

2025년 상반기 K-디지털 트레이닝

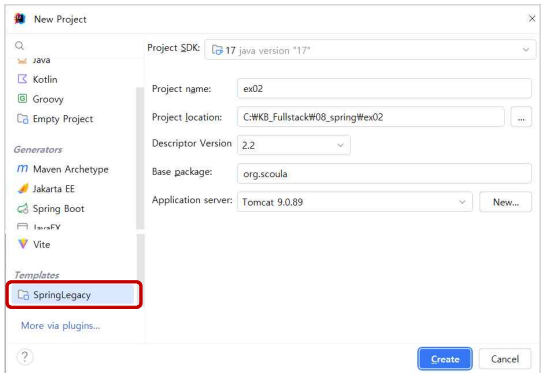
# 의존성 주입 (심화1)

[KB] IT's Your Life

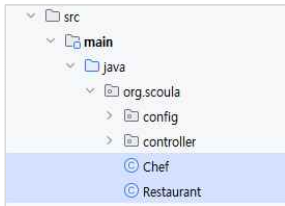
## ✓ SpringLegacy 템플릿을 이용하여 ex02프로젝트 만드시오.

### ○ File > New > Project...

- Templates: Spring Legacy
- setting.gradle: ex02
  - gradle sync!



- ✓ 다음과 같이 클래스 생성하고 각각을 어노테이션을 이용하여 싱글톤으로 생성 후, Restaurant내에 Chef 싱글톤 객체를 주입하시오.
- ✓ Restaurant은 lombok을 이용해 생성자, getter/setter/toString()을 재정의하시오.
  - org.scoula
    - Chef.java
    - Restaurant.java



### Chef.java

```
package org.scoula;  
  
import org.springframework.stereotype.Component;  
  
@Component  
public class Chef {  
}
```

### Restaurant.java

```
package org.scoula;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Component;

import lombok.Data;
import lombok.Setter;

@Component
@Data
public class Restaurant {
    @Autowired
    private Chef chef;
}
```

### RootConfig.java

- ✓ org.scoula패키지 아래 @Component 어노테이션이 붙은 클래스에 대해 싱글톤 객체를 생성할 수 있도록 설정하시오.

```
package org.scoula.config;

import org.springframework.context.annotation.ComponentScan;
import org.springframework.context.annotation.Configuration;

@Configuration
@ComponentScan(basePackages = {"org.scoula"})
public class RootConfig {

}
```

2025년 상반기 K-디지털 트레이닝

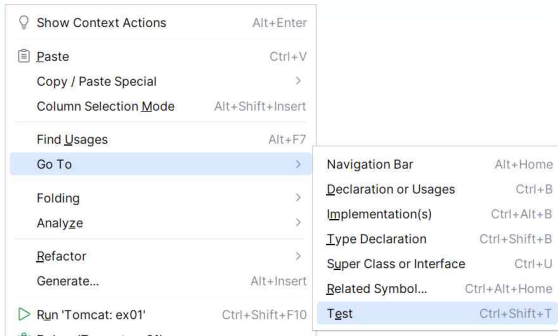
## 의존성 주입 (심화2)

[KB] IT's Your Life

## ✓ 테스트 코드를 통해 주입된 객체를 확인하시오.

### ○ 테스트 클래스 만들기

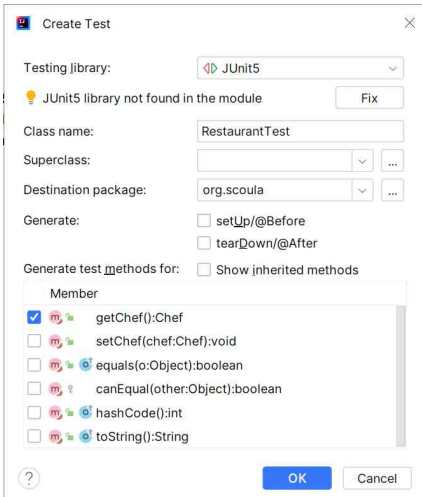
- Restaurant.java 편집창 >> Go To > Test





## ✓ 테스트 코드를 통한 확인

### ○ 테스트 클래스 만들기



## test :: RestaurantTest.java

```
package org.scoula;
```

```
import org.junit.jupiter.api.Test;
```

```
import static org.junit.jupiter.api.Assertions.*;
```

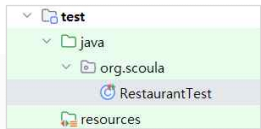
```
class RestaurantTest {
```

```
    @Test
```

```
    void getChef() {
```

```
    }
```

```
}
```



## test :: RestaurantTest.java

```
...
import static org.junit.jupiter.api.Assertions.assertNotNull;

@ExtendWith(SpringExtension.class)
@ContextConfiguration(classes = {RootConfig.class})
@Log4j2
class RestaurantTest {
    @Autowired
    private Restaurant restaurant;

    @Test
    void getChef() {
        assertNotNull(restaurant);
        log.info(restaurant);
        log.info("-----");
        log.info(restaurant.getChef());
    }
}
```



```
INFO : org.scoula.RestaurantTest - Restaurant(chef=org.scoula.Chef@25230246)
INFO : org.scoula.RestaurantTest - -----
INFO : org.scoula.RestaurantTest - org.scoula.Chef@25230246
```

## ✓ @Log4j2

- 테스트 코드에서도 Lombok을 이용해서 로그를 기록하는 Logger 변수 생성할 수 있도록 로그 파일을 아래와 같이 설정하시오.
- 로그 설정 파일
  - src/main/resources/log4j2.xml
    - 실제 운영을 위한 로거 설정
  - src/test/resources/log4j2.xml
    - 테스트를 위한 로거 설정
    - 없으면 운영을 위한 로거 설정이 적용

