**@Lazy**

• **Spring’s** default behavior to load singleton beans eagerly is also valid

for both annotation-based declaration using **@Component** and for Javabased configuration using **@Bean and @Configuration.**

• For these two kinds of configuration **@Lazy** is used to specify that bsingleton eans should be loaded lazily.

• **org.springframework.context.annotation.Lazy** is an annotation that can be used both at class and method level.

• **@Lazy** has an attribute **value** which is true in default.

@Component, @Bean ve @Configuration icerisinde kullanılabilir.

Yukarıda doküman pathi belirtilmistir oradan bakılabilir.

• When used with **@Component** and **@Bean**, **@Lazy** specifies that the singleton bean should be loaded lazily.

• When used with **@Configuration @Lazy** specifies that all of the

singletons that will be created in methods with **@Bean** should be loaded lazily.

• Methods with **@Bean** can override what **@Configuration** specifies by declaring value attribute **of @Lazy** as false.

• Using **@Lazy** with false in attribute value is very rare.

• **@Lazy can also be used at injection points with @Autowired and @Inject.**

**• In this case the injected object’s proxy is created and injected not real object itself.**

**• The creation of the real object is delayed until it is reached on the enclosing object.**

**• This mechanism is known as proxy pattern of GoF**.

**Lazy olmayan bir componente lazy olan bir component inject ediliyorsa lazy olan componentte aslında eager ile load edilecektir.**

Eğer iki bean de lazy load ise cagırmadığız sürece run ettiğimiz de ikisi de gelmez ama diyelim ki A beanini cagırdık o zaman B beani de gelir cünkü birbirine inject oldukları icin.

Lazy’li bir componenti eagerli bir bean ile initalize edersek lazyli component eager olarak calısır. Örnek olarak Bean b lazy olsun Bean A eager B ‘yi A’ya inject edip calıstırdıgımız da Bean B, Bean A’ da eager olarak calısır.

Hibernate, JPA, JDBC’ler Lazy’leri Proxy ile halleder.