redis session 实现多ip同步

# 1、springboot

## 1、准备工作，添加redis缓存

### 1、pom依赖

*<?*xml version="1.0" encoding="UTF-8"*?>*<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">  
 <modelVersion>4.0.0</modelVersion>  
  
 <groupId>sinoSpringBootNginxSessionSueecss</groupId>  
 <artifactId>sinospringbootnginxsessionsueecss</artifactId>  
 <version>0.0.1-SNAPSHOT</version>  
 <packaging>jar</packaging>  
  
 <name>sinospringbootnginxsessionsueecss</name>  
 <description>Demo project for Spring Boot</description>  
  
 <parent>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-parent</artifactId>  
 <version>1.5.9.RELEASE</version>  
 <relativePath/> <!-- lookup parent from repository -->  
 </parent>  
  
 <properties>  
 <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>  
 <project.reporting.outputEncoding>UTF-8</project.reporting.outputEncoding>  
 <java.version>1.8</java.version>  
 </properties>  
  
 <dependencies>  
 <dependency>  
 <groupId>org.springframework.session</groupId>  
 <artifactId>spring-session</artifactId>  
 </dependency>  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-thymeleaf</artifactId>  
 </dependency>  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-web</artifactId>  
 </dependency>  
  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-test</artifactId>  
 <scope>test</scope>  
 </dependency>  
  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-redis</artifactId>  
 </dependency>  
  
 <dependency>  
 <groupId>org.springframework.session</groupId>  
 <artifactId>spring-session-data-redis</artifactId>  
 </dependency>  
  
  
 </dependencies>  
  
 <build>  
 <plugins>  
 <plugin>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-maven-plugin</artifactId>  
 </plugin>  
 </plugins>  
 </build>  
  
  
</project>

## 2、添加redis 配置

server.port=8082  
  
########################################################  
###REDIS (RedisProperties) redis»ù±¾ÅäÖÃ£»  
########################################################  
# database name  
spring.redis.database=0  
# server host1  
spring.redis.host=127.0.0.1  
# server password  
#spring.redis.password=  
#connection port  
spring.redis.port=6379  
# pool settings ...  
spring.redis.pool.max-idle=8  
spring.redis.pool.min-idle=0  
spring.redis.pool.max-active=8  
spring.redis.pool.max-wait=-1  
# name of Redis server  
#spring.redis.sentinel.master=  
# comma-separated list of host:port pairs  
#spring.redis.sentinel.nodes=

## 3、添加启动redis 配置缓存类 @EnableRedisHttpSession

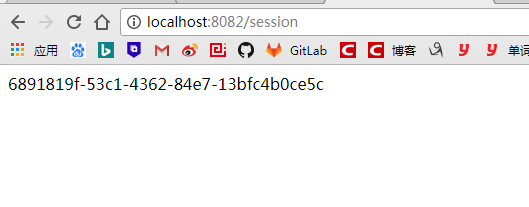
/\*\*  
 \* @Description  
 \* @Author HealerJean  
 \* @Date 2017/12/4 11:15.  
 如果需要添加失效时间可以使用以下的写法：  
 @EnableRedisHttpSession(maxInactiveIntervalInSeconds = 60) //1分钟失效  
 \*/  
@Configuration  
@EnableRedisHttpSession  
public class RedisSessionConfig {  
  
}

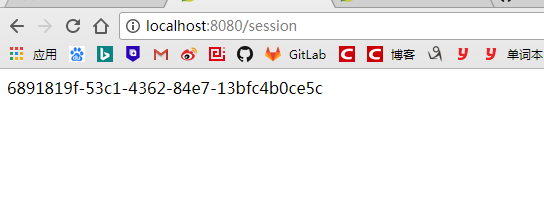
## 4、测试controller

@RestController  
public class HomeController {  
   
  
  
  
 @GetMapping("session")  
 public String getSession(){  
 HttpServletRequest request = ((ServletRequestAttributes) RequestContextHolder.getRequestAttributes()).getRequest();

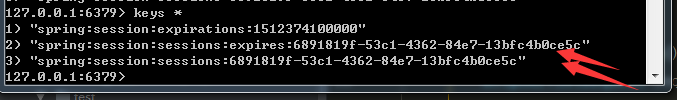
return request.getSession().getId();  
 }  
}

## 5、测试浏览器（测试两个项目）





## 6、观察redis 缓存，成功



## 7、注意事项

所有实体类实现Serializable接口

public class UserInfo implements Serializable

# 2、springMvc redis

## 1、web.xml中配置过滤器

|  |
| --- |
| <!-- redis-session -->  <filter>  <filter-name>springSessionRepositoryFilter</filter-name>  <filter-class>org.springframework.web.filter.DelegatingFilterProxy</filter-class>  </filter>  <filter-mapping>  <filter-name>springSessionRepositoryFilter</filter-name>  <url-pattern>/\*</url-pattern>  </filter-mapping> |

## 2、spring xml配置redis

|  |
| --- |
| <?xml version=*"1.0"* encoding=*"UTF-8"*?>  <beans xmlns=*"http://www.springframework.org/schema/beans"*  xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xmlns:mvc=*"http://www.springframework.org/schema/mvc"*  xmlns:context=*"http://www.springframework.org/schema/context"*  xmlns:aop=*"http://www.springframework.org/schema/aop"* xmlns:tx=*"http://www.springframework.org/schema/tx"*  xsi:schemaLocation=*"http://www.springframework.org/schema/beans*  *http://www.springframework.org/schema/beans/spring-beans-3.2.xsd*  *http://www.springframework.org/schema/mvc*  *http://www.springframework.org/schema/mvc/spring-mvc-3.2.xsd*  *http://www.springframework.org/schema/context*  *http://www.springframework.org/schema/context/spring-context-3.2.xsd*  *http://www.springframework.org/schema/aop*  *http://www.springframework.org/schema/aop/spring-aop-3.2.xsd*  *http://www.springframework.org/schema/tx*  *http://www.springframework.org/schema/tx/spring-tx-3.2.xsd "*>    <!-- 连接池配置 -->  <bean id=*"jedisPoolConfig"* class=*"redis.clients.jedis.JedisPoolConfig"*>  <!-- 最大连接数 -->  <property name=*"maxTotal"* value=*"30"* />  <!-- 最大空闲连接数 -->  <property name=*"maxIdle"* value=*"10"* />  <!-- 每次释放连接的最大数目 -->  <property name=*"numTestsPerEvictionRun"* value=*"1024"* />  <!-- 释放连接的扫描间隔（毫秒） -->  <property name=*"timeBetweenEvictionRunsMillis"* value=*"30000"* />  <!-- 连接最小空闲时间 -->  <property name=*"minEvictableIdleTimeMillis"* value=*"1800000"* />  <!-- 连接空闲多久后释放, 当空闲时间>该值 且 空闲连接>最大空闲连接数 时直接释放 -->  <property name=*"softMinEvictableIdleTimeMillis"* value=*"10000"* />  <!-- 获取连接时的最大等待毫秒数,小于零:阻塞不确定的时间,默认-1 -->  <property name=*"maxWaitMillis"* value=*"1500"* />  <!-- 在获取连接的时候检查有效性, 默认false -->  <property name=*"testOnBorrow"* value=*"true"* />  <!-- 在空闲时检查有效性, 默认false -->  <property name=*"testWhileIdle"* value=*"true"* />  <!-- 连接耗尽时是否阻塞, false报异常,ture阻塞直到超时, 默认true -->  <property name=*"blockWhenExhausted"* value=*"false"* />  </bean>    <!-- redis连接工厂 -->  <bean id=*"connectionFactory"* class=*"org.springframework.data.redis.connection.jedis.JedisConnectionFactory"*>  <property name=*"hostName"* value=*"127.0.0.1"*/>  <property name=*"port"* value=*"6379"*/>  <property name=*"password"* value=*""*/>  <property name=*"timeout"* value=*"20000"*/>  <property name=*"poolConfig"* ref=*"jedisPoolConfig"*></property>  </bean>      <bean class=*"org.springframework.session.data.redis.config.annotation.web.http.RedisHttpSessionConfiguration"*>  <property name=*"maxInactiveIntervalInSeconds"* value=*"3600"*></property>  </bean>    </beans> |

## 3、代码位置

