

Day2 微服务入门之编写HelloWorld

1 打卡任务

作业：

- 1、在provider服务中增加一个greeting方法，接受POST请求，请求参数为request body中传递的Person对象，包含name和gender两个字段。name为String类型；gender为枚举Gender类型，有MALE/FEMALE两个值。返回值为GreetingResponse类型，包含msg和timestamp两个字段，msg为根据gender不同生成的问候语 Hello, Mr.xxx (MALE) / Hello, Ms.xxx (FEMALE)，timestamp为java.util.Date类型的时间戳。
- 2、在consumer服务中增加一个greeting方法，接收外部请求的触发，以调用provider服务的greeting方法，并返回provider的应答消息。

打卡：

- 1、调用provider服务的greeting方法成功，并截图
- 2、调用consumer服务的greeting方法成功，并截图

打卡任务基于Day2的demo项目：



Demo-Day2.zip

2 准备工作

- 1、CSEJavaSDK需要在Java 1.8环境运行。请确保本地环境安装了Java 1.8开发环境及maven。
- 2、配置maven setting文件，确保本地开发环境能够下载CSEJavaSDK的依赖包。

配置方法参考 https://support.huaweicloud.com/devg-cse/cse_03_summary.html

如果您之前使用的是默认的maven setting文件的话，也可以直接使用这份配置文件



settings.xml

覆盖它

3、获取AK/SK

参考文档 https://support.huaweicloud.com/usermanual-iam/zh-cn_topic_0079477318.html，获取自己的AK/SK。

AK/SK需要配置在microservice.yaml中，用于连接华为云上的CSE服务时的认证。

配置方式参考Day2课程培训资料。

4、本地成功运行 Day2 的provider、consumer demo工程项目。

3 在 provider 服务中开发 greeting 方法

1、创建作为请求的Person类、作为返回值的GreetingResponse类以及Gender枚举类型

```
C GreetingResponse.java ×
1 package microservice.demo.training21days.provider.service;
2
3 import java.util.Date;
4
5 public class GreetingResponse {
6     private String msg;
7
8     private Date timestamp;
9
10    public String getMsg() { return msg; }
13
14    public void setMsg(String msg) { this.msg = msg; }
17
18    public Date getTimestamp() { return timestamp; }
21
22    public void setTimestamp(Date timestamp) { this.timestamp = timestamp; }
25
26    @Override
27    public String toString() {...}
34 }
```

```

C Person.java x
1 package microservice.demo.training21days.provider.service;
2
3 public class Person {
4     private String name;
5
6     private Gender gender;
7
8     public String getName() { return name; }
11
12     public void setName(String name) { this.name = name; }
15
16     public Gender getGender() { return gender; }
19
20     public void setGender(Gender gender) { this.gender = gender; }
23
24     @Override
25     public String toString() {...}
32 }

E Gender.java x
1 package microservice.demo.training21days.provider.service;
2
3 public enum Gender {
4     MALE,
5     FEMALE
6 }

```

2、在HelloService中创建greeting方法

```

1... @PostMapping(path = "/greeting")
2... public GreetingResponse greeting(@RequestBody Person person) {
3...     GreetingResponse greetingResponse = new GreetingResponse();
4...
5...     if (Gender.MALE.equals(person.getGender())) {
6...         greetingResponse.setMsg("Hello, Mr." + person.getName());
7...     } else {
8...         greetingResponse.setMsg("Hello, Ms." + person.getName());
9...     }
10...     greetingResponse.setTimestamp(new Date());
11...     return greetingResponse;
12... }

```

3、运行provider服务并进行调用

直接启动有可能会因为与服务中心记录的服务契约不一致而报错，此时可以考虑删除服务中心内的provider服务记录，或升级provider服务的版本。

观察服务契约，可以看到greeting方法的请求参数person在body中传递：

```
consumes:
- "application/json"
produces:
- "application/json"
paths:
  .. /greeting:
    .. post:
      .. .. operationId: "greeting"
      .. .. parameters:
      .. .. - in: "body"
      .. ..   name: "person"
      .. ..   required: true
      .. ..   schema:
      .. ..     $ref: "#/definitions/Person"
      .. .. responses:
      .. ..   200:
      .. ..     description: "response of 200"
      .. ..     schema:
      .. ..       $ref: "#/definitions/GreetingResponse"
```

Person的结构如下:

```
.. Person:
  .. type: "object"
  .. properties:
  ..   name:
  ..     type: "string"
  ..   gender:
  ..     type: "string"
  ..     enum:
  ..       - "MALE"
  ..       - "FEMALE"
  ..   x-java-class: "microservice.demo.training21days.provider.service.Gender"
  ..   x-java-class: "microservice.demo.training21days.provider.service.Person"
```

接口接收请求、返回应答的Content-Type都是application/json, 因此, 发送请求时的body中应该拼写一个json串。

4 在 consumer 服务中开发 greeting 方法

- 1、定义Person、GreetingResponse、Gender类, 保持类名、包名、内部属性与provider一致。
- 2、在HelloService代理接口中增加一个greeting方法:

```
public interface HelloService {
  .. String sayHello(String name);

  .. GreetingResponse greeting(Person person);
}
```

- 3、在HelloConsumerService中增加一个greeting方法，接收请求中的Person参数并通过RPC代理将它发送到provider服务中：

```
..@Path("/greeting")
..@POST
..public GreetingResponse greeting(Person person) {
..    return helloService.greeting(person);
..}
```

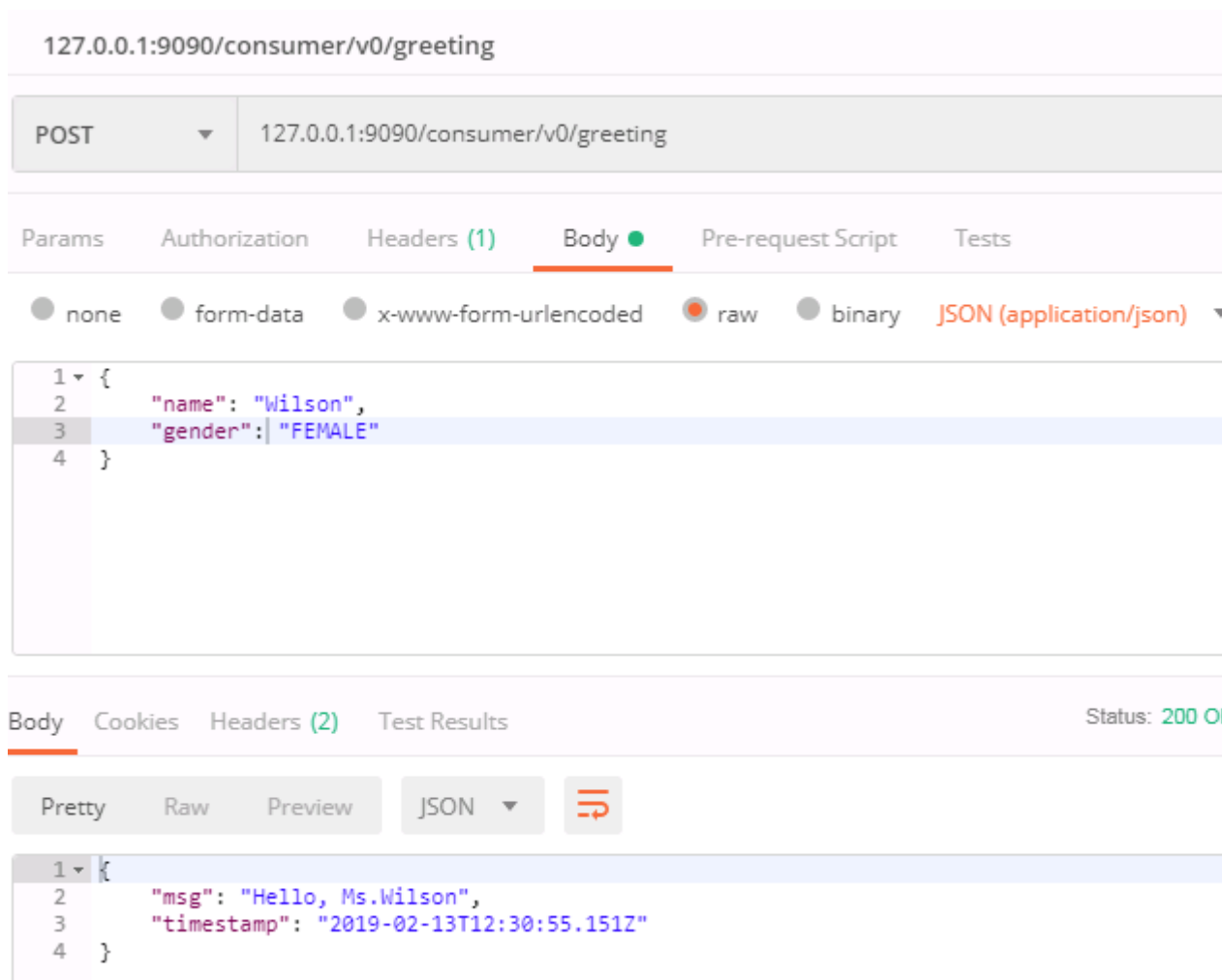
- 4、启动consumer服务，调用它的greeting接口，可以看到正常返回结果。

5 打卡截图

调用provider服务的结果如下：

The screenshot displays a REST client interface. The top section shows the URL `http://127.0.0.1:8080/provider/v0/greeting` and the method `POST`. Below this, the `Body` tab is selected, showing a JSON request body: `{ "name": "Davis", "gender": "MALE" }`. The bottom section shows the `Body` tab for the response, displaying a JSON response body: `{ "msg": "Hello, Mr.Davis", "timestamp": "2019-02-13T11:38:16.192Z" }`. The status bar at the bottom right indicates `Status: 200`.

调用consumer服务的结果如下：



6 总结

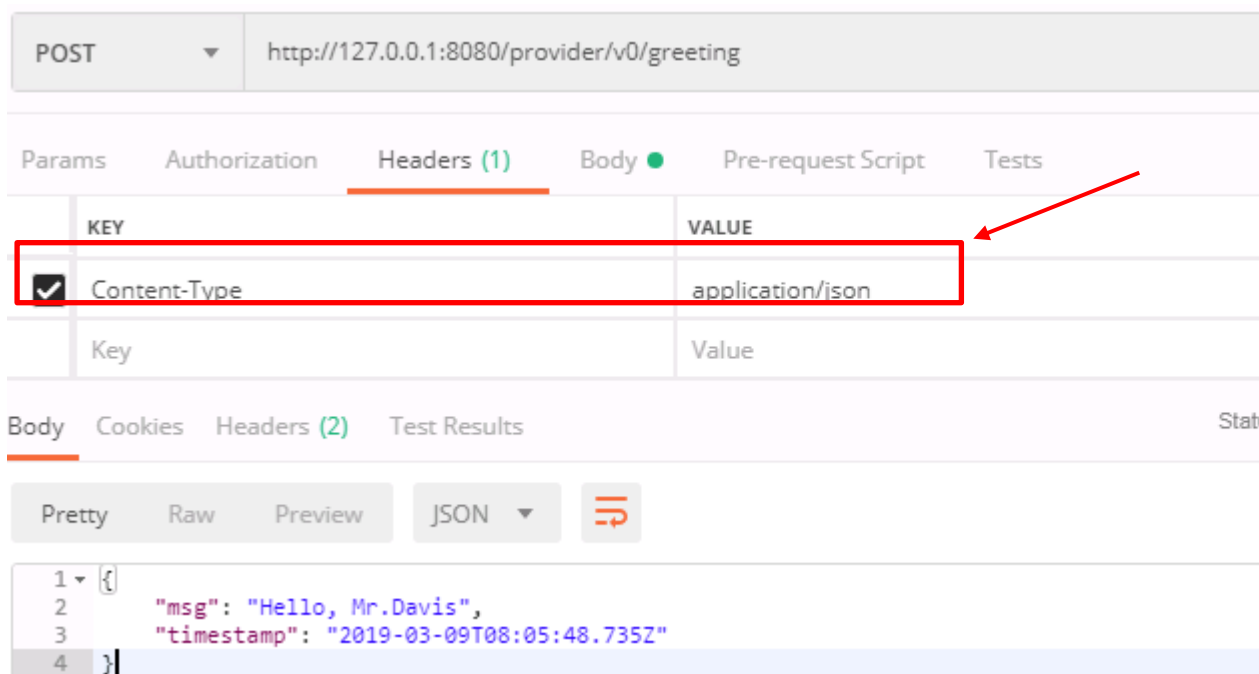
- 1、Java对象类型的参数也会在服务契约中明确描述出来，CSEJavaSDK会根据契约对参数进行序列化和反序列化，用户在业务代码中直接操作Java对象即可，不需要再考虑数据流的序列化问题。因此，参数类型必须是明确的，可以被服务契约声明出来的。例如，Object类型、抽象类型、没有指定具体属性类型的泛型都是不能作为参数的。参考<https://docs.servicecomb.io/java-chassis/zh-CN/build-provider/interface-constraints.html>。
- 2、consumer端和provider端使用的参数类型不一定要来自于同一个jar包中，但两端的参数的类名和包名最好保持完全一致（如同本次打卡作业）。事实上只要参数类型的完整类名是一致的，CSEJavaSDK就会认为这是同一个Java类型。如果consumer端的参数类型和provider端参数类型不在同一个包中，也可以调用通过，但是中间

多了 契约声明类型 -> json串 -> 代码中的参数类型 的转换过程，增加了运行开销。对于这种情况，CSEJavaSDK在加载provider端契约时会打印提示信息，如“Bad practice, low performance, convert from class

microservice.demo.training21days.provider.service2.Person to class

microservice.demo.training21days.provider.service.Person”。

- 3、当在postman中选择body的类型为application/json时，postman会自动在header中加上Content-Type:application/json。微服务在解析body中承载的数据时，会根据Content-Type来判断应该如何处理数据，因此该header不能去掉，否则将导致微服务解析body失败。



- 4、对于实践过程中碰到的问题，可以参考开源的[ServiceComb-Java-Chassis文档](#)、[CSEJavaSDK文档](#)，可以去[华为云论坛CSE板块](#)搜索资料，有意见和建议也可以去[开源代码库](#)提issue。

善用搜索引擎可以更好地搜索相关信息：



域名解析问题 site:bbs.huaweicloud.com

全部

新闻

图片

视频

地图

更多

找到约 9,720 条结果（用时 0.23 秒）

部署任务失败，日志提示DNS解析问题？_云问答_云社区-华为云

<https://bbs.huaweicloud.com/ask/1522461230685923> ▼

烟花易冷. 2018-04-01 21:56. 0. 0. 域名解析不生效的表现是使用ping命令无法解析IP地址，DNS解析失败的原因有很多，具体请参照如下信息：

服务部署启动后报告域名解析失败_微服务引擎_云论坛_华为云

<https://bbs.huaweicloud.com/forum/thread-11806-1-1.html> ▼

2018年10月17日 - 有些时候，在Windows笔记本电脑上做开发的时候也会碰到是由于Netty在Windows多网卡环境下做DNS域名解析的时候，选错了 ...

参考答案：



Demo-Day2-Homework.zip