掌骏传媒短信接口SDK使用说明

快速跳转对应语言

C#

Java

Python

Go

Node.js

PHP

C#

SDK版本为 **.NET Framework 4.5**

文档中使用的开发工具为 **Visual Studio 2022**

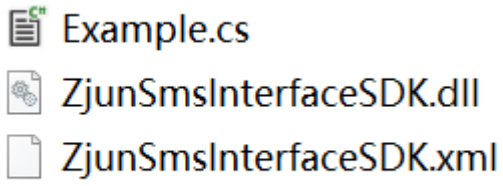
一、DLL文件引用

解压SDK的压缩包

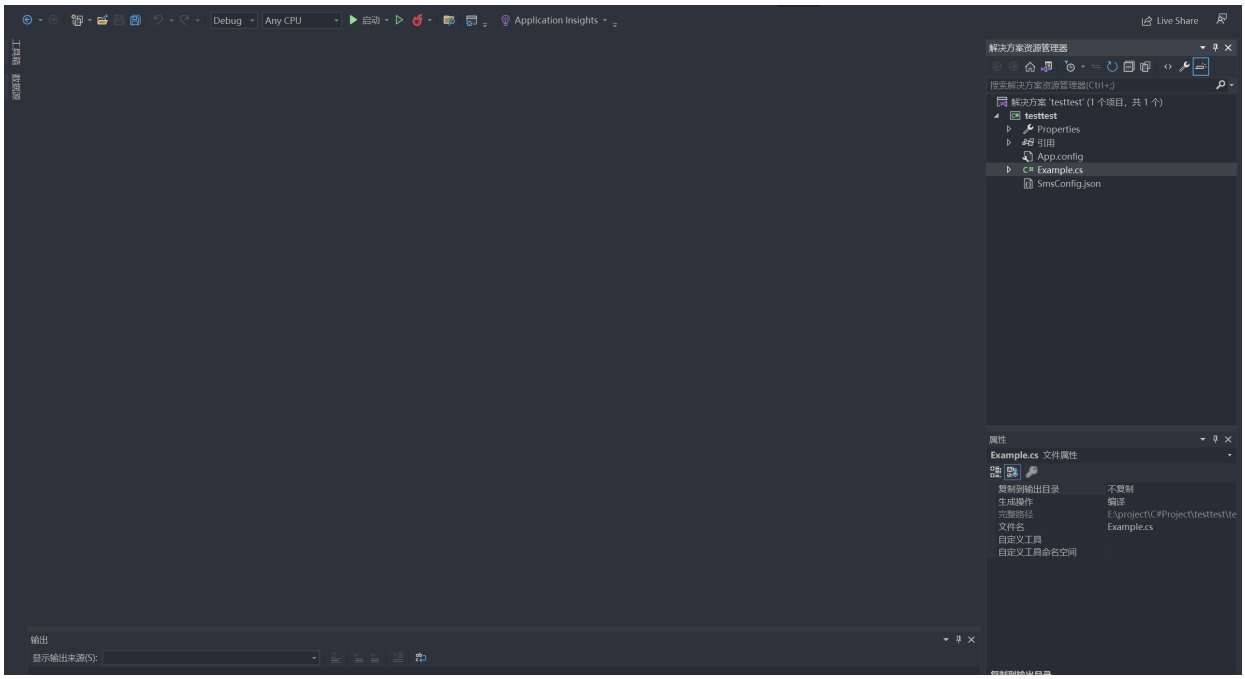
【ZjunSmsInterfaceSDK.dll】文件就是要引用的文件

【ZjunSmsInterfaceSDK.xml】是类库的注释，不需要引用

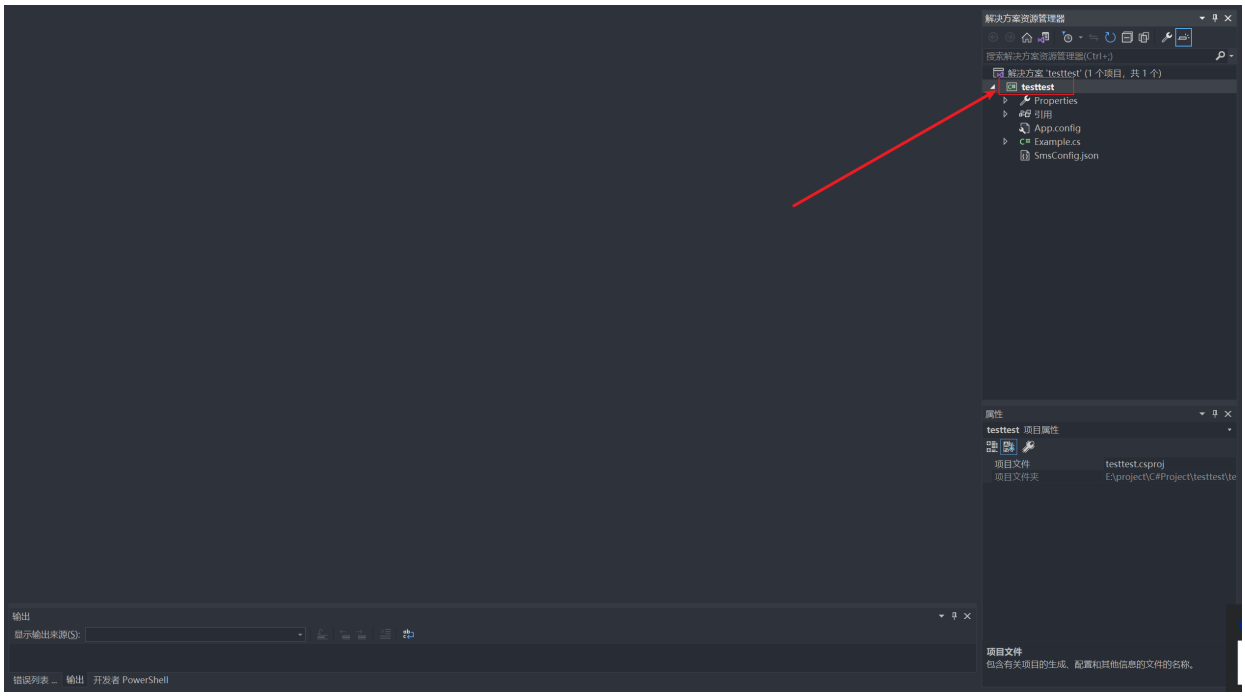
【Example.cs】是类库使用的例子



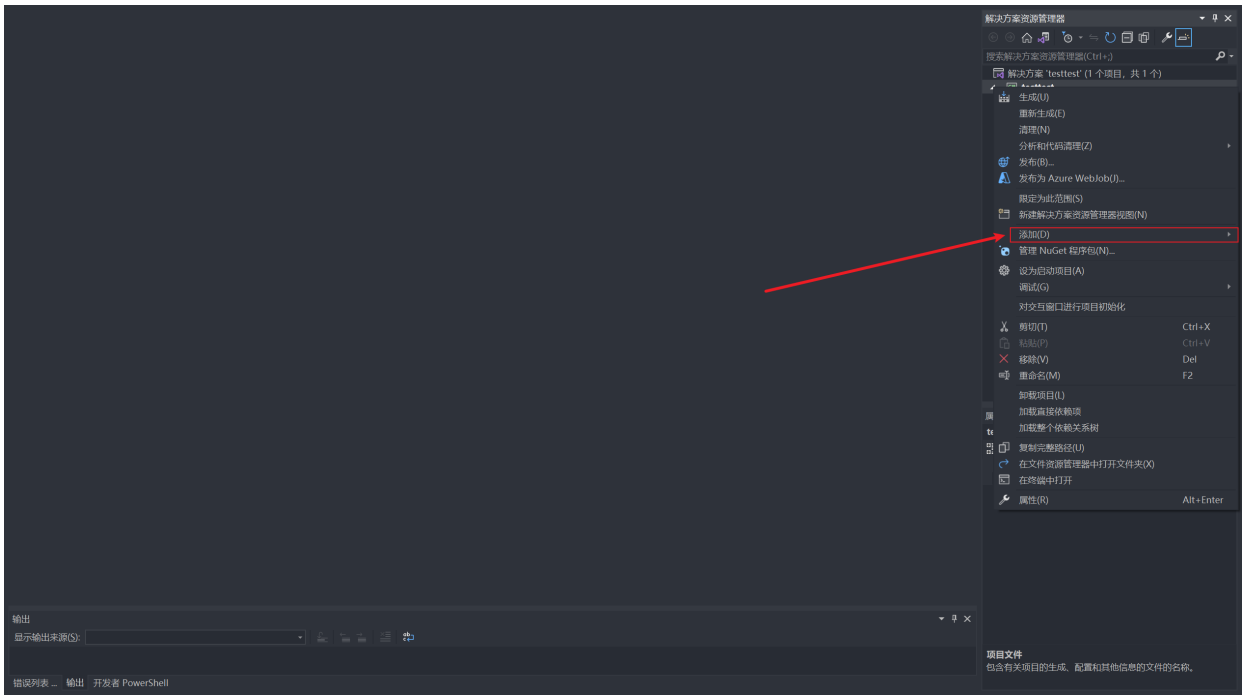
第一步：打开要引用dll文件的项目



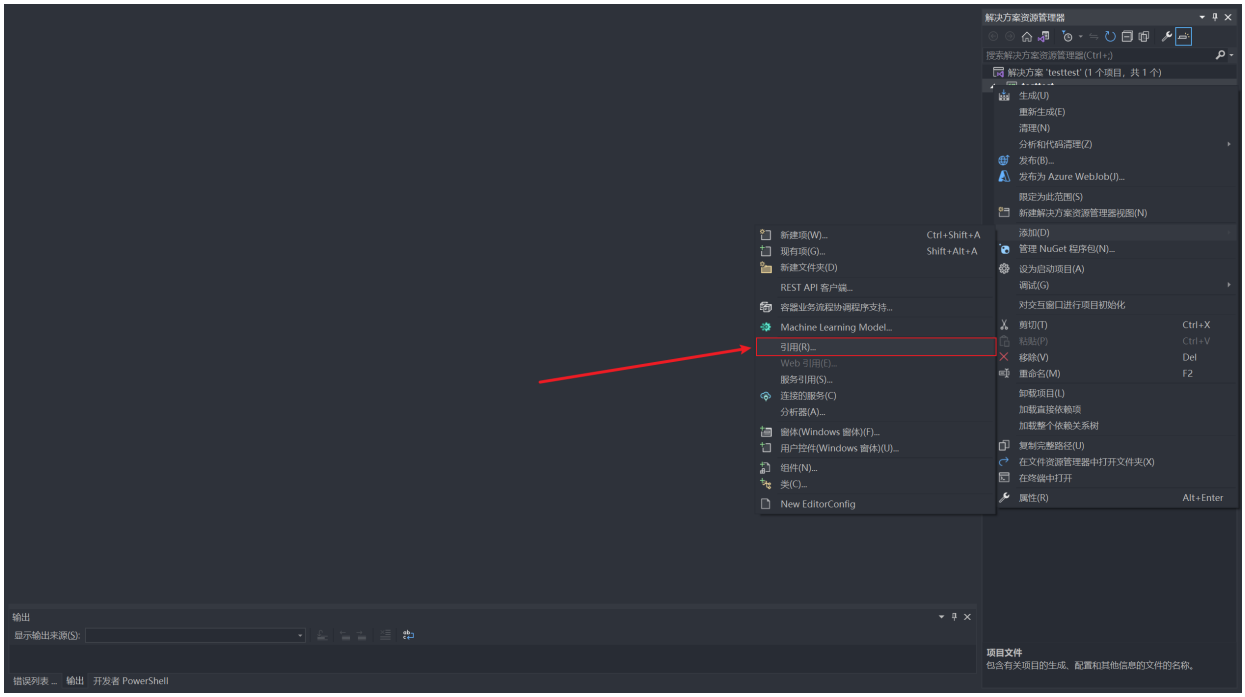
第二步：右键项目



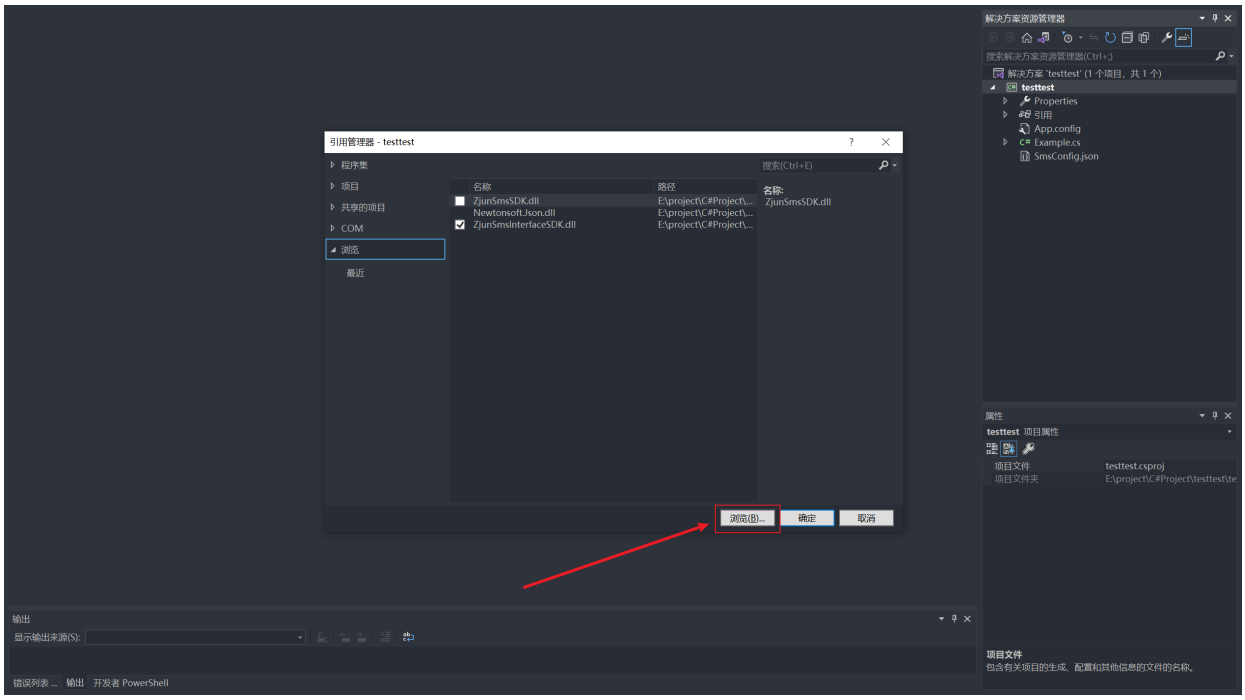
第三步：点击【添加】选项



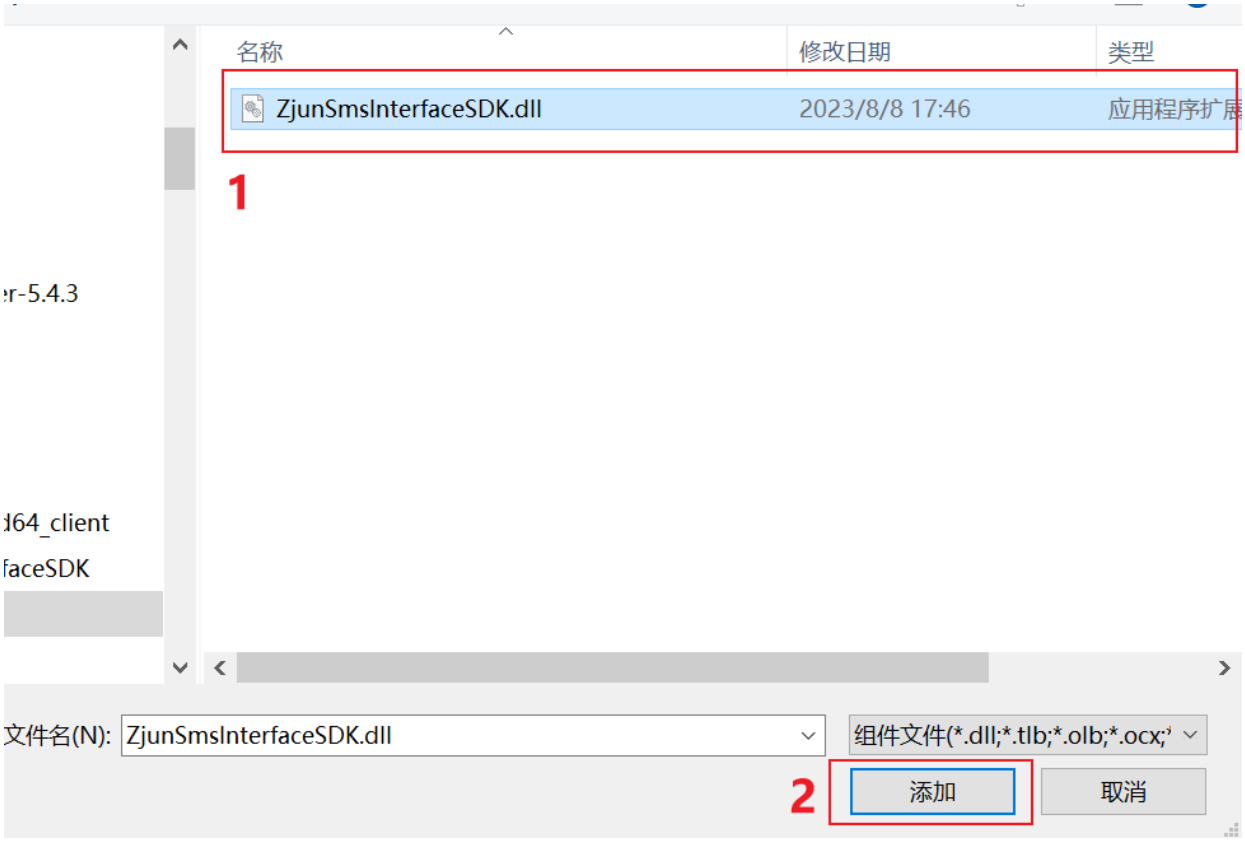
第四步：点击【引用】选项



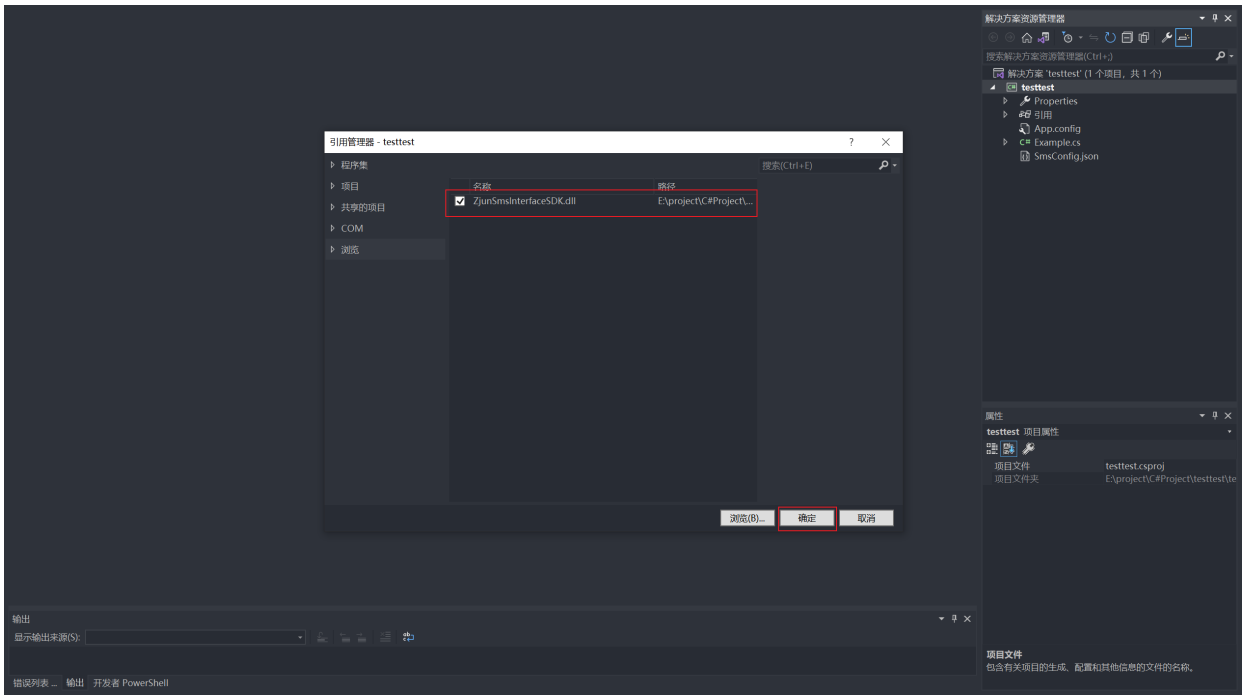
第五步：点击【浏览】按钮



第六步：选择要引用的文件，点击【添加】按钮



第七步：这样就是成功引用了，然后点击【确定】按钮



二、发送短信

第一步：引用命名空间，命名空间为【com.izjun.sms】

```
ZjunSmsInterfaceSDK    Example.Example    Main(string[] args)
1   using System;
2   using System.Collections.Generic;
3   using com.izjun.sms; // 引用命名空间
4
5   namespace Example
6   {
        0 个引用
7       internal class Example
8       {
            0 个引用
9           static void Main(string[] args)
10          {
11              // 账号
12              string account = "test";
13              // 密码
14              string passwod = "test";
15              // ip端口或域名
16              string address = "192.168.2.1:8080";
17              // 创建短信发送类，传入账号、密码、ip端口或域名
18              SmsSend smsSend = new SmsSend(account, passwod, address);
19
20              // 群发短信
21              List<string> phoneList = new List<string>();
22              phoneList.Add("15011111111");
23              phoneList.Add("15022222222");
24              string content = "【签名】验证码：190190";
25              string res = smsSend.SendSmsMass(phoneList, content);
26              // 打印返回值
27              Console.WriteLine(res);
28              Console.ReadLine();
29          }
30      }
31  }
```

第二步：创建SmsSend对象，需要传入账号、密码、ip端口或域名（账号密码会在创建账号后提供给到，ip端口或域名在短信接口说明文档中获取）,address只需要ip+端口或者是域名，前面不需要http://或https://，ip之后的资源定位符也不需要

```
using System;
using System.Collections.Generic;
using com.izjun.sms; // 引用命名空间

namespace Example
{
    0 个引用
    internal class Example
    {
        0 个引用
        static void Main(string[] args)
        {
            // 账号
            string account = "test";
            // 密码
            string passwod = "test";
            // ip端口或域名
            string address = "192.168.2.1:8080";
            // 创建短信发送类，传入账号、密码、ip端口或域名
            SmsSend smsSend = new SmsSend(account, passwod, address);

            // 群发短信
            List<string> phoneList = new List<string>();
            phoneList.Add("15011111111");
            phoneList.Add("15022222222");
            string content = "【签名】验证码: 190190";
            string res = smsSend.SendSmsMass(phoneList, content);
            // 打印返回值
            Console.WriteLine(res);
            Console.ReadLine();
        }
    }
}
```

第三步：调用方法发送短信

```
using System;
using System.Collections.Generic;
using com.izjun.sms; // 引用命名空间

namespace Example
{
    0 个引用
    internal class Example
    {
        0 个引用
        static void Main(string[] args)
        {
            // 账号
            string account = "test";
            // 密码
            string passwod = "test";
            // ip端口或域名
            string address = "192.168.2.1:8080";
            // 创建短信发送类，传入账号、密码、ip端口或域名
            SmsSend smsSend = new SmsSend(account, passwod, address);

            // 群发短信
            List<string> phoneList = new List<string>();
            phoneList.Add("15011111111");
            phoneList.Add("15022222222");
            string content = "【签名】验证码: 190190";
            string res = smsSend.SendSmsMass(phoneList, content);
            // 打印返回值
            Console.WriteLine(res);
            Console.ReadLine();
        }
    }
}
```

下面可直接复制代码

```
using System;
using System.Collections.Generic;
using com.izjun.sms; // 引用命名空间

namespace Example
```

```
6   {
7       internal class Example
8       {
9           static void Main(string[] args)
10          {
11              // 账号
12              string account = "test";
13              // 密码
14              string passwod = "test";
15              // ip端口或域名
16              string address = "192.168.2.1:8080";
17              // 创建短信发送类，传入账号、密码、ip端口或域名
18              SmsSend smsSend = new SmsSend(account, passwod, address);
19
20              // 群发短信
21              List<string> phoneList = new List<string>();
22              phoneList.Add("15011111111");
23              phoneList.Add("15022222222");
24              string content = "【签名】验证码：190190";
25              // 发送短信并获取返回值
26              string res = smsSend.SendSmsMass(phoneList, content);
27              // 打印返回值
28              Console.WriteLine(res);
29
30
31              // 一对一短信
32              // 号码与短信内容数据1
33              Dictionary<string, string>  message1 = new
    Dictionary<string, string>();
34              message1.Add("phone", "15011111111");
35              message1.Add("content", "【签名】实际发送内容");
36              // 号码与短信内容数据2
37              Dictionary<string, string> message2 = new Dictionary<string,
    string>();
38              message2.Add("phone", "15022222222");
39              message2.Add("content", "【签名】实际发送内容");
40
41              // 号码与短信内容数据集合
42              List<Dictionary<string, string>> messageList = new
    List<Dictionary<string, string>>() {};
43              messageList.Add(message1);
44              messageList.Add(message2);
45
46              // 发送短信并获取返回值
47              res = smsSend.SendSmsOne(messageList);
```

```
            // 打印返回值
            Console.WriteLine(res);

            Console.ReadLine();
        }
    }
}

```

三、方法说明

1. 短信发送

需先创建SmsSend对象，例如
SmsSend send = new SmsSend(“test”,“test”,“192.168.2.1:8080”);

（1）群发短信

send.SendSmsMass(phoneList, content)

参数名	参数类型	说明
phoneList	List	手机号码列表
content	string	短信内容

（2）群发定时短信

send.SendSmsMass(phoneList, content, sendTime)

参数名	参数类型	说明
phoneList	List	手机号码列表
content	string	短信内容
sendTime	string	定时时间

（3）群发短信，携带其他参数

send.SendSmsMass(phoneList, content, extcode, callData)

参数名	参数类型	说明
phoneList	List	手机号码列表
content	string	短信内容
extcode	string	通道扩展码
callData	string	用户回传数据，用户若传递此参数将在回执推送时回传给用户。

（4）群发短信，完整参数

send.SendSmsMass(phoneList, content, sendTime, extcode, callData)

参数名	参数类型	说明
phoneList	List	手机号码列表
content	string	短信内容
sendTime	string	定时时间
extcode	string	通道扩展码
callData	string	用户回传数据，用户若传递此参数将在回执推送时回传给用户。

（5）一对一短信发送

send.SendSmsOne(messageList)

参数名	参数类型	说明
messageList	List<Dictionary<string, string>>	手机号码与短信内容等列表

messageList由多个字典对象构成的数组，具体参数列表：

参数名	参数类型	说明
phone	string	手机号码
content	string	短信内容
extcode	string	可选，通道扩展号
callData	string	可选，用户回传数据，用户若传递此参数将在回执推送时回传给用户。

（6）一对一短信定时发送

send.SendSmsOne(messageList， sendTime)

参数名	参数类型	说明
messageList	List<Dictionary<string, string>>	手机号码与短信内容等列表
sendTime	string	定时时间

2. 获取状态、回复、余额

（1）获取状态

send.GetReport()

（2）获取回复

send.GetUpstream()

（3）获取余额

send.GetBalance()

3. 短信模板

（1）提交短信模板

send.CreateTemplate(content)

参数名	参数类型	说明
content	string	模板内容

（2）查询短信模板

send.QueryTemplate()

> 该文档只提供SDK使用说明，有关接口的详细说明请查阅接口文档

Java

SDK使用JDK版本为 **JDK 1.8**

文档中使用的开发工具为 **IntelliJ IDEA**

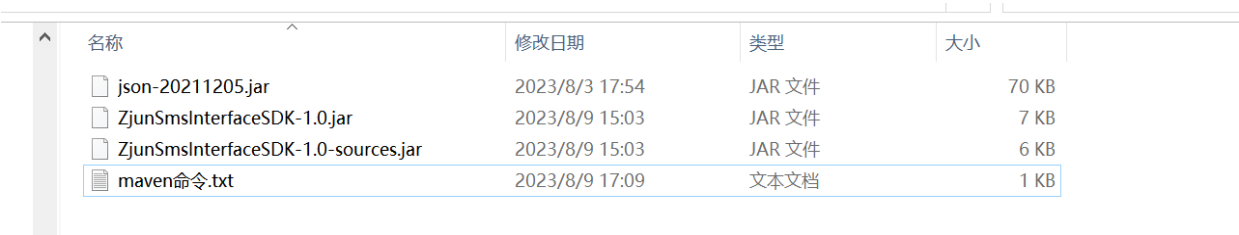
一、jar包安装

【ZjunSmsInterfaceSDK-1.0.jar】文件就是要按照的jar包

【ZjunSmsInterfaceSDK-1.0-sources.jar】是源码jar包

【json-20211205.jar】是JSON转换要使用的jar包，使用maven安装方式直接引入依赖就可以了，不需要使用这个jar包

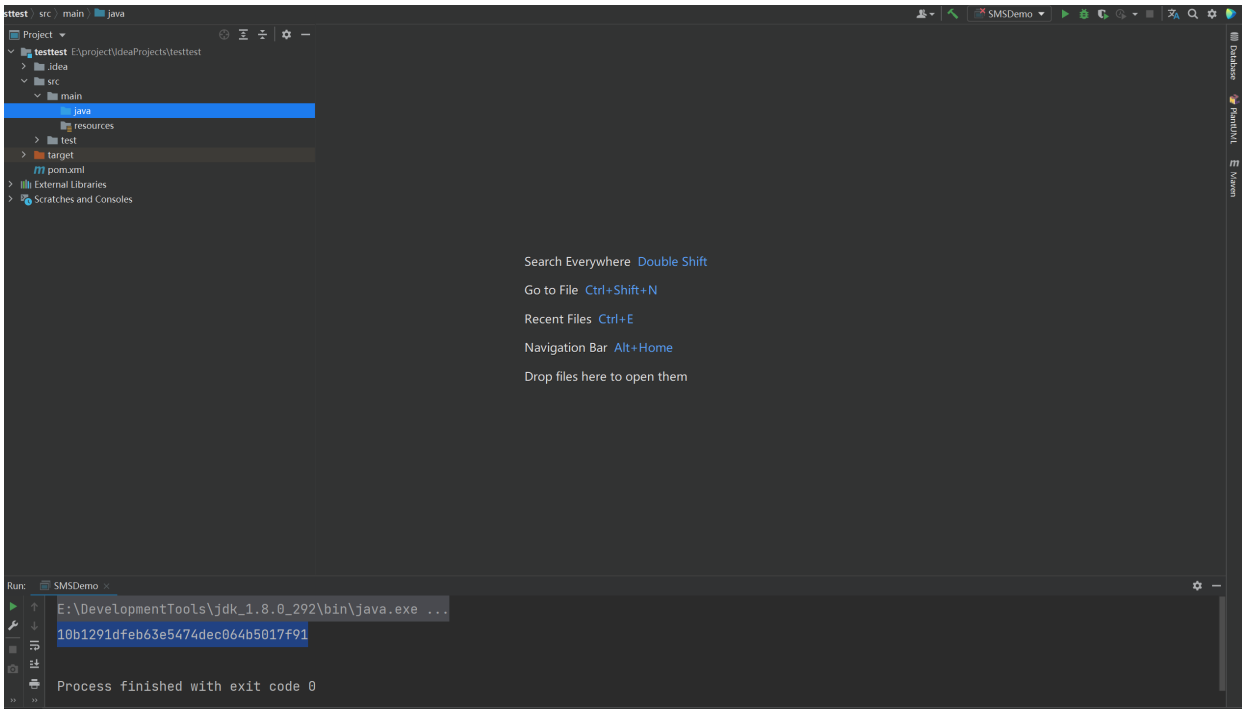
【maven命令.txt】中保存maven安装jar包时需执行的命令



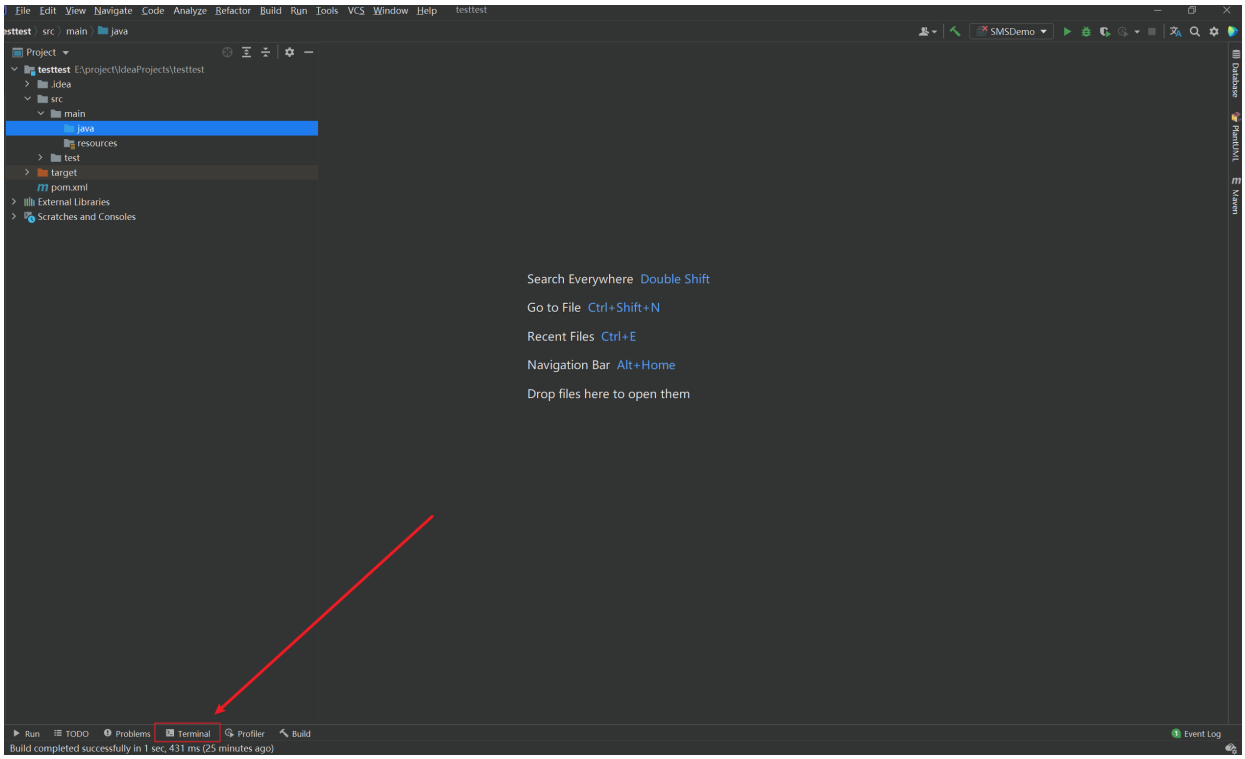
名称	修改日期	类型	大小
json-20211205.jar	2023/8/3 17:54	JAR 文件	70 KB
ZjunSmsInterfaceSDK-1.0.jar	2023/8/9 15:03	JAR 文件	7 KB
ZjunSmsInterfaceSDK-1.0-sources.jar	2023/8/9 15:03	JAR 文件	6 KB
maven命令.txt	2023/8/9 17:09	文本文档	1 KB

第一种方式：使用maven安装

第一步：打开要安装jar包的项目



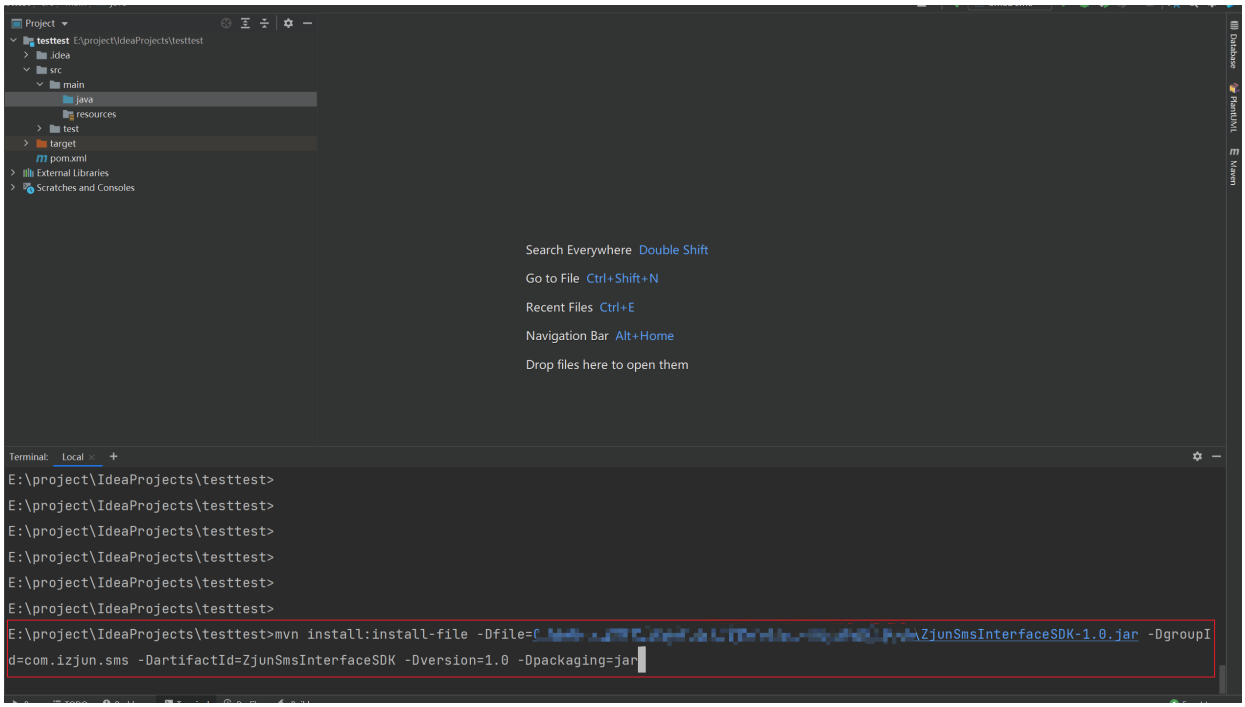
第二步：点击控制台

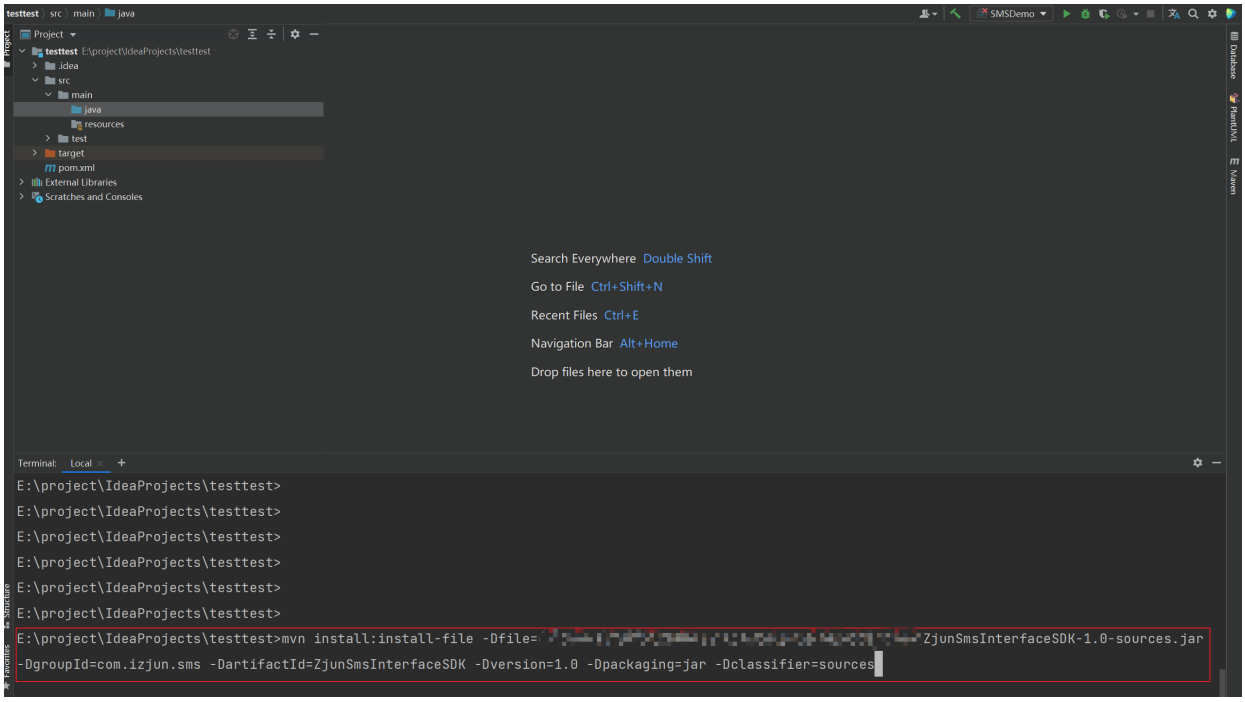


第三步：执行mvn命令，mvn命令在【maven命令.txt】文件中，如果执行时提示mvn命令无法找到，需要将maven的bin包的路径添加到系统环境变量中

将命令中的<jar包路径>与<源码jar包路径>换成实际的路径

然后在控制台中执行

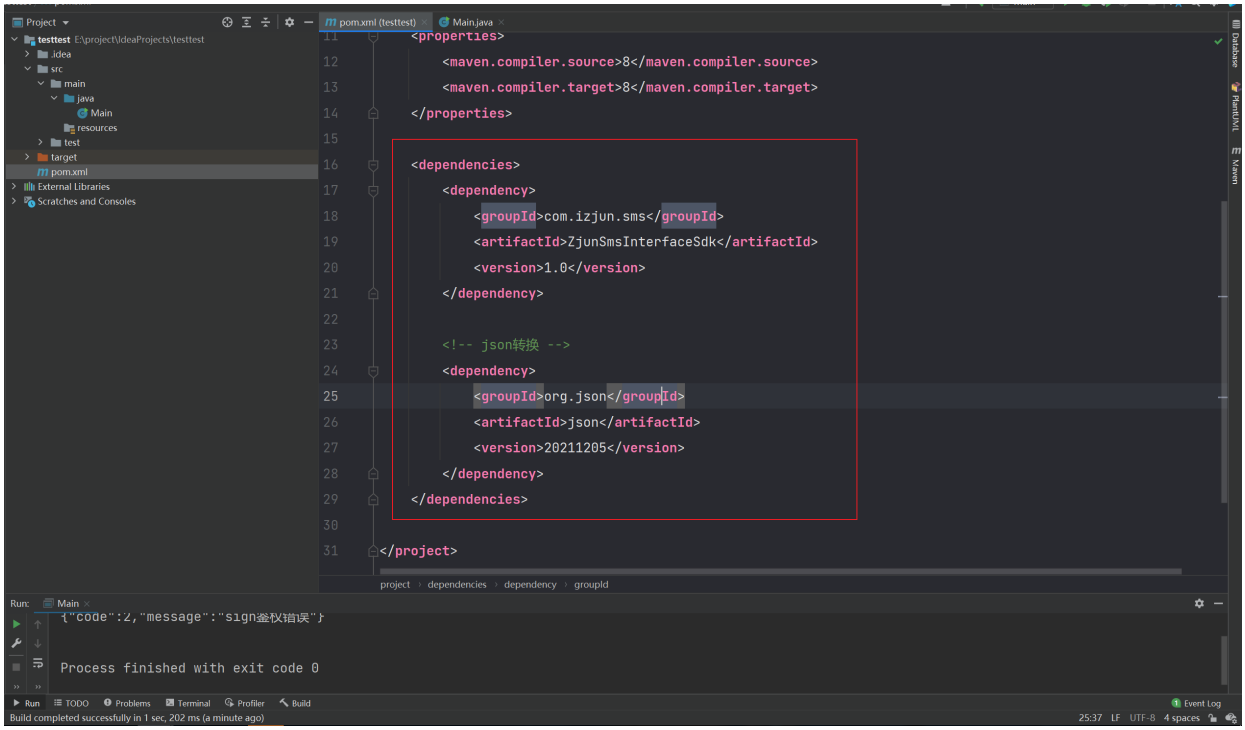




第四步：在pom.xml文件中添加依赖

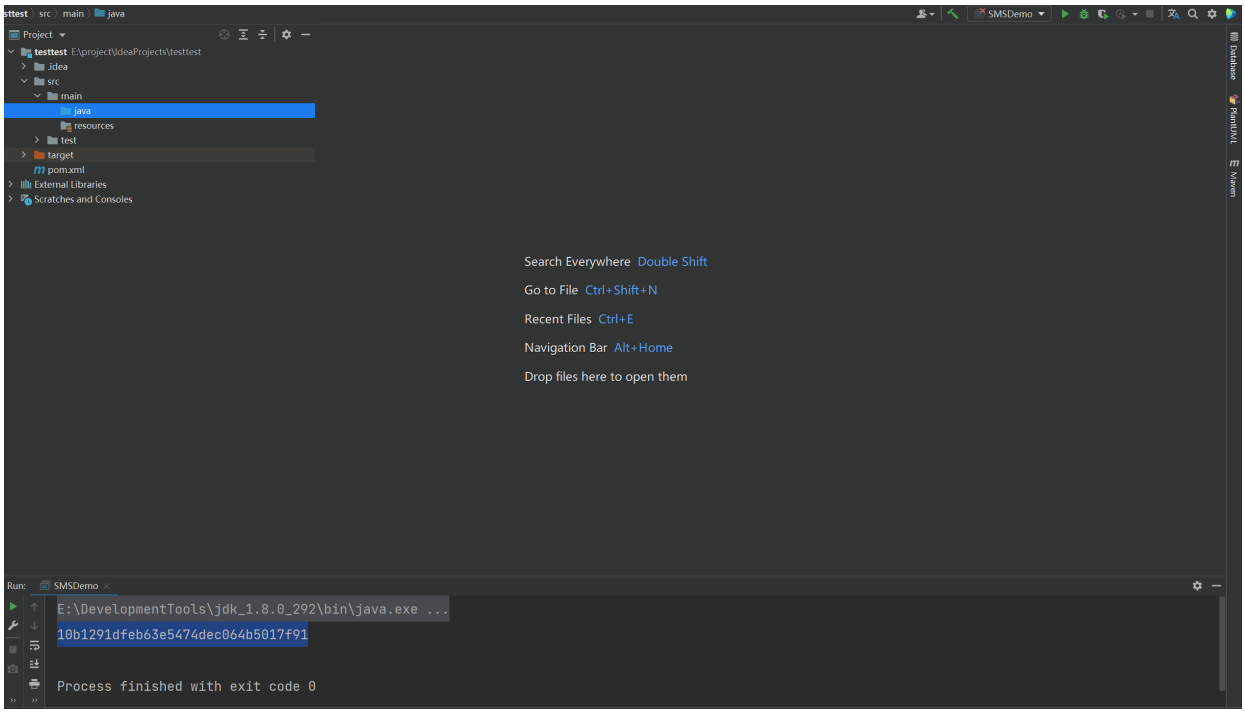
```
<dependencies>
    <dependency>
        <groupId>com.izjun.sms</groupId>
        <artifactId>ZjunSmsInterfaceSdk</artifactId>
        <version>1.0</version>
    </dependency>

    <!—— json转换 ——>
    <dependency>
        <groupId>org.json</groupId>
```

```
        <artifactId>json</artifactId>
        <version>20211205</version>
    </dependency>
</dependencies>
```

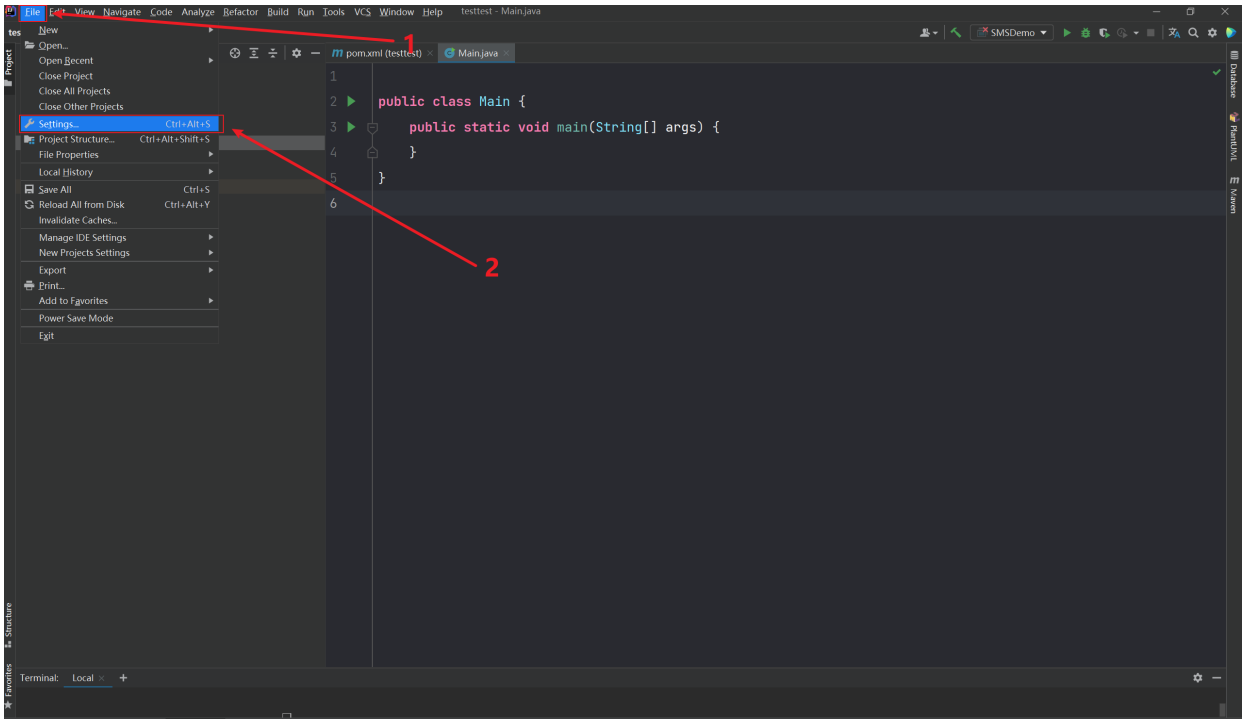


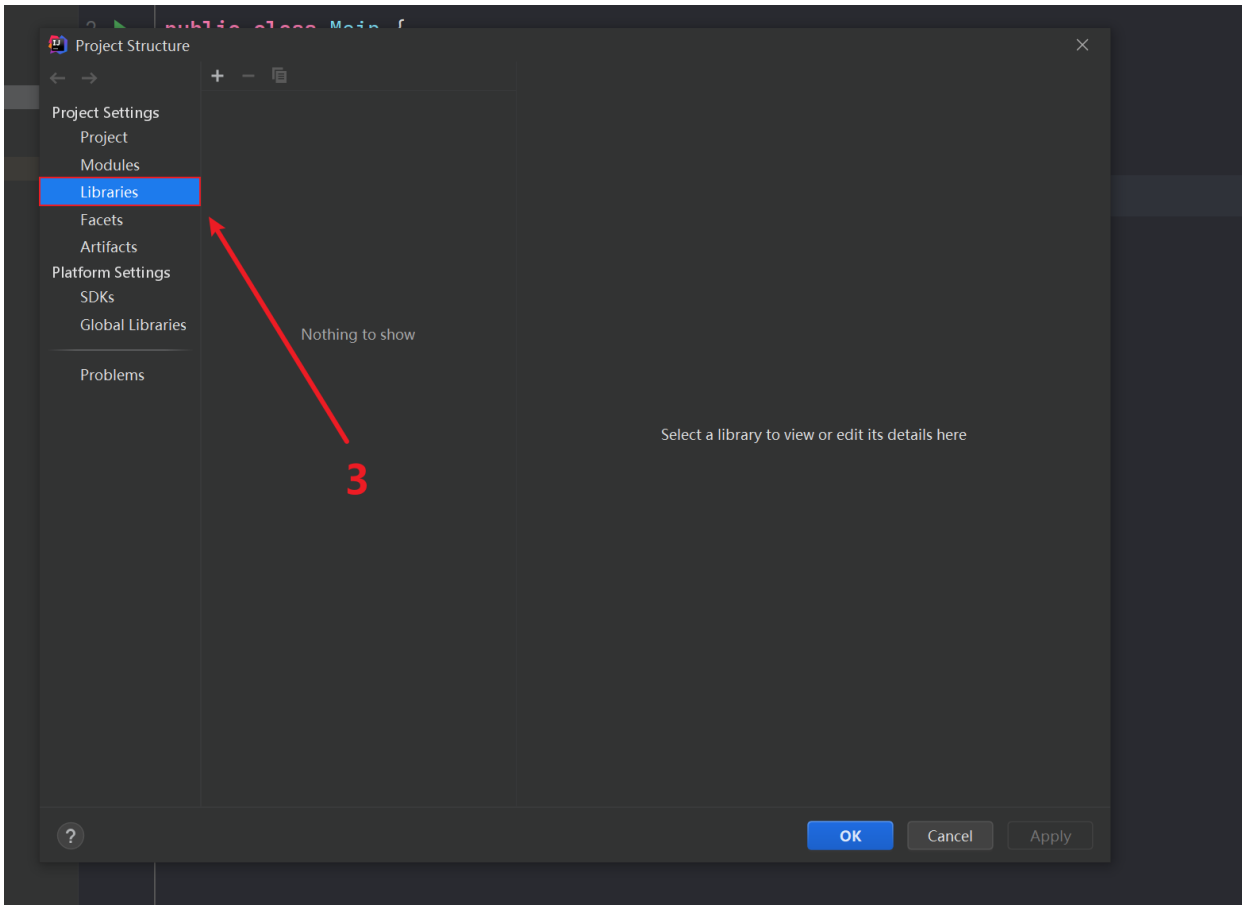
第二种方式：直接导入jar包

第一步：打开要安装jar包的项目

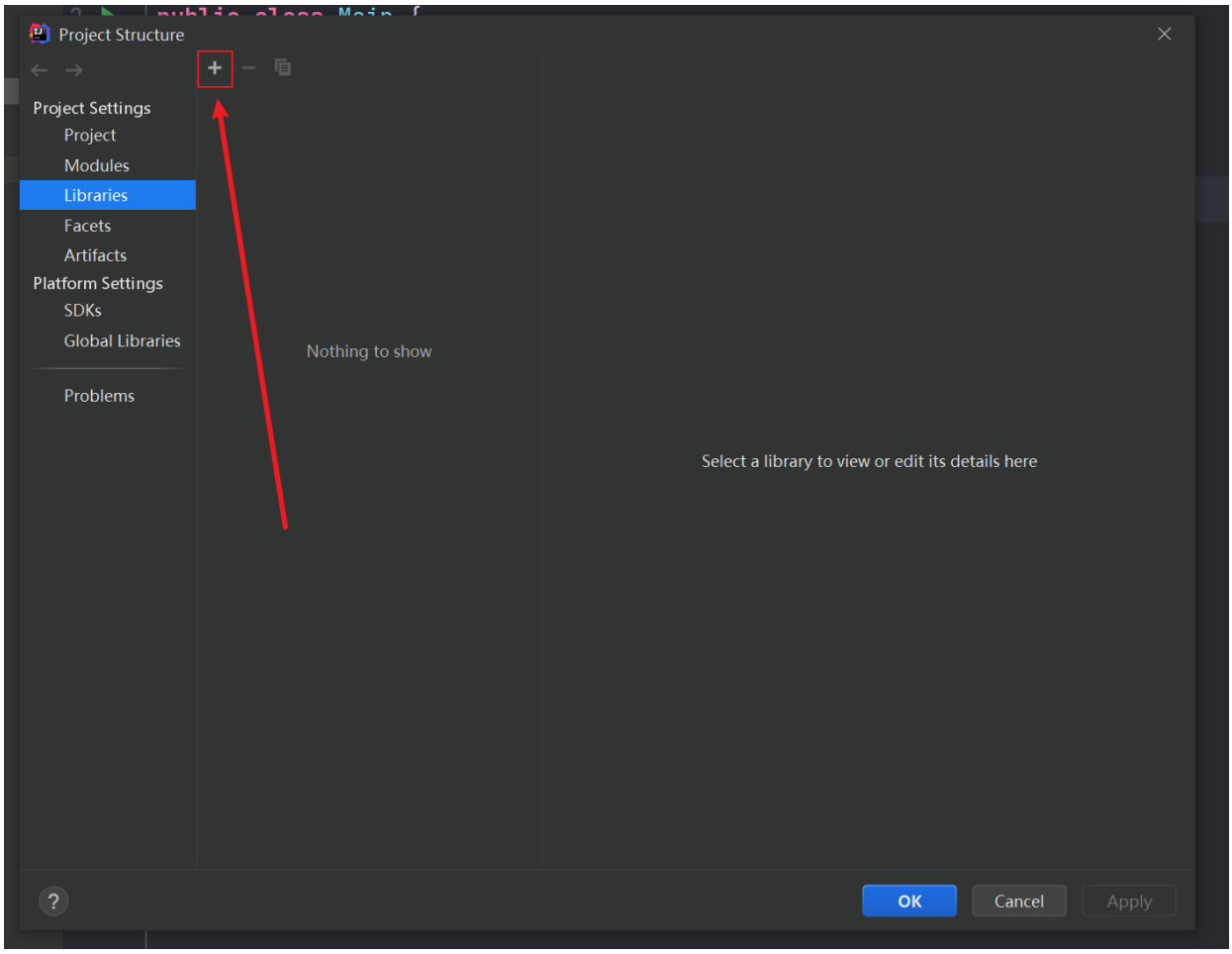


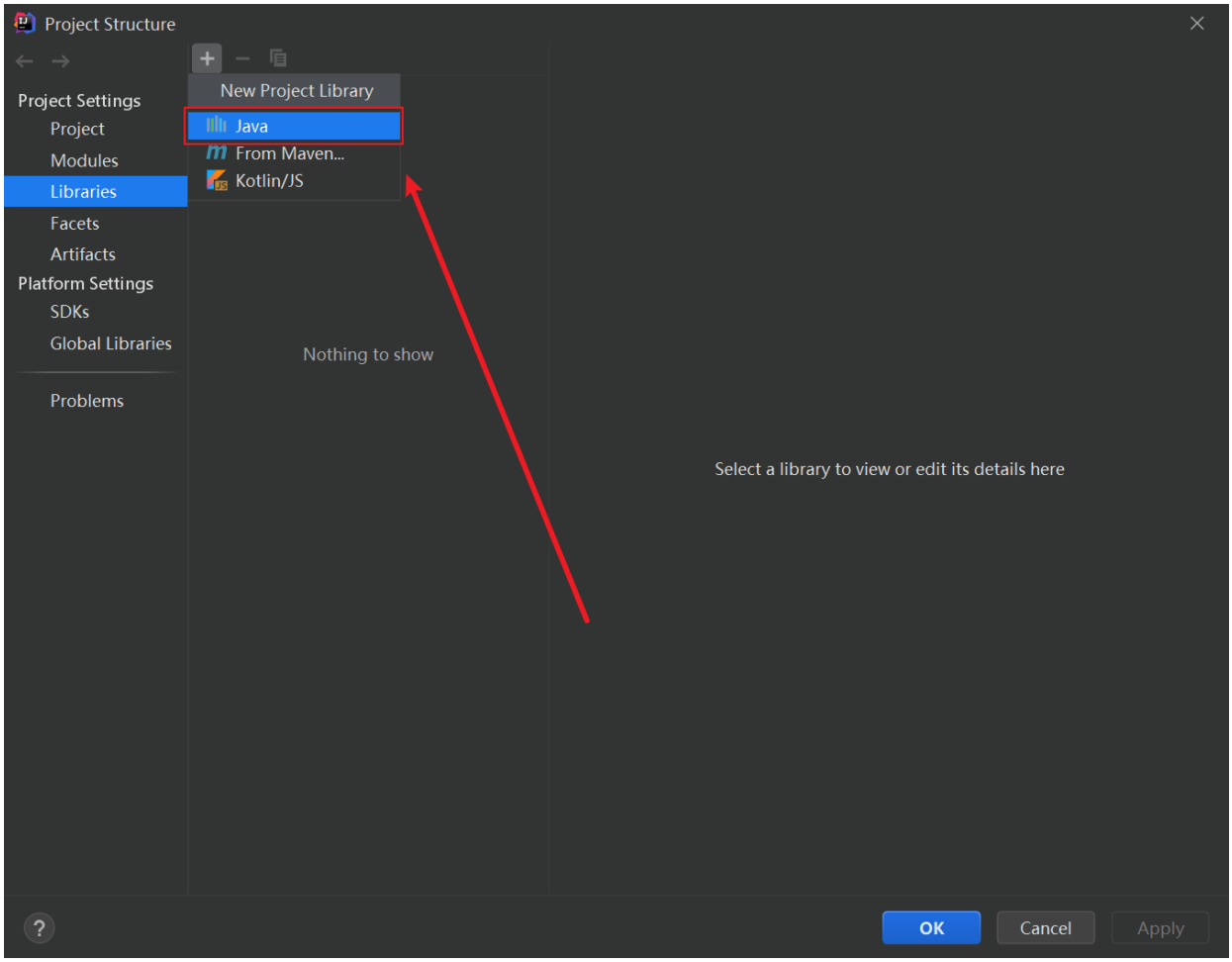
第二步：点击 File>Project Structure>Libraries


File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
testtest - Main.java
New
Open...
Open Recent
Close Project
Close All Projects
Close Other Projects
Settings... Ctrl+Alt+S
Project Structure... Ctrl+Alt+Shift+S
File Properties
Local History
Save All Ctrl+S
Reload All from Disk Ctrl+Alt+Y
Invalidate Caches...
Manage IDE Settings
New Projects Settings
Export
Print...
Add to Favorites
Power Save Mode
Exit
pom.xml (testtest)
Main.java
SMSDemo
public class Main {
public static void main(String[] args) {
}
}
1
2
Terminal: Local

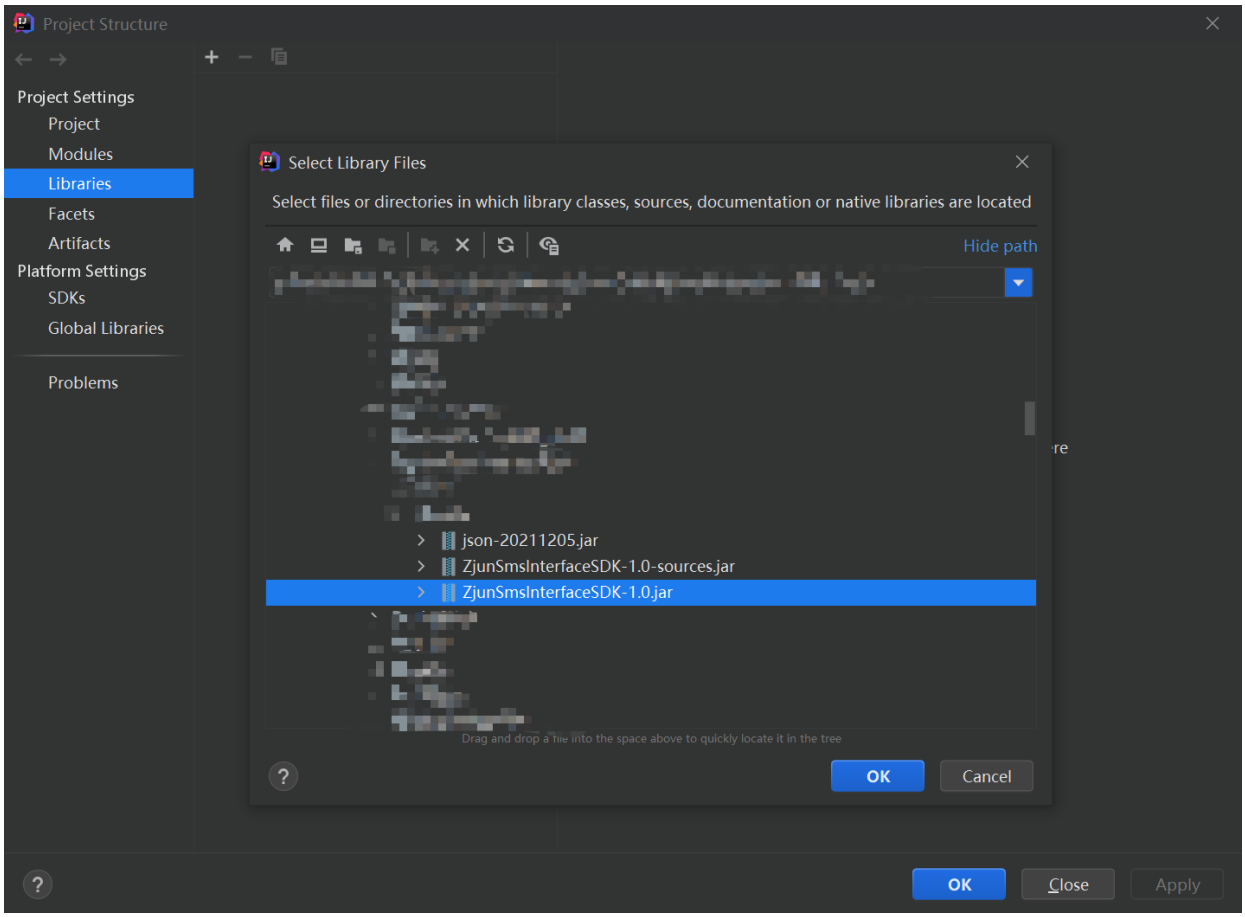

Project Structure
Project Settings
Project
Modules
Libraries
Facets
Artifacts
Platform Settings
SDKs
Global Libraries
Problems
Nothing to show
Select a library to view or edit its details here
3
OK
Cancel
Apply

第三步：点击 +号>Java

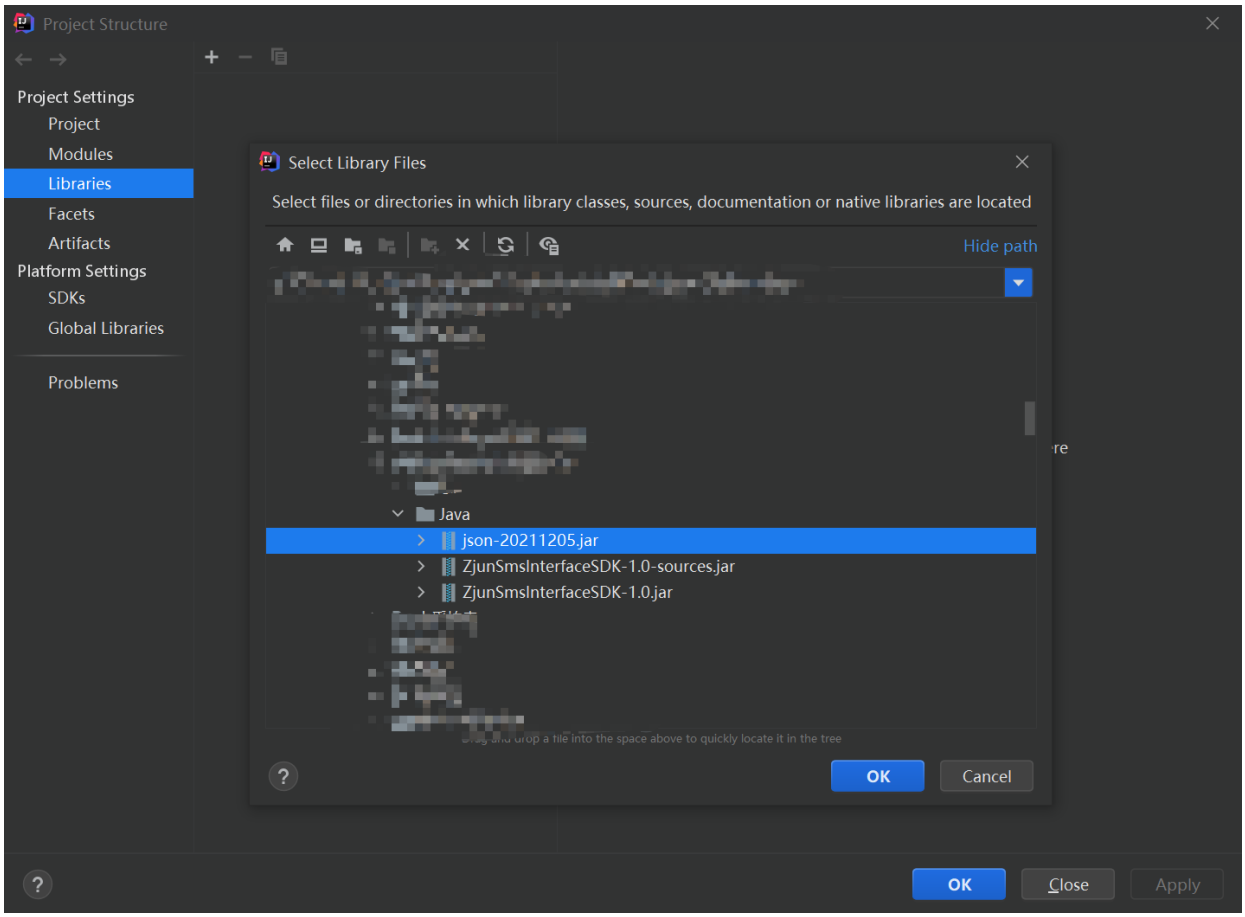

Project Structure
Project Settings
Project
Modules
Libraries
Facets
Artifacts
Platform Settings
SDKs
Global Libraries
Problems
Nothing to show
Select a library to view or edit its details here
OK
Cancel
Apply


Project Structure
Project Settings
Project
Modules
Libraries
Facets
Artifacts
Platform Settings
SDKs
Global Libraries
Problems
New Project Library
Java
From Maven...
Kotlin/JS
Nothing to show
Select a library to view or edit its details here
OK
Cancel
Apply

第四步:选择ZjunSmsInterfaceSDK.jar包,然后点击OK>OK,然后点击Apply之后再点击OK

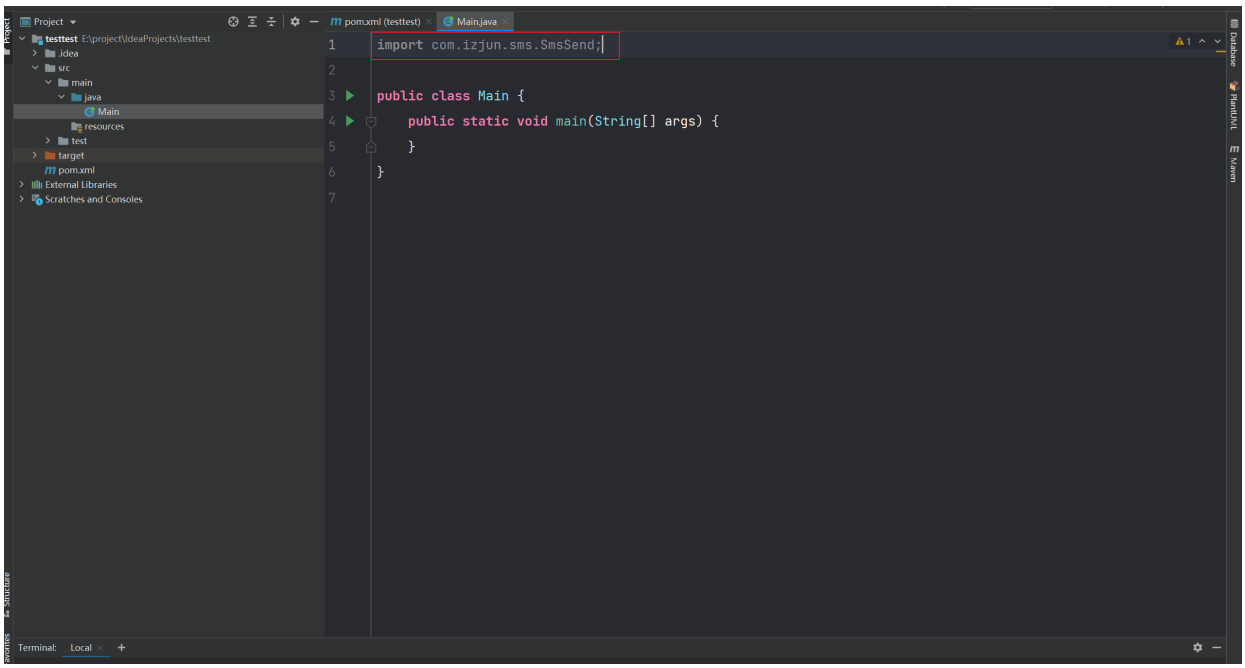


第五步:选择json-20211205.jar包,然后点击OK>OK,然后点击Apply之后再点击OK

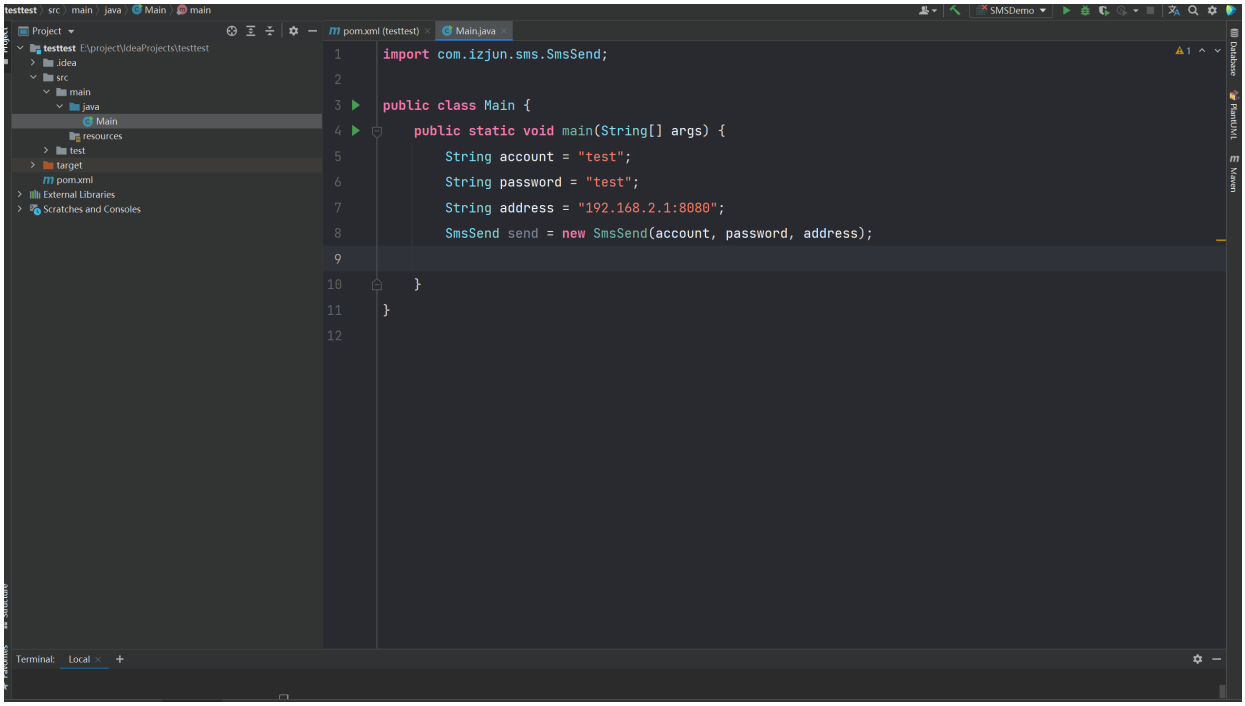


二、发送短信

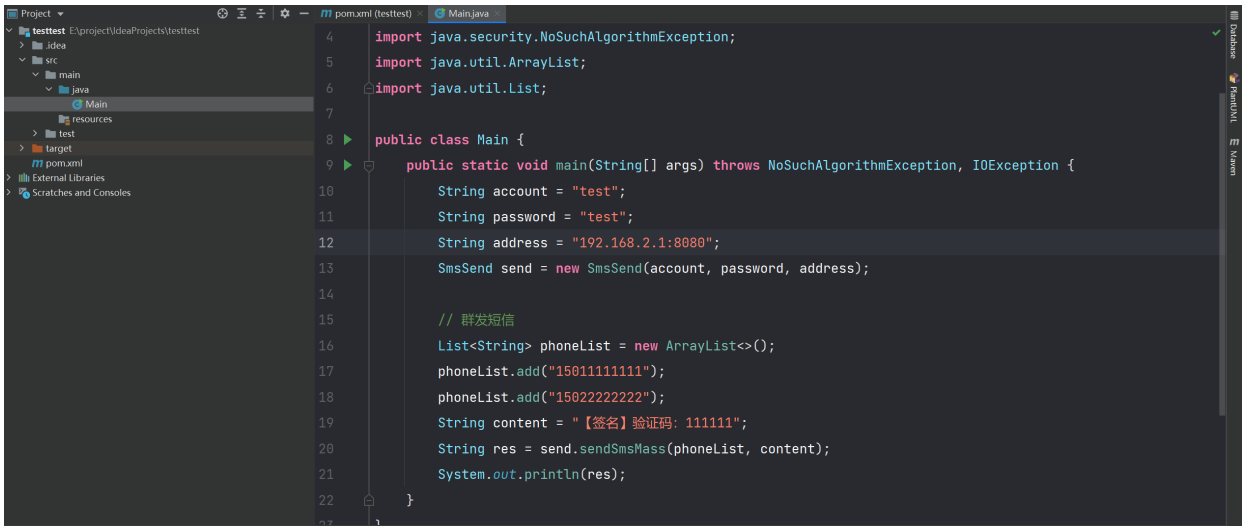
第一步：导入依赖



第二步：创建SmsSend对象，需要传入账号、密码、ip端口或域名（账号密码会在创建账号后提供给到，ip端口或域名在短信接口说明文档中获取）,address只需要ip+端口或者是域名，前面不需要http://或https://，ip之后的资源定位符也不需要



第三步：调用方法发送短信



下面可直接复制代码

```
import com.izjun.sms.SmsSend;

import java.io.IOException;
import java.security.NoSuchAlgorithmException;
import java.util.ArrayList;
import java.util.List;

public class Main {
```

```java
    public static void main(String[] args) throws NoSuchAlgorithmException, IOException {
        String account = "test";
        String password = "test";
        String address = "192.168.2.1:8080";
        SmsSend send = new SmsSend(account, password, address);

        // 群发短信
        List<String> phoneList = new ArrayList<>();
        phoneList.add("15011111111");
        phoneList.add("15022222222");
        String content = "【签名】验证码：111111";
        String res = send.sendSmsMass(phoneList, content);
        System.out.println(res);

        // 一对一短信
        // 手机号与内容1
        Map<String, Object> message1 = new HashMap<>();
        message1.put("phone", "15011111111");
        message1.put("content", "【签名】实际发送内容");
        // 手机号与内容2
        Map<String, Object> message2 = new HashMap<>();
        message1.put("phone", "15022222222");
        message1.put("content", "【签名】实际发送内容");

        // messageList参数
        List<Map<String, Object>> messageList = new ArrayList<>();
        messageList.add(message1);
        messageList.add(message2);

        res = send.sendSmsOne(messageList);
        System.out.println(res);
    }
}

```

三、方法说明

1. 短信发送

需先创建SmsSend对象，例如
SmsSend send = new SmsSend(“test” , “test” , “192.168.2.1:8080”);

（1）群发短信

send.sendSmsMass(phoneList, content)

参数名	参数类型	说明
phoneList	List	手机号码列表
content	String	短信内容

（2）群发定时短信

send.sendSmsMass(phoneList, content, sendTime)

参数名	参数类型	说明
phoneList	List	手机号码列表
content	String	短信内容
sendTime	String	定时时间

（3）群发短信，携带其他参数

send.sendSmsMass(phoneList, content, extcode, callData)

参数名	参数类型	说明
phoneList	List	手机号码列表
content	String	短信内容
extcode	String	通道扩展码
callData	String	用户回传数据，用户若传递此参数将在回执推送时回传给用户。

（4）群发短信，完整参数

send.sendSmsMass(phoneList, content, sendTime, extcode, callData)

参数名	参数类型	说明
phoneList	List	手机号码列表
content	String	短信内容
sendTime	String	定时时间
extcode	String	通道扩展码
callData	String	用户回传数据，用户若传递此参数将在回执推送时回传给用户。

（5）一对一短信发送

send.sendSmsOne(messageList)

参数名	参数类型	说明
messageList	List<Map<String, String>>	手机号码与短信内容等列表

messageList由多个字典对象构成的数组，具体参数列表：

参数名	参数类型	说明
phone	String	手机号码
content	String	短信内容
extcode	String	可选，通道扩展号
callData	String	可选，用户回传数据，用户若传递此参数将在回执推送时回传给用户。

（6）一对一短信定时发送

send.sendSmsOne(messageList， sendTime)

参数名	参数类型	说明
messageList	List<Map<String, String>>	手机号码与短信内容等列表
sendTime	String	定时时间

2. 获取状态、回复、余额

（1）获取状态

send.getReport()

（2）获取回复

send.getUpstream()

（3）获取余额

send.getBalance()

3. 短信模板

（1）提交短信模板

send.createTemplate(content)

参数名	参数类型	说明
content	String	模板内容

（2）查询短信模板

send.queryTemplate()

> 该文档只提供SDK使用说明，有关接口的详细说明请查阅接口文档

Python

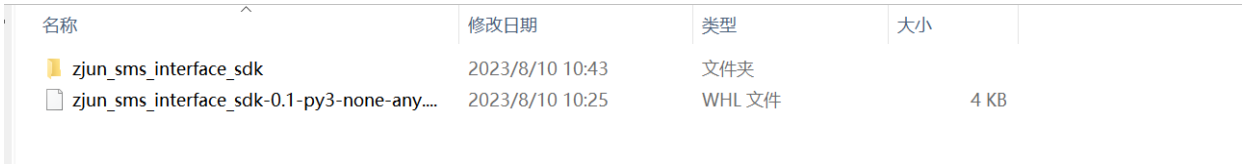
SDK使用python版本为 **python3**

文档中使用的开发工具为 **pycharm 2022**

一、类库安装

【zjun_sms_interface_sdk】文件夹里包含的是源代码，如果无法使用pip安装或者安装失败的可以使用源代码

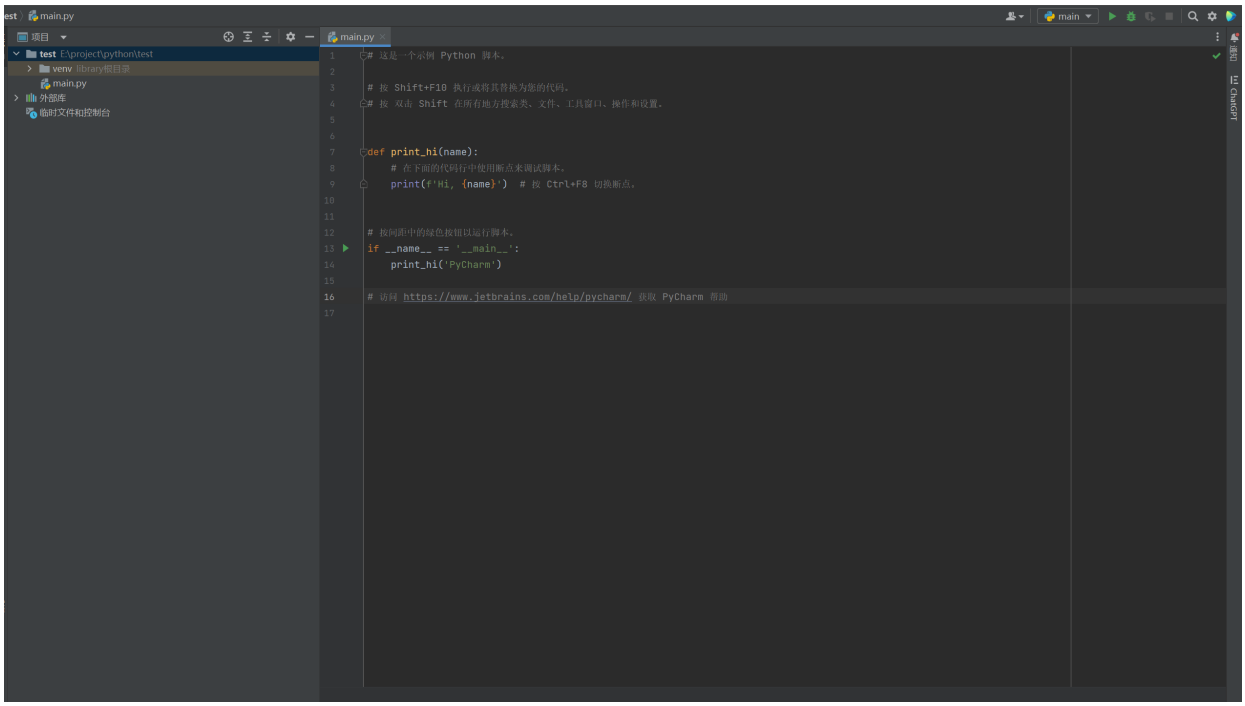
【zjun_sms_interface_sdk-0.1-py3-none-any.whl】是库安装文件



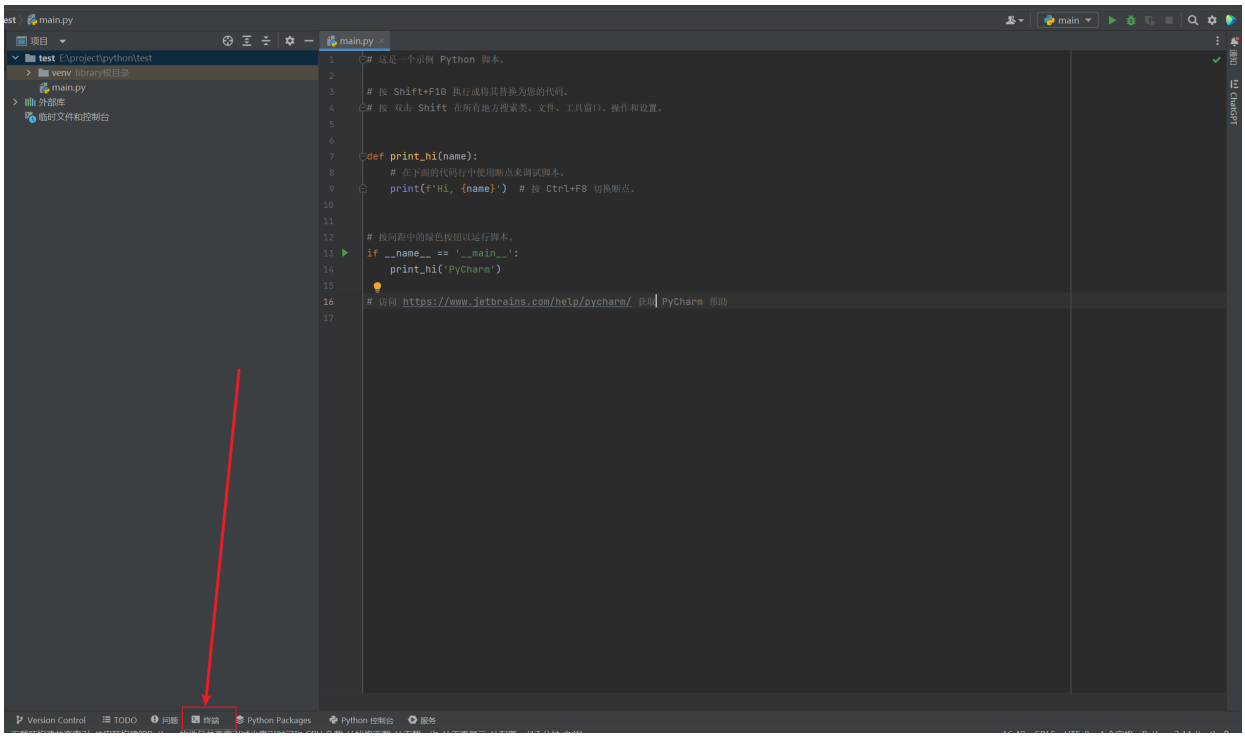
名称	修改日期	类型	大小
zjun_sms_interface_sdk	2023/8/10 10:43	文件夹	
zjun_sms_interface_sdk-0.1-py3-none-any....	2023/8/10 10:25	WHL 文件	4 KB

第一种方式：使用pip安装

第一步：打开要安装类库的项目



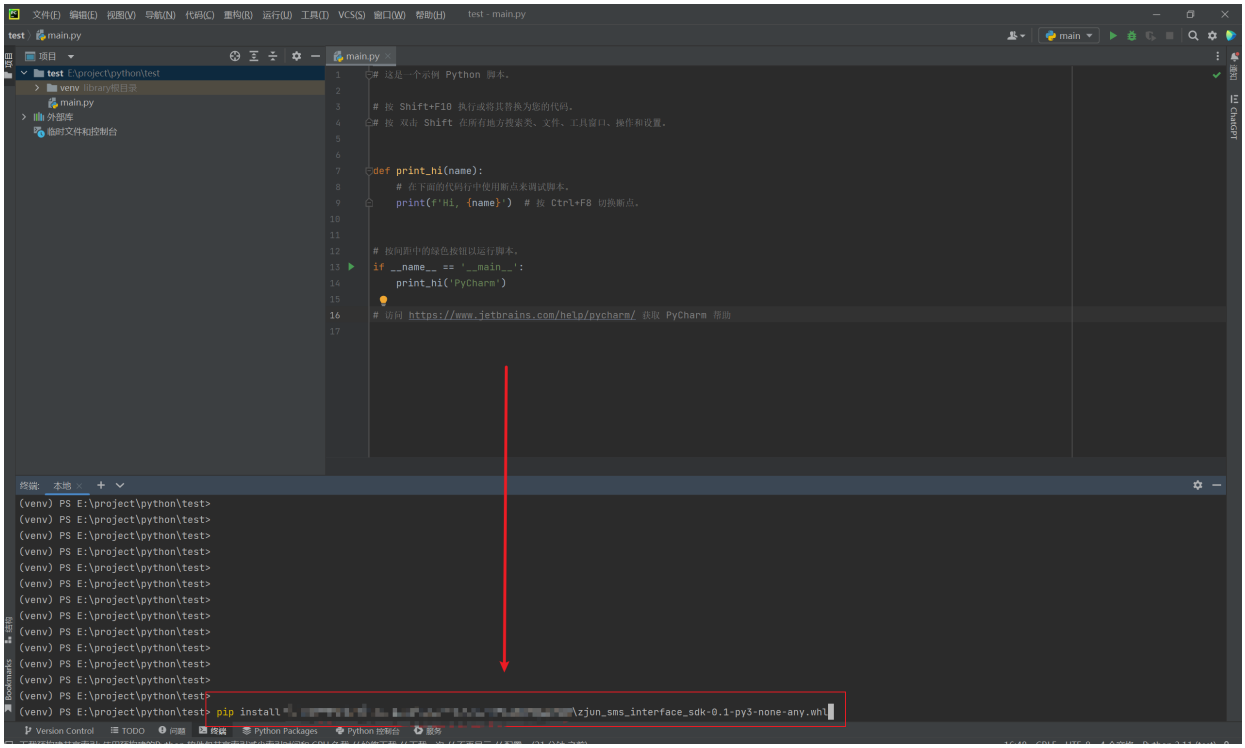
第二步：点击终端



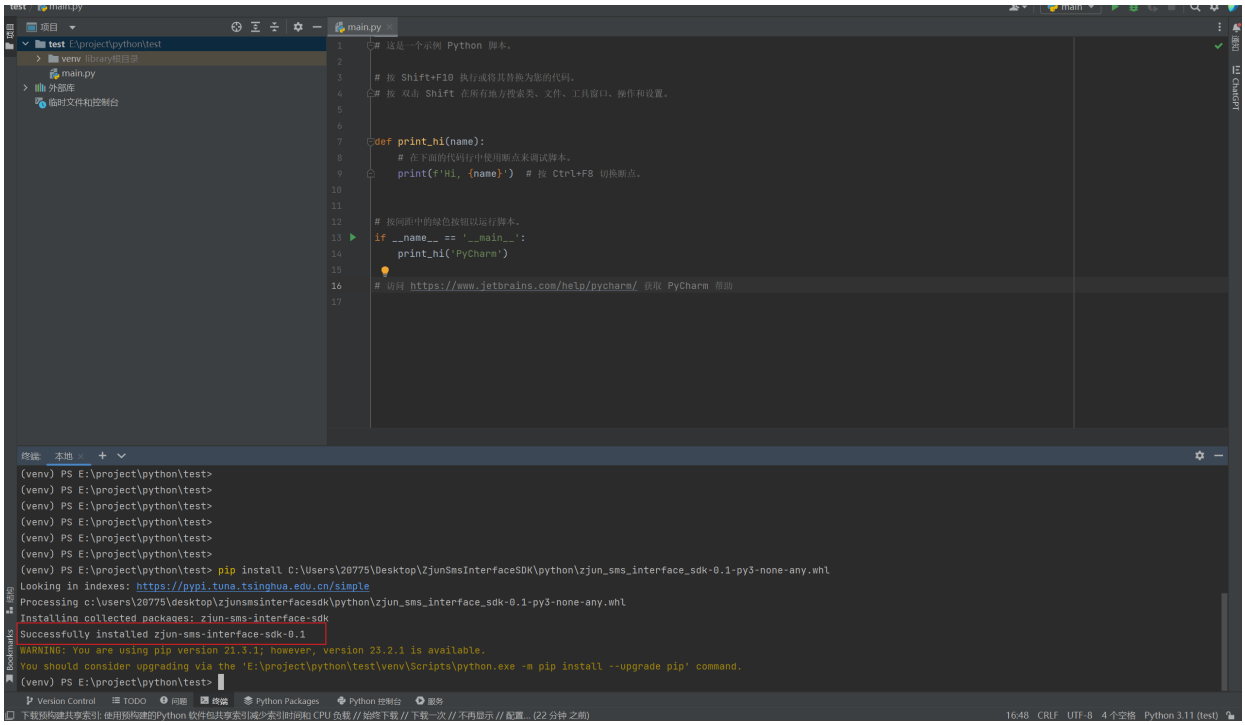
第三步：执行pip命令

pip命令：pip install <实际路径>/zjun_sms_interface_sdk-0.1-py3-none-any.whl

将命令中的<实际路径>换成你zjun_sms_interface_sdk-0.1-py3-none-any.whl文件所在的目录



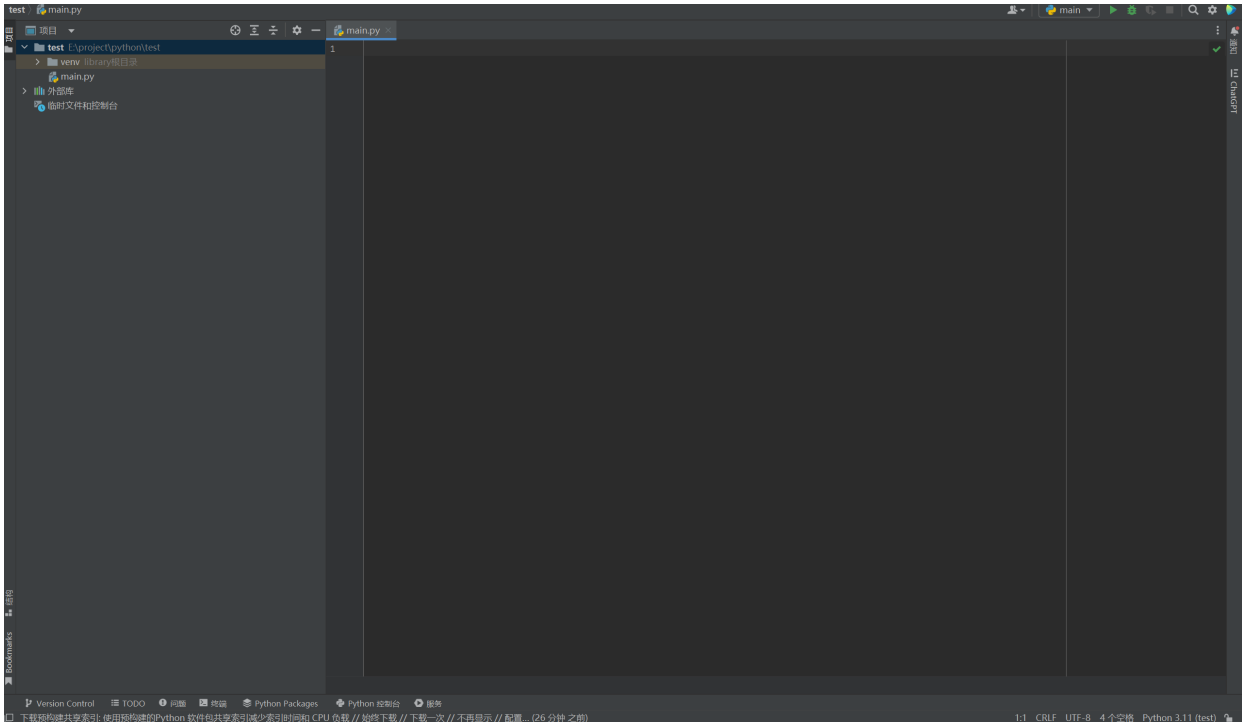
这样就是安装成功



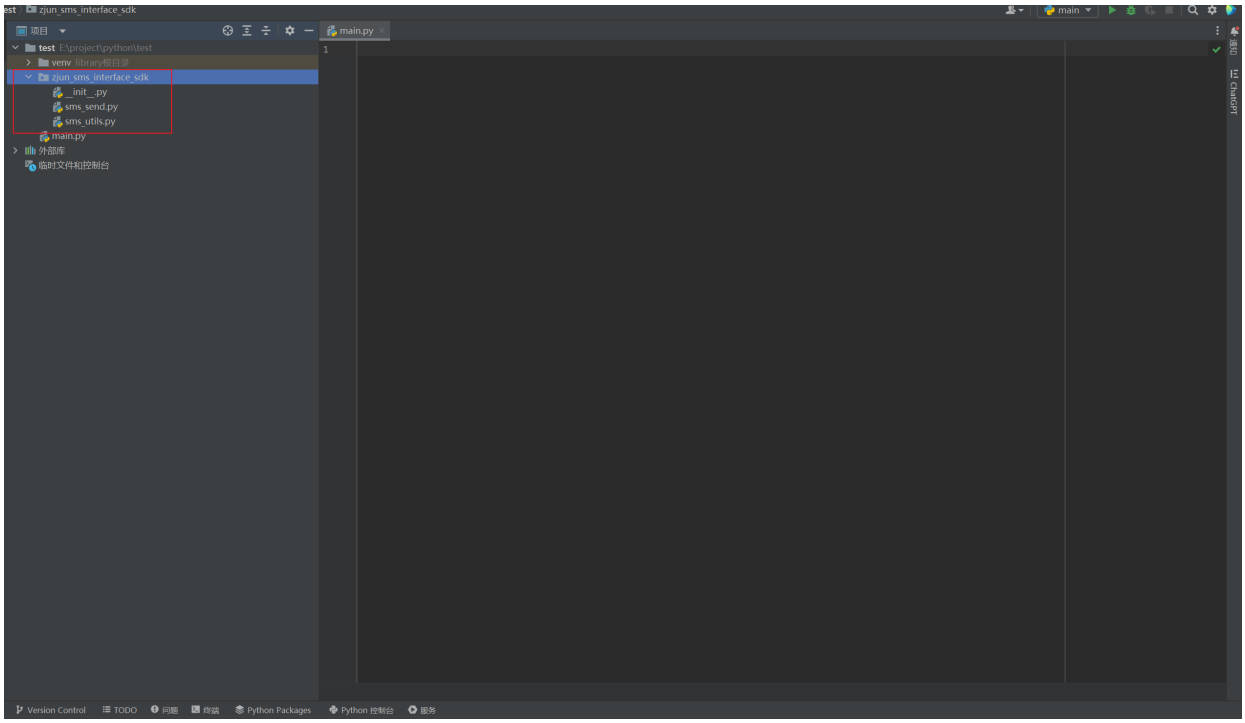
第二种方式：复制源代码

这种方式只需将文件夹复制到项目中然后使用就可以了，但是需要自己安装需要的库

第一步：打开项目

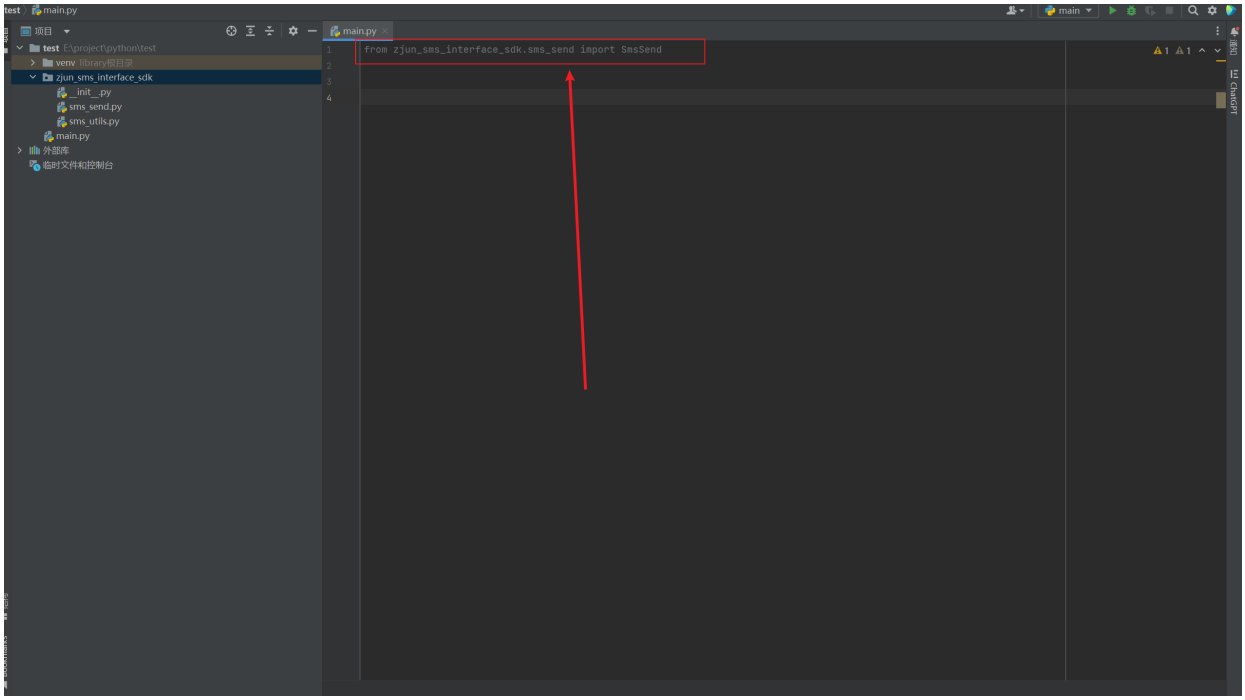


第二步：将【zjun_sms_interface_sdk】文件夹复制粘贴到项目下



二、发送短信

第一步：引用库



第二步：创建SmsSend对象，需要传入账号、密码、ip端口或域名（账号密码会在创建账号后提供给到，ip端口或域名在短信接口说明文档中获取）,address只需要ip+端口或者是域名，前面不需要http://或https://，ip之后的资源定位符也不需要

```python
from zjun_sms_interface_sdk.sms_send import SmsSend

# 账号
account = "test"
# 密码
password = "test"
# ip端口或域名
address = "192.168.2.1:8080"
# 短信发送对象
send = SmsSend(account, password, address)

```

第三步：调用方法发送短信

```python
from zjun_sms_interface_sdk.sms_send import SmsSend

# 账号
account = "test"
# 密码
password = "test"
# ip端口或域名
address = "192.168.2.1:8080"
# 短信发送对象
send = SmsSend(account, password, address)

# 手机号码列表
phone_list = ["15011111111", "15022222222"]
# 内容
content = "【签名】短信内容"

# 群发，send_time、ext_code、call_data如果不传就填空字符串
res = send.send_sms_mass(phone_list, content, "", "", "")
print(res)

```

下面可直接复制代码

```python
from zjun_sms_interface_sdk.sms_send import SmsSend

```

```python
# 账号
account = "test"
# 密码
password = "test"
# ip端口或域名
address = "192.168.2.1:8080"
# 短信发送对象
send = SmsSend(account, password, address)

# 手机号码列表
phone_list = ["15011111111", "15022222222"]
# 内容
content = "【签名】短信内容"

# 群发，send_time、ext_code、call_data如果不传就填空字符串
res = send.send_sms_mass(phone_list, content, "", "", "")
print(res)

```

三、方法说明

1. 短信发送

需先创建SmsSend对象，例如
send = SmsSend(“test”，“test”，“192.168.2.1:8080”);

（1）群发短信

send.send_sms_mass(phoneList, content, send_time, ext_code, call_data)

参数名	参数类型	说明
phoneList	list[str]	手机号码列表
content	str	短信内容
send_time	str	定时时间，不传的话就传空字符串
ext_code	str	通道扩展码，不传的话就传空字符串
call_data	str	用户回传数据，用户若传递此参数将在回执推送时回传给用户。不传的话就传空字符串

（2）一对一短信发送

send.send_sms_one(message_list, send_time)

参数名	参数类型	说明
message_list	list[dict[str:str]]	手机号码与短信内容等列表
send_time	str	定时时间，不传的话就传空字符串

message_list由多个字典对象构成的列表，具体参数列表：

参数名	参数类型	说明
phone	str	手机号码
content	str	短信内容
extcode	str	可选，通道扩展号
callData	str	可选，用户回传数据，用户若传递此参数将在回执推送时回传给用户。

2. 获取状态、回复、余额

（1）获取状态

send.get_report()

（2）获取回复

send.get_upstream()

（3）获取余额

send.get_balance()

3. 短信模板

（1）提交短信模板

send.create_template(content)

参数名	参数类型	说明
content	str	模板内容

（2）查询短信模板

send.query_template()

> 该文档只提供SDK使用说明，有关接口的详细说明请查阅接口文档

Go

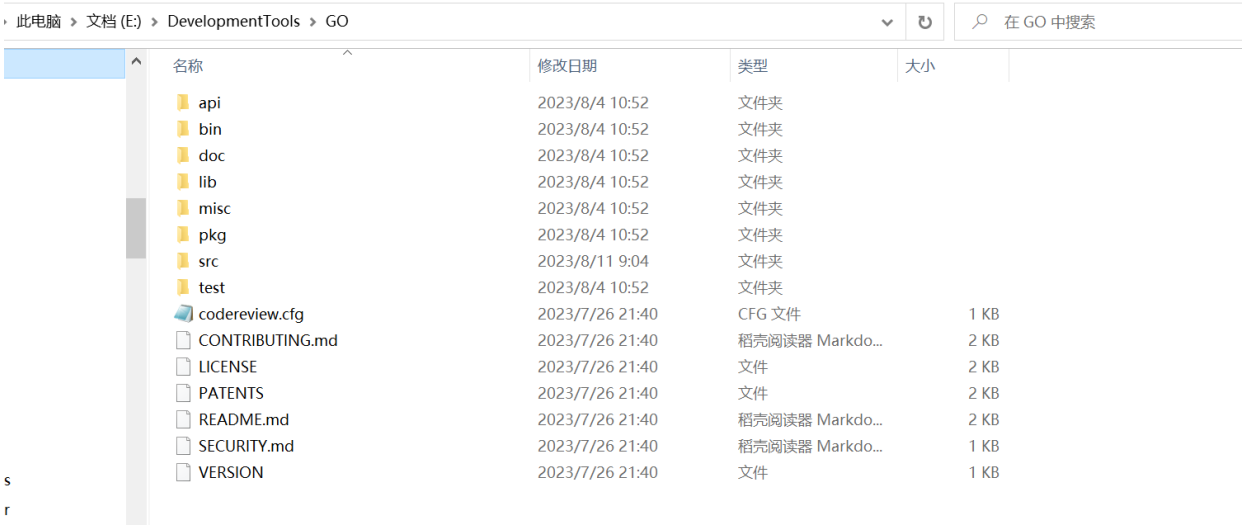
SDK使用go版本为 **go 1.20**

文档中使用的开发工具为 **GoLand 2023**

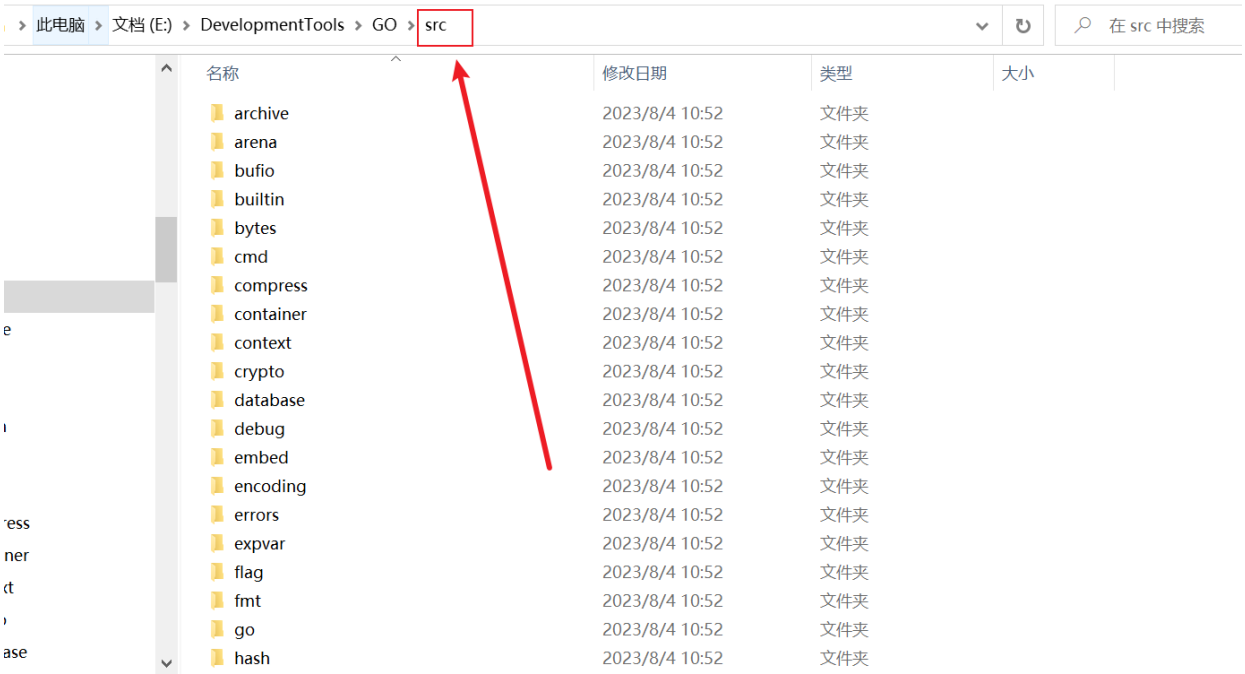
一、类库安装

【izjun】文件夹里包含的是源代码，直接将整个文件夹复制到GOPATH路径下的src目录下就可以了，具体看下面步骤

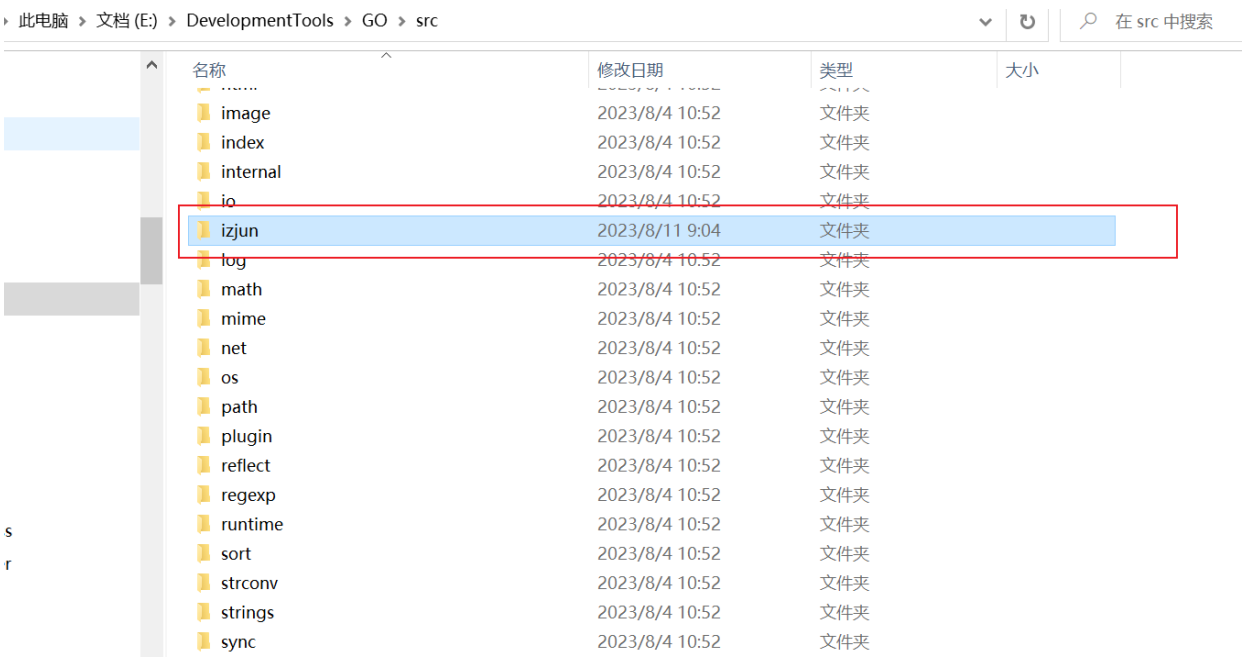
第一步：打开文件资源管理器，切换到Go的安装目录下



第二步：切换到src目录下

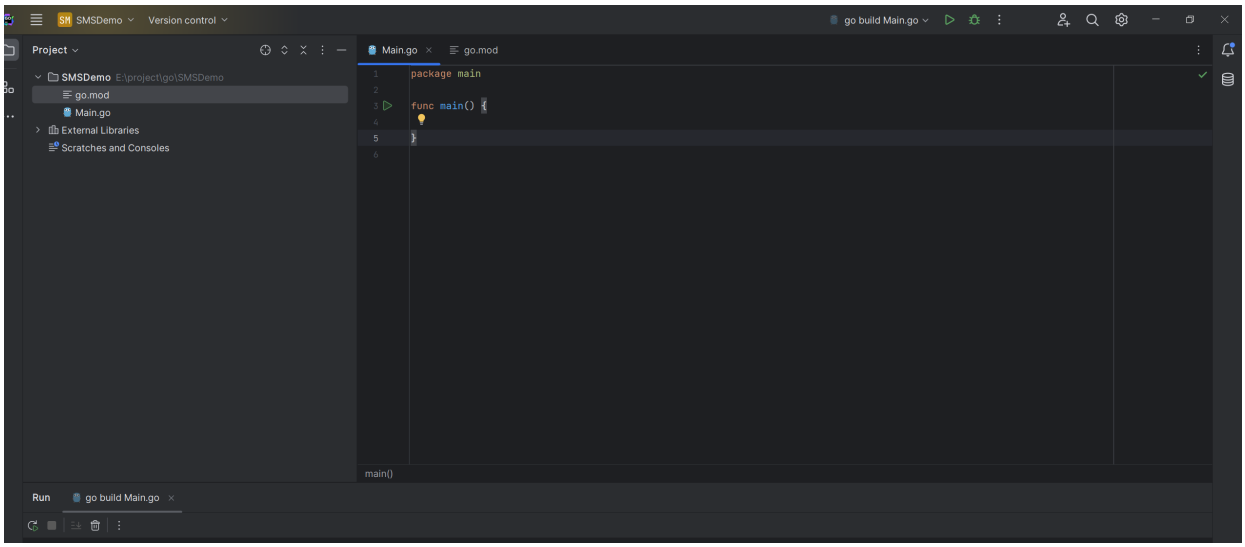


第三步：将【izjun】文件夹整个复制到src目录下

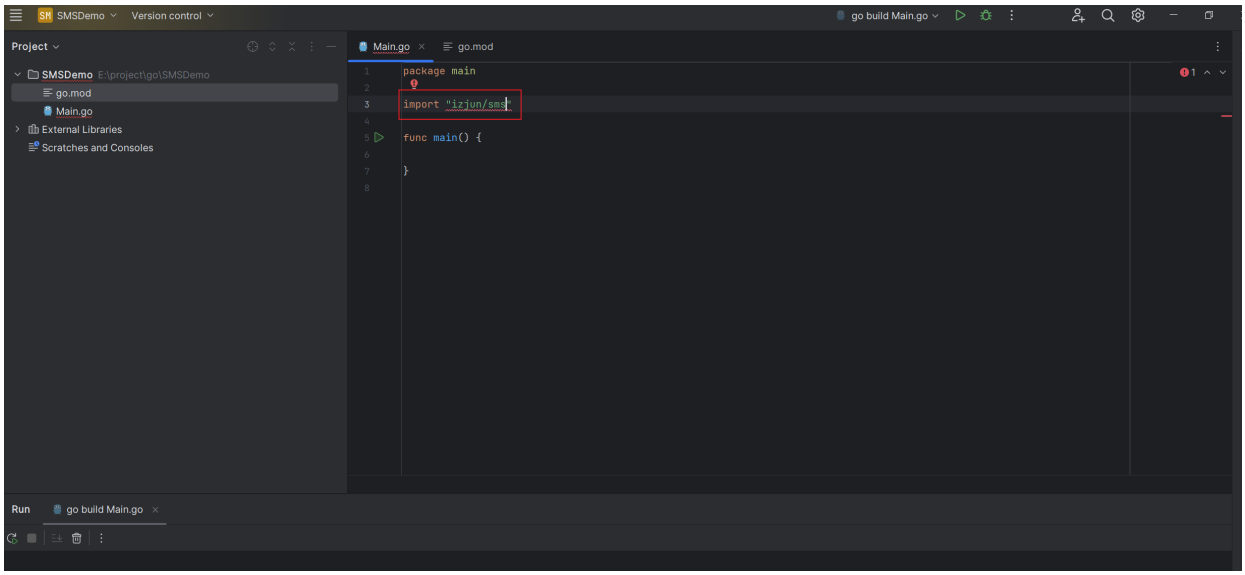


二、发送短信

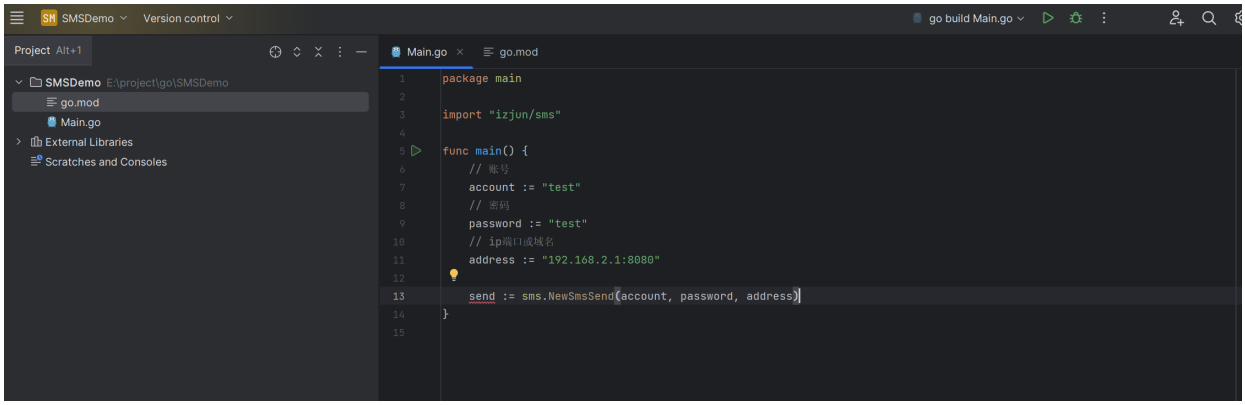
第一步：打开项目



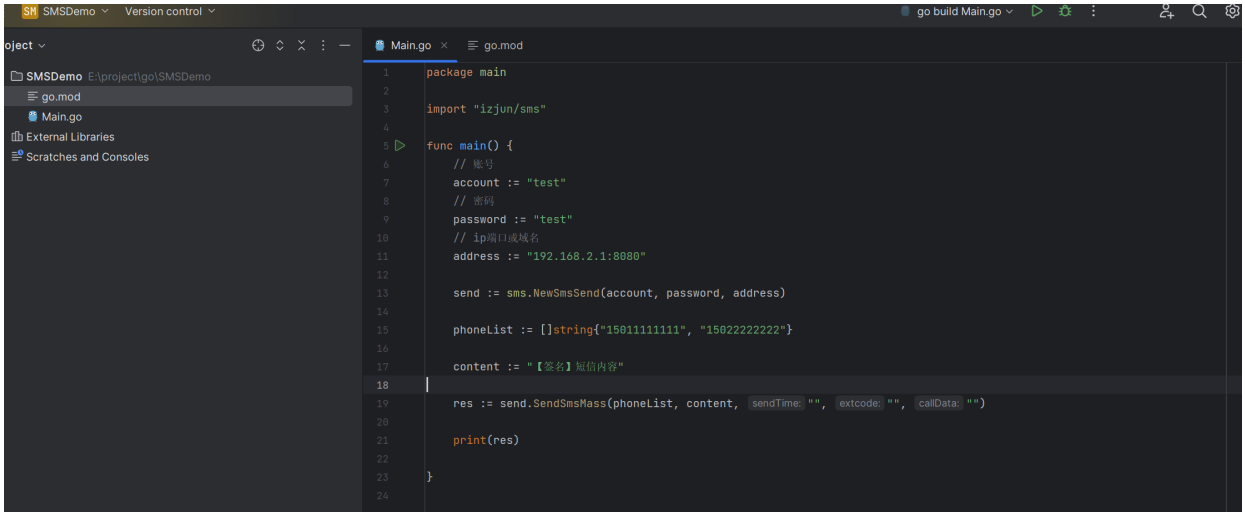
第二步：引用库



第三步：创建send对象，需要传入账号、密码、ip端口或域名（账号密码会在创建账号后提供给到，ip端口或域名在短信接口说明文档中获取）,address只需要ip+端口或者是域名，前面不需要http://或https://，ip之后的资源定位符也不需要



第四步：调用方法发送短信



下面可直接复制代码

```go
package main

import "izjun/sms"

func main() {
    // 账号
    account := "test"
    // 密码
    password := "test"
    // ip端口或域名
    address := "192.168.2.1:8080"

    send := sms.NewSmsSend(account, password, address)

    phoneList := []string{"15011111111", "15022222222"}

    content := "【签名】短信内容"

    // 群发短信，其他参数不需要可传空字符串
    res := send.SendSmsMass(phoneList, content, "", "", "")

    print(res)

    // 手机号与内容列表
    messageList := []map[string]string{
        map[string]string{
            "phone":   "15011111111",
            "content": "【签名】短信内容",
            "extcode": "123",
            "callData": "abc",
        },
        map[string]string{
            "phone":   "15022222222",
            "content": "【签名】短信内容2",
            "extcode": "456",
            "callData": "efg",
        },
    }
    // 一对一发送，其他参数不需要可传空字符
    res = send.SendSmsOne(messageList, "")
    print(res)

}

```

三、方法说明

1. 短信发送

需先创建SmsSend对象，例如
send := NewSmsSend(“test”，“test”，“192.168.2.1:8080”);

（1）群发短信

send.SendSmsMass(phoneList, content, sendTime, extcode, callData)

参数名	参数类型	说明
phoneList	[]string	手机号码列表
content	string	短信内容
sendTime	string	定时时间，不传的话就传空字符串
extcode	string	通道扩展码，不传的话就传空字符串
callData	string	用户回传数据，用户若传递此参数将在回执推送时回传给用户。不传的话就传空字符串

（2）一对一短信发送

send.SendSmsOne(messageList, sendTime)

参数名	参数类型	说明
messageList	[]map[string]string	手机号码与短信内容等列表
sendTime	string	定时时间，不传的话就传空字符串

messageList由多个map对象构成的列表，具体参数列表：

参数名	参数类型	说明
phone	string	手机号码
content	string	短信内容
extcode	string	可选，通道扩展号
callData	string	可选，用户回传数据，用户若传递此参数将在回执推送时回传给用户。

2. 获取状态、回复、余额

（1）获取状态

send.GetReport()

（2）获取回复

send.GetUpstream()

（3）获取余额

send.GetBalance()

3. 短信模板

（1）提交短信模板

send.CreateTemplate(content)

参数名	参数类型	说明
content	string	模板内容

（2）查询短信模板

send.QueryTemplate()

> 该文档只提供SDK使用说明，有关接口的详细说明请查阅接口文档

Node.js

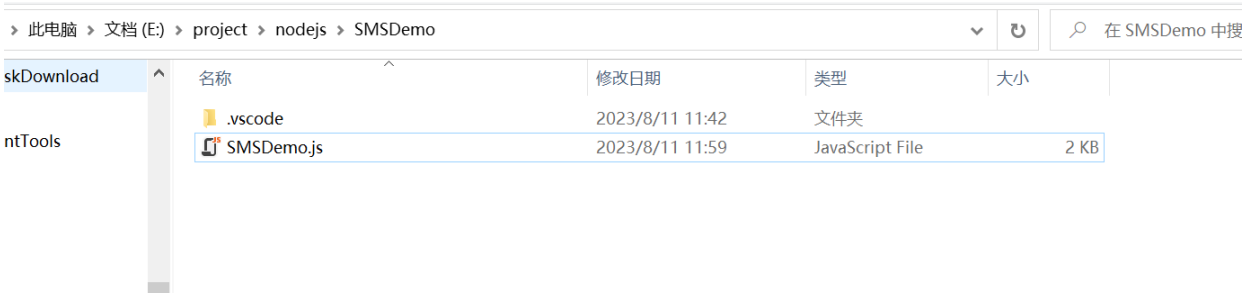
SDK使用node.js版本为 **v14**

文档中使用的开发工具为 **Visual Studio Code**

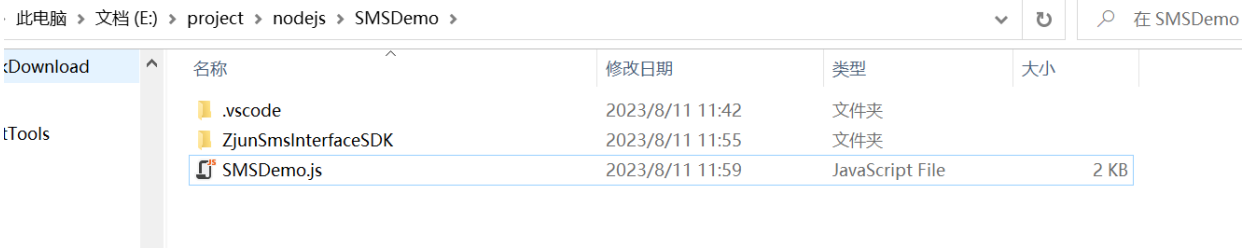
一、类库安装

【ZjunSmsInterfaceSDK】文件夹里包含的是源代码，直接将整个文件夹复制到项目中使用即可

第一步：打开文件资源管理器，切换到项目路径下

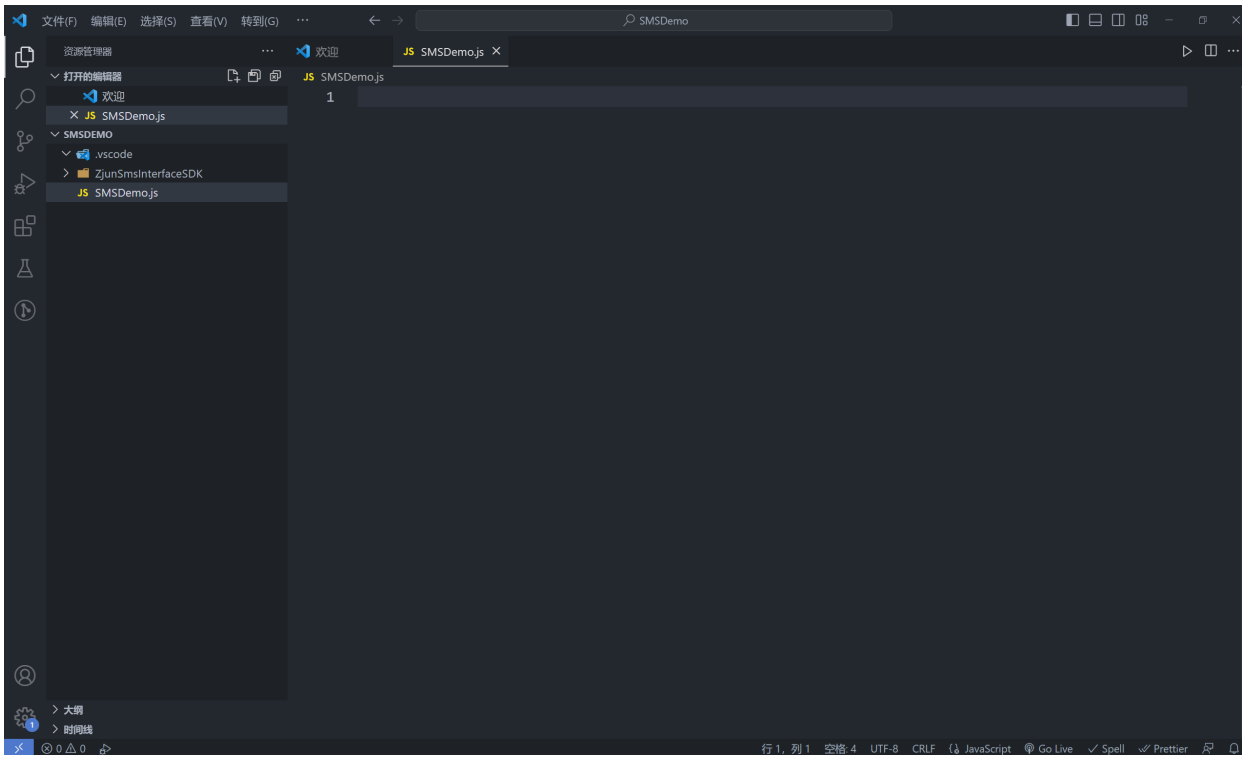


第二步：将【ZjunSmsInterfaceSDK】文件夹复制到该目录下

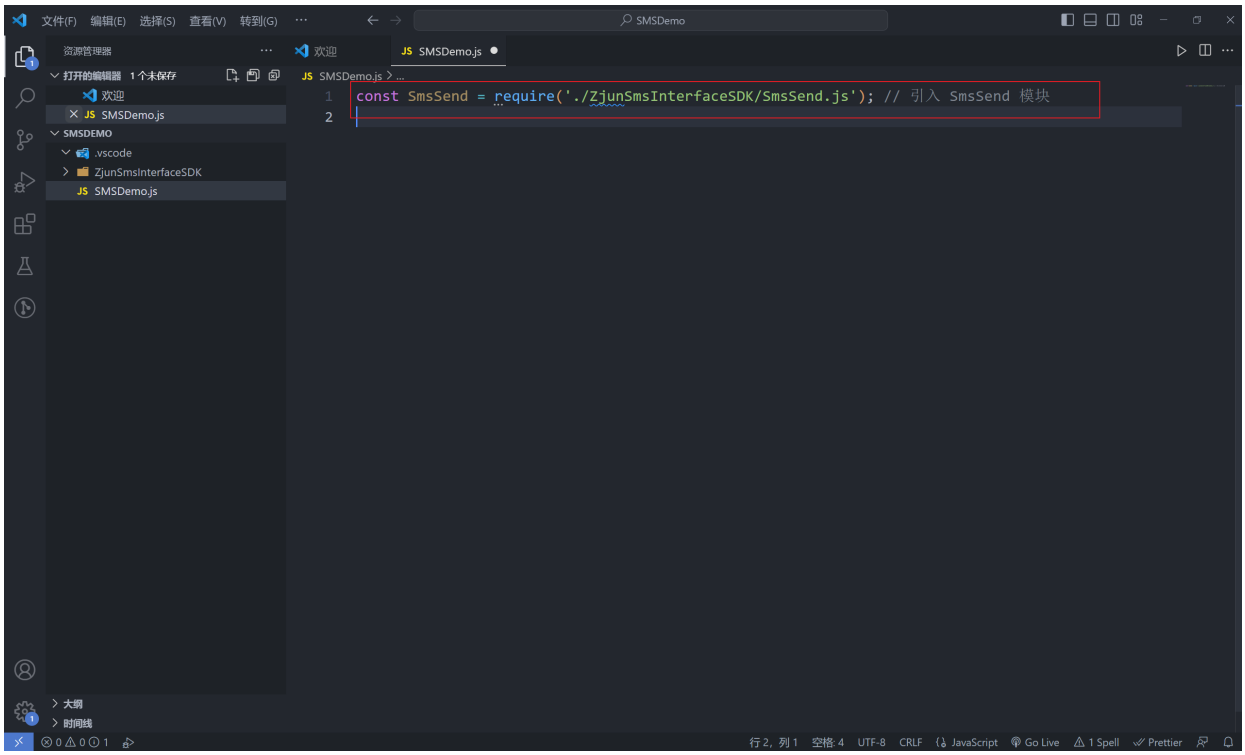


二、发送短信

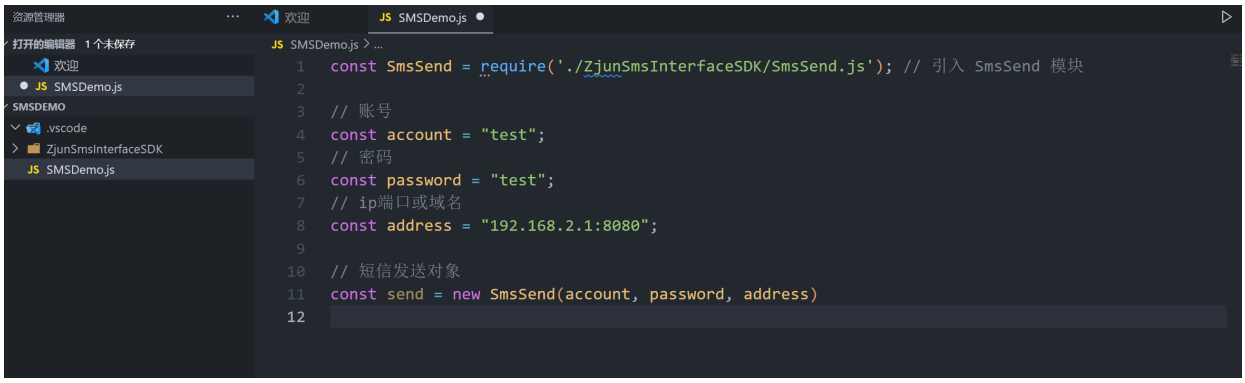
第一步：打开项目



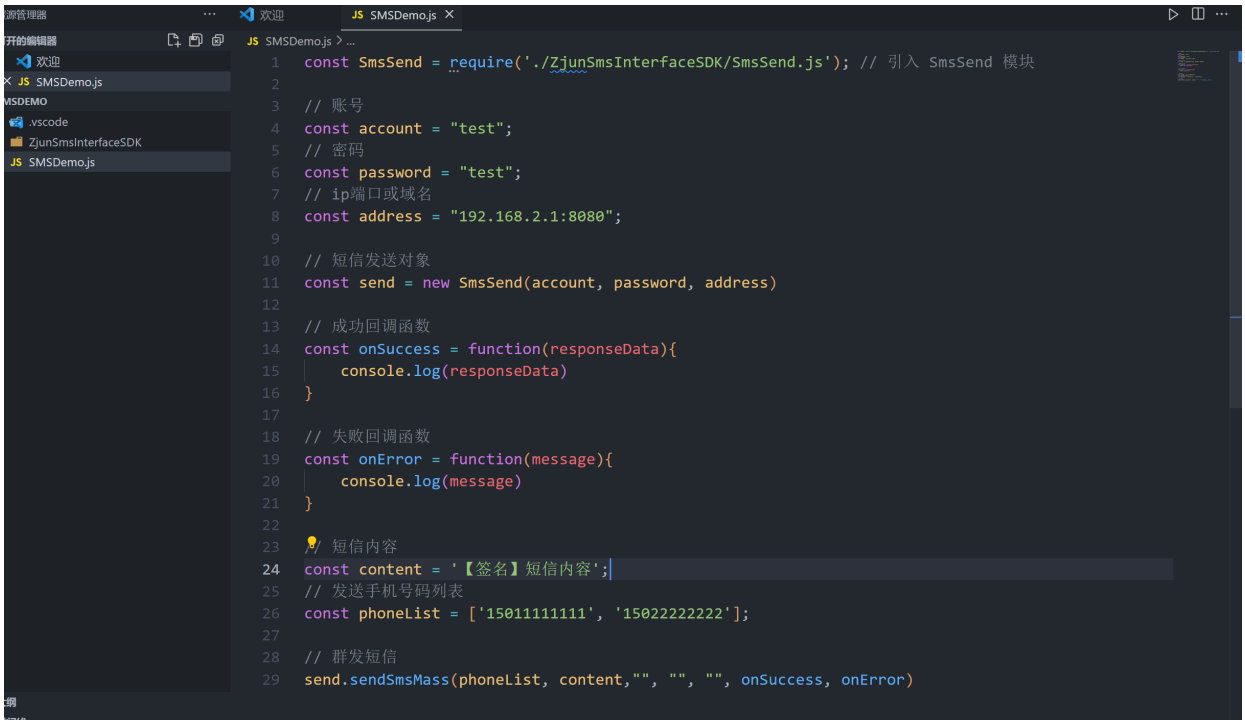
第二步：引用库，其中SmsSend.js的路径为相对路径



第三步：创建send对象，需要传入账号、密码、ip端口或域名（账号密码会在创建账号后提供给到，ip端口或域名在短信接口说明文档中获取）,address只需要ip+端口或者是域名，前面不需要http://或https://，ip之后的资源定位符也不需要



第四步：调用方法发送短信



下面可直接复制代码

```
const SmsSend = require('./ZjunSmsInterfaceSDK/SmsSend.js'); // 引入 SmsSend 模块

// 账号
const account = "test";
// 密码
const password = "test";
// ip端口或域名
const address = "192.168.2.1:8080";

// 短信发送对象
const send = new SmsSend(account, password, address)

// 成功回调函数
const onSuccess = function(responseData){
```

```
    console.log(responseData)
}

// 失败回调函数
const onError = function(message){
    console.log(message)
}

// 短信内容
const content = '【签名】短信内容';
// 发送手机号码列表
const phoneList = ['15011111111', '15022222222'];

// 群发短信
send.sendSmsMass(phoneList, content,"", "", "", onSuccess, onError)

const messageList = [
    {
        phone: "15011111111",
        content: "【签名】短信内容1",
        extcode: "123",
        callData: "e2fc714c4727ee9395f324cd2e7f331f"
    },
    {
        phone: "15022222222",
        content: "【签名】短信内容2",
        extcode: "223",
        callData: "9df62e693988eb4e1e1444ece0578579"
    }

]

// 点对点短信
send.sendSmsOne(messageList, "", onSuccess, onError)
```

三、方法说明

1. 短信发送

需先创建SmsSend对象，例如
send = SmsSend(“test” , “test” , “192.168.2.1:8080”);

（1）群发短信

send.sendSmsMass(phoneList, content, sendTime, extcode, callData, onSuccess, onError)

参数名	参数类型	说明
phoneList	Array[String]	手机号码列表
content	String	短信内容
sendTime	String	定时时间，不传的话就传空字符串
extcode	String	通道扩展码，不传的话就传空字符串
callData	String	用户回传数据，用户若传递此参数将在回执推送时回传给用户。不传的话就传空字符串
onSuccess	Function	请求成功的回调函数，返回响应数据
onError	Function	请求错误的回调函数，返回错误消息

（2）一对一短信发送

send.sendSmsOne(messageList, sendTime, onSuccess, onError)

参数名	参数类型	说明
messageList	Array[Object]	手机号码与短信内容等列表
sendTime	String	定时时间，不传的话就传空字符串
onSuccess	Function	请求成功的回调函数，返回响应数据
onError	Function	请求错误的回调函数，返回错误消息

messageList由多个对象构成的列表，具体参数列表：

参数名	参数类型	说明
phone	String	手机号码
content	String	短信内容
extcode	String	可选，通道扩展号
callData	String	可选，用户回传数据，用户若传递此参数将在回执推送时回传给用户。

2. 获取状态、回复、余额

（1）获取状态

send.getReport(onSuccess, onError)

参数名	参数类型	说明
onSuccess	Function	请求成功的回调函数，返回响应数据
onError	Function	请求错误的回调函数，返回错误消息

（2）获取回复

send.getUpstream(onSuccess, onError)

参数名	参数类型	说明
onSuccess	Function	请求成功的回调函数，返回响应数据
onError	Function	请求错误的回调函数，返回错误消息

（3）获取余额

send.getBalance(onSuccess, onError)

参数名	参数类型	说明
onSuccess	Function	请求成功的回调函数，返回响应数据
onError	Function	请求错误的回调函数，返回错误消息

3. 短信模板

（1）提交短信模板

send.createTemplate(content, onSuccess, onError)

参数名	参数类型	说明
content	String	模板内容
onSuccess	Function	请求成功的回调函数，返回响应数据
onError	Function	请求错误的回调函数，返回错误消息

（2）查询短信模板

send.queryTemplate(onSuccess, onError)

参数名	参数类型	说明
onSuccess	Function	请求成功的回调函数，返回响应数据
onError	Function	请求错误的回调函数，返回错误消息

> 该文档只提供SDK使用说明，有关接口的详细说明请查阅接口文档

PHP

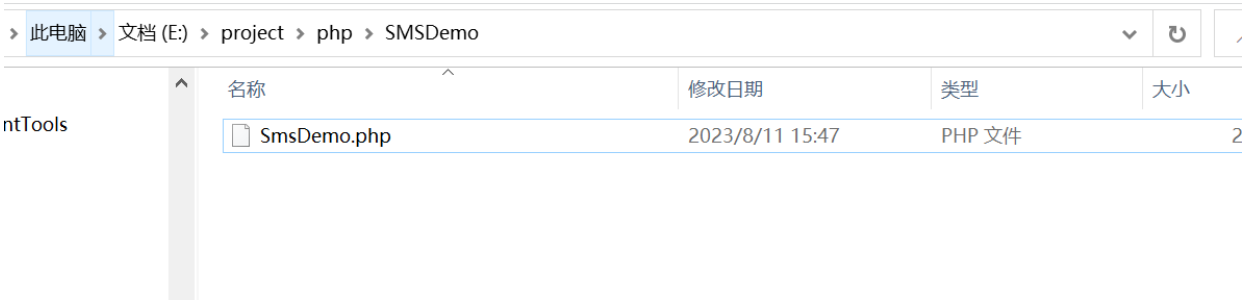
SDK使用php版本为 **php5**

文档中使用的开发工具为 **Visual Studio Code**

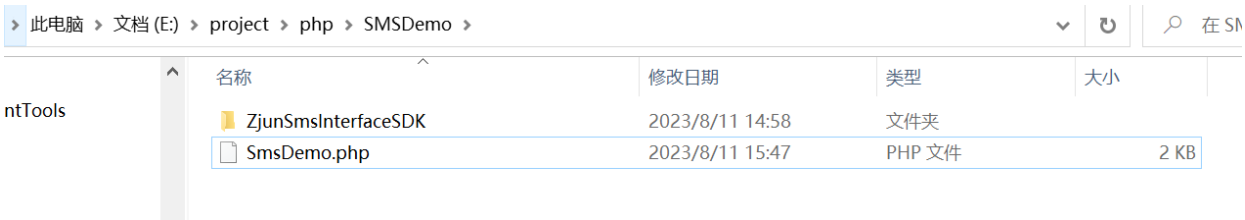
一、类库安装

【ZjunSmsInterfaceSDK】文件夹里包含的是源代码，直接将整个文件夹复制到项目中使用即可

第一步：打开文件资源管理器，切换到项目路径下

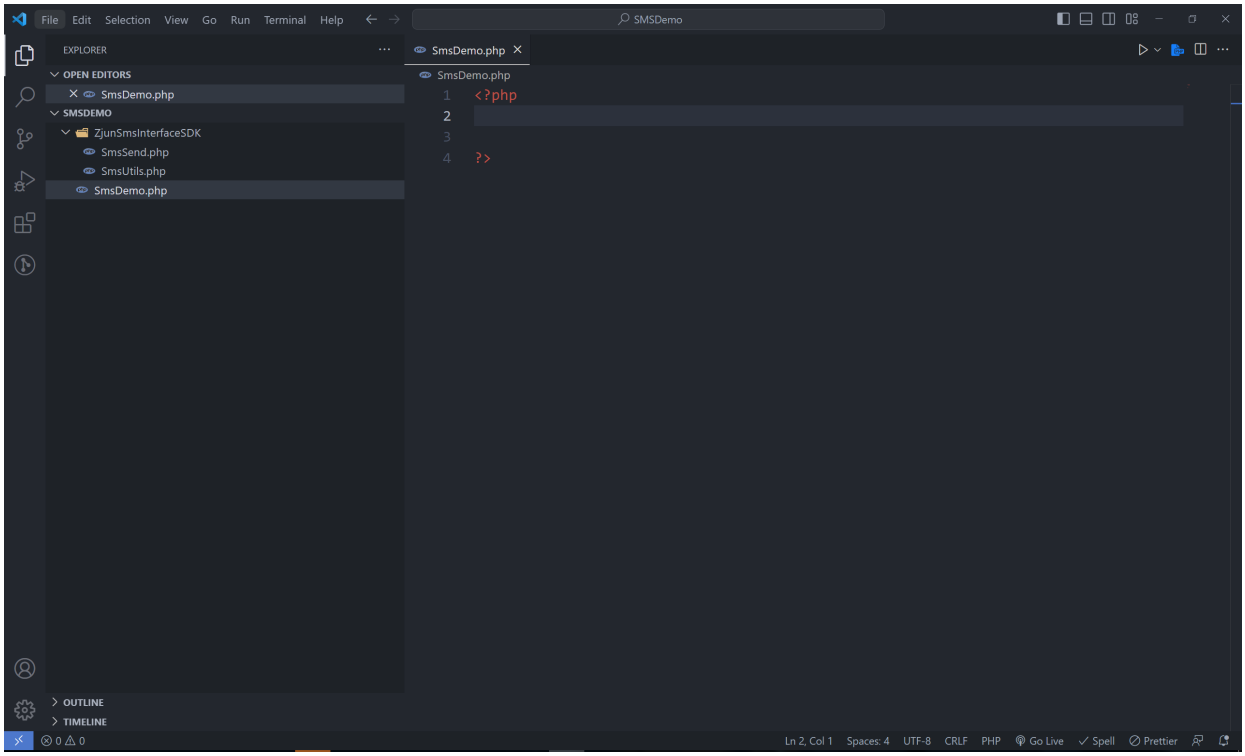


第二步：将【ZjunSmsInterfaceSDK】文件夹复制到该目录下

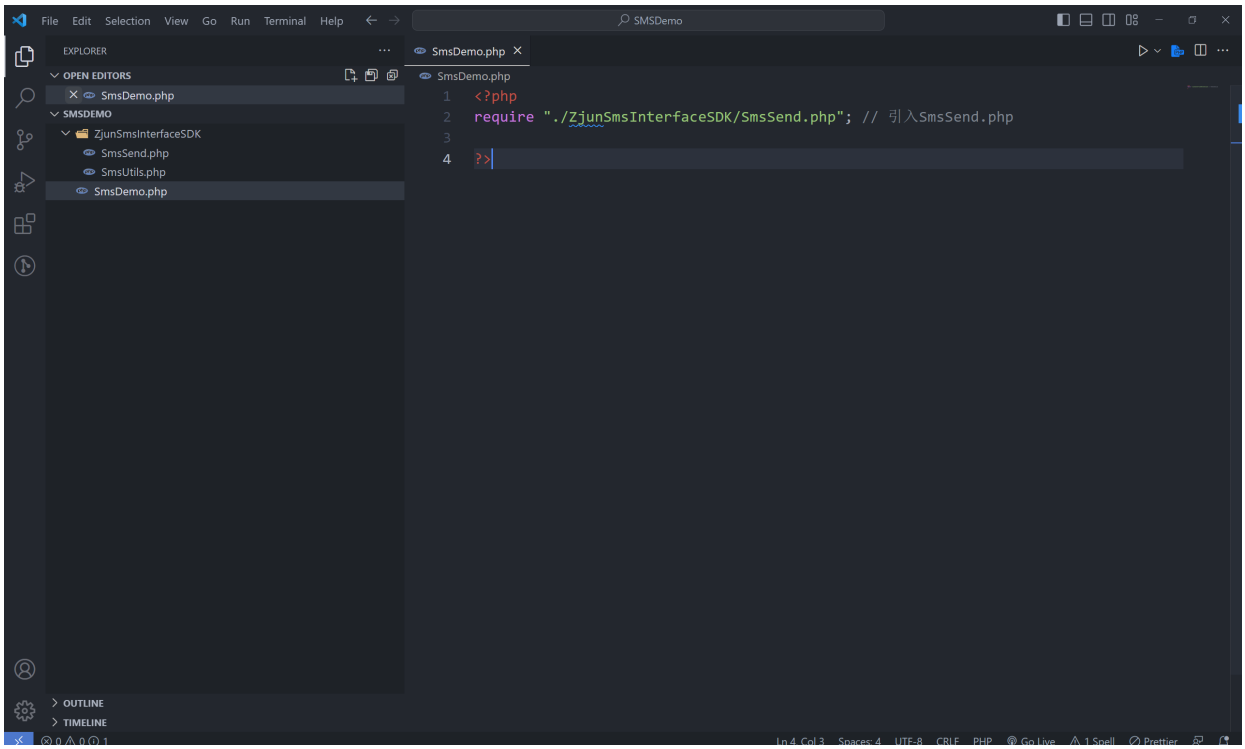


二、发送短信

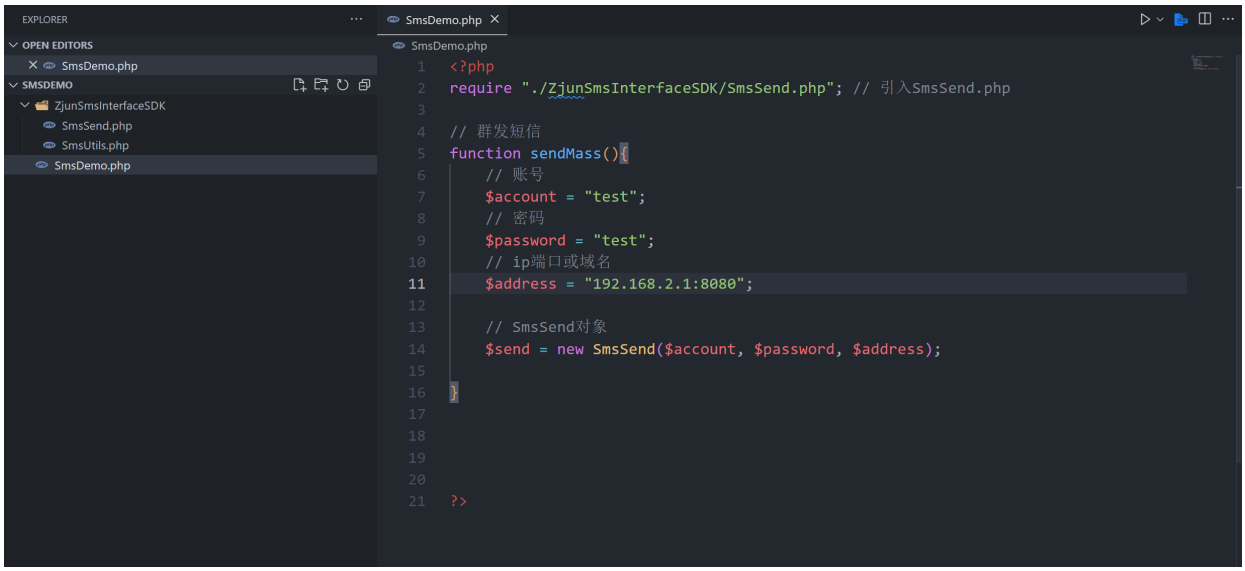
第一步：打开项目



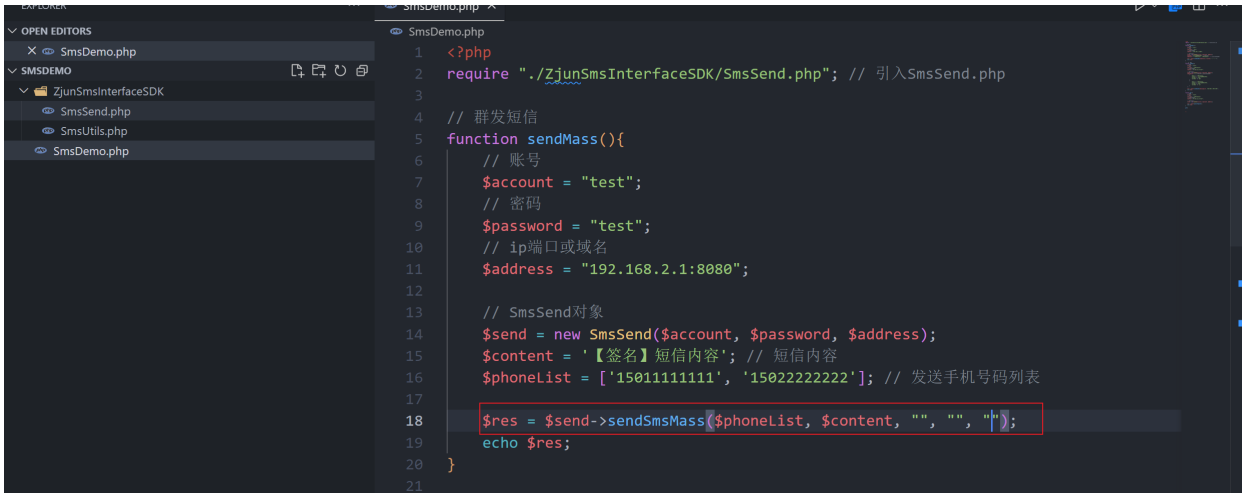
第二步：引用库



第三步：创建send对象，需要传入账号、密码、ip端口或域名（账号密码会在创建账号后提供给到，ip端口或域名在短信接口说明文档中获取）,address只需要ip+端口或者是域名，前面不需要http://或https://，ip之后的资源定位符也不需要



第四步：调用方法发送短信



下面可直接复制代码

```php
<?php
require "./ZjunSmsInterfaceSDK/SmsSend.php"; // 引入SmsSend.php

// 群发短信
function sendMass(){
    // 账号
    $account = "test";
    // 密码
    $password = "test";
    // ip端口或域名
    $address = "192.168.2.1:8080";

    // SmsSend对象
    $send = new SmsSend($account, $password, $address);
    $content = '【签名】短信内容'; // 短信内容
    $phoneList = ['15011111111', '15022222222']; // 发送手机号码列表

    $res = $send->sendSmsMass($phoneList, $content, "", "", "");
    echo $res;
}

// 一对一短信
function sendOne(){
    // 账号
    $account = "test";
    // 密码
    $password = "test";
    // ip端口或域名
    $address = "192.168.2.1:8080";

    // SmsSend对象
```

```php
    $send = new SmsSend($account, $password, $address);
    $messageList = [ // 短信内容和发送手机号码列表
        [
            'phone' => '15011111111',
            'content' => '【签名】短信内容1',
            'extcode' => '222',
            'callData' => 'bbb'
        ],
        [
            'phone' => '15022222222',
            'content' => '【签名】短信内容2',
            'extcode' => '333',
            'callData' => 'ccc'
        ]
    ];

    $res = $send->sendSmsOne($messageList, "");
    echo $res;
}
?>
```

三、方法说明

1. 短信发送

需先创建SmsSend对象，例如
$send = new SmsSend(“test”,“test”,“192.168.2.1:8080”);

（1）群发短信

$send->sendSmsMass(phoneList, content, sendTime, extcode, callData)

参数名	参数类型	说明
phoneList	array[string]	手机号码列表
content	string	短信内容
sendTime	string	定时时间，不传的话就传空字符串
extcode	string	通道扩展码，不传的话就传空字符串
callData	string	用户回传数据，用户若传递此参数将在回执推送时回传给用户。不传的话就传空字符串

（2）一对一短信发送

$send->sendSmsOne(messageList, sendTime)

参数名	参数类型	说明
messageList	array[array[]]	手机号码与短信内容等列表
sendTime	string	定时时间，不传的话就传空字符串

messageList由多个一维数组构成二维数组，具体参数列表：

参数名	参数类型	说明
phone	string	手机号码
content	string	短信内容
extcode	string	可选，通道扩展号
callData	string	可选，用户回传数据，用户若传递此参数将在回执推送时回传给用户。

2. 获取状态、回复、余额

（1）获取状态

$send->getReport()

（2）获取回复

$send->getUpstream()

（3）获取余额

$send->getBalance()

3. 短信模板

（1）提交短信模板

$send->createTemplate(content)

参数名	参数类型	说明
content	string	模板内容

（2）查询短信模板

```
$send->queryTemplate()
```

> 该文档只提供SDK使用说明，有关接口的详细说明请查阅接口文档