

# Laboratorium 1 – 03.10.2018

## Zadania na rozgrzewkę

1. Wygenerować projekt Spring Boot na <http://start.spring.io> i zaimportować go do IntelliJ Idea.
2. Utworzyć bean `HelloWorld`, który po zainicjalizowaniu wypisze „Hello world” na konsolę.
3. Utworzyć bean `GoodbyeWorld`, który przed zakończeniem działania aplikacji wypisze „Goodbye” na konsolę.

## Zadania DI

4. Toolbox:
  - 4.1. Utworzyć interfejs `Tool`.
  - 4.2. Utworzyć beana `Toolbox`. Uruchomić aplikację, upewnić się, że bean jest tworzony.
  - 4.3. Wstrzyknąć do `Toolbox` beana typu `Tool`. Uruchomić aplikację. Jaki jest wynik?
  - 4.4. Utworzyć beana, który implementuje interfejs `Tool`. Uruchomić aplikację, upewnić się, że bean jest tworzony.
  - 4.5. Utworzyć więcej beanów implementujących `Tool`. Uruchomić aplikację. Jaki jest wynik?
  - 4.6. Zmienić `Toolbox` tak, żeby wstrzyknięta była konkretna implementacja `Tool` – podstawowe narzędzie. (Można to osiągnąć na dwa sposoby!)
  - 4.7. Zmienić kod `Toolbox` tak, żeby oprócz podstawowego narzędzia wstrzykiwane były wszystkie beany typu `Tool`.
  - 4.8. Ustawić beany na liście w kolejności odwrotnej do alfabetycznej.
  - 4.9. Zmienić kod tak, żeby podstawowe narzędzie było zależne od profilu (np. inne dla hydraulika, inne dla stolarza.) Sprawdzić działanie aplikacji w zależności od profilu.
  - 4.10. Stworzyć beana typu `Tool` za pomocą anonimowej klasy.
    - 4.10.1. (opcjonalnie) dodać metodę, za pomocą której beany `Tool` będą się „przedstawiać”. Utworzyć domyślną implementację tej metody w interfejsie.
5. Ying-Yang
  - 5.1. Utworzyć beany `Ying` i `Yang`.
  - 5.2. Wstrzyknąć `Ying` do `Yang`, a `Yang` do `Ying`:
    - 5.2.1. Używając anotacji na konstruktorze,
    - 5.2.2. Używając anotacji na setterach,
    - 5.2.3. Używając anotacji na polach.
  - 5.3. Jak zachowa się aplikacja w każdym z tych przypadków? Dlaczego?
  - 5.4. Jak zachowa się aplikacja w przypadku różnych kombinacji?
6. Singletony/Prototypy
  - 6.1. Utworzyć beana `SingletonBean` ze scope singleton. (Czy potrzebna jest dodatkowa anotacja?)

6.2. Utworzyć beana `PrototypeBean` ze scope `prototype`.

6.3. Utworzyć beana `ScopesShowcase`:

6.3.1. Do `ScopesShowcase` wstrzyknąć `SingletonBean` i `PrototypeBean`,

6.3.2. Stworzyć metodę `showcase()`, która pokaże, jakie obiekty zostały wstrzyknięte.

6.3.3. W `ScopesShowcase` utworzyć metody pozwalające na pobranie z kontekstu Sprigna `SingletonBean` i `PrototypeBean`. Nie korzystać z wcześniej wstrzykniętych wartości. Wywołać metody w `showcase()` i pokazać, jakie obiekty zostały zwrócone.

6.3.4. Wywołać metodę `showcase()` dwa razy. Co można dzięki temu zaobserwować? Dlaczego?