

千锋教育Java教研院 关注公众号【Java架构栈】下载所有课程代码课件及工具 让技术回归本该有的纯净!

|千锋教育|千锋Java|公众号:Java架构栈

作者:Wilson

Redis分布式Session

Spring-Session

spring-session 解决分布式 session 的共享问题。

介绍

Spring Session 是 Spring 的项目之一，GitHub地址 <https://github.com/spring-pro>。

Spring Session 提供了一套创建和管理 Servlet HttpSession 的完美方案。

功能

spring Session 提供了 API 和实现，用于管理用户的 Session 信息。除此之外，它还提供了如下特性：

- 将 session 所保存的状态卸载到特定的外部 session 存储汇总，如 Redis 中，他们能够以独立于应用服务器的方式提供高质量的集群。
- 控制 sessionid 如何在客户端和服务端之间进行交换，这样的话就能很容易地编写 Restful API，因为它可以从 HTTP 头信息中获取 sessionid，而不必再依赖于 cookie。
- 在非 Web 请求的处理代码中，能够访问 session 数据，比如在 JMS 消息的处理代码中。
- 支持每个浏览器上使用多个 session，从而能够很容易地构建更加丰富的终端用户体验。
- 当用户使用 WebSocket 发送请求的时候，能够保持 HttpSession 处于活跃状态。

依赖

```
<!-- spring session 和 redis starter集成包 目的用于分布式session管理 -->
<dependency>
  <groupId>org.springframework.session</groupId>
  <artifactId>spring-session-data-redis</artifactId>
</dependency>
```

入口类或Config

```

/**
 * @EnableRedisHttpSession注解，开启redis集中式session管理，把所有的session都存放到了redis中
 * maxInactiveIntervalInSeconds：设置 Session 失效时间
 * 使用 Redis Session 之后，原 Spring Boot 中的 server.session.timeout 属性不再生效。
 */
@Configuration
@EnableRedisHttpSession(maxInactiveIntervalInSeconds = 1000*60)
public class RedisSessionConfig {
}

```

测试

```

@RestController
public class RedisSessionTest {

    //登录
    @RequestMapping("/login")
    public String login(@RequestParam("uname") String uname,
        @RequestParam("upass") String upass,
        HttpServletRequest request){
        String msg="login failure!";

        if (uname!=null && "admin".equals(uname) && "123".equals(upass)){
            User u=new User();
            u.setUname(uname); u.setUpass(upass);
            request.getSession().setAttribute("user",u);
            msg="login successful!";
        }
        return msg;
    }
    @Value("${server.port}")
    private String port;
    //取Session
    @GetMapping("getSession")
    public Map<String, Object> getSession(HttpSession session){
        Map<String, Object> map=new HashMap<>();
        map.put("user", session.getAttribute("user"));
        map.put("port", port);
        return map;
    }
}

```

总结

分别启动两个微服务项目。发现测试运行。

springboot配置session共享是非常方便的，只需要EnableRedisHttpSession注解即可。

解读

RedisSession 在创建时设置 3 个变量 creationTime, maxInactiveInterval, lastAccessedTime。maxInactiveInterval 默认值为 1800秒, 表示 1800s 之内该 session 没有被再次使用, 则表明该 session 已过期。每次 session 被访问都会更新 lastAccessedTime 的值, session 的过期计算公式:
 $\text{当前时间} - \text{lastAccessedTime} > \text{maxInactiveInterval}$ 。

获取 session

spring session在 redis 里面保存的数据包括:

- SET 类型的

```
spring:session:expireations:[min]
```

min 表示从 1970 年 1 月 1 日 0 点 0 分经过的分钟数, SET 集合的 member 为 expires:[sessionId],表示 members 会在 min 分钟后过期。

- String 类型的

```
spring:session:sessions:expires:[sessionId]
```

该数据的 TTL 表示 sessionId 过期的剩余时间, 即 maxInactiveInterval。

- Hash 类型的

```
spring:session:sessions:[sessionId]
```

session 保存的数据, 记录了 creationTime, maxInactiveInterval, lastAccessedTime, attribute。前两个数据是用于 session 过期管理的辅助数据结构。

干锋教育Java教研院/关注公众号【Java架构栈】下载所有课程代码课件及工具 让技术回归本该有的纯粹!

|干锋教育|干锋Java|公众号:Java架构栈

作者:Wilson