****

**《分布式计算》**

RESTful Webservice编程

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **学院名称** | **：** | 数据科学与计算机学院 | | | | | |
| **成员** | **：** | 陈伟宸 | | | | | |
| **时间** | **：** | 2016 | 年 | 9 | 月 | 17 | 日 |

1. 输入是URL，输出是XML

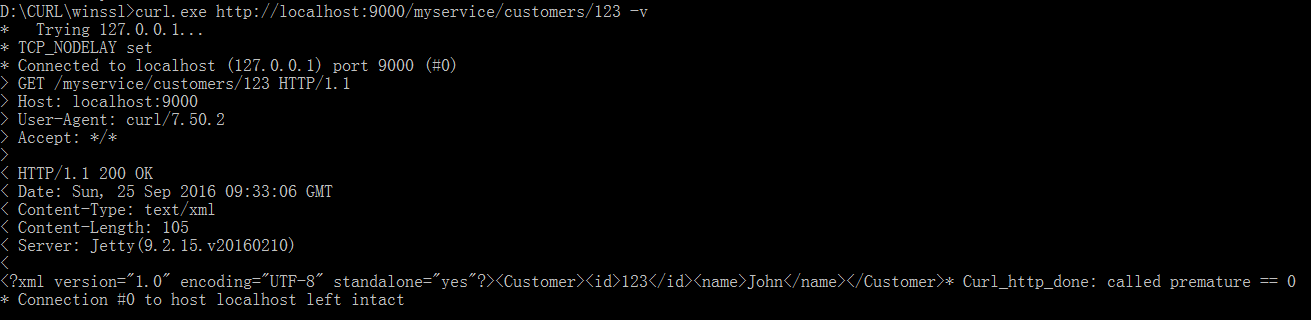
2. 需要申明HTTP协议方法和URI；

这些申明为：

@GET

@Path("/customers/{id}/")

协议的方法对应@GET，协议的地址对应@Path

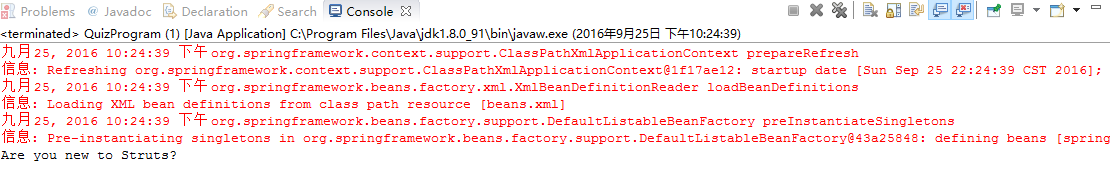
3. 

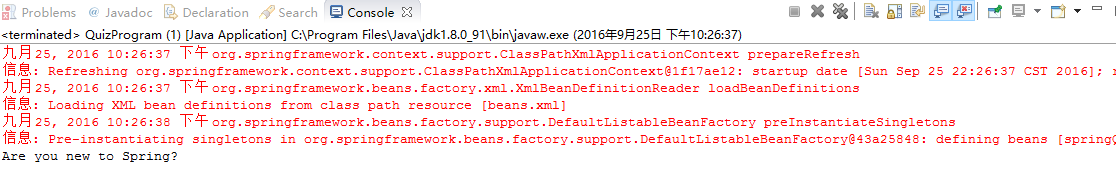
4. GET方法的查询字符串在URL中的，而POST方法的查询字符串实在HTTP消息主体中的

5. 会，当数据量增大时，由于map会维护数据，增加新用户的时间会边长

6. IoC将需要的资源从主体中分离，在需要时向主体中注入，使得主体不需要考虑资源的生成与维护，降低了耦合程度

7. 代码见IOC\_test.zip，程序运行结果如下：





8. 在标签

<bean id="customerBean" class="demo.jaxrs.server.CustomerService" />

中，class属性申明了根资源

9. Hello类代码：

**package** demo.jaxrs.server;

**import** javax.ws.rs.GET;

**import** javax.ws.rs.Path;

**import** javax.ws.rs.PathParam;

**import** javax.ws.rs.Produces;

@Path("/hello/{username}")

**public** **class** Hello {

@GET

@Produces("text/xml")

**public** String sayHello(@PathParam("username") String userName) {

**return** "hello " + userName;

}

}

beans.xml如下：

<?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:jaxrs=*"http://cxf.apache.org/jaxrs"*

xsi:schemaLocation=*"*

*http://www.springframework.org/schema/beans*

*http://www.springframework.org/schema/beans/spring-beans.xsd*

*http://cxf.apache.org/jaxrs*

*http://cxf.apache.org/schemas/jaxrs.xsd"*>

<!-- do not use import statements if CXFServlet init parameters link to this beans.xml -->

<import resource=*"classpath:META-INF/cxf/cxf.xml"* />

<import resource=*"classpath:META-INF/cxf/cxf-servlet.xml"* />

<jaxrs:server id=*"customerService"* address=*"/service1"*>

<jaxrs:serviceBeans>

<ref bean=*"customerBean"* />

</jaxrs:serviceBeans>

<!-- 引入json 输入/输出支持 -->

<jaxrs:providers>

<bean class=*"org.codehaus.jackson.jaxrs.JacksonJaxbJsonProvider"*/>

<bean class=*"org.apache.cxf.jaxrs.provider.JAXBElementProvider"*/>

</jaxrs:providers>

</jaxrs:server>

<jaxrs:server id=*"helloService"* address=*"/service2"*>

<jaxrs:serviceBeans>

<ref bean=*"helloBean"* />

</jaxrs:serviceBeans>

<!-- 引入json 输入/输出支持 -->

<jaxrs:providers>

<bean class=*"org.codehaus.jackson.jaxrs.JacksonJaxbJsonProvider"*/>

<bean class=*"org.apache.cxf.jaxrs.provider.JAXBElementProvider"*/>

</jaxrs:providers>

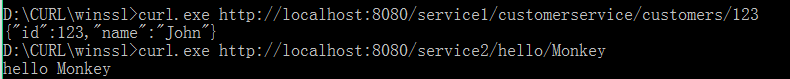
</jaxrs:server>

<bean id=*"helloBean"* class=*"demo.jaxrs.server.Hello"* />

<bean id=*"customerBean"* class=*"demo.jaxrs.server.CustomerService"* />

</beans>

curl测试结果如下：



10. Web Application Description Language



11. 由Accept域决定

12. Java API for RESTful Web Services

13. @Produces用于指定输出格式，@Consumes用于指定输入媒体的类型

小结

1. 学习了RESTful Webservice基本编程

2. 学习了CXF与Spring的集成

3. 理解了IOC和DI的思想

4. 学习了Spring的基本用法

5. 学习了JAX-RS application 编程