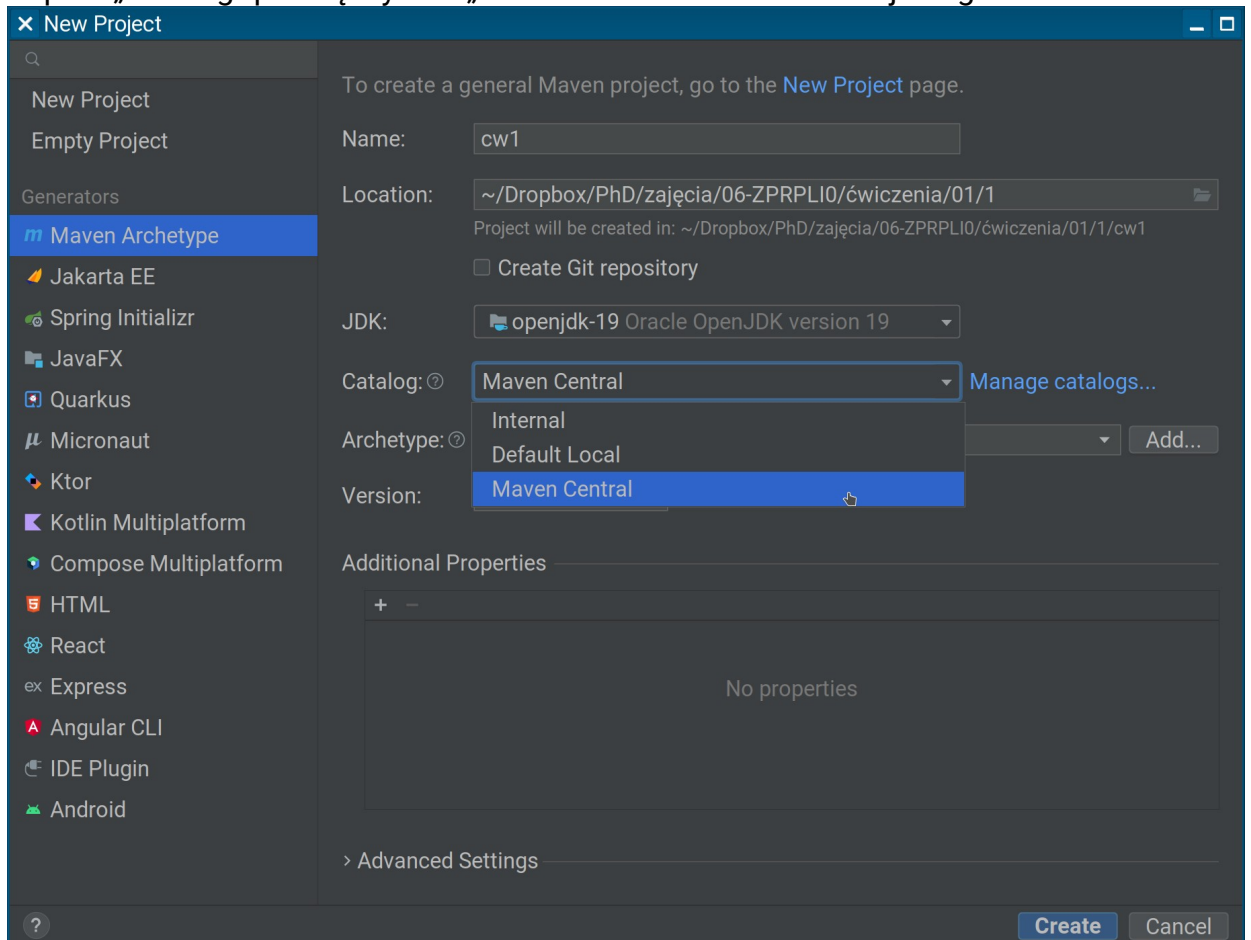


3. Z grupy „Generators” proszę wybrać „Maven Archetype”

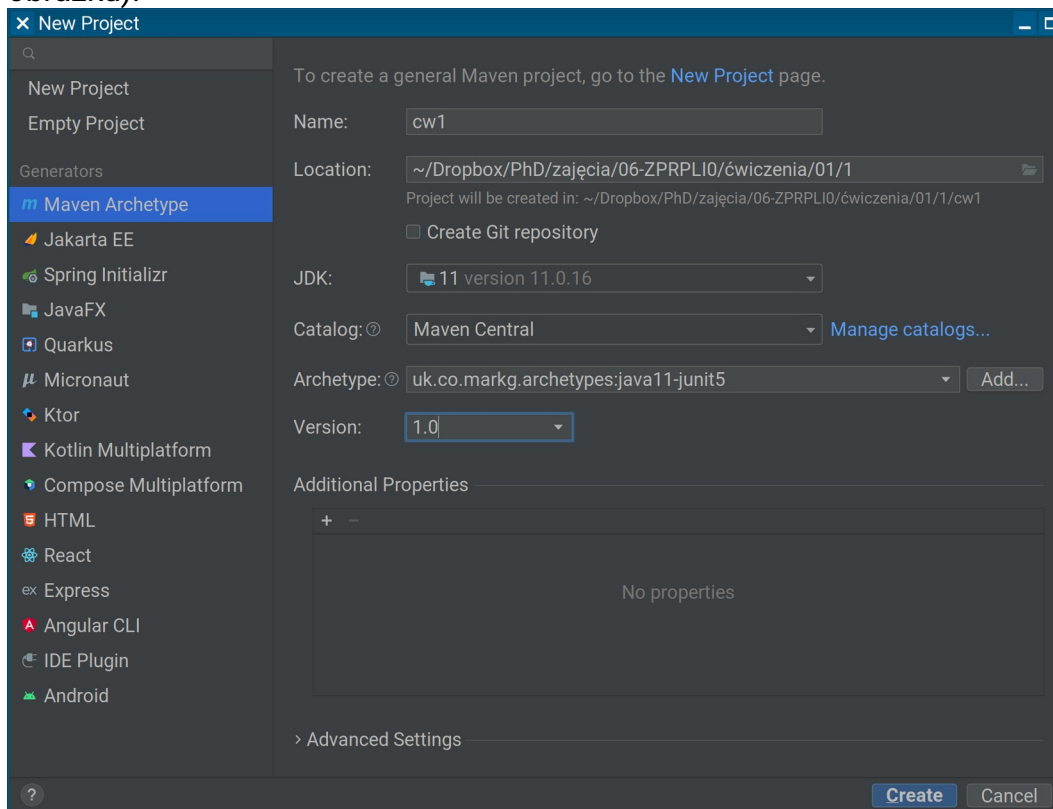


4. Proszę ustawić nazwę na „cw1”, a lokalizację pliku relatywnie względem repozytorium git na „<git>/ćwiczenia/01/1”.

5. W polu „Catalog” proszę wybrać „Maven Central” z menu rozwijalnego.



- Po pewnej chwili (archetypy muszą się ściągnąć) proszę kliknąć w listę rozwijalną w polu „Archetype” i wyszukać archetyp java11-junit5 (taki jak na załączonym obrazku).



- Proszę kliknąć na przycisku „Create”, aby utworzyć projekt.
- Dużo się powinno stać, IDEA pewnie wyświetli stado komunikatów. Zignorować.
- Proszę zapisać utworzony projekt w repozytorium git.

Potencjalne problemy:

- Projekt pokazuje wiele błędów (w tym że nie ma takich wtyczek). To ponieważ normalne. W zintegrowanym z IDE terminalu można wpisać komendę `mvn package` i zobaczyć czy działa. To powinno ściągnąć wszystkie potrzebne zależności, uruchomić załączek testu, kompilację i budowę paczki. Można też próbować ponownie załadować projekt (prawy klawisz myszy na pliku pom.xml, Maven → Reload Project). To niejednokrotnie pomaga.
- Maven nie działa / wyrzuca kryptyczne błędy (nie o braku zależności, te są dość jasne). Powodów może być wiele, ale najbardziej prawdopodobne:
 - nieprawidłowa wersja Maven (sprawdzić: `mvn -version`, lepiej mieć najnowszą)
 - nieprawidłowa / niezgodna wersja Javy (jw. `mvn -version` pokazuje z którego JDK korzysta, to powinno być możliwe najwyższe – będziemy używali funkcjonalności Javy 17+):
 - ustawić zmienną środowiskową `JAVA_HOME` na lokalizację odpowiedniego JDK (uwaga: katalog zawierający m.in. podkatalog `bin/`, a w nim komenda `javac`)
 - zainstalować odpowiednie JDK (polecam OpenJDK) jeżeli nie ma komendy `javac`
 - w ogóle nie ma komendy `mvn`
 - [zainstalować Mavena](#)
 - `mvn` jest, ale się nie uruchamia
 - sprawdzić i ustawić zmienną środowiskową `JAVA_HOME`