

# Spring Boot & Logging



# Spring Boot ve Logging

- Spring Boot log altyapısı olarak **logback**, **log4j2** ve **java util logging** kütüphanelerini desteklemektedir
- **Default** olarak **logback** kullanılmaktadır
- Spring Boot içerisinde log API'si olarak Apache **Commons Logging API** kullanılmaktadır
- Dolayısı ile yukarıda bahsedilen herhangi bir log kütüphanesine geçiş **sadece konfigürasyon değişikliği** ile gerçekleşecektir

# Spring Boot ve Logging

- Application.properties içerisinde logger'lar ve log düzeyleri, log çıktısının formatı, log dosyalarının lokasyonu vb yönetilebilir

```
logging.level.root=INFO
logging.level.org.hibernate.type=TRACE
logging.level.org.springframework.web=DEBUG
```

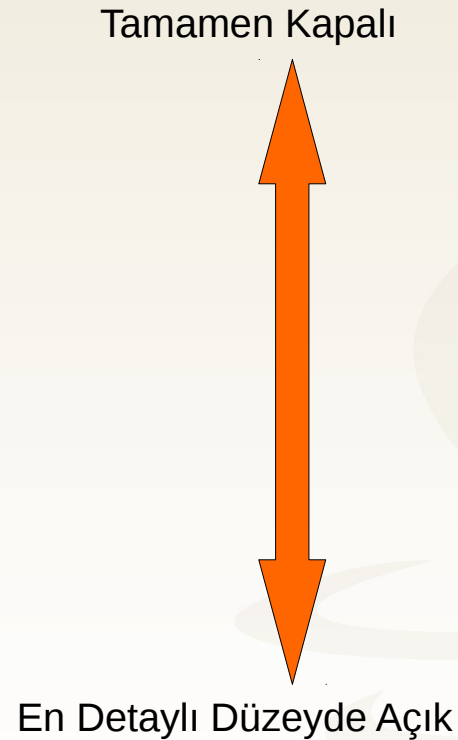
- Kullanılan log altyapısına göre ilgili **log konfigürasyon dosyaları** classpath'den yüklenebilir

```
logging.config=logback-spring.xml
```

logback-spring.xml,  
logback.xml,  
log4j2-spring.xml,  
log4j2.xml,  
logging.properties

# Spring Boot ve Logging

- OFF
- FATAL
- ERROR
- WARN
- INFO
- DEBUG
- TRACE



# Spring Boot ve Logging

- Log çıktısı default **console**'a yazılır, ancak istenirse **log dosyalarına** da yazılabilir

```
logging.file=petclinic.log  
logging.path=/tmp
```

logging.file tanımlı ise path devreye girmez,  
sadece path tanımlı olduğu durumda ise spring.log  
isimli log dosyası belirtilen dizinde oluşturulur

- Log property tanımları fiziksel log altyapısından bağımsızdır

# İletişim

- **Harezmi** Bilişim Çözümleri
- Kurumsal Java Eğitimleri
- <http://www.java-egitimleri.com>
- [info@java-egitimleri.com](mailto:info@java-egitimleri.com)

