

# Setter Injection with Annotations and Autowiring



# Spring Injection Types

- Constructor Injection
- Setter Injection
- Field Injection

**Inject dependencies by calling  
setter method(s) on your class**

# Autowiring Example

Coach

FortuneService

- Injecting FortuneService into a Coach implementation
- Spring will scan @Components
- Any one implements FortuneService interface???
- If so, let's inject them. For example: *HappyFortuneService*

# Development Process - Setter Injection

1. Create setter method(s) in your class for injections
2. Configure the dependency injection with **@Autowired** Annotation

*Step-By-Step*

# Step 1: Create setter method(s) in your class for injections

File: TennisCoach.java

```
@Component
public class TennisCoach implements Coach {

    private FortuneService fortuneService;

    public TennisCoach() {
    }

    public void setFortuneService(FortuneService fortuneService) {
        this.fortuneService = fortuneService;
    }

    ...
}
```

# Step 2: Configure the dependency injection with Autowired Annotation

File: TennisCoach.java

```
@Component
public class TennisCoach implements Coach {

    private FortuneService fortuneService;

    public TennisCoach() {
    }

    @Autowired
    public void setFortuneService(FortuneService fortuneService) {
        this.fortuneService = fortuneService;
    }

    ...
}
```

**Inject dependencies by calling**

**ANY method on your class**

**Simply give: @Autowired**

# Step 2: Configure the dependency injection with Autowired Annotation

File: TennisCoach.java

```
@Component
public class TennisCoach implements Coach {

    private FortuneService fortuneService;

    public TennisCoach() {
    }

    @Autowired
    public void doSomeCrazyStuff(FortuneService fortuneService) {
        this.fortuneService = fortuneService;
    }

    ...
}
```