

# Spring Bean Oluşturma Yöntemleri



# Bean Oluşturma Yöntemleri: Constructor Çağırarak

- Bean tanımından nesne oluşturma en temel yolu doğrudan **sınıf constructor**'ini çağırmasıdır

```
<bean id="foo" class="x.y.z.Foo"/>
```

Foo foo = new Foo();

# Bean Oluşturma Yöntemleri: Constructor Çağırarak

```
<bean id="foo" class="x.y.Foo">  
  <constructor-arg ref="bar"/>  
  <constructor-arg ref="baz"/>  
</bean>
```

Foo foo = new Foo(bar, baz);

# Bean Oluşturma Yöntemleri: JAVA Eğitimleri

## FactoryBean

- Spring Container, **FactoryBean** arayüzünü implement eden sınıflara farklı davranır
- FactoryBean arayüzünü implement eden sınıftan bir bean tanımı gördüğünde **FactoryBean.getObject()** metodundan dönen nesne asıl bean olur
- Spring'in kendi içerisinde de **yaygın biçimde kullanılan bir yapıdır**
- Nesnenin **yaratılması birden fazla adımdan oluşuyor** ise bu yöntem idealdir

# Bean Oluşturma Yöntemleri: FactoryBean



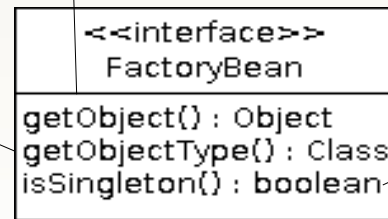
```
<bean id="foo" class="x.y.z.FooFactoryBean"/>
```

Bean instance'ını döner

```
public Object getObject() {  
    Foo f = new Foo();  
    f.setName("my foo");  
    return f;  
}
```

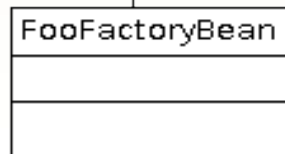
Oluşturulacak  
Bean instance'ın  
sınıfını döner

```
public Object  
    getObjectType() {  
        return Foo.class  
    }
```



Bean scope'unun singleton  
olup olmadığını belirtir

```
public boolean isSingleton() {  
    return true;  
}
```



# İletişim

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

