1、Maven添加jar

|  |
| --- |
| <cxf.version>3.1.7</cxf.version>  <dependency>  <groupId>org.apache.cxf</groupId>  <artifactId>cxf-rt-frontend-jaxws</artifactId>  <version>${cxf.version}</version>  </dependency>  <dependency>  <groupId>org.apache.cxf</groupId>  <artifactId>cxf-rt-transports-http</artifactId>  <version>${cxf.version}</version>  </dependency> |

2、修改web.xml

|  |
| --- |
| <!--配置 cxf 的核心的过滤器-->  <servlet>  <servlet-name>CXFServlet</servlet-name>  <servlet-class>org.apache.cxf.transport.servlet.CXFServlet</servlet-class>  <load-on-startup>1</load-on-startup>  </servlet>  <servlet-mapping>  <servlet-name>CXFServlet</servlet-name>  <url-pattern>/webservice/\*</url-pattern>  </servlet-mapping> |

在web.xml中添加以上内容，顺序需要注意，在springmvc核心控制器之前

3、添加代码，并配置cxf

|  |
| --- |
| @WebService  public interface TestWS {  @WebMethod  public String sayHello(@WebParam(name = "username") String username);  } |

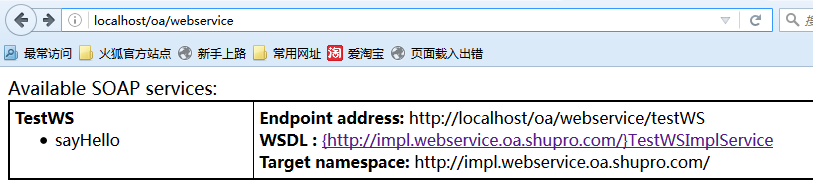
|  |
| --- |
| @WebService  public class TestWSImpl implements TestWS{  @Autowired  SysUserService sysUserService;    @Override  public String sayHello(String username) {  System.out.println("hello:" + username);    return username;  }    // @Override  // public String sayHello(String username) {  // SysUser sysUser = sysUserService.login("test", "123");  // System.out.println("hello:" + sysUser);  // Result result = new Result(0,sysUser,"成功");  //  // return JsonUtil.obj2JsonStr(result);  // }  } |

配置spring-webservice.xml

|  |
| --- |
| <?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:aop=*"http://www.springframework.org/schema/aop"*  xmlns:tx=*"http://www.springframework.org/schema/tx"*  xmlns:jaxws=*"http://cxf.apache.org/jaxws"*  xmlns:http-conf = *"http://cxf.apache.org/transports/http/configuration"*  xsi:schemaLocation=*"*  *http://www.springframework.org/schema/beans*  *http://www.springframework.org/schema/beans/spring-beans.xsd*  *http://www.springframework.org/schema/aop*  *http://www.springframework.org/schema/aop/spring-aop.xsd*  *http://www.springframework.org/schema/tx*  *http://www.springframework.org/schema/tx/spring-tx.xsd*  *http://cxf.apache.org/jaxws*  *http://cxf.apache.org/schemas/jaxws.xsd*  *http://cxf.apache.org/transports/http/configuration*  *http://cxf.apache.org/schemas/configuration/http-conf.xsd"*>    <!-- cxf3.x 不需要导入这些xml  <import resource="classpath:META-INF/cxf/cxf.xml" />  <import resource="classpath:META-INF/cxf/cxf-extension-soap.xml" />  <import resource="classpath:META-INF/cxf/cxf-servlet.xml" /> -->    <!-- 测试发布WebService -->  <jaxws:endpoint id=*"testWS"*  implementor=*"com.shupro.oa.webservice.impl.TestWSImpl"*  address=*"/testWS"* />  </beans> |

将spring-webservice.xml文件添加入web.xml，简单方式是添加到spring.xml文件中

4、浏览器测试



5、客户端调用测试

|  |
| --- |
| public static void main(String[] args) {  JaxWsProxyFactoryBean factoryBean=new JaxWsProxyFactoryBean();  factoryBean.getInInterceptors().add(new LoggingInInterceptor());  factoryBean.getOutInterceptors().add(new LoggingOutInterceptor());  factoryBean.setServiceClass(TestWS.class);  factoryBean.setAddress("http://localhost/oa/webservice/testWS");  TestWS impl=(TestWS) factoryBean.create();  System.out.println(impl.sayHello("张三"));  } |