**Spring ve Spring Boot arasındaki farklar**

Geçtiğimiz birkaç yılda eklenen özellikler nedeniyle Spring kütüphanesi giderek daha karmaşık bir hal almaya başladı. Sıfırdan yeni bir Spring projesine başlamak için bir sürü prosedürden geçilmesi gibi durumlar olabiliyor. Sıfırdan başlamayı önlemek ve zamandan tasarruf etmek için Spring Boot diye bir teknoloji dünyamıza girdi. Spring Boot yine Spring kütüphanesini kullanan bir teknolojidir.

Spring Kütüphanesi

Spring; bir web uygulaması geliştirmek için en yaygın olarak kullanılan Java EE (Java Enterprise Edition) kütüphanelerinden biridir. Java platformu için ayrıntılı bir programlama ve yapılandırma modeli sunar. Java EE kapsamında uygulama geliştirmeyi basitleştirmeyi amaçlar ve biz geliştiricilerin daha üretken olmasına yardımcı olur.

Diğer kütüphanelerin aksine Spring uygulamanın birkaç alanına odaklanır ve kapsamda birçok özellik sunar.

Bilindiği üzere Spring kütüphanesinin en önemli özelliği **Dependency Injection** dır.

Spring Boot

Spring kütüphanesi bize esneklik uygulamaya odaklanırken, Spring Boot kod uzunluğunu kısaltmayı ve bir web uygulaması geliştirmenin en kolay yolunu bize sunmayı amaçlamaktadır. Spring Boot, uygulama geliştirme için gerekli olan süreyi bir hayli kısaltır. Neredeyse hiçbir konfigürasyon yapmadan tek başına bir uygulama oluşturulmasına yardımcı olur.

Oto konfigürasyon Spring Boot için özel bir özelliktir.

Spring kütüphanesinin yararları

* Spring kütüphanesi bir web uygulamasının tüm katmanlarına uygulanabilir.
* Gevşek bağlılık (Loose Coupling) ve kolay test edilebilirlik sağlar.
* XML ve  Annotation konfigürasyonlarını destekler.
* Singleton ve Factory sınıflarının ortadan kaldırılması için gerekli yeteneğe sahiptir.
* Bildirimsel (Declarative) programlamayı destekler.

Spring Boot’un yararları

* Bağımsız (stand-alone) uygulamalar oluşturur.
* Gömülü olarak Tomcat, Jetty veya Undertow birlikte gelir.
* XML konfigürasyonuna ihtiyaç duymaz.
* LOC (Lines of Code) ‘u azaltmayı hedefler.
* Başlatması kolaydır.
* Özelleştirme ve yönetim basittir.

Bu nedenle Spring Boot bir kütüphane olmayıp, Spring tabanlı hazır bir proje başlatıcıdır. Otomatik yapılandırma gibi özelliklerle sizi uzun kod yazmaktan kurtarır ve gereksiz yapılandırmalardan kurtulmanızı sağlar.

* **Spring MVC** , Spring Framework tarafından yönetilen ve Servlets merkezli eksiksiz bir HTTP odaklı MVC çerçevesidir. JavaEE yığınındaki JSF'ye eşdeğerdir. İçindeki en popüler öğeler, @Controllerfarklı HTTP istekleri kullanarak erişebileceğiniz yöntemleri uyguladığınız ek açıklamalarla eklenmiş sınıflardır . @RestControllerREST tabanlı API'leri uygulamaya eşdeğerdir .
* **Spring boot** , uygulamaları hızlı bir şekilde kurmak için bir araçtır, Spring destekli uygulamalar oluşturmak için kutudan çıkarılmış bir konfigürasyon sunar. Bildiğiniz gibi, Spring , *yay çekirdeği* , *bahar verileri* , *bahar ağı* (bu arada Spring MVC'yi içerir) ve benzeri gibi [şemsiyesi](https://spring.io/projects) altında çok çeşitli modülleri entegre eder . Bu araçla Spring'e kaç tanesini kullanacağını söyleyebilir ve onlar için hızlı bir kurulum elde edersiniz (daha sonra kendiniz değiştirmenize izin verilir).
* Spring boot ile spring mvc farkı: Spring özellik Java dünyasında programcılara kolaylık sağlayan bir programlama çatısıdır. Spring boot size hiç bir ayar dosyasıyla uğraşmadan, çoğu ayarı otomatik yaparak size anahtar teslimi çalışma ortamı sağlar. Spring MVC, web uygulamaları için geliştirilmiş, çok gelişmiş ayarlar yapabileceğiniz bir çalışma çatısıdır.
* Spring MVC, Spring Data, Hibernate ve Tomcat kullanarak bir web uygulaması geliştiriyorsanız, tüm bu bileşenlerle tek tek uğraşmanız gerekir. Spring boot ile bir proje hazırladığınızda, sonuçta projenizi bir jar dosyasıyla yayınlayabilirsiniz. Yani içinde Spring, Hibernate,Tomcat ve diğer tüm bileşenleri barındırır.
* Spring Boot, bir uygulamayı yapılandırabilen, oluşturabilen ve çalıştırabilen basit, hepsi bir arada bir araç kullanarak kurumsal uygulamaları çalıştırmak için bir çerçevedir.
* Spring MVC, söz konusu uygulamayı oluşturmak için bir çerçevedir. Web uygulaması API uç noktaları, görselleştirmeler, veritabanı varlıkları ve diğer şeyler oluşturmak için kullanılır. Spring Boot ile yapılandırıp çalıştırabilir veya sadece standart Spring konfigürasyon sistemini kullanabilirsiniz.
* Spring tabanlı uygulamalar çok fazla XML konfigürasyonuna sahiptir. Spring Boot, Spring tabanlı uygulama geliştirmenin en kısa ve kullanışlı olması amacıyla geliştirilmiş bir frameworktur. Geliştiriciler için büyük kolaylıklar sağlar. Spring MVC’yi kullandığımızda, diğer şeyler arasında bileşen tarama, dağıtım programı servleti, bir view çözümleyici, web jarları(statik içerik sunmak için) yapılandırmamız gerekir.