**《Spring Boot基础教程》**

**第25节**

**使用Spring Session实现集群-redis**

**session集群的解决方案：**

1.扩展指定server

利用Servlet容器提供的插件功能，自定义HttpSession的创建和管理策略，并通过配置的方式替换掉默认的策略。缺点：耦合Tomcat/Jetty等Servlet容器，不能随意更换容器。

2.利用Filter

利用HttpServletRequestWrapper，实现自己的 getSession()方法，接管创建和管理Session数据的工作。spring-session就是通过这样的思路实现的。

**Spring Boot中spring session支持方式：**

JDBC、MongoDB、**Redis**、Hazelcast、HashMap

1. **添加依赖**

<!-- spring session -->

<dependency>

<groupId>org.springframework.session</groupId>

<artifactId>spring-session</artifactId>

</dependency>

<!-- redis -->

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-redis</artifactId>

</dependency>

1. **配置**

# spring session使用存储类型

#spring.session.store-type=redis

# spring session刷新模式：默认on-save

#spring.session.redis.flush-mode=on-save

#spring.session.redis.namespace=

# session超时时间，单位秒

#server.session.timeout=30

#redis

#spring.redis.host=localhost

#spring.redis.port=6379

#spring.redis.password=123456

#spring.redis.database=0

#spring.redis.pool.max-active=8

#spring.redis.pool.max-idle=8

#spring.redis.pool.max-wait=-1

#spring.redis.pool.min-idle=0

#spring.redis.timeout=0

1. **测试**

@RequestMapping(value = "/index")

public String index(ModelMap map, HttpSession httpSession) {

map.put("title", "第一个应用：sessionID=" + httpSession.getId());

System.out.println("sessionID=" + httpSession.getId());

return "index";

}

**更多课程信息，请关注 龙果学院 官方网站**[**http://www.roncoo.com/**](http://www.roncoo.com/)

**或关注 龙果 微信公众号 RonCoo\_com**

****