

IoC and DI (기본편)

Recap

- JavaBeans 는 중요하지 않은 개념이다?
- EJB는 프레임워크인가? 2.0, 3.0 릴리즈가 무엇이냐!

Spring Beans vs JavaBeans

- JavaBean이 중요하지 않은 것 아님.
- JavaBean과 Spring Bean의 차이가 조금 있음.
- JavaBean은 그때 말한 것처럼 특정 규약을 지킨 Java 진영의 클래스의 개념이고, 그 이상의 개념은 아님.
- 하지만 Spring bean은 Spring IoC 컨테이너에서 관리되는 객체들을 말함. 아마 예진이가 말한 Spring MVC에서 사용되는 Bean은 Spring Bean일 것 같음.
- Spring Bean에 대해선 다음주에 더 자세하게 얘기해줄게...

EJB is...

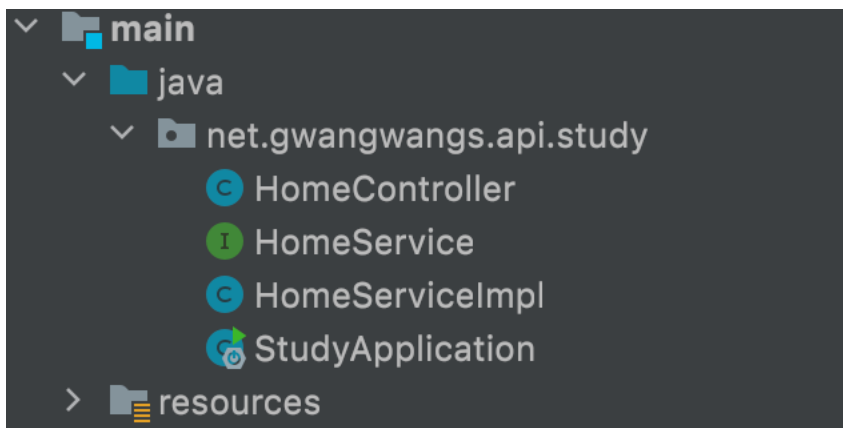
- EJB는 프레임워크 X
- EJB는 Class file이고, API임
- Java의 Class들도 생성되고 지워지고, 업데이트 되듯 EJB 또한 계속해서 발전하고, 업데이트 됨.
- 그렇기 때문에 2.0, 3.0과 같은 릴리즈 버전들이 있는 것.

IoC and DI

의존성이란?

클래스 간에 의존 관계가 존재

= 한 클래스가 바뀔 때 다른 클래스가 영향을 받는다는 것



```
package net.gwangwangs.api.study;

import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RestController;

@RestController
public class HomeController {
    //HomeService의 메소드를 사용하기 위해서 어떻게 해야하나?
}

```

```
package net.gwangwangs.api.study;

import org.springframework.stereotype.Service;

@Service
public interface HomeService {
    int get();
    void put(int a);
}

```

HomeController 클래스에서
HomeServiceImpl가 구현한 메소드를 이용하기 위해선
어떻게 해야할까?


```

package net.gwangwangs.api.study;

import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RestController;

@RestController
public class HomeController {
    //HomeService의 메소드를 사용하기 위해서 어떻게 해야하나?
    @GetMapping("/")
    String getHome(){
        return "";
    }
}

```

```

package net.gwangwangs.api.study;

import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RestController;

@RestController
public class HomeController {
    //HomeService의 메소드를 사용하기 위해서 어떻게 해야하나?
    HomeService hs = new HomeServiceImpl();

    @GetMapping("/")
    String getHome(){
        hs.put(a: 10);
        System.out.println(hs.get());
        return "";
    }
}

```

나는 치킨집을 열고 싶다.

무엇이 필요할까?

나는 치킨집을 열고 싶다.

무엇이 필요할까?

인테리어

튀김기

소스

앞치마

라이더

배달의 민족

가장 중요한 “**닭**”

나는 치킨집을 열고 싶다.

무엇이 필요할까?

인테리어

튀김기

소스

앞치마

라이더

배달의 민족

가장 중요한 “**답**”

나는 이 물품 들과 ‘의존’관계에 있는 것

나는 이 물품 들에 의존하고 있는 것

메뉴판

후라이드 15,000원

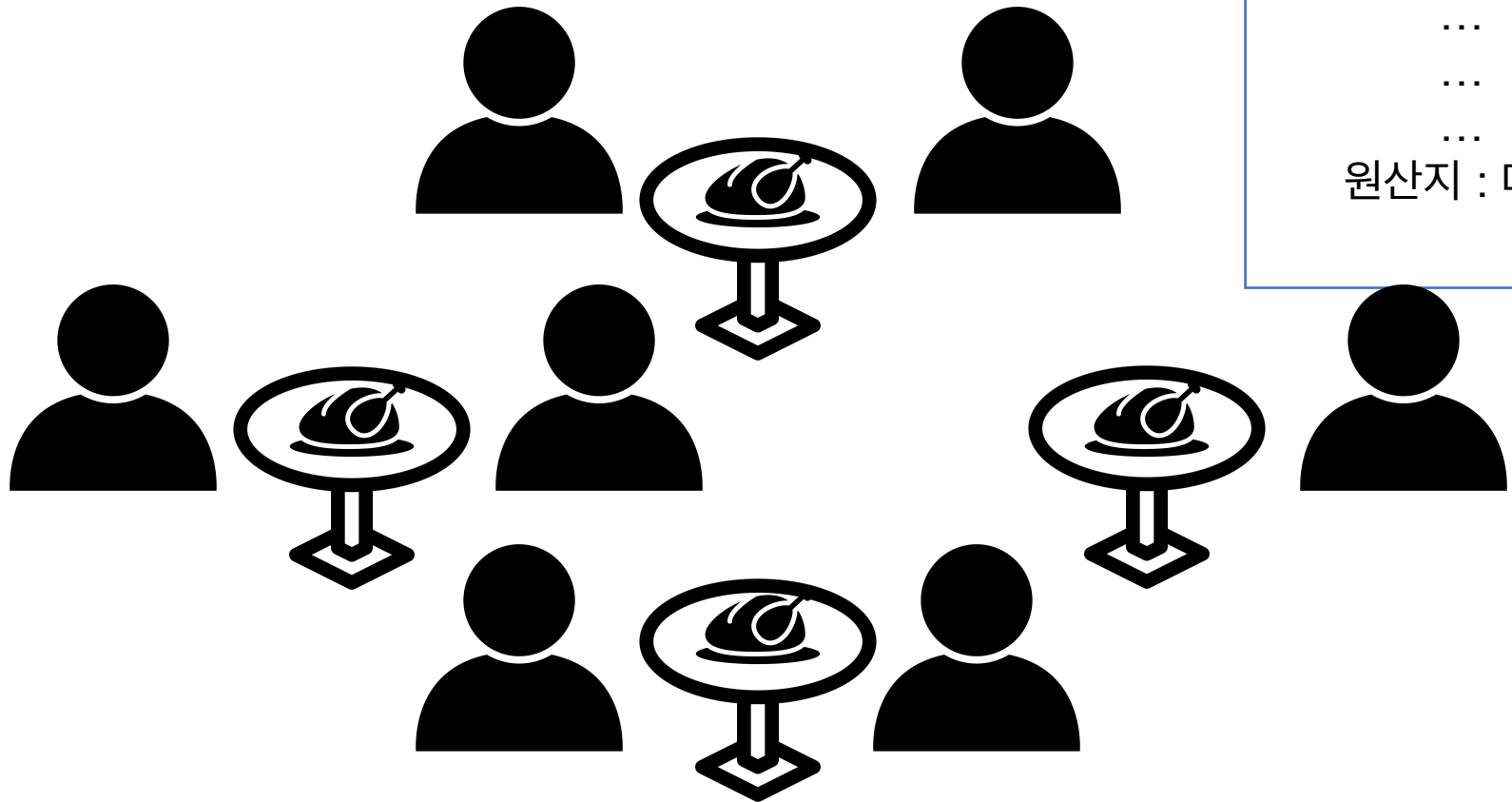
양념 15,000원

...

...

...

원산지 : 미국



미국산 닭고기 수입금지 /농림부, 델라웨어서 조류독감 발생따라

등록 :2004-02-09 00:00

메뉴판

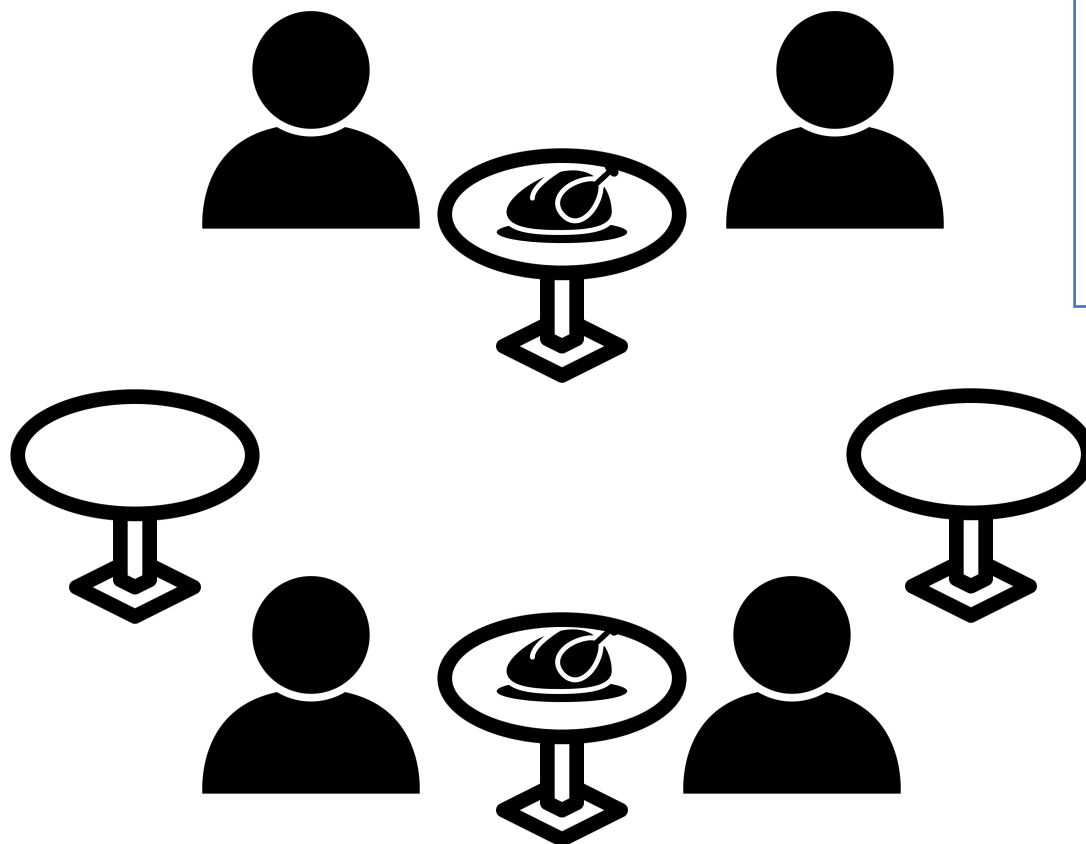
후라이드 15,000원

양념 15,000원

...

...

원산지 : 미국



메뉴판

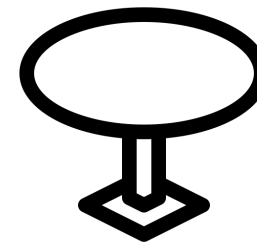
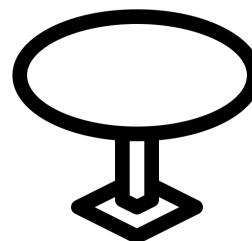
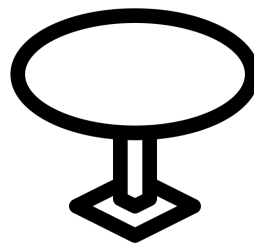
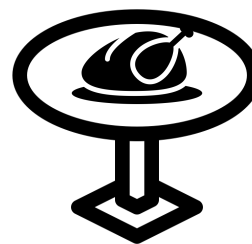
후라이드 15,000원

양념 15,000원

...

...

원산지 : 미국



메뉴판

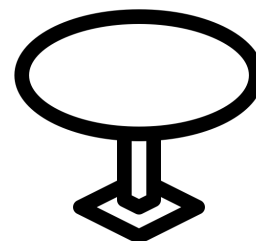
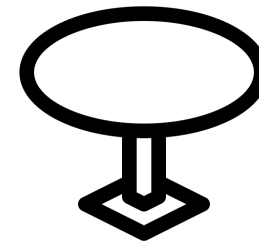
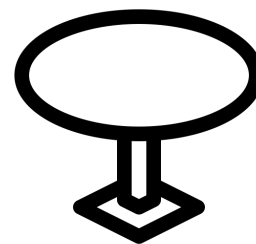
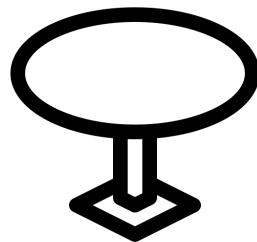
후라이드 15,000원

양념 15,000원

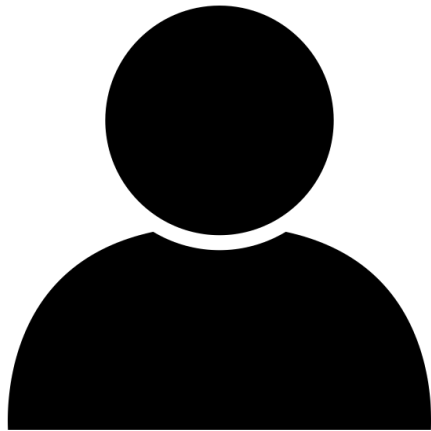
...

...

원산지 : 미국



나의 치킨집은 닭에 대한 의존성이 매우 높음



나



닭의 원산지
닭의 신선도
...
닭에 대한 세부적인 고민



Spring의 의존성 주입 방식



ApplicationContext



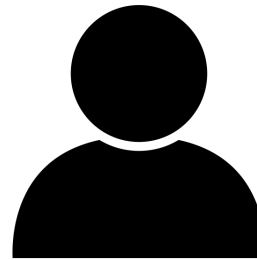
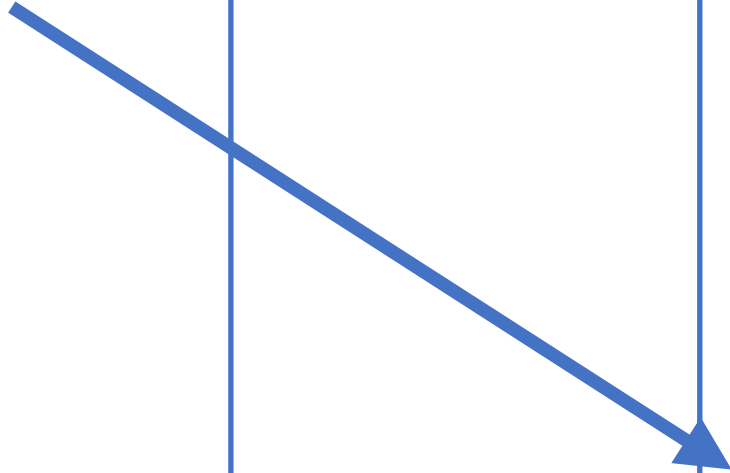
미국



한국



칠레



치킨가게1 메뉴판

후라이드 15,000원

양념 15,000원

...

...

원산지 : 미국

치킨가게2 메뉴판

후라이드 15,000원

양념 15,000원

...

...

원산지 : 미국

치킨가게3 메뉴판

후라이드 15,000원

양념 15,000원

...

...

원산지 : 미국

미국산 닭고기 수입금지 /농림부, 델라웨어서 조류독감 발생따라

등록 :2004-02-09 00:00



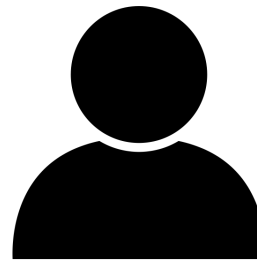
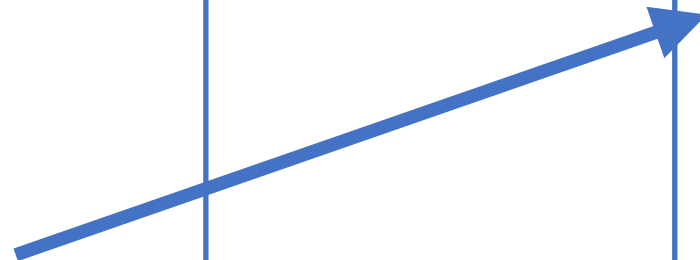
미국



한국



칠레



치킨가게1 메뉴판

후라이드 15,000원

양념 15,000원

...

...

원산지 : 미국

치킨가게2 메뉴판

후라이드 15,000원

양념 15,000원

...

...

원산지 : 미국

치킨가게3 메뉴판

후라이드 15,000원

양념 15,000원

...

...

원산지 : 미국



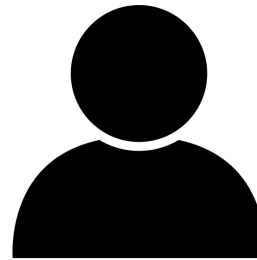
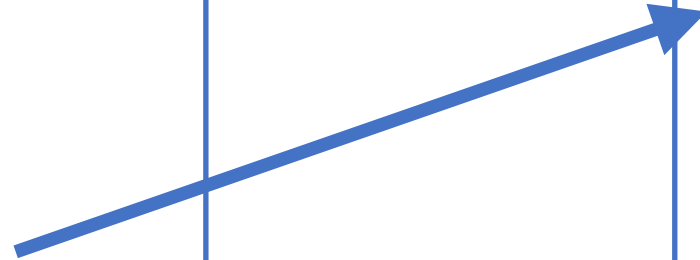
미국



한국



칠레



치킨가게1 메뉴판

후라이드 15,000원

양념 15,000원

...

...



원산지 : 칠레

치킨가게2 메뉴판

후라이드 15,000원

양념 15,000원

...

...



원산지 : 칠레

치킨가게3 메뉴판

후라이드 15,000원

양념 15,000원

...

...



원산지 : 칠레



미국



한국



칠레



치킨가게 1
메뉴판

세상에서 가장 맛있는 건강한 치킨메뉴 MENU			
황금불고기치킨 16,000 원	황금불고기치킨 17,500 원	바삭한치킨 16,000 원	치킨강판 18,000 원
순살치킨 17,000 원	순살치킨 18,000 원	황금불고기치킨 17,000 원	바삭한치킨 16,000 원
자매치킨 17,500 원	소고기치킨 17,000 원	황금불고기치킨 19,000 원	바삭한치킨 17,500 원
바삭한치킨 19,000 원	바삭한치킨 19,000 원	바삭한치킨 19,000 원	바삭한치킨 19,000 원

치킨가게 2
메뉴판

세상에서 가장 맛있는 건강한 치킨메뉴 MENU			
황금불고기치킨 16,000 원	황금불고기치킨 17,500 원	바삭한치킨 16,000 원	치킨강판 18,000 원
순살치킨 17,000 원	순살치킨 18,000 원	황금불고기치킨 17,000 원	바삭한치킨 16,000 원
자매치킨 17,500 원	소고기치킨 17,000 원	황금불고기치킨 19,000 원	바삭한치킨 17,500 원
바삭한치킨 19,000 원	바삭한치킨 19,000 원	바삭한치킨 19,000 원	바삭한치킨 19,000 원

치킨가게 3
메뉴판

세상에서 가장 맛있는 건강한 치킨메뉴 MENU			
황금불고기치킨 16,000 원	황금불고기치킨 17,500 원	바삭한치킨 16,000 원	치킨강판 18,000 원
순살치킨 17,000 원	순살치킨 18,000 원	황금불고기치킨 17,000 원	바삭한치킨 16,000 원
자매치킨 17,500 원	소고기치킨 17,000 원	황금불고기치킨 19,000 원	바삭한치킨 17,500 원
바삭한치킨 19,000 원	바삭한치킨 19,000 원	바삭한치킨 19,000 원	바삭한치킨 19,000 원

```

public class Main {
    class chickenIsRight1{
        Chicken chicken = new AmericanChicken();
        Chicken get(){
            return chicken;
        }
    }
    class chickenIsRight2{
        Chicken chicken = new AmericanChicken();
        Chicken get(){
            return chicken;
        }
    }
    class chickenIsRight3{
        Chicken chicken = new AmericanChicken();
        Chicken get(){
            return chicken;
        }
    }
}

```

```

//미국 닭 조류독감
class chickenIsRight1{
    Chicken chicken = new ChileChicken();
    Chicken get(){
        return chicken;
    }
}
class chickenIsRight2{
    Chicken chicken = new ChileChicken();
    Chicken get(){
        return chicken;
    }
}
class chickenIsRight3{
    Chicken chicken = new ChileChicken();
    Chicken get(){
        return chicken;
    }
}

```

```
class chickenIsRight1{
    @Autowired
    Chicken chicken;
    Chicken get(){
        return chicken;
    }
}
class chickenIsRight2{
    @Autowired
    Chicken chicken;
    Chicken get(){
        return chicken;
    }
}
class chickenIsRight3{
    @Autowired
    Chicken chicken;
    Chicken get(){
        return chicken;
    }
}
```

프랜차이즈

Chicken chicken = new AmericanChicken();



```
class chickenIsRight1{
    @Autowired
    Chicken chicken;
    Chicken get(){
        return chicken;
    }
}
class chickenIsRight2{
    @Autowired
    Chicken chicken;
    Chicken get(){
        return chicken;
    }
}
class chickenIsRight3{
    @Autowired
    Chicken chicken;
    Chicken get(){
        return chicken;
    }
}
```

프랜차이즈

Chicken chicken = new ChileChicken();



```
class chickenIsRight1{
    @Autowired
    Chicken chicken;
    Chicken get(){
        return chicken;
    }
}
class chickenIsRight2{
    @Autowired
    Chicken chicken;
    Chicken get(){
        return chicken;
    }
}
class chickenIsRight3{
    @Autowired
    Chicken chicken;
    Chicken get(){
        return chicken;
    }
}
```

ApplicationContext

Chicken chicken = new ChileChicken();

실제 ApplicationContext 코드가 이런진 않고, instance
를 만들어서 필드에 넣어준다는 것을 이해하기 위해

DI가 중요한 이유

- 클래스간의 결합도가 약해져, 리팩토링이 쉬워진다.
- 클래스간의 결합도가 약해져 단위 별 클래스를 테스트하기 편해진다.
- 클래스 간 관계에 상관 없이 비즈니스 로직 구현에만 신경쓸 수 있게 한다.

IoC

IoC란

프로그램 흐름의 제어권이 역전 되었다

= 애플리케이션의 제어권이 개발자에게서 프레임워크로 넘어갔다.

기존 방식

개발자가 main 메소드에
객체를 생성하는 명령문을 넣고,
메소드를 호출하며
실행하는 형식.

개발자가 실행의 흐름을 제어하고 있는 것.

IoC가 적용된 방식

객체의 생성, 제거(관리)를
프레임워크 컨테이너가 한다.

애플리케이션의 제어권을
‘개발자’가 아닌 ‘프레임워크’가 가져갔기 때문에,
이를 ‘제어의 역전’이라고 하는 것

Library

개발자가 전체적인 흐름을 만들며
미리 정의되어 있는 기능들을
개발자가 적절히 가져다가 사용하는 방식.

Framework

애플리케이션의 수행 흐름을
프레임워크가 관리하고,
개발자가 그 안에 필요한 코드를 추가적으로
넣어 프로그램을 수행하게 하는 것

Framework vs Library

Framework와 Library의 실행 주체가 각자 다름.

Framework의 실행 주체는 Framework 자신.

Framework에는 IoC가 적용이 되어 있다는 것.

ApplicationContext

- Spring에선 ApplicationContext라는 컨테이너가 IoC를 담당하고 있다.
- 객체의 생성과 객체 사이의 실행 관계를 설정하는 관점(의존성 주입 관점)에서 바라보면 해당 컨테이너를 Bean Factory라고 함.
- Bean Factory보다는 그래도 많은 일을 한다.
- Bean Factory에 애플리케이션 개발에 필요한 여러가지 컨테이너 기능을 추가한 것이 ApplicationContext

다음주!

- => 코드 중심으로 볼 것
- ApplicationContext, BeanFactory의 구조 및 동작방식
- Bean의 Life-Cycle, Scope
- Spring에서의 의존성 주입 방식과 자세한 흐름