



**POLITECHNIKA
BYDGOSKA**

Wydział Telekomunikacji,
Informatyki i Elektrotechniki

CHMURY OBLICZENIOWE

Informatyka Stosowana

Instrukcja 01 - Tworzenie aplikacji Java Spring

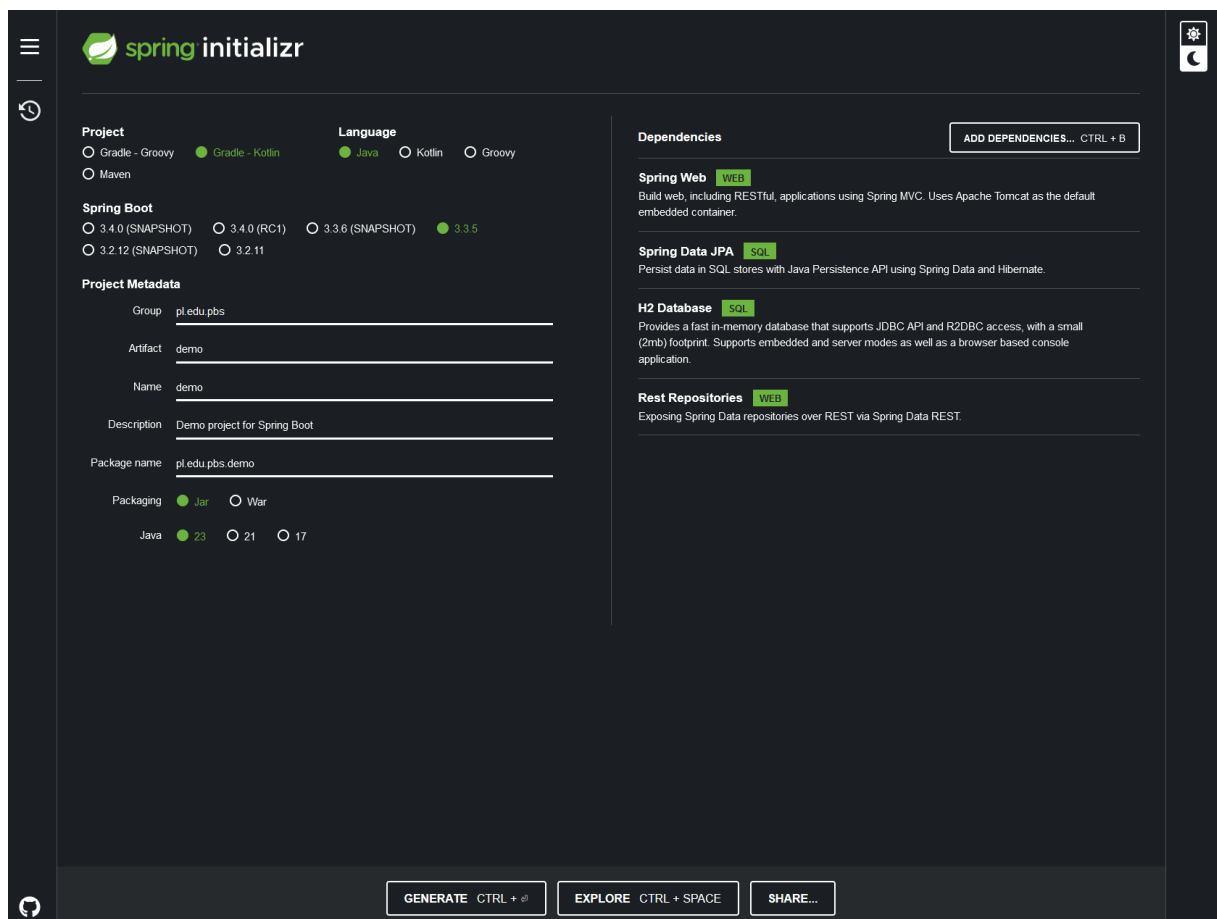
Kacper Krzyżniewski
Filip Pawłowski
Szymon Janiak

Cel laboratorium

Zapoznanie się z procesem inicjalizacji, konfiguracji oraz uruchomienia przykładowej aplikacji Java Spring. Dodatkowym celem jest praktyczne wykorzystanie narzędzia Spring Initializr oraz środowiska programistycznego IntelliJ IDEA do uruchomienia wygenerowanej aplikacji.

Przykładowe rozwiązanie

Pracę nad utworzeniem aplikacji Spring należy rozpocząć od wygenerowania podstawowej struktury projektu. W tym celu należy wykorzystać narzędzie Spring Initializr dostępne pod adresem <https://start.spring.io/>. Proces inicjalizacji przedstawiony został na rysunku 1.1.



The screenshot displays the Spring Initializr web application interface. The main configuration area is divided into several sections:

- Project:** Includes radio buttons for **Gradle - Groovy**, **Gradle - Kotlin** (selected), and **Maven**.
- Language:** Includes radio buttons for **Java** (selected), **Kotlin**, and **Groovy**.
- Spring Boot:** Includes radio buttons for various versions, with **3.3.5** selected.
- Project Metadata:** Includes input fields for **Group** (pl.edu.pbs), **Artifact** (demo), **Name** (demo), **Description** (Demo project for Spring Boot), and **Package name** (pl.edu.pbs.demo).
- Packaging:** Includes radio buttons for **Jar** (selected) and **War**.
- Java:** Includes radio buttons for **23** (selected), **21**, and **17**.
- Dependencies:** A section on the right with a button **ADD DEPENDENCIES... CTRL + B** and a list of selected dependencies: **Spring Web** (WEB), **Spring Data JPA** (SQL), **H2 Database** (SQL), and **Rest Repositories** (WEB).

At the bottom, there are three buttons: **GENERATE CTRL + G**, **EXPLORE CTRL + SPACE**, and **SHARE...**

Rysunek 1: Inicjalizacja projektu przy użyciu Spring Initializr

W formularzu generowania projektu należy określić następujące parametry:

- Project: Gradle - Kotlin
- Language: Java
- Spring Boot: 3.3.5
- Project Metadata:
 - Group: pl.edu.pbs
 - Artifact: demo
 - Name: demo
 - Description: Demo project for Spring Boot
 - Package name: pl.edu.pbs.demo
 - Packaging: Jar
 - Java: 23
- Dependencies:
 - Spring Web
 - Spring Data JPA
 - H2 Database
 - Rest Repositories

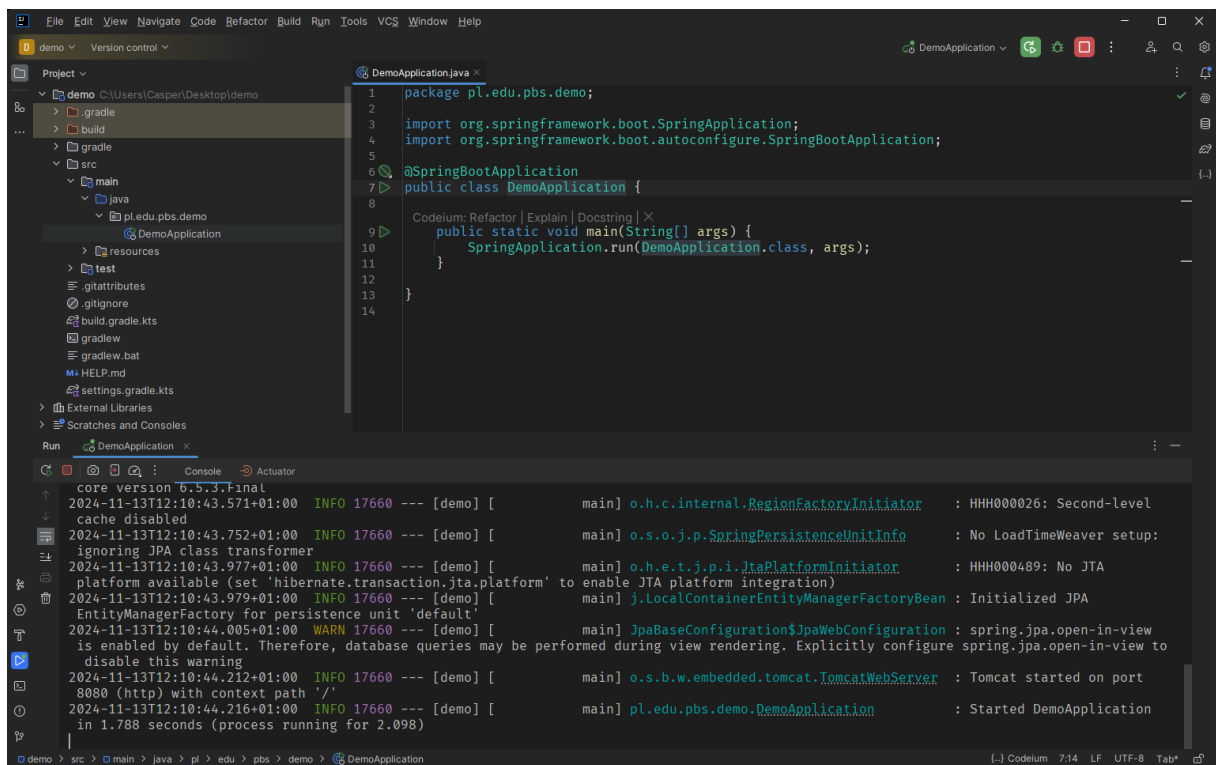
Po wygenerowaniu projektu, należy go zaimportować do środowiska IntelliJ IDEA. W tym celu należy:

- Rozpakować pobrany plik archiwalny ZIP
- W IntelliJ IDEA wybrać opcję "Open"
- Wskazać folder z rozpakowanym projektem
- Poczekać na zaindeksowanie projektu i pobranie zależności

Aby uruchomić aplikację należy:

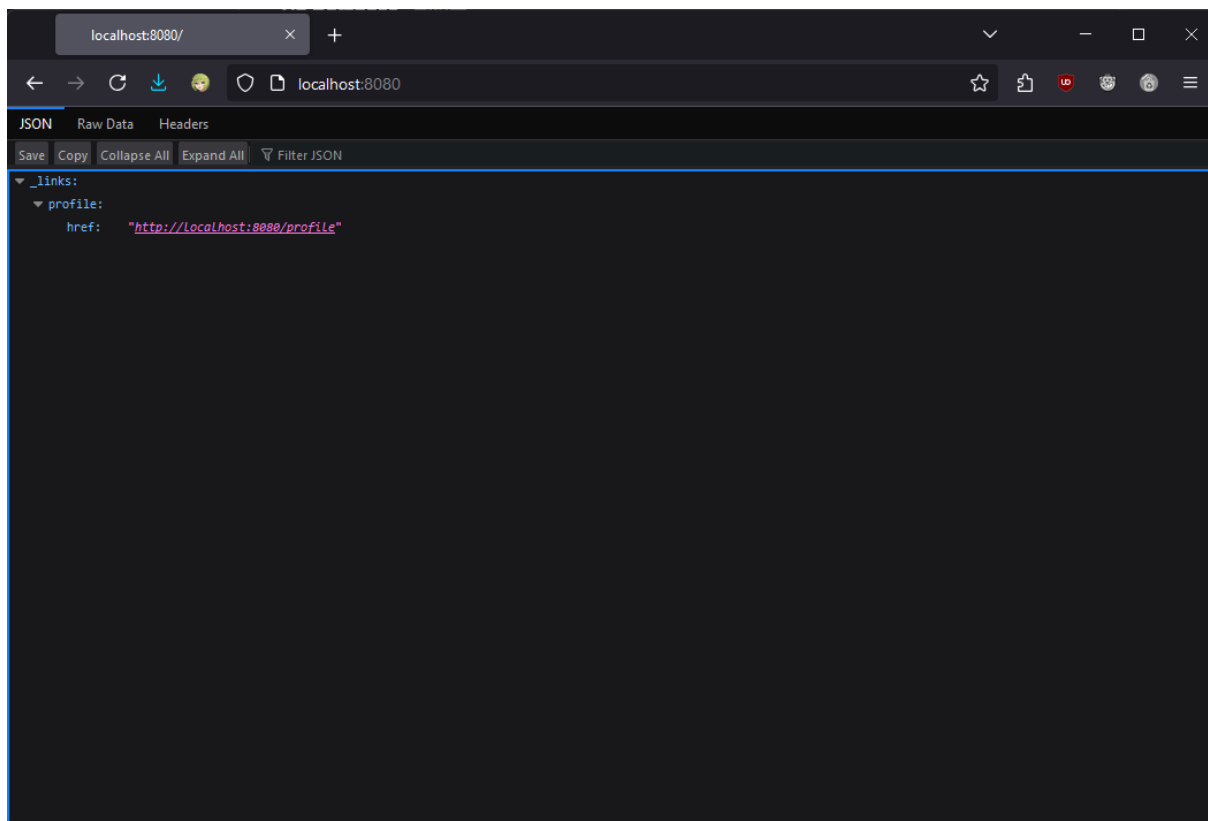
- Odnaleźć klasę główną z adnotacją @SpringBootApplication
- Kliknąć zielony przycisk "Run" obok metody main
- Poczekać na uruchomienie serwera (domyślnie na porcie 8080)

Rezultat poprawnego otwarcia i uruchomienia projektu w środowisku IDE przedstawiony został na rysunku 1.2.



Rysunek 2: Otwarty projekt w środowisku IntelliJ IDEA

Po wykonaniu powyższych kroków, aplikacja powinna być dostępna pod adresem `http://localhost:8080`, co zostało przedstawione na rysunku 1.3.



Rysunek 3: Widok środowiska REST w przeglądarce