

How to get bean using application context in spring boot

i am developing a spring boot project and i want to get the bean by its name using `applicationContext` . I have tried many solution on web but could not succeed. My Requirement is that i have a controller

ControllerA

and inside controller a i have a method `getBean(String className)` , i want to get instance of registered bean. I have hibernate entities and i want to get an instance of bean by passing the name of class only in `getBean` method.

please help if someone know the solution

[spring](#) [spring-boot](#)

asked Dec 4 '15 at 12:47



Qasim

1,211 7 14 34

4 Answers

You can Autowire the `ApplicationContext`, either as a field

```
@Autowired
private ApplicationContext context;
```

or a method

```
@Autowired
public void context(ApplicationContext context) { this.context = context; }
```

Finally use

```
context.getBean(SomeClass.class)
```

answered Dec 4 '15 at 13:16



reos

4,055 3 10 27

4 Field injection is considered as bad practice. – [luboskrnac](#) Dec 4 '15 at 22:42

[@luboskrnac](#): Yes.Curious to understand why. – [Priyanka.Patil](#) Mar 17 '16 at 10:04

2 Just google "Field vs Constructor injection". Also Oliver Gierke (Spring Data Team lead) opposes field injection. – [luboskrnac](#) Mar 17 '16 at 10:52

Thanks for sharing ... – [Buminda](#) Nov 21 '17 at 6:30

You can use **`ApplicationContextAware`**.

`ApplicationContextAware`:

Interface to be implemented by any object that wishes to be notified of the `ApplicationContext` that it runs in. Implementing this interface makes sense for example when an object requires access to a set of collaborating beans.

There are a few methods for obtaining a reference to the application context. You can implement `ApplicationContextAware` as in the following example:

```
package hello;

import org.springframework.beans.BeansException;
import org.springframework.context.ApplicationContext;
import org.springframework.context.ApplicationContextAware;

@Component
public class ApplicationContextProvider implements ApplicationContextAware {

    private ApplicationContext applicationContext;

    @Override
    public void setApplicationContext(ApplicationContext applicationContext) throws
```

```

BeansException {
    this.applicationContext = applicationContext;
}

public ApplicationContext getContext() {
    return applicationContext;
}
}

```

Spring will call `setApplicationContext()` and **set current** applicationcontext.

Once you get the reference to Application context, you get fetch the bean whichever you want by using `getBean()`.

answered Jul 8 '16 at 20:14



[Sundararaj Govindasamy](#)

3,293 2 16 42

actually you want to get the object from the Spring engine, where the engine already maintaining the object of your required class at that starting of the spring application(Initialization of the Sprint engine).Now the thing is you just have to get that object to a reference.

in a service class

```

@Autowired
private ApplicationContext context;

SomeClass sc = (SomeClass)context.getBean(SomeClass.class);

```

now in the reference of the sc you are having the object. Hope explained well. If any doubt please let me know.

edited May 9 '17 at 11:56



[KarelG](#)

3,455 2 20 39

answered May 9 '17 at 11:33



[Debendra Parida](#)

26 1 6

If you are inside of Spring bean (in this case `@Controller` bean) you shouldn't use Spring context instance at all. Just autowire `className` bean directly.

BTW, avoid using field injection as it's considered as bad practice.

answered Dec 4 '15 at 22:42



[luboskrnac](#)

12.4k 4 40 59