

# Spring Package 구조는?

## Spring Package 구조는 어떻게 하는게 가장 좋을까?

### 1) Package By Layer

| Package By Layer는 기능 대신 층별로 다양한 어플리케이션을 나눈다.

#### 특징

- 낮은 응집성과 낮은 모듈성을 갖는다.
- 높은 결합도를 갖는다.

#### EX)

- Controller
  - UserController
  - AdminController
  - BoardController
- Service
  - UserService
  - AdminService
  - BoardService
- Dao
  - UserDao
  - AdminDao
  - BoardDao
- Dto
  - UserDto
  - AdminDto

- BoardDto

## 2) Package By Feature

Package By Feature는 작은 기능에 관련지어 아이템을 위치시키는 패키지 구조 방법이다.

### 특징

- 패키지간 높은 응집력과 높은 모듈성을 가질 수 있다.
- 패키지간 적은 결합성을 갖는다.
- 불필요하다고 생각된 기능은 디렉토리(패키지)를 제거함으로써 간단하게 처리할 수 있다.
- 기능에 관련되어 있다면, Java 코드 뿐만 아니라 다른 파일도 해당 패키지에 넣을 수 있어 보기가 편하다.

### 파일 명명 예시

- Controller : abcAction.java
- Dto : abc.java
- Dao : abcDao.java
- Service : abcServiceImpl.java

**Package by feature**은 다른 패키지에 속해있는 아이템(class)를 사용할 수 없다라는 의미보단 기본 제한 범위로써 **패키지의 프라이빗을 더 선호**한다는 의미를 갖고 있다. 또한, **필요하다면 제한 범위를 Public으로 확장**시켜 사용 할 수 있다.

### EX)

- User
  - Controller
  - Service
  - Dao

- Dto
- Admin
  - Controller
  - Service
  - Dao
  - Dto
- Board
  - Controller
  - Service
  - Dao
  - Dto


위의 내용을 기반으로 기능별 패키지가 더 괜찮을 것 같다고 생각한다.

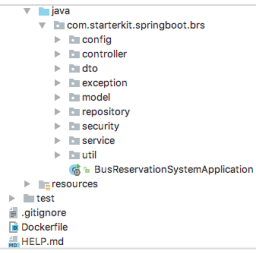
1. 높은 모듈성
2. 쉬운 코드 안내 (기능별로 구분되어있어 찾기가 쉽다.)
3. 높은 추상화
  - 높은 추상화는 프로그래밍 영구적인 가이드중 하나이다.  
⇒ 쉽게 문제를 접근할 수 있고 실행의 세부사항보다 전반적인 서비스의 기초를 강조할 수 있다.
  - 어플리케이션이 그 자체로 문서화가 될 수 있다.
4. 기능과 층을 모두 분리 할 수 있다.
5. 범위의 최소화
6. 좋은 확장 방법 (확장 시 깔끔하게 확장 할 수 있다.)

## REFERENCE

## Spring Boot 2.0-Project Structure and Best Practices (Part 2)


In continuation to the first part of the series on Spring Boot v2.0, this is the second and concluding part where we will go through the application project structure along with some important snippets and

 <https://medium.com/the-resonant-web/spring-boot-2-0-project-structure-and-best-practices-part-2-7137bdcba7d3>



## Java Practices→Package by feature, not layer

Package by feature, not layer The first question in building an application is "How do I divide it up into packages?". For typical business applications, there seems to be two ways of answering this question. Package-by-feature uses packages to reflect the feature set.

 <http://www.javapractices.com/topic/TopicAction.do?Id=205>