

Day14-CSE实战之其他服务何如接入之 helloworld

1 打卡任务

作业：

1. 在springboot服务中增加两个个greeting方法，分别为providerGreeting和consumerGreeting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。

打卡：

2. 调用provider服务的greeting方法成功，并截图
3. 调用consumer服务的greeting方法成功，并截图

开发基于Day14 demo进行开发

2 准备任务

1. 能正常运行Day14的provider和consumer

3 为 springboot 服务增加 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

```

public class GreetingResponse {
    private String msg;
    private Date timestamp;

    public String getMsg() { return msg; }

    public void setMsg(String msg) { this.msg = msg; }

    public Date getTimestamp() { return timestamp; }

    public void setTimestamp(Date timestamp) { this.timestamp = timestamp; }
}

```

```

public enum Gender {
    MALE,
    FEMALE
}

```

```

2
3 public class person {
4     private Gender gender;
5     private String name;
6
7
8     public Gender getGender() { return gender; }
9     public void setGender(Gender gender) { this.gender = gender; }
0     public String getName() { return name; }
1     public void setName(String name) { this.name = name; }
2

```

2. helloController中创建consumerGreeting和providerGreeting方法

```

@RestController
@RequestMapping("/consumer/v0")
public class ConsumerController {
    @RequestMapping("/hello/{name}")
    public String ProviderHello(@PathVariable(value = "name") String name) {
        RestTemplate restTemplate = new RestTemplate();
        String s = restTemplate.getForObject(url: "http://provider/provider/v0/hello/" + name, String.class);
        return s;
    }

    @RequestMapping("/greeting")
    public GreetingResponse ConsumerGreeting(@RequestBody Person p) {
        RestTemplate restTemplate = new RestTemplate();
        GreetingResponse response = restTemplate.postForObject(url: "http://provider/provider/v0/greeting", p, GreetingResponse.class);
        return response;
    }
}

```

3. 运行provider服务并进行调用

运行springboot服务后，访问/provider/v0/greeting接口

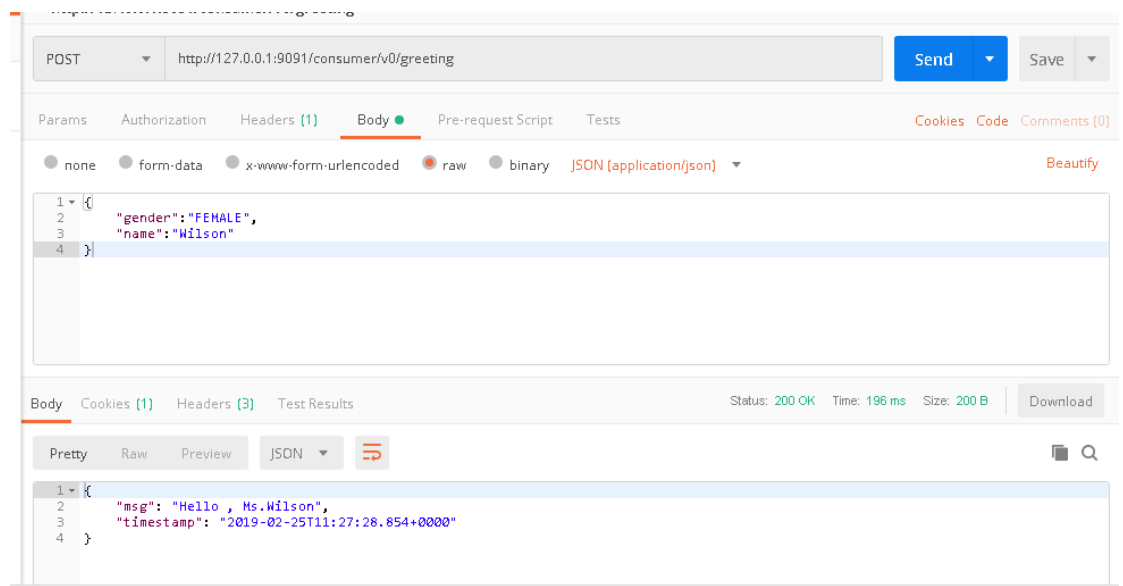
提示 :请启动 Day11 的 consumer 和 provider ,在 Day11demo 中 ,springboot 的 consumer

访问 Day11 的 provider , Day11 的 consumer 访问 springboot 的 provider 接口

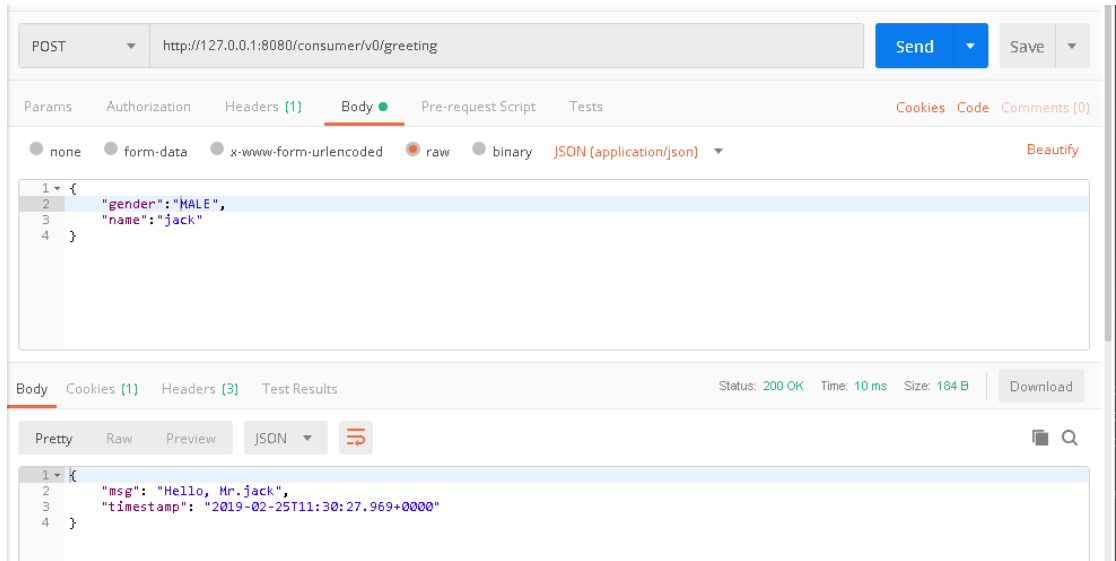
4 打卡截图

在 Day11 和修改后的 spring 全部启动后

1. 访问springboot的consumer接口



2. 访问Day11中的consumer端接口



5 总结

1. 在springboot时，需要在springboot中设置http_proxy。
2. 启动mesher前，我们需要将mesher的监听地址修改为运行机器的外部地址(非127.0.0.1)，如外部地址为：192.168.0.1，这里不能使用127.0.0.1
3. 在接入cse后，通过服务名进行访问非直接使用ip+port访问