**AWS动手实验Lambad for HTTP GET**

**实验目的**

使用Lambda完成HTTP中的Get请求

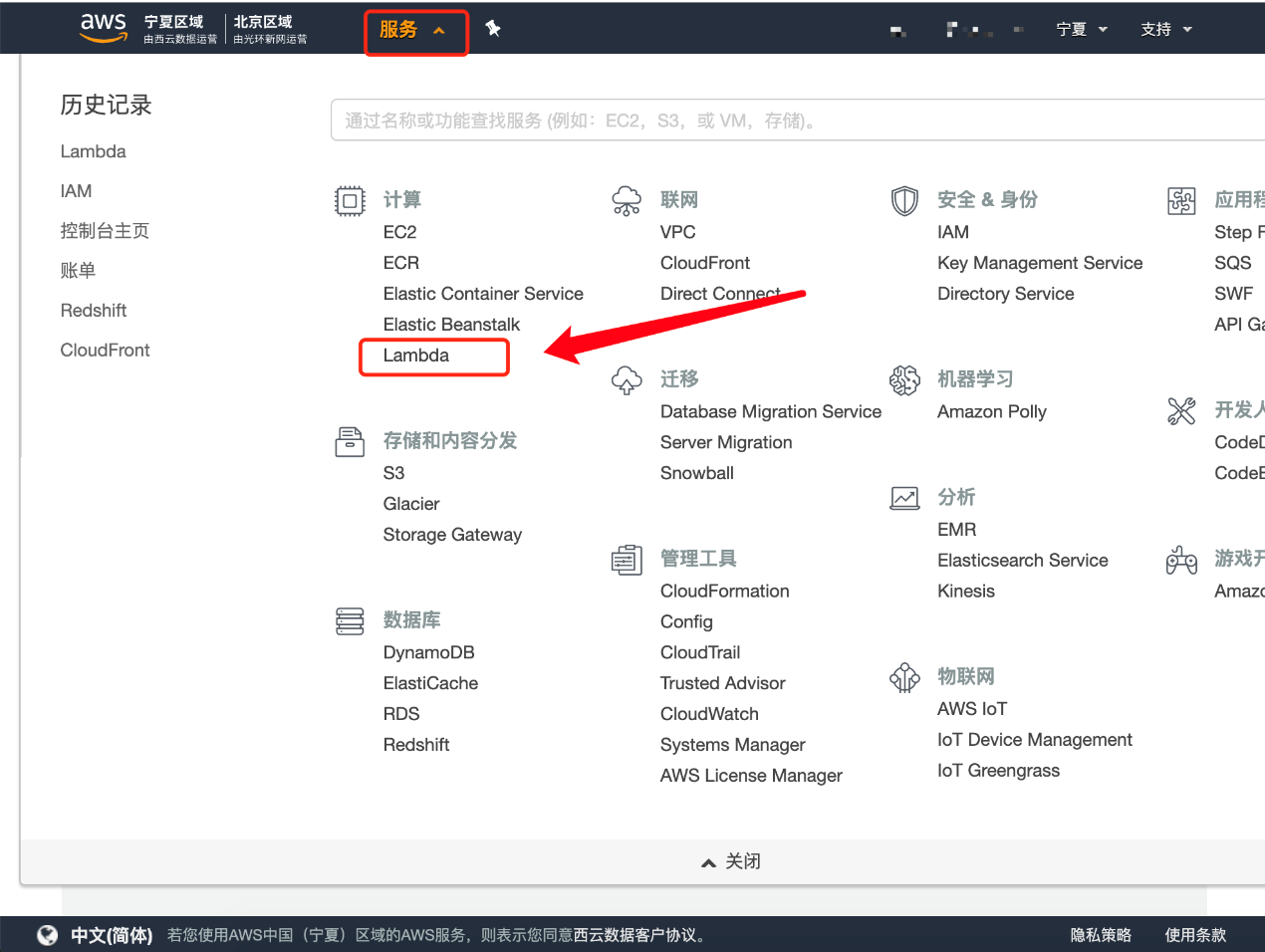
**实验要求**

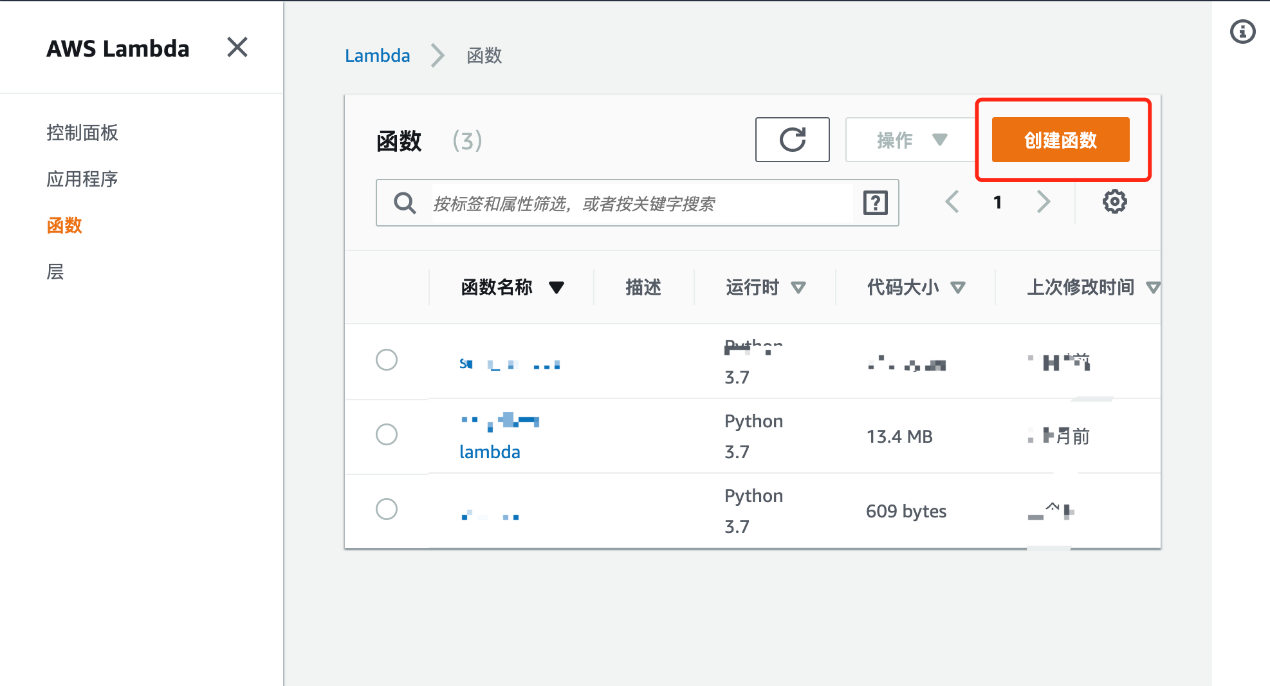
了解Amazon Web Services  
了解无服务器架构  
了解Python  
了解AWS SDK for Python  
了解HTPP中的GET方法

**实验原理**

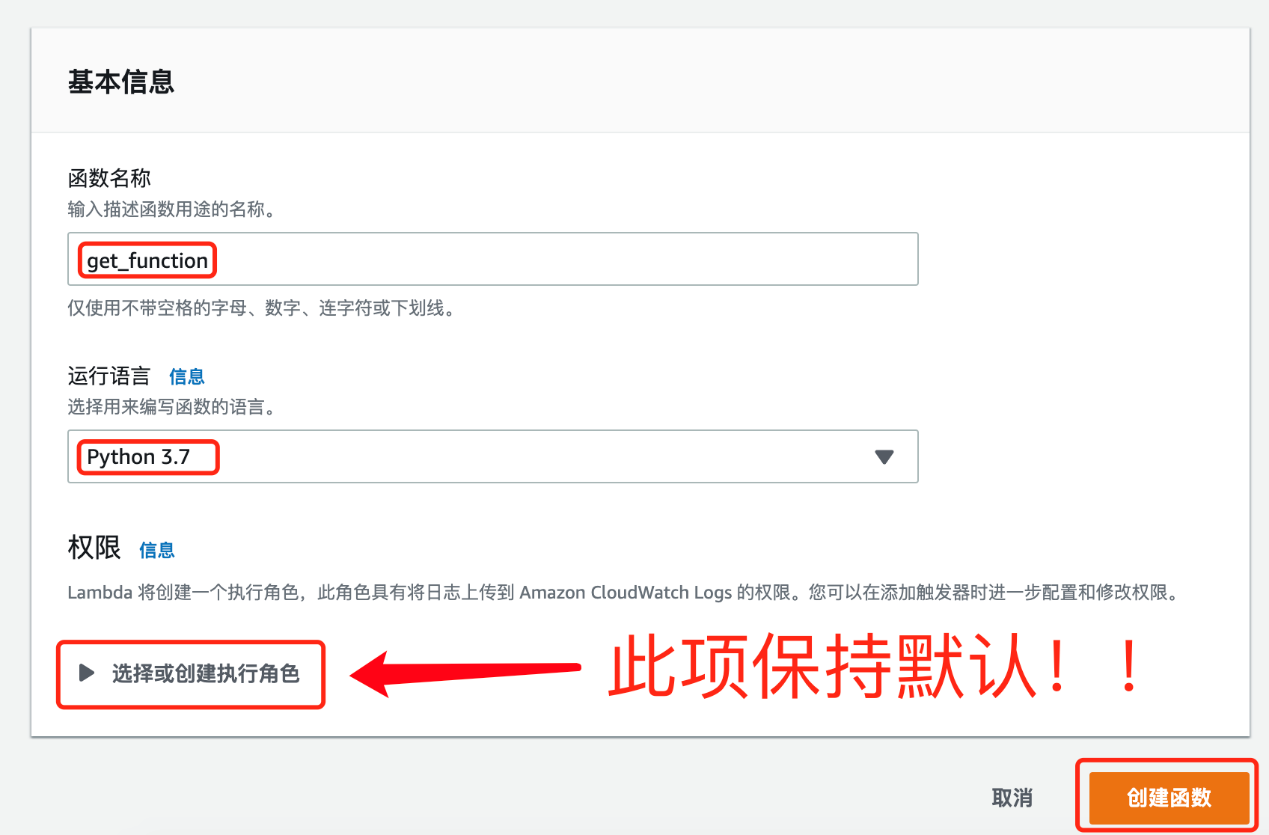
AWS Lambda是一种计算服务，可让您在不配置或管理服务器的情况下运行代码。AWS Lambda仅在需要时执行您的代码并自动扩展，从每天几个请求到每秒数千个。您只需要为您消耗的计算时间付费，当您的代码未运行时不收取任何费用。借助AWS Lambda，您几乎可以为任何类型的应用程序或后端服务运行代码，并且不必进行任何管理。  
HTTP定义了与服务器交换的不同方法，最基本的方法有四种，分别是GET、POST、PUT、DELETE。  
GET方法是最常用的方法，通常用于请求服务器发送某个资源，而且应该是安全的和等幂的。

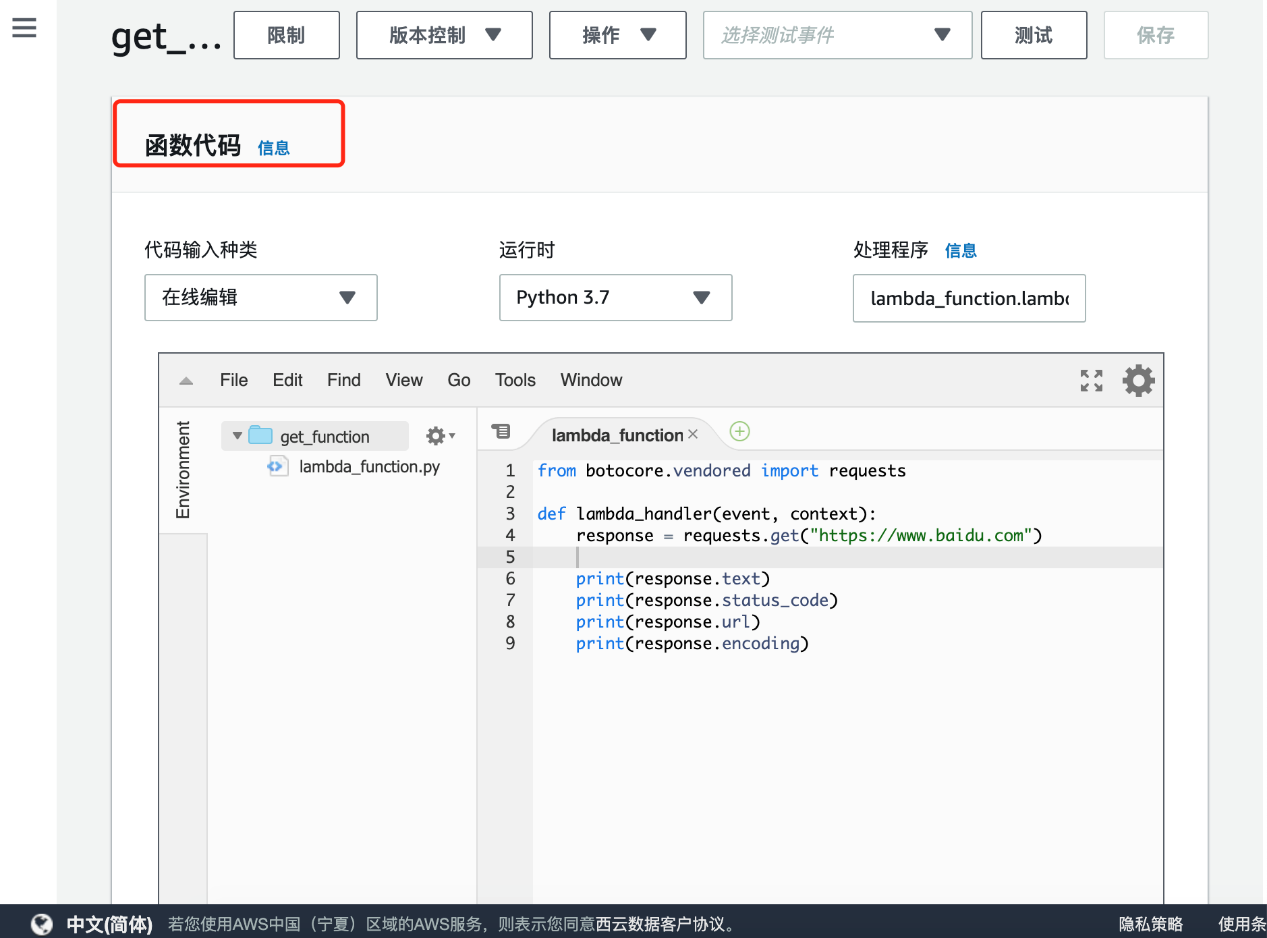
**实验步骤**

1、点击【服务】，选择【Lambda】  


2、在Lambda左侧导航窗格中点击【函数】后再点击【创建函数】，如下：  


3、在“创建函数”页，选择【从头开始制作】，如下：  


4、在下方“基础信息”中配置以下信息：  
函数名称：【get\_function】  
运行语言：【Python3.7】  
权限：保持默认配置  
完成后点击【创建函数】，如下：  


5、在“函数代码”模块，编写代码。图中代码是发送HTTP请求，如下：  


6、代码如下：

from botocore.vendored import requests

def lambda\_handler(event, context):

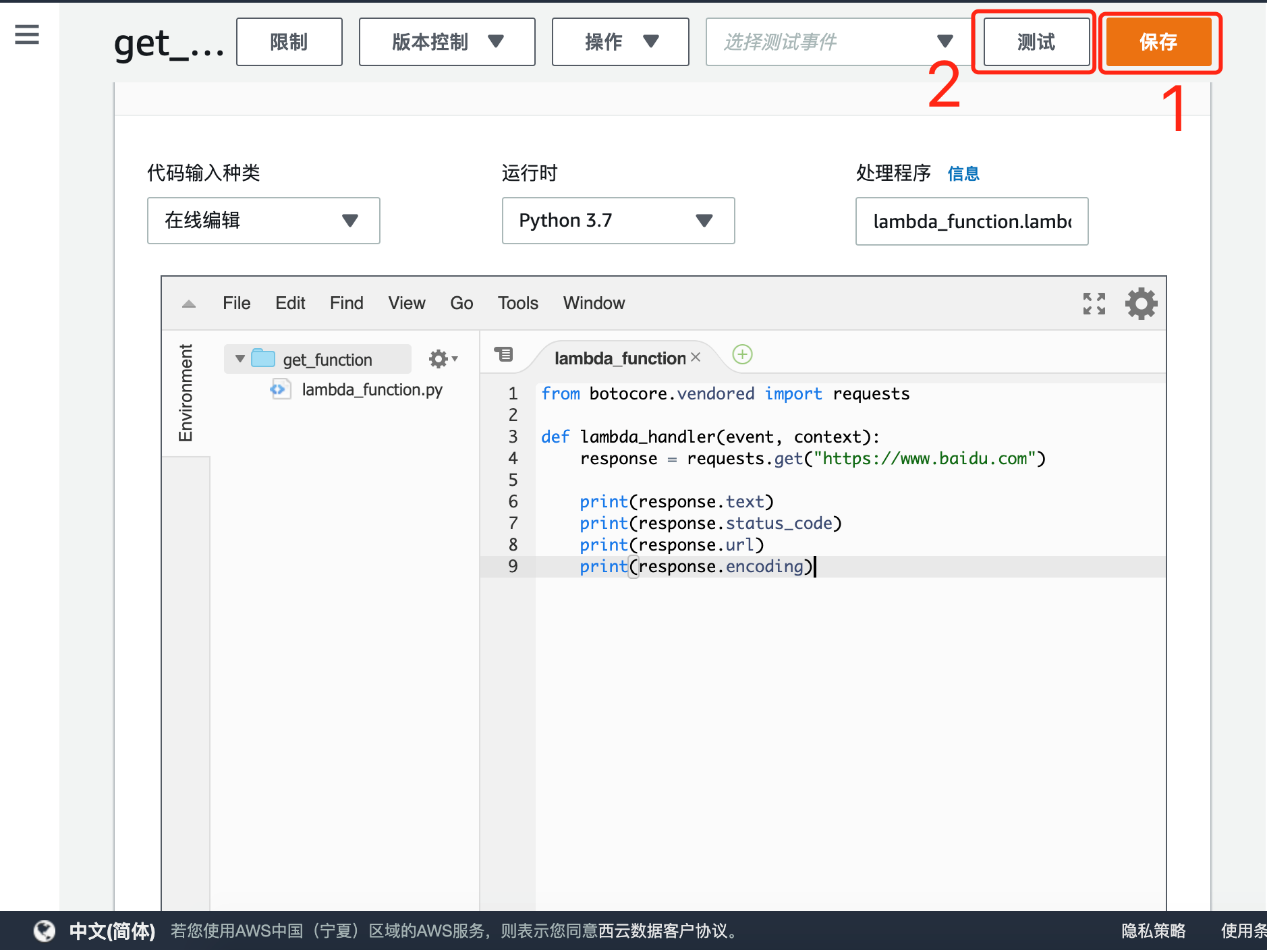
response = requests.get("https://www.baidu.com")

print(response.text)

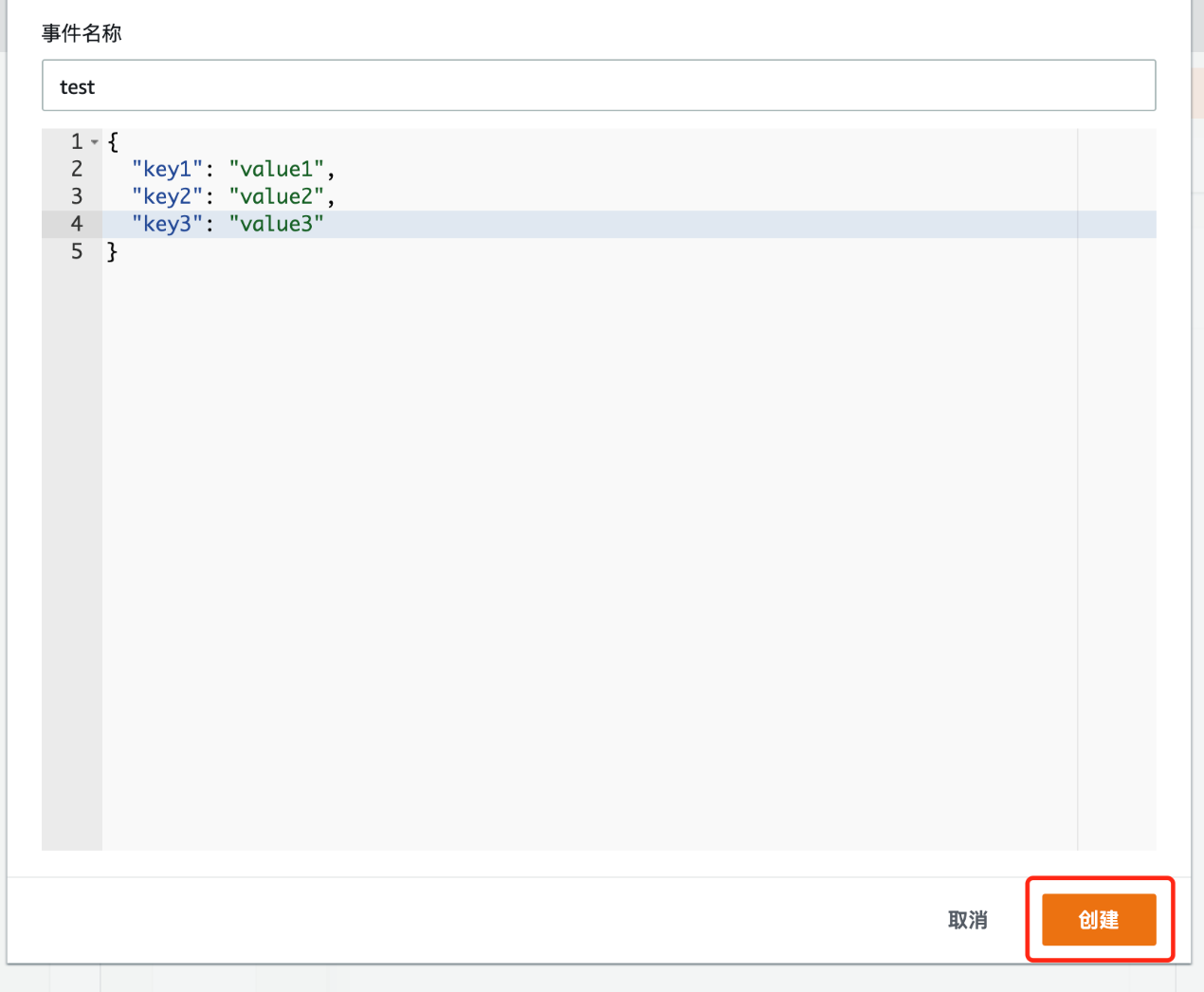
print(response.status\_code)

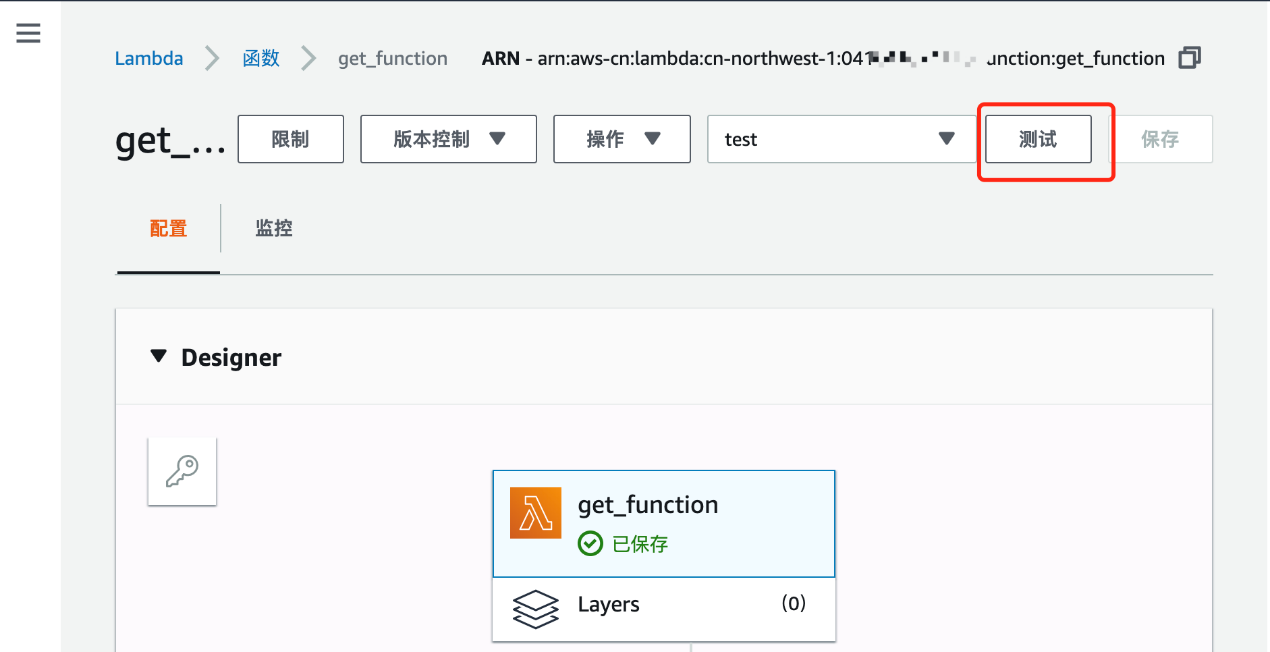
print(response.url)

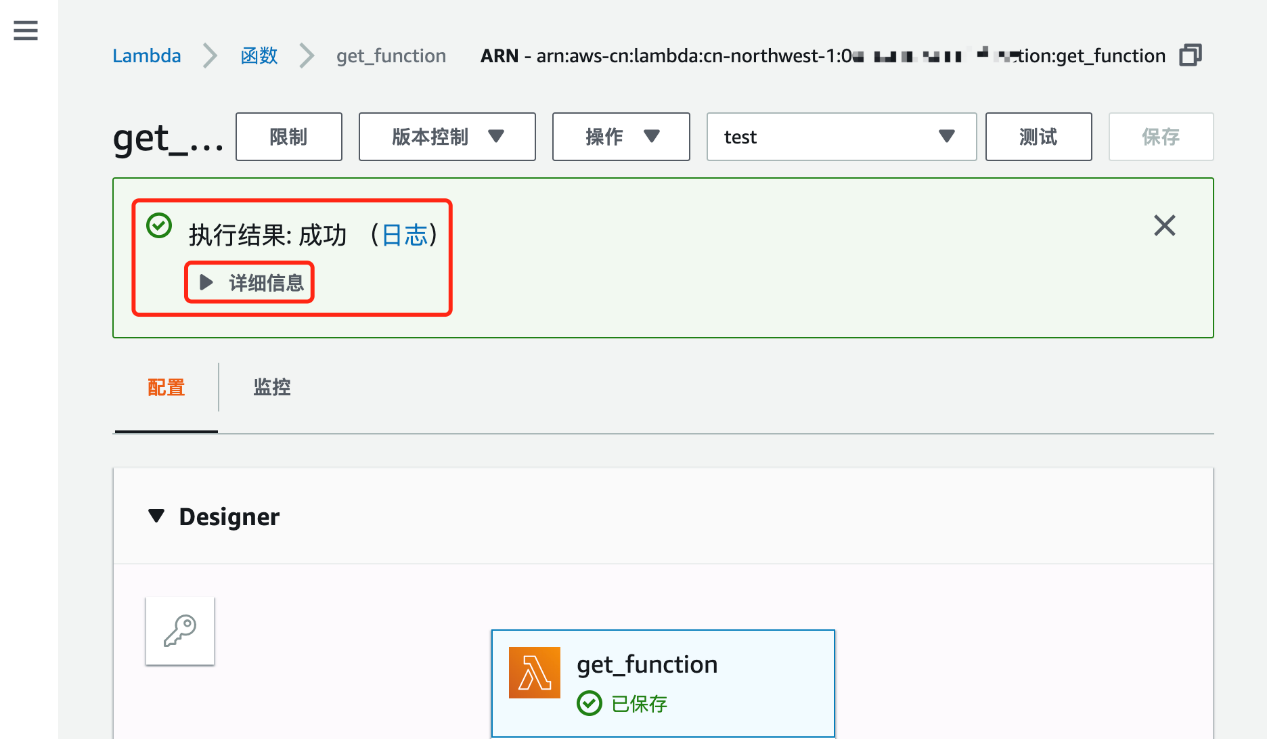
print(response.encoding)

7、编写完成代码后，点击上方菜单栏中的【保存】，然后点击【测试】，如下：  


8、在“配置测试事件”弹窗页中，配置事件的名称为【test】，如下：  


9、下拉菜单栏，点击【创建】，如下：  


10、创建完成后，点击【测试】，如下：  


11、显示执行成功，点击【详细信息】以查看详细信息，如下：  


12、在日志输出中，可以看到我们GET “https://www.baidu.com”详细信息，如下：  


**实验结果**

Lambda函数执行成功  
