作业 1：服务开发和调用

BY2006105 陈天宇

本文档主要用于说明实验流程和代码解释。

This document is for illustration of experiments and explaination of Code.

```
class Person
{
    String name;
    int age;
    boolean gender;
    set/getName();
    set/getAge();
    String sayHello()
    {
        Return "Hello World!" + name;
    }
}
```

基于 Apach Dubbo 将上述类的方法对外提供 RPC 服务并调用

Config the zookeeper registry

1. download from https://zookeeper.apache.org/releases.html
2. follow the instructions on https://phoenixnap.com/kb/install-apache-zookeeper

Start the service provider

```
mvn clean package
mvn -Djava.net.preferIPv4Stack=true -Dexec.mainClass=org.apache.dubbo.samples.provider.Application exec:java
```

If all things are ready, you will see the information below:

```
[INFO] Scanning for projects...
[INFO]
[INFO] ----------------< org.apache.dubbo:dubbo-samples-api >----------------
[INFO] Building dubbo-samples-api 1.0-SNAPSHOT
[INFO] --------------------------------[ jar ]---------------------------------
[INFO]
[INFO] --- exec-maven-plugin:3.0.0:java (default-cli) @ dubbo-samples-api ---
log4j:WARN No appenders could be found for logger (org.apache.dubbo.common.logger.LoggerFactory).
log4j:WARN Please initialize the log4j system properly.
log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more info.
dubbo service started
```

build successfully

Invoke the service consumer

Of course you can start a new node and refer to the service above. For convenience, you can start another terminal (another process) and type the following commands.

```
cd dubbo-samples-api
mvn -Djava.net.preferIPv4Stack=true -Dexec.mainClass=org.apache.dubbo.samples.client.Application exec:java
```

After referring to the remote service, you will see the results:

```
log4j:WARN Please initialize the log4j system properly.
log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more info.
Set new name:Tom
Tom
Set new age: 22
The age of Tom is 22
Set new gender:false
Tomis a girl? false
Hello, world!Tom
```

RPC

基于 Java Spring 将上述类的方法对外提供 RESTful 服务并调用

Setup

```
cd spring-restful-demo
mvn clean package
mvn spring-boot:run
```

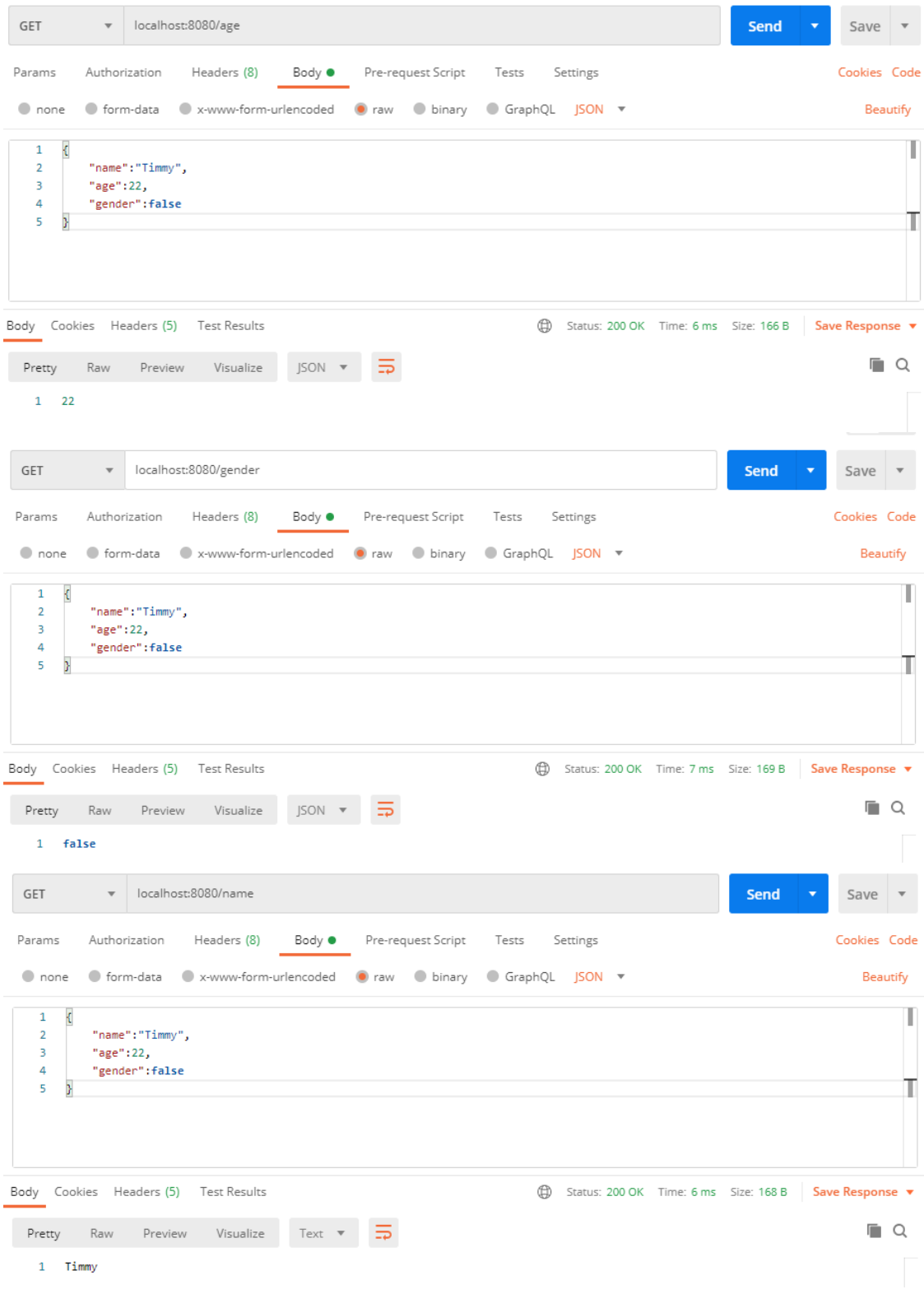
The service will start on localhost:8080

Usage

The Postman software is recommended for APIs Tests.

The results can be seen under the spring-restful-demo/results folder.

Here are some samples：


GET
localhost:8080/age
Send
Save
Params
Authorization
Headers (8)
Body
Pre-request Script
Tests
Settings
Cookies
Code
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
Beautify
{
"name":"Timmy",
"age":22,
"gender":false
}
Body
Cookies
Headers (5)
Test Results
Status: 200 OK
Time: 6 ms
Size: 166 B
Save Response
Pretty
Raw
Preview
Visualize
JSON
22
GET
localhost:8080/gender
Send
Save
Params
Authorization
Headers (8)
Body
Pre-request Script
Tests
Settings
Cookies
Code
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
Beautify
{
"name":"Timmy",
"age":22,
"gender":false
}
Body
Cookies
Headers (5)
Test Results
Status: 200 OK
Time: 7 ms
Size: 169 B
Save Response
Pretty
Raw
Preview
Visualize
JSON
false
GET
localhost:8080/name
Send
Save
Params
Authorization
Headers (8)
Body
Pre-request Script
Tests
Settings
Cookies
Code
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
Beautify
{
"name":"Timmy",
"age":22,
"gender":false
}
Body
Cookies
Headers (5)
Test Results
Status: 200 OK
Time: 6 ms
Size: 168 B
Save Response
Pretty
Raw
Preview
Visualize
Text
Timmy

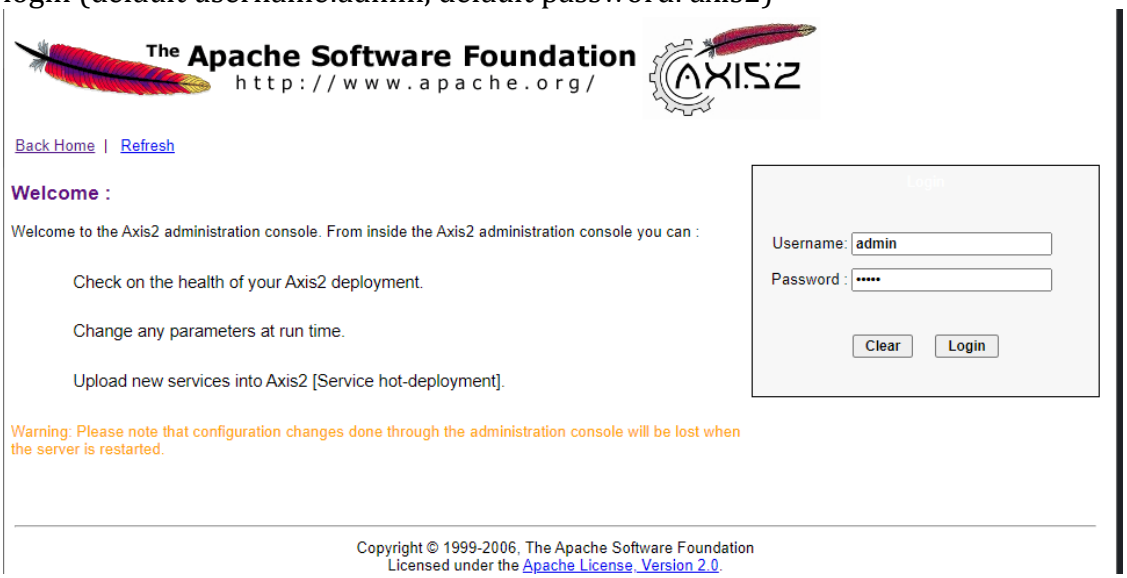
基于 Apache Axis2 将上述类的方法对外提供 Web 服务，生成 WSDL 文件，以及调用服务

pre-requesitions

1. apache tomcat seervlet engine
2. apache ant
3. apache maven

Setup

1. start tomcat engine and download the war-version of apache axis2.
2. copy the axis2.war file into {tomcat_root_dir}/webapp
3. visit the default admin page http://localhost:8080/axis2/axis2-admin and login (default username:admin, default password: axis2)



4. switch into this folder and run apache ant to generate wsdl

```
cd axis-demo
mvn install
ant generate.wsdl
ant generate.service
```

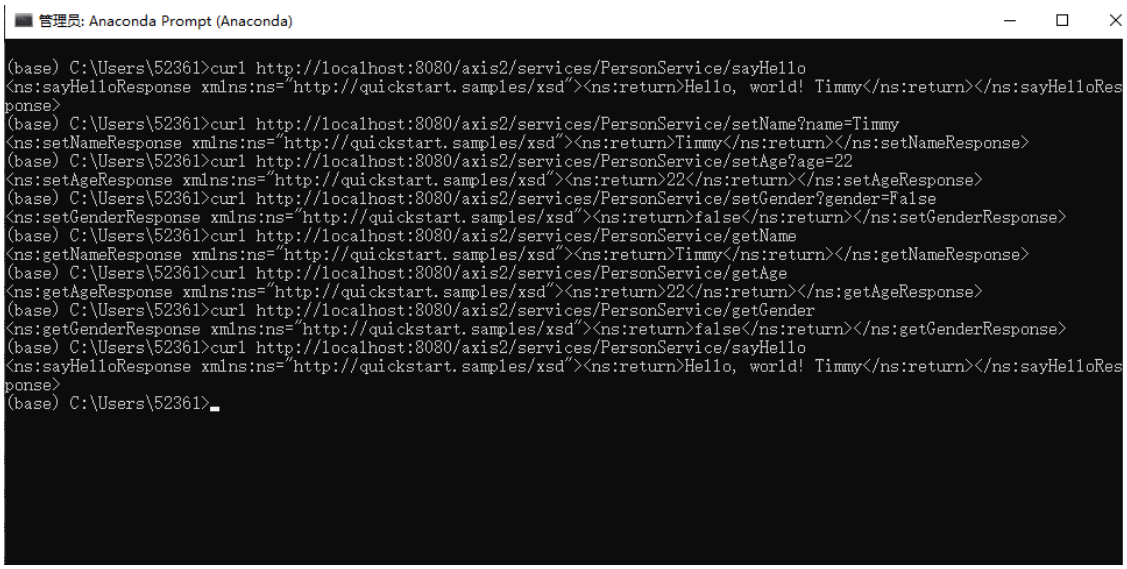
after generation, you will find the PersonService.aar and PersonService.wsdl under axis2-demo/quickstart/build/ 5. open the upload page http://localhost:8080/axis2/axis2-admin/upload and upload the PersonService.aar file. 6. The PersonService can be seen under the servicelist http://localhost:8080/axis2/axis2-admin/listServices

service_list

Invoke the service

Here Curl is used to make tests.

1. set person name
2. set person age
3. set person gender
4. get person name
5. get person age
6. get person gender
7. sayHello


管理员: Anaconda Prompt (Anaconda)
(base) C:\Users\52361>curl http://localhost:8080/axis2/services/PersonService/sayHello
<ns:sayHelloResponse xmlns:ns="http://quickstart.samples/xsd"><ns:return>Hello, world! Timmy</ns:return></ns:sayHelloRes
ponse>
(base) C:\Users\52361>curl http://localhost:8080/axis2/services/PersonService/setName?name=Timmy
<ns:setNameResponse xmlns:ns="http://quickstart.samples/xsd"><ns:return>Timmy</ns:return></ns:setNameResponse>
(base) C:\Users\52361>curl http://localhost:8080/axis2/services/PersonService/setAge?age=22
<ns:setAgeResponse xmlns:ns="http://quickstart.samples/xsd"><ns:return>22</ns:return></ns:setAgeResponse>
(base) C:\Users\52361>curl http://localhost:8080/axis2/services/PersonService/setGender?gender=False
<ns:setGenderResponse xmlns:ns="http://quickstart.samples/xsd"><ns:return>false</ns:return></ns:setGenderResponse>
(base) C:\Users\52361>curl http://localhost:8080/axis2/services/PersonService/getName
<ns:getNameResponse xmlns:ns="http://quickstart.samples/xsd"><ns:return>Timmy</ns:return></ns:getNameResponse>
(base) C:\Users\52361>curl http://localhost:8080/axis2/services/PersonService/getAge
<ns:getAgeResponse xmlns:ns="http://quickstart.samples/xsd"><ns:return>22</ns:return></ns:getAgeResponse>
(base) C:\Users\52361>curl http://localhost:8080/axis2/services/PersonService/getGender
<ns:getGenderResponse xmlns:ns="http://quickstart.samples/xsd"><ns:return>false</ns:return></ns:getGenderResponse>
(base) C:\Users\52361>curl http://localhost:8080/axis2/services/PersonService/sayHello
<ns:sayHelloResponse xmlns:ns="http://quickstart.samples/xsd"><ns:return>Hello, world! Timmy</ns:return></ns:sayHelloRes
ponse>
(base) C:\Users\52361>

results