<http://blog.csdn.net/u014430366/article/details/65633679>

本文环境指的 Spring Boot下1.4.2版本下   
pom.xml (核心内容)

<parent>

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

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

<version>1.4.2.RELEASE</version>

<relativePath/>

</parent>

<dependencies>

<dependency>

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

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

</dependency>

<dependencies>

如果单纯想用RestTemplate 引上面的应该是多余了，博主没去查阅具体那个包，可以自行去Maven 中央仓库搜索有没有专门的RestTemplate包

RestTemplate有简写方法，但不具备通用性且无法做细节设置，故本文不再赘述有兴趣可访问，下文只提通用性方法   
[RestTemplate 官方 API](http://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/client/RestTemplate.html)

举例：

restTemplate.getForObject(url, String.class);

// 向url发送 Get类型请求，将返回结果 以String类型显示

// 能很明显发现无法设置编码格式等，乱码等问题无法解决

# 实例

* application/json类型请求

RestTemplate restTemplate=new RestTemplate();

String url="http://www.testXXX.com";

/\* 注意：必须 http、https……开头，不然报错，浏览器地址栏不加 http 之类不出错是因为浏览器自动帮你补全了 \*/

HttpHeaders headers = new HttpHeaders();

/\* 这个对象有add()方法，可往请求头存入信息 \*/

headers.setContentType(MediaType.APPLICATION\_JSON\_UTF8);

/\* 解决中文乱码的关键 , 还有更深层次的问题 关系到 StringHttpMessageConverter，先占位，以后补全\*/

HttpEntity<String> entity = new HttpEntity<String>(body, headers);

/\* body是Http消息体例如json串 \*/

restTemplate.exchange(url, HttpMethod.POST, entity, String.class);

/\*上面这句返回的是往 url发送 post请求 请求携带信息为entity时返回的结果信息

String.class 是可以修改的，例如User.class，这样你就可以有 User resultUser接返回结果了\*/

1

另post、get、delete……等其他http请求类型只需将exchange参数修改为HttpMethod.GET 等等。

* application/x-www-form-urlencoded类型请求

RestTemplate restTemplate=new RestTemplate();

String url="http://www.testXXX.com";

/\* 注意：必须 http、https……开头，不然报错，浏览器地址栏不加 http 之类不出错是因为浏览器自动帮你补全了 \*/

String bodyValTemplate = "%E6%B5%8B%E8%AF%95%E7%9A%84%E5%8F%82%E6%95%B01&var2=test+val2";

HttpHeaders headers = new HttpHeaders();

headers.setContentType(MediaType.APPLICATION\_FORM\_URLENCODED);

HttpEntity entity = new HttpEntity(bodyValTemplate, headers);

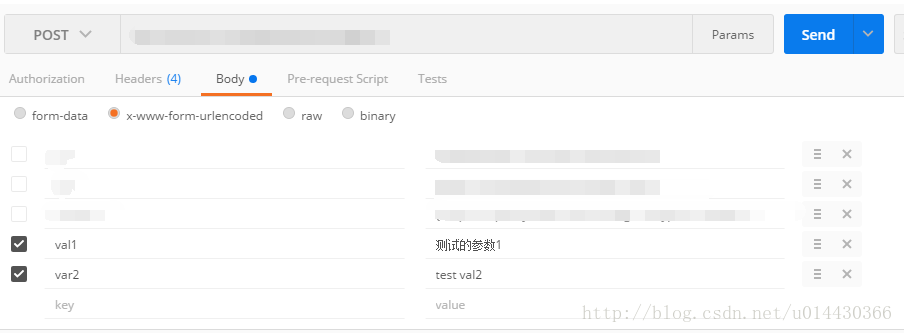
restTemplate.exchange(url, HttpMethod.POST, entity, String.class);

/\*

\* bodyValTemplate 实际值为urlencod集

\* Tips: URLEncoder.encode() 这个方法能将普通字符编码成 urlencode

\*/

上方请求在PostMan里等效于图   


**RestTemplate还有专门支持异步请求的特殊类 AsyncRestTemplate，具体用法和RestTemplate类似（需要 ListenableFutureCallback 基础）**

版权声明：本文为CSDN博主-------SolidCocoi原创文章，允许转载，但声明原文Url。