FUDAN-adweb

MyEclipse 创建基于 X-File 的 Web service 及调用 实例详解

whh

2013-3-6

一二.

1.	环境配置	. 2
2.	创建 Web service project	2
3.	编写 Web service 服务端代码	.4
4.	部署并测试 WebSericeServer	7
5.	创建 Web service 客户端	10
6.	附:Web service 传递自定义数据的实现	14
7.	附录:涉及到的工程的工程目录	19

1. 环境配置

- MyEclipse 7.0
- ➢ JDK 1.6

2. 创建 Web service project

2.1 在 MyEclipse 中选择 File→New→Other,选择 Web Service Project。

🤣 New	
Select a wizard 🛛 🛁 🔷	
Create a Web Service Project	
Wizards:	
type filter text	
 > Image > Java Enterprise Projects S EJB Project (Optional Maven Support) S Enterprise Application Project Web Project (Optional Maven Support) Web Service Project (Optional Maven Support) > Aven > Report Development > Spring > > Show All Wizards. 	
⑦ < Back Next > Finish Cancel]

2.2 Project Name: WebServiceDemo; Framework: X-File; 其他选项默认。

Wew Web Services	s Project				
New Web Service	New Web Service Project				
Web service projec	t creation details				
Web Project Detail	 				
Project Name:	WebServiceDemo				
Location:	Use default location				
Directory:	D:\workspace\Myeclipse_workspace\WebService Browse				
Source folder:	src				
Web root folder:	WebRoot				
Context root URL:	/WebServiceDemo				
Maven					
🔲 Add Maven sup	port				
Learn more about	Maven4MyEclipse				
Web Service & 12E	© LAX MC				
Framework:	XFire (deprecated)				
J2EE specification:	● Java EE 5.0 ○ J2EE 1.4 ○ J2EE 1.3				
?	< Back Next > Finish Cancel				

2.3 选择 next,保持默认值,选择 next。

2.4 Project Library Configuration,选择如下图,点击 finish

🚱 New Web Services Project
Project Library Configuration
Add MyEclipse XFire and User libraries to project
Select the libraries to add to project buildpath Shown @ MyEclipse Libraries - «MyEclipse-Library» Ø XFire 1.2 Core Libraries - «MyEclipse-Library» Ø XFire 1.2 HTTP Client Libraries - «MyEclipse-Library» Ø XFire 1.1 JAXB1 Libraries (deprecated) - «MyEclipse-Library» Ø XFire 1.2 XMLBeans Libraries - «MyEclipse-Library»
? < Back

2.5 创建完成后看到的工程目录如下

4	🚰 WebServiceDemo
	🕮 src
	JRE System Library [MyEclipse 7.0M1]
	▷ 🛋 Java EE 5 Libraries
	XFire 1.2 Core Libraries
	XFire 1.2 HTTP Client Libraries
	🔺 🚌 WebRoot
	META-INF
	> Solution WEB-INF
	牙 index.jsp
	🔺 🗁 WebServices
	👗 services.xml

2.6 完成以上 5 步后, 查看 WebRoot/WEB-INF/web.xml, 这里指明了当遇到/services/*请

求时,将选用 XFireConfigurableServlet 来处理



3. 编写 Web service 服务端代码

3.1 在刚刚创建的 WebServiceDemo 上右键,选择 New→Other→Web services→Web service,创建代码。

🔗 New 🍯 👘 👘	
Select a wizard	-
Wizards:	
type filter text	
🖻 🗁 Web	*
Gewies	
Web Service	
₩ WSDL	
Web-Struts	
> 🗁 XML	
Plug-in Development	=
Diser Assistance	*
Show All Wizards.	
? < Back Next >	Finish Cancel

3.2 选择通过 JAVA CLASS 创建 Web Service。

🛞 New Web	o Service
New Web Select web	Service service creation scenario
Project:	WebServiceDemo 👻
Info: Framework	Java EE 5 Web/XFire project : JAX-WS Ø XFire (deprecated)
Strat egy:	© Create web service from WSDL document (Top-down scenario)
0	< Back Next > Finish Cancel

3.3 创建 MyWebService,通过 New 创建 edu.fudan.adweb.demo package,向导将为工程自动创建对应的接口类与实现类。

Wew Web Service	3 82-9 24	# 4 * J	· .			
New Web Service - Bottom-up Scenario						
Create web service f	rom Java			2		
Web service project:	WebServiceDemo	-				
Web service name:	MyWebService					
Java Implementatio	n					
Java source folder:	src		•	New		
Java package:	edu.fudan.adweb.demo		Browse	New		
Service interface:	IMyWebService					
Service impl. class:	MyWebServiceImpl					
SOAP & Servlet Cor	figuration					
SOAP style/use: w	vrapped/literal 👻					
Servlet scope:	application 👻					
(2)	< Back	Next >	Finish	Cancel		
U.S. C.S. C.S. C.S. C.S. C.S. C.S. C.S.	Buck					

工程目录如下:



3.4 创建成功后查看 WebServices/services.xml,发现配置文件多了以下内容:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <beans xmlns="http://xfire.codehaus.org/config/1.0">
3
40
      <service>
5
          <name>MyWebService</name>
6
          <serviceClass>edu.fudan.adweb.demo.IMyWebService</serviceClass>
7\Theta
           <implementationClass>
8
               edu.fudan.adweb.demo.MyWebServiceImpl
9
          </implementationClass>
10
          <style>wrapped</style>
11
          <use>literal</use>
12
          <scope>application</scope>
13
      </service></beans>
```

3.5 修改 MyWebServiceImpl.java 文件,实现方法内容。

```
1 package edu.fudan.adweb.demo;
 2 //Generated by MyEclipse
3
 4 public class MyWebServiceImpl implements IMyWebService {
 5
 60
       public String example(String message) {
7
           return getAuthor() + ":" + message;
8
       3
       private String getAuthor(){
90
10
           return "whh";
11
       }
12 }
```

4. 部署并测试 WebSericeServer

4.1. 部署 WebServiceServer



在部署向导中选择项目、服务器

🕜 Pro	oject Deployment	ts				×
Man Dep	age Deployme	ents y J2EE projec	ts.	I		- J
Proje Dep i	ect WebServiceD)emo	•			
	Server	Туре	Location			Add
						Kemove
						Redeploy
•		III			Þ	Browse
Dep	loyment Status					
						<u>م</u>
?						ОК



Project Deployment	ts		x		
Manage Deployme	ents y J2EE projects	•	Í,		
Project WebServiceD Deployments)emo	•			
Server	Туре	Location	Add		
✓ 💐 Tomcat 6.x	Exploded	D:\software\Tomcat 6.0\webap;	Remove		
			Redeploy		
•		4	Browse		
Deployment Status					
Deployment failure on Tomcat 6.x. Could not copy all resources to D:\software \Tomcat 6.0\webapps\WebServiceDemo. If a file is locked, you can wait until the lock times out to redeploy, or stop the server and redeploy, or manually remove the deployment at D:\software\Tomcat 6.0\webapps\WebServiceDemo					
0			ОК		

4.2. 启动服务器

- - -	1	▼ 🎱 🍃 🌏 🔝 🖓 ▼ MyEclipse Derby	•	6	• 💰 • 🎋	•
mplia	X	wyeciipse romcat	۲,			
	×	Tomcat 6.x	Þ	0	Start	
	₿	Configure Server			Stop	
ents	는 번	Manage Deployments		8	Configure	
	_			-		

- 3age) {
- 4.3. 测试 WebService
 - 4.3.1 通过 MyEclipse 自带 Explorer 测试

MyEclipse 提供了一个 Web Service Explorer 来测试 Web Service



在 Explorer 中选择 WSDL Main, 在右边的 Registry Name 中输入 hello(这是因为 我 们 定 义 的 service 中 example 方 法 需 要 参 数), 在 URL 中 输 入 <u>http://localhost:8080/WebServiceDemo/services/MyWebService?wsdl</u>, 单击"Go"。

web Services Explorer	* M @ [] \$ \$ \$
९इ. Navigator 🔗 🖉	Actions
品 <mark>UDDI Main</mark> 由	1 Open Registry
	Specify the name and inquiry URL of the registry to be opened. Selecting a favorite registry will automatically fill in the required information. Press Ge to open the registry.
	Favorite Registry
	Registry Name
	hello
	Inquiry URL
	http://localhost:8080/WebServiceDemo/services/MyWebService?wsdl
	Go Reset
	1 Status
	IWAB01391 The registry named hello located at http://localhost.8080/WebServiceDemo/services/MyWebService?wsdl was opened successfully.
	E

4.3.2 直接通过外部浏览器测试

在	外	部	浏	览	器	地	址	栏	中	输	入	:
											•	

<u>http://localhost:8080/WebServiceDemo/services/MyWebService?wsdl</u>,打开如下

则说明部署成功。

C n localnost:8080/webservices/Mywebservicer/wsdi	22 Q =
2 复且 2 我的收藏 2 技术收藏	
This XML file does not appear to have any style information associated with it. The document tree is shown below.	
v (vdlidefinitions xnln:tns="http://demo.adveb.fudan.edu" xnln:vsdlsoap="http://schemas.xnlsoap.org/vsdl/soap/" xnln:soap12="http://vww.v3.org/2003/05/soap-envelope" xnln:scad="http://www.v3.org/2001/ME.Schema" xnls:scad=nell="http://schemas.xnlsoap.org/vsdl/soap=coling/" xnln:scagencl2="http://www.v3.org/2003/05/soap-envelope" xnls:scad="http://schema" xnls:scad=nell="http://schemas.xnlsoap.org/vsdl/soap=coling/" xnln:scagencl2="http://www.v3.org/2003/05/soap-envelope" xnls:scad="http://schema" xnls:scad=nell="http://schemas.xnlsoap.org/vsdl/soap=coling/" xnln:scagencl2="http://www.v3.org/2003/05/soap=envelope" xnls:scad="http://schema" xnls:scad=nell="http://schemas.xnlsoap.org/vsdl/soap=coling/" xnls:scagencl2="http://www.v3.org/2003/05/soap=envelope" xnls:scad="http://schema" xnls:scad=nell="http://schemas.xnlsoap.org/vsdl/soap=coling/" xnls:scagencl2="http://schemas.xnls:scad="http://schemas" xnls:scad="http://schemas.xnls:scad="http://schemas" xnls:scad="http://schemas" xnls:scad="http://schemas" xnls:scad="http://schemas" xnls:scad="http://schemas" xnls:scad="http://schemas" xnls:scad="http://schemas	
<pre>v'xadischema xmlns:xxdd"http://www.v3.org/2001/XMLSchema' attributeFormDefault="qualified" elementFormDefault="qualified" targetNamespace="http://demo.adweb.fudam.edu"> v(xxdischemant mame="example") v(xxdischemant mame="example")</pre>	
<pre>v(xsd:sequence) (xsd:sequence) (xsd:sequence) (/xsd:sequence) (/vsd:sequence)</pre>	
<pre>\v_startowstart.yp/> \v_startowstart.yp/> \v_startowstart.yp/>\v_startowstart.yp/>\v_startowstart.yp/>\v_startowstart.yp/ \v_startowstart.yp/>\v_startowstart.yp/ \v_startowstart.yp/>\v_startowstart.yp/ \v_startowstart.yp/ \v_startowstar</pre>	
♥ Gzslsequence>	
<pre>/xsticomplextype/ /xstichema> /xstichema></pre>	
▼(vsdl:nessage name="parameters" element="tns:example">(vsdl:nessage) (vsdl:nessage)	
▼(vsdl:nessage name"parameters" element="tns:exampleResponse"> ⟨vsdl:part name"parameters" element="tns:exampleResponse"> ⟨vsdl:nessage⟩	
▼(vsdl)potType name="MyMeDServicePotType") √ (vsdl)opt=tion name"=cxample% (vsdl)input name"=cxample%pouert" message="tms:exampleRequest"> /vsdl:input)<br (vsdl)input name"=cxampleRequest" message="tms:exampleRequest">(vsdl)input name"=cxampleRequest" message="tms:exampleRequest"> <td></td>	
<pre>{/wsl:operation></pre>	
<pre>(vedisoap:tbinding style= document transport= http://schemas.xmisoap.org/soap/http:// v(vedisoap:operation name="scample") (vedisoap:operation soapActions"/) v(vedisoap:operation scampleschert")</pre>	
<pre>veduringut name = samplaneques/ <veducationaphody use="literal"></veducationaphody></pre>	

5. 创建 Web service 客户端

经过以上几步,我们已经成功发布了一个 WebService,现在需要创建一个客户端来调用 该服务。调用 web service 的客户端可以是一个简单的 Java Project,也可以是另外一个 web service,在此步骤中我们创建一个 Java Project 来调用在以上步骤创建的 web service。

5.1 创建 Java Project: File→New→Java Project. Project Name: TestWebService

🚱 New Java Project	
Create a Java Project	
Create a Java project in the workspace or in an	external location.
Project name: TestWebService	
Contents	
Oreate new project in workspace	
○ Create project from existing source	
Directory: D:\workspace\Myeclipse_workspa	ce\TestWebService Browse
IRE	
Use default JRE (Currently 'MyEclipse 7.0M	1') <u>Contigure JREs</u>
O Use a project specific JRE: MyEc	lipse 7.0M1 👻
O Use an execution environment JRE: J2SE-	1.5 •
Project layout	
Use project folder as root for sources and	class files
Oreate separate folders for sources and cl	ass files <u>Configure default</u>
Working sets	
Add project to working sets	
Working sets:	▼ Select
⑦ < Back N	lext > Finish Cancel

5.2 引入 MyEclipse Libraries,右键 TestWebServic→Build Path→Add Library

type filter text Java Build Path → ♥ ♥ ↔ ♥ ♥ ↔ ♥ ♥ ↔ ♥ ♥ ↔ ♥ ♥ ↔ ♥ ♥ ♥ ♥
Resource Image: Constraint of the second consecond conseconstraint of the second constraint of the
Project References Refactoring History Run/Debug Settings

选择 MyEclipse Libraries

🤣 Add Library	
Add Library Select the library type to add.	5
JRE System Library JUnit Maven Managed Dependencies MyEclipse Libraries Plug-in Dependencies Tapesuy Framework (plugin default) User Library WTP EAR Libraries (MyEclipse incompatible) WTP JSF Libraries (MyEclipse incompatible) WTP Server Runtime (MyEclipse incompatible) WTP Web App Libraries (MyEclipse incompatible)	
⑦ < Back Next > Finish	Cancel

选择 XFile 1.2 Core Libraries 和 XFire 1.2 HTTP Client Libraries

🤣 Add Library	
Add MyEclipse Libraries Select a MyEclipse Library to add to classpath	à
Selected MyEclipse Library:	
 Spring 2.5 Web Libraries Spring 2.5 J2EE Libraries Spring Webflow 1.0 Core Spring JavaConfig 1.0 Core XFire 1.2 Core Libraries XFire 1.2 JAXB2 Libraries XFire 1.1 JAXB1 Libraries (deprecated) YEire 1.2 YMI Beans Libraries XFire 1.2 HTTP Client Libraries XFire 1.2 Security Libraries XFire 1.2 Miscellaneous Libraries JAX-WS 2.1 Runtime Libraries (Project Metro 1.1) JAX-WS 2.1 API Libraries 	
? < Back Next > Finish	Cancel

5.3 创建 WebServiceClient.java

Create a new Java o Source folder: Package:	lass. TestWebService/src edu.fudan.adweb.test	Browse
Source folder: Package: Enclosing type:	TestWebService/src edu.fudan.adweb.test	Browse
Package:	edu.fudan.adweb.test	
Enclosing type:		Browse
		Browse
Name:	WebServiceClient	
Modifiers:	● public	ected
Superclass:	java.lang.Object	Browse
Interfaces:		Add
		Remove
Which method stub	would you like to create?	
[public static void main(String[] args)	
	Constructors from superclass	
Do you want to add	Innerited abstract methods	up here)
Do you want to add	Generate comments	
	_	
?	Finish	Cancel
₿/**		
package edu.fudar	.adweb.test;	
import java.net.U	RL;	
import org.codeha	us.xfire.client.Client;	
B/**		
* @author whh		
* */		
public class Webs	erviceClient {	
* Oparam arc	3	
*/		
public static	void main(String[] args) {	
// TODO #	uto-generated method stub	
try {		lhost:8080/WebServiceDemo/services/MyWebSer
try { Clier	t client = new client(new UKL("nttp://loca	
try { Clier Objec	<pre>t client = new Client(new UKL("nttp://ioca t[] results = client.invoke("example", new control to the terminal of term</pre>	Object[] {"hello"});
try { Clier Objec Syste	<pre>t client = new client(new UKL["nttp://loca t[] results = client.invoke("example", new m.out.println((String) results[0]); MalformedURLException e) {</pre>	<pre>Object[] {"hello"});</pre>
try { Clier Objec Syste } catch { // TC	<pre>t client = new client(new UKL("nttp://loca t[] results = client.invoke("example", new m.out.println((String) results[0]); MalformedURLException e) { DO Auto-generated catch block</pre>	<pre>Object[] {"hello"});</pre>
try { Clier Objec Syste } catch (// TC e.pri	<pre>t client = new client(new Ukl("nttp://loca t[] results = client.invoke("example", new m.out.println((String) results[0]); MalformedURLException e) { DO Auto-generated catch block ntStackTrace();</pre>	Object[] {"hello"});
try { Clier Objec Syste } catch // TC e.pri } catch // T	<pre>t client = new client(new Ukl("nttp://loca t[] results = client.invoke("example", new m.out.println((String) results[0]); MalformedURLException e) { DO Auto-generated catch block ntStackTrace(); Exception e) { DO Auto-generated catch block </pre>	Object[] {"hello"});
try { Clier Objec Syste } catch (// TC e.pri e.pri	<pre>t client = new client(new Ukl("nttp://loca t[] results = client.invoke("example", new m.out.println((String) results[0]); MalformedURLException e) { DO Auto-generated catch block ntStackTrace(); Exception e) { DO Auto-generated catch block ntStackTrace();</pre>	Object[] {"hello"});
try { Clier Objec Syste catch (// TC e.pri } catch (// TC e.pri } }	<pre>t client = new client(new Ukl("nttp://loca t[] results = client.invoke("example", new m.out.println((String) results[0]); MalformedURLException e) { DO Auto-generated catch block ntStackTrace(); Exception e) { DO Auto-generated catch block ntStackTrace();</pre>	Object[] {"hello"});
<pre>try { Clier Objec Syste } catch (</pre>	<pre>t client = new client(new Ukl("nttp://loca t[] results = client.invoke("example", new m.out.println((String) results[0]); MalformedURLException e) { DO Auto-generated catch block ntStackTrace(); Exception e) { DO Auto-generated catch block ntStackTrace(); </pre>	Object[] {"hello"});
<pre>try { Clier Objec Syste catch (</pre>	<pre>t client = new client(new Ukl("nttp://loca t[] results = client.invoke("example", new m.out.println((String) results[0]); MalformedURLException e) { DO Auto-generated catch block ntStackTrace(); Exception e) { DO Auto-generated catch block ntStackTrace();</pre>	Object[] {"hello"});

数。见 3.5)

5.4 在确认服务器启动的情况下,运行 HelloWebService.java,右键点击 Run As→Java 13/20

Application,在控制台中看到如下输出

Problems @ Javadoc 😣 Declaration 🚍 Console 🛛 <terminated> WebServiceClient [Java Application] D:\software\MyEclipse 7 whh:hello

至此,通过 Java Project 调用 web service 测试完成。

6. 附: Web service 传递自定义数据的实现

在以上过程中,web service 返回的为简单的 String,可以通过 5.3 中的 Client 直接调用 并得到返回值,但是某些时候(在实际项目中)我们可能需要 web service 返回较复杂 的值,这个时候需要通过生成客户端的方式来调用 web service。

在下面的实例中,我们让 web service 返回一个我们自定义的 MyData 数据的 ArrayList。

6.1 首先,我们在创建 MyWebService(见 3.3)后,另外创建一个 edu.fudan.adweb.mydata

package 和 MyData 类,代码如下:

```
10/**..
 4 package edu.fudan.adweb.mydata;
 50/**
 6
   * @author whh
 8 */
 9 public class MyData {
      private String name;
10
      private int ID;
12⊖
      public void setName(String name) {
13
           this.name = name;
14
      - }
150
      public String getName() {
16
           return name;
17
       }
180
      public void setID(int iD) {
19
           ID = iD;
20
      }
210
      public int getID() {
          return ID;
22
23
       3
24 }
25
此时工程目录为:
```

```
⊿ 🥵 WebServiceDemo
```

⊿ (∄ src

- 🔺 🌐 edu.fudan.adweb.demo
 - IMyWebService.java
 - MyWebServiceImpl.java
- edu.fudan.adweb.mydata
 - ⊳ 🚺 MyData.java
- JRE System Library [MyEclipse 7.0M1]
- 👂 🛋 Java EE 5 Libraries
- XFire 1.2 Core Libraries
- XFire 1.2 HTTP Client Libraries
- > >> WebRoot
- > > WebServices

6.2 之后修改 IMyWebService.java 和 MyWebServiceImpl.java,增加 getMyDatas()方法。

```
1 package edu.fudan.adweb.demo;
2
3 import java.util.ArrayList;
4
5 import edu.fudan.adweb.mydata.MyData;
6
7 //Generated by MyEclipse
8
9 public interface IMyWebService {
10
11
       public String example(String message);
12
13
       public ArrayList<MyData> getMyDatas();
14 }
```

MyWebServiceImpl.java:

```
1 package edu.fudan.adweb.demo;
 2
 3@import java.util.ArrayList;
 4
 5 import edu.fudan.adweb.mydata.MyData;
 6 //Generated by MyEclipse
 7
 8 public class MyWebServiceImpl implements IMyWebService {
 9
100
       public String example(String message) {
11
           return getAuthor() + ":" + message;
12
       - }-
130
      private String getAuthor() {
          return "whh";
14
15
       - }
160
       public ArrayList<MyData> getMyDatas() {
           // TODO Auto-generated method stub\
17
           ArrayList<MyData> ret = new ArrayList<MyData>();
18
19
20
           MyData d1 = new MyData();
21
           d1.setID(1);
22
           d1.setName("data1");
23
           MyData d2 = new MyData();
24
25
           d2.setID(2);
26
           d2.setName("data2");
27
28
           ret.add(d1);
29
           ret.add(d2);
30
31
           return ret;
32
       3
33 }
```

6.3 重新部署后可以在 wsdl 文件中看到如下内容 (方框中),可以看到我们定义的返回

值。



6.4 采用生成客户端的方式来调用 web service 时,不能够在 Java Project 中生成,我们 需要另外建立一个 web service project(参考步骤 2.1 至 2.5),新建的 web service project 名字为 WebServiceClient.

```
WebServiceClient
    🗯 src
  JRE System Library [MyEclipse 7.0M1]
  Java EE 5 Libraries
  XFire 1.2 Core Libraries
  XFire 1.2 HTTP Client Libraries
  WebRoot
     META-INF
    WEB-INF
         🕞 lib
         🕅 web.xml
      🔐 index.jsp
  WebServices
      👗 services.xml
    ......
          · -
```

6.5 接下来我们需要通过 wsdl 文件生成一个客户端。(6.5 整个步骤中需要保证 Tomcat

服务器运行中)

在刚刚创建的 WebServiceClient 上右键,选择 New→Other→Web services→Web service client,创建代码。

🕖 New	# 4 + 3		
Select a wizard			→
Create a new web service client			
Wizards:			
type filter text			
👂 🗁 Swing			*
Display="block">Display="block">Display=block			
D 🔁 UML			
🛛 🗁 Web	_		
🔺 🗁 Web Services			
🏄 Web Service			
🔊 Web Service Client			
2 W3DL	-		=
Veb-Struts			-
Show All Wizards.			
(?) < Back Next	> Fin	ish	Cancel

根据向导配置如下: WSDL URL 为之前发布的 web service 地址(和 4.3 种 URL 地址

相同), java source folder 和 java package 都是点击 New 根据向导提示生成。

New Web Service C	lient 🧬 🖉 👘 🐞 👘 🖤 👘 👘	10.1	- • ×
New Web Service	Client est client from WSDL document		Ð
Web service project:	WebServiceClient		
Service Definition			
🔘 WSDL File		-	Browse
WSDL URL	http://localhost:8080/WebServiceDemo/services/MyWe	bServic 👻	
Java source folder:	src_MyWebService_client	•	New
Java package:	edu.fudan.adweb.client Br	owse	New
Note: The generated	d web service client will use Java 5, JSR-181 web service	annotation	s
-Service Customization	on		
JAXB2 binding file:			Browse
0	< Back Next >	Finish	Cancel

一路 next 后 finish,最后配置好后工程目录如下:(src_MyWebService_client 下文件 均是自动生成的)

WebServiceClient

/ 进 进

- ▲ ﷺ src_MyWebService_client
 - 4 🔠 edu.fudan.adweb.client
 - MyWebServiceClient.java
 - MyWebServicePortType.java
 - 🖻 🌐 edu.fudan.adweb.demo
 - edu.fudan.adweb.mydata
- JRE System Library [MyEclipse 7.0M1]
- Java EE 5 Libraries
- XFire 1.2 Core Libraries
- XFire 1.2 HTTP Client Libraries
- XFire 1.2 JAXB2 Libraries
- 🔺 🚌 WebRoot

 - > >> WEB-INF
 - 🔐 index.jsp
- WebServices
 - 👗 services.xml
- 6.6 接下来,我们在 WebServiceClient src 文件夹下创建 edu.fudan.adweb.test package 和

TestClient.java 文件,内容如下:

- WebServiceClient
 - 4 / 🖽 src
 - edu.fudan.adweb.test
 TestClient.java
 - ▲ ﷺ src_MyWebService_client
 - 🔺 🔠 edu.fudan.adweb.client
 - MyWebServiceClient.java
 - MyWebServicePortType.java
 - edu.fudan.adweb.demo
 - 🖻 🌐 edu.fudan.adweb.mydata
 - JRE System Library [MyEclipse 7.0M1]
 - Java EE 5 Libraries
 - XFire 1.2 Core Libraries
 - XFire 1.2 HTTP Client Libraries
 - XFire 1.2 JAXB2 Libraries
 - > >> WebRoot
 - VebServices

TestClient.java:

```
whh
```

```
10/**[.]
 4 package edu.fudan.adweb.test;
 5
 6=import java.util.ArrayList;
 7 import java.util.List;
 8
 9 import edu.fudan.adweb.client.MyWebServiceClient;
10 import edu.fudan.adweb.client.MyWebServicePortType;
11 import edu.fudan.adweb.mydata.MyData;
120/**
13 * @author whh
14
15 */
16 public class TestClient {
17
180
        /**
19
        * @param args
        */
20
210
       public static void main(String[] args) {
22
           // TODO Auto-generated method stub
23
           MyWebServiceClient client = new MyWebServiceClient();
24
           MyWebServicePortType service = client.getMyWebServiceHttpPort();
25
26
            List<MyData> datas = new ArrayList<MyData>();
27
            datas = service.getMyDatas().getMyData();
28
29
            for(MyData d : datas) {
30
                System.out.println(d.getName().getValue() + ":" + d.getID());
31
            }
32
        }
33
34 }
35
```

其中,MyWebServiceClient 和 MyWebServicePortType 均是根据 wsdl 生成客户端时

自动生成的文件。

FUDAN-adweb

运行 TestClient, 控制台输出如下:



至此,根据 wsdl 文件生成客户端,传递自定义数据的测试完成。

7. 附录: 涉及到的工程的工程目录

整个过程中一共建立的三个工程,分别如下:

➤ Web Service Project: WebServiceDemo 实现了服务端代码,定义了自定义数据类型

- a 澾 WebServiceDemo
 - 🔺 进 src
 - a 🌐 edu.fudan.adweb.demo
 - IMyWebService.java
 - MyWebServiceImpl.java
 - edu.fudan.adweb.mydata
 - ⊳ 🚺 MyData.java
 - JRE System Library [MyEclipse 7.0M1]
 - 🖻 📥 Java EE 5 Libraries
 - XFire 1.2 Core Libraries
 - XFire 1.2 HTTP Client Libraries
 - 🔺 🚌 WebRoot
 - D 🗁 META-INF
 - 🛛 🏷 WEB-INF
 - J index.jsp
 - WebServices
 - 👗 services.xml
- ▶ Java Project: TestWebService 实现了调用服务端的客户端代码
 - ▲ ≧ TestWebService
 - 🔺 进 src
 - 🔺 🌐 edu.fudan.adweb.test
 - VebServiceClient.java
 - JRE System Library [MyEclipse 7.0M1]
 - XFire 1.2 Core Libraries
 - ▷ 🛋 XFire 1.2 HTTP Client Libraries
- ▶ Web Service Project: WebServiceClient 实现了通过 wsdl 文件自动生成客户端调用
 - 服务端的代码,接受了服务端传递过来的自定义数据。
 - ▲ 2 WebServiceClient
 - 🔺 / 🕮 src
 - a 🌐 edu.fudan.adweb.test
 - I TestClient.java
 - ▲ ﷺ src_MyWebService_client
 - a A edu.fudan.adweb.client
 - MyWebServiceClient.java
 - MyWebServicePortType.java
 - 🖻 🌐 edu.fudan.adweb.demo
 - edu.fudan.adweb.mydata
 - JRE System Library [MyEclipse 7.0M1]
 - 👂 🛋 Java EE 5 Libraries
 - XFire 1.2 Core Libraries
 - XFire 1.2 HTTP Client Libraries
 - XFire 1.2 JAXB2 Libraries
 - 🔺 🚌 WebRoot
 - META-INF
 - 🖻 📂 WEB-INF
 - 🔐 index.jsp
 - WebServices
 - 👗 services.xml