

Spring

Spring Annotations:

1. `@Controller` -> UI/REST
2. `@Configuration` -> configure beans
3. `@Component` -> Basic Annotation for visibility to `ComponentScan`
4. `@Bean` -> initialize an object based on custom logic
5. `@Autowired` -> Injection
6. `@Repository` -> Database interaction
7. `@RestController` -> REST
8. `@ComponentScan` -> It scans the packages to initialize beans
9. `@Service` -> Class that acts as service

```
// @Component
public class Cat {
    String sound;
    // other methods
}
```

```
@Configuration
public class PetConfig {
```

```
    @Bean
    public Cat getCat() {
        Cat cat = new Cat();
        cat.setSound("meow");

        return cat;
    }
}
```

```
@Service
public class PetService {
    @Autowired
    private Cat cat;

    public void makeNoise() {
        System.out.println(cat.getSound());
    }
}
```


Spring Boot

1. Simple Java Process
2. RESTful service
3. SOAP service
4. Scheduler
5. Messaging Based Applications
6. Streaming based Applications

- Open source
- Opinionated Framework
- Starter dependencies
- Production-Ready Observability
- Java, Groovy, Kotlin
- Maven, Gradle
- Auto Configuration

spring-core
spring-mvc
spring-tx
spring-aop
spring-kafka
spring-amqp

Starter Dependencies:

RESTful using spring:

- spring-context
- spring-core
- spring-mvc
- spring-web
- spring-aop
- tomcat
- jackson

RESTful using spring-boot:

- spring-boot-starter-web

Auto Configuration:

classpath:
oracle-driver
kafka
rabbitmq
redis

usersms - handson







