

The word "Spring" is written in a white, stylized, handwritten-style font on a dark blue background.

# 6강\_DI (Dependency injection)

DI를 이용한 프로그래밍 방법과 스프링에서의 DI에 대해서 학습합니다.

---

6-1 DI (Dependency Injection)란?

6-2 스프링 DI설정 방법

## 6-1 : DI(Dependency Injection)란?



배터리 일체형



배터리가 떨어지면  
장난감을 새로 구입해야 한다.



배터리 분리형



배터리가 떨어지면  
배터리만 교체하면 된다.



배터리 분리형



배터리가 떨어지면  
배터리만 교체하면 된다.

## 6-1 : DI(Dependency Injection)란?

```
public class ElectronicCarToy {  
    private Battery battery;  
  
    public ElectronicCarToy() {  
        battery = new NormalBattery();  
    }  
}
```

배터리 일체형



배터리가 떨어지면  
장난감을 새로 구입해야 한다.

```
public class ElectronicRobotToy {  
    private Battery battery;  
  
    public ElectronicRobotToy() {  
    }  
  
    public void setBattery(Battery battery) {  
        this.battery = battery;  
    }  
}
```

배터리 분리형



배터리가 떨어지면  
배터리만 교체하면 된다.

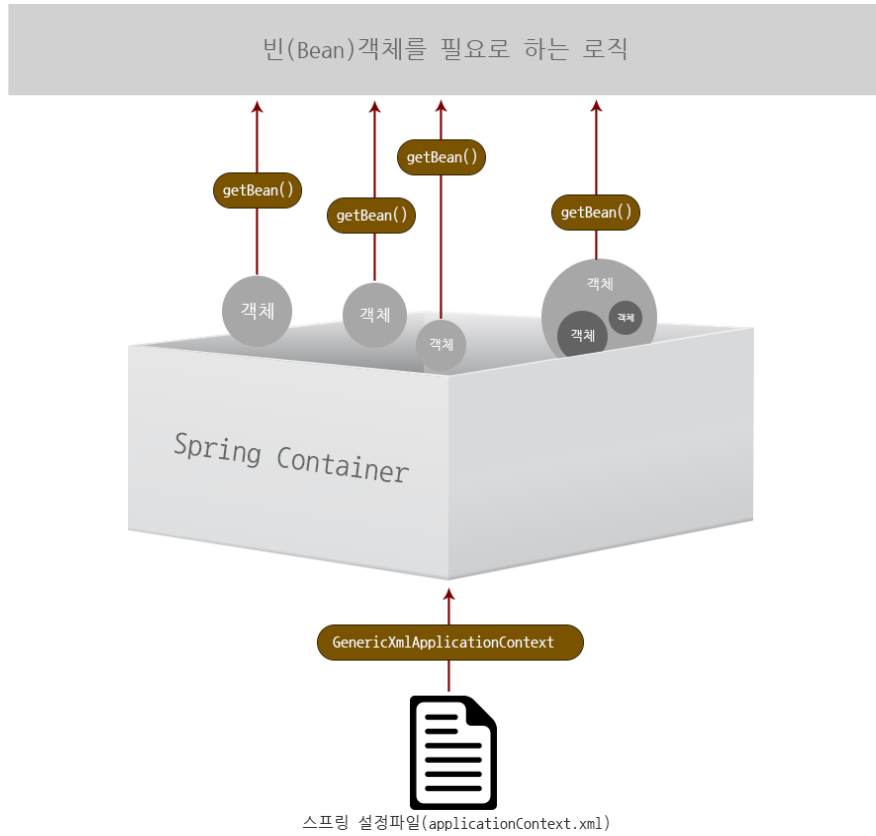
```
public class ElectronicRadioToy {  
    private Battery battery;  
  
    public ElectronicRadioToy(Battery battery) {  
        this.battery = battery;  
    }  
  
    public void setBattery(Battery battery) {  
        this.battery = battery;  
    }  
}
```

배터리 분리형

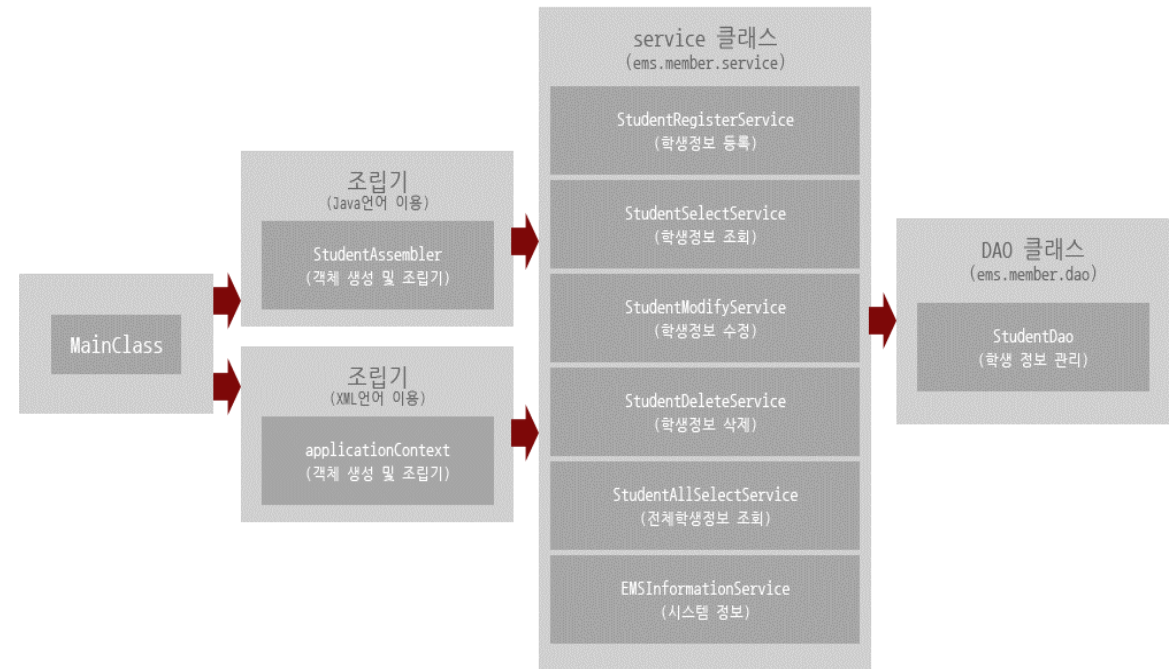


배터리가 떨어지면  
배터리만 교체하면 된다.

## 6-2 : 스프링 DI설정 방법



스프링 컨테이너 생성 및 빈(Bean)객체 호출 과정



클래스 구조

## 6-2 : 스프링 DI설정 방법

```
public StudentRegisterService(StudentDao studentDao) {  
    this.studentDao = studentDao;  
}
```

```
public StudentModifyService(StudentDao studentDao) {  
    this.studentDao = studentDao;  
}
```

```
public StudentDeleteService(StudentDao studentDao) {  
    this.studentDao = studentDao;  
}
```

```
public StudentSelectService(StudentDao studentDao) {  
    this.studentDao = studentDao;  
}
```

```
public StudentAllSelectService(StudentDao studentDao) {  
    this.studentDao = studentDao;  
}
```

```
<bean id="studentDao" class="ems.member.dao.StudentDao" ></bean>
```

```
<bean id="registerService" class="ems.member.service.StudentRegisterService">  
    <constructor-arg ref="studentDao" ></constructor-arg>  
</bean>
```

```
<bean id="modifyService" class="ems.member.service.StudentModifyService">  
    <constructor-arg ref="studentDao" ></constructor-arg>  
</bean>
```

```
<bean id="deleteService" class="ems.member.service.StudentDeleteService">  
    <constructor-arg ref="studentDao" ></constructor-arg>  
</bean>
```

```
<bean id="selectService" class="ems.member.service.StudentSelectService">  
    <constructor-arg ref="studentDao" ></constructor-arg>  
</bean>
```

```
<bean id="allSelectService" class="ems.member.service.StudentAllSelectService">  
    <constructor-arg ref="studentDao" ></constructor-arg>  
</bean>
```