EJB 2.0 -> Spring->Spring Boot czyli najlżejsza wersja

EJB 3.0 przewyższa Springa ale na szczęście powstał Spring Boot (a więc zarządzanie przez pluginy mavenowe)

**Spring Boot to**

To podział na paczki i zarządzanie by nie dochodziło do konfliktów

Wbudowany serwer webowy Tomcat

@Component – klasa jest Spring beanem

**Zasady**

1.Wstrzykiwanie zależności i rodzaje

2.Beans

3.DI + beans

4. Testowanie

**Wstrzykiwanie zależności – dependency injection**

Z tej techniki spring wyrósł

1. Di przez obiekt (dodatkowy param w konstruktorze)
2. Di przez metodę (przekazanie obiektu (zależności) w seterze po utworzeniu obiektu

**Spring Bean**

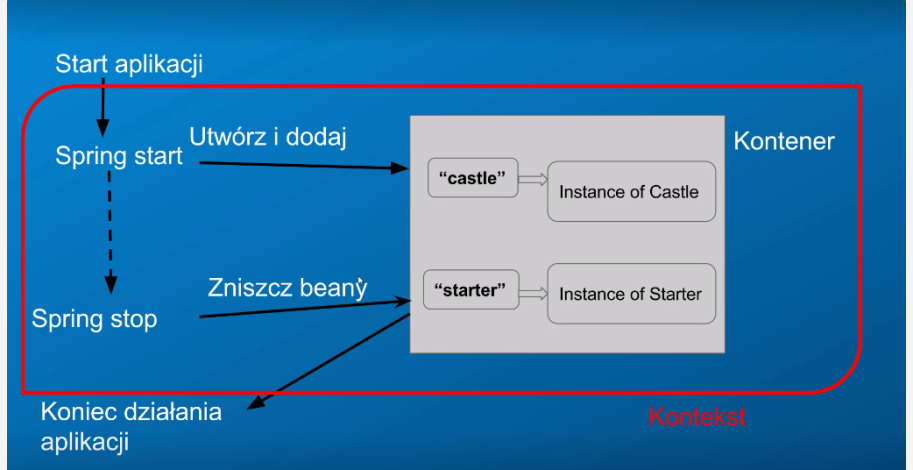
Klasy oznaczone adnotacją @Component

Spring startuje. Skanuje pliki źródłowe. Szuka np. adnotacji @Component

Utworzy obiekt tej klasy i doda do kontenera. A klasa jest dodawana do kontekstu.

**Kontekst**

Odpala metodę run na klasach implementujących CommandLineRunner



**Kontener**

to zwykła mapa, gdzie jako klucz mamy nazwę naszego beana czyli naszej klasy, a jako wartość mamy instancję naszej klasy

W momencie zakończenia aplikacji beany z kontenera są usuwane.

**Cykl życia Spring Beans**

Kontekst tworzy beany i je usuwa – to cykl ich życia.

@PostConstruct metoda uruchamia się od razu po utworzeniu beana

@PreDestroy metoda uruchamia się na chwilę przez usunięciem beana