

L'autowire con l'annotation @Resource

Cosa imparerai

- Come implementare l'autowire con l'annotation JSR-250 @Resource

Metadati Annotation-based

Spring supporta anche l'injection utilizzando l'annotazione JSR-250 **@Resource** (javax.annotation.Resource) al posto dell'annotation @Autowired.

L'annotation @Resource si applica su variabili di istanza o sui metodi set delle variabili.

Queste variabili devono essere dei bean.

È un pattern molto comune in Java EE, ad es. nei bean gestiti da JSF...

```
@Component(value = "ordine")
public class OrdineService {
    @Resource
    private ClienteService cliente;

    @Autowired
    private ProdottoService prodotti;

    public String sayHello() {
        return "CIAO sono un ordine del cliente " +
            cliente.getClient(1231) + " " +
            prodotti.sayHello();
    }
}
```

Metadati Annotation-based

Per poter utilizzare questa annotation (e tutte quelle del package `javax.annotation.`), se usiamo Maven è necessario impostare la dipendenza dall'artifact *javax.annotation-api*.

Esempio di dipendenza

```
<!-- https://mvnrepository.com/artifact/javax.annotation/javax.annotation-api -->  
<dependency>  
  <groupId>javax.annotation</groupId>  
  <artifactId>javax.annotation-api</artifactId>  
  <version>1.3.2</version>  
</dependency>
```

Metadati Annotation-based

L'annotation segue la semantica *by-name* per individuare il bean da iniettare e si può usare in due modi:

❑ Inserendo solo l'annotation

```
@Component(value = "cliente")
public class ClienteService {

    public ClienteService() {
        super();
    }

    public String getCliente(Long id) {
        String ret = "Mario Rossi";

        return ret;
    }
}
```

```
@Component(value = "ordine")
public class OrdineService {
    @Resource
    private ClienteService cliente;

    @Autowired
    private ProdottoService prodotti;

    public String sayHello() {
        return "CIAO sono un ordine del cliente " +
            cliente.getCliente(1231) + " " +
            prodotti.sayHello();
    }
}
```

Metadati Annotation-based

- ❑ Specificando l'attributo «name» che deve coincidere con il valore dell'attributo «value» del bean da iniettare (se è stato impostato...)

```
@Component(value = "cliente")
public class ClienteService {

    public ClienteService() {
        super();
    }

    public String getClienti(Long id) {
        String ret = "Mario Rossi";

        return ret;
    }
}
```

```
@Component(value = "ordine")
public class OrdineService {
    @Resource(name = "clienteService")
    private ClienteService cliente;

    @Autowired
    private ProdottoService prodotti;

    public String sayHello() {
        return "CIAO sono un ordine del cliente " +
            cliente.getClienti(1231) + " " +
            prodotti.sayHello();
    }
}
```

```
pt. org.springframework.beans.factory.NoSuchBeanDefinitionE
org.springframework.beans.factory.NoSuchBeanDefinitionE
or creating bean with name 'ordine': Injection of resour
tion: No bean named 'clienteService' available
```

```
@Component(value = "ordine")
public class OrdineService {
    @Resource(name = "cliente")
    private ClienteService cliente;

    @Autowired
    private ProdottoService prodotti;

    public String sayHello() {
        return "CIAO sono un ordine del cliente " +
            cliente.getClienti(1231) + " " +
            prodotti.sayHello();
    }
}
```

```
CIAO sono un ordine del cliente Mario Rossi TV 32''
```

Metadati Annotation-based

```
@Component
public class ClienteService {

    public ClienteService() {
        super();
    }

    public String getCliente(Long id) {
        String ret = "Mario Rossi";

        return ret;
    }
}
```

```
@Component(value = "ordine")
public class OrdineService {
    @Resource(name = "clienteService")
    private ClienteService cliente;

    @Autowired
    private ProdottoService prodotti;

    public String sayHello() {
        return "CIAO sono un ordine del cliente " +
            cliente.getCliente(1231) + " " +
            prodotti.sayHello();
    }
}
```

org.springframework.beans.factory.NoSuchBeanDefinitionException: No bean named 'clienteService' available

```
@Component(value = "ordine")
public class OrdineService {
    @Resource
    private ClienteService cliente;

    @Autowired
    private ProdottoService prodotti;

    public String sayHello() {
        return "CIAO sono un ordine del cliente " +
            cliente.getCliente(1231) + " " +
            prodotti.sayHello();
    }
}
```

CIAO sono un ordine del cliente Mario Rossi TV 32''



Di cosa abbiamo parlato in questa lezione

- Come implementare l'autowire con l'annotation JSR-250 @Resource