# SpringBoot环境集成TestNG

## 准备工作

开发环境Eclipse安装TestNG插件和Maven插件。

## 代码集成

1. 拷贝src/test/resources和src/test/java下的文件到项目对应的目录。
2. [修改src/test/java下com.wafer.wtp.base.TestAPI的类注解@SpringBootTest](mailto:修改src/test/java下com.wafer.wtp.base.TestAPI的类注解@SpringBootTest)，参数classes的值为当前项目的启动类(TargetApplication.class)。

(低版本SpringBoot使用@SpringApplicationConfiguration(classes= TargetApplication.class)注解)

1. 修改src/test/java下com.wafer.wtp.base.AbstractTestNGTest类中变量值。

DP\_DATA\_TYPE:接口测试数据源类型(rest,json,excel)

DP\_REST\_PATH:数据类型为rest时，数据来源地址(http://localhost:8000/api/v1/dss/1)

1. 拷贝testng.xml到项目根路径下。
2. 修改pom.xml文件，添加对应依赖和Maven配置。

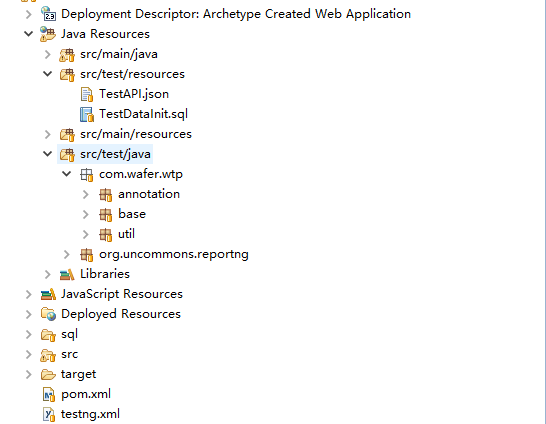
添加依赖如下：

|  |
| --- |
| <dependency>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-starter-test</artifactId>  <scope>test</scope>  </dependency>  <dependency>  <groupId>net.sf.json-lib</groupId>  <artifactId>json-lib</artifactId>  <version>2.4</version>  <classifier>jdk15</classifier>  </dependency>  <!-- 添加testNG依赖 -->  <dependency>  <groupId>org.testng</groupId>  <artifactId>testng</artifactId>  <version>6.9.10</version>  <scope>test</scope>  </dependency>  <!-- 依赖reportNg 关联testNg -->  <dependency>  <groupId>org.uncommons</groupId>  <artifactId>reportng</artifactId>  <version>1.1.4</version>  <scope>test</scope>  <exclusions>  <exclusion>  <groupId>org.testng</groupId>  <artifactId>testng</artifactId>  </exclusion>  </exclusions>  </dependency>  <!-- 依赖Guice -->  <dependency>  <groupId>com.google.inject</groupId>  <artifactId>guice</artifactId>  <version>4.0</version>  <scope>test</scope>  </dependency>  <!-- 依赖velocity -->  <dependency>  <groupId>velocity</groupId>  <artifactId>velocity-dep</artifactId>  <version>1.4</version>  </dependency>  <!-- 依赖poi读取Excel数据 -->  <dependency>  <groupId>org.apache.poi</groupId>  <artifactId>poi</artifactId>  <version>3.14</version>  </dependency> |

添加Maven配置如下：

|  |
| --- |
| <build>  <plugins>  <plugin>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-maven-plugin</artifactId>  </plugin>  <!-- 运行项目中src/test/Java文件夹下的testcase -->  <plugin>  <groupId>org.apache.maven.plugins</groupId>  <artifactId>maven-surefire-plugin</artifactId>  <configuration>  <!-- 不对xls进行转码 -->  <properties>  <property>  <name>usedefaultlisteners</name>  <value>false</value>  </property>  <property>  <name>listener</name>  <value>org.uncommons.reportng.HTMLReporter, org.uncommons.reportng.JUnitXMLReporter</value>  </property>  </properties>  <suiteXmlFiles>  <suiteXmlFile>testng.xml</suiteXmlFile>  </suiteXmlFiles>  <!-- 系统参数默认不初始化数据mvn test -DinitData=true -->  <systemPropertyVariables>  <init>false</init>  <filePath>classpath:TestDataInit.sql</filePath>  </systemPropertyVariables>  </configuration>  </plugin>  </plugins>  <testResources>  <testResource>  <directory>src/test/java</directory>  <includes>  <include>\*\*/\*.xls</include>  <include>\*\*/\*.json</include>  </includes>  <filtering>false</filtering>  </testResource>  <testResource>  <directory>src/test/resources</directory>  <includes>  <include>\*\*/\*.xls</include>  <include>\*\*/\*.json</include>  </includes>  <filtering>false</filtering>  </testResource>  </testResources>  </build> |

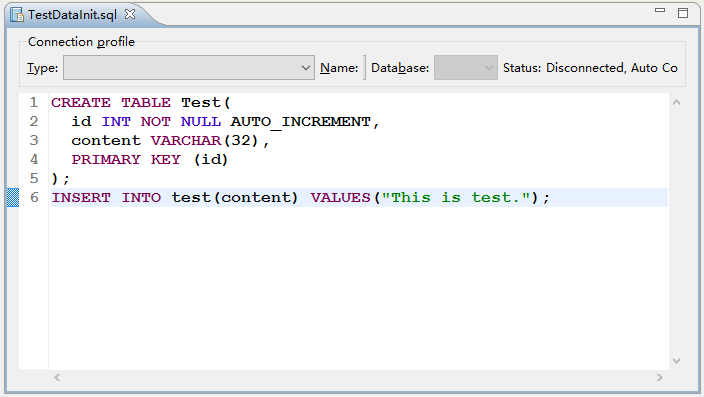
代码集成之后的工程目录如下所示:



## 接口测试

### 1．测试数据初始化

在src/test/resources目录中的TestDataInit.sql文件中维护数据库初始化sql脚本，如果需要执行初始化脚本，在mvn test命令后添加 -Dinit=true 参数即可执行。

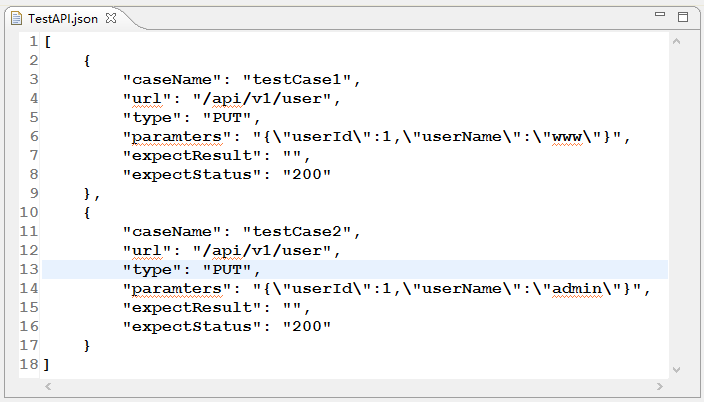


### 2．测试接口数据准备

测试接口数据支持json文件和Excel文件两种方式，两种方式只能选择一种。

#### JSON数据维护

在src/test/resources目录中的TestAPI.json维护接口测试数据。



1. caseName：测试用例的名称
2. url：接口请求地址
3. type：请求类型（GET,POST,PUT,DELETE）
4. paramters：请求需要的参数，如果请求没有参数，则不填
5. exceptResult：请求期望返回的结果，如果不需要校验返回结果，则不填
6. exceptStatus：期望请求执行的状态，如果不需要校验请求状态，则不填

为方便维护数据，接口数据文件可以根据模块划分。例如：

用户管理接口测试数据可以维护在TestAPI\_UserManage.json；

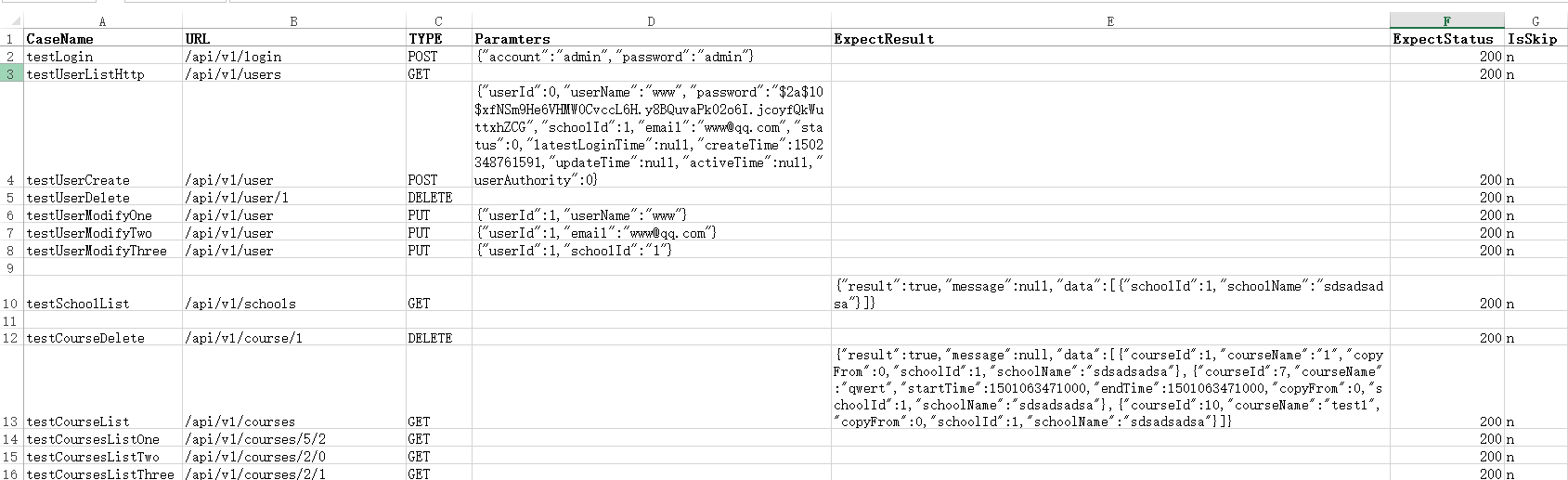
项目管理接口测试数据可以维护在TestAPI\_ProjectManage.json；

……

注意:所有jon数据文件必须以TestAPI开头，以.json结尾。

#### Excel数据维护

在src/test/resources目录中的TestAPI.xls维护接口测试数据。



1. CaseName：测试用例的名称
2. URL：接口请求地址
3. TYPE：请求类型（GET,POST,PUT,DELETE）
4. Paramters：请求需要的参数，如果请求没有参数，则不填
5. ExceptResult：请求期望返回的结果，如果不需要校验返回结果，则不填
6. ExceptStatus：期望请求执行的状态，如果不需要校验请求状态，则不填
7. IsSkip:标记这条Case是否跳过(不执行)（y,n）

为方便维护数据，接口数据文件可以根据模块划分。例如：

用户管理接口测试数据可以维护在TestAPI\_UserManage.xls；

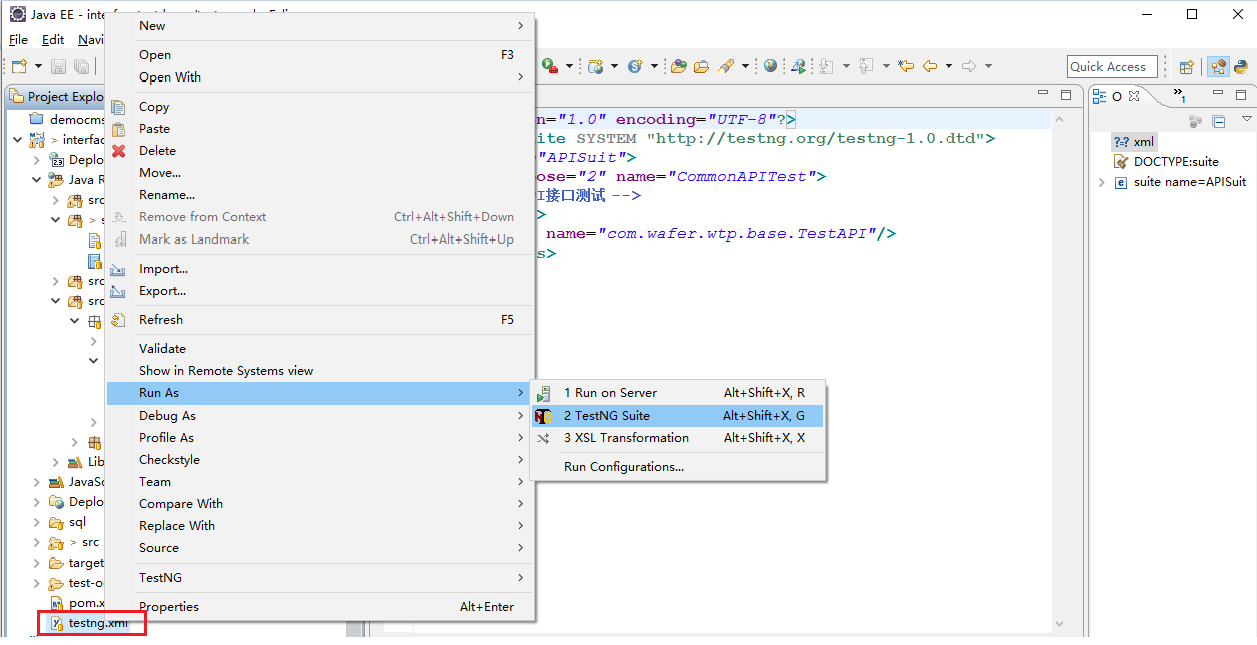
项目管理接口测试数据可以维护在TestAPI\_ProjectManage.xls；

……

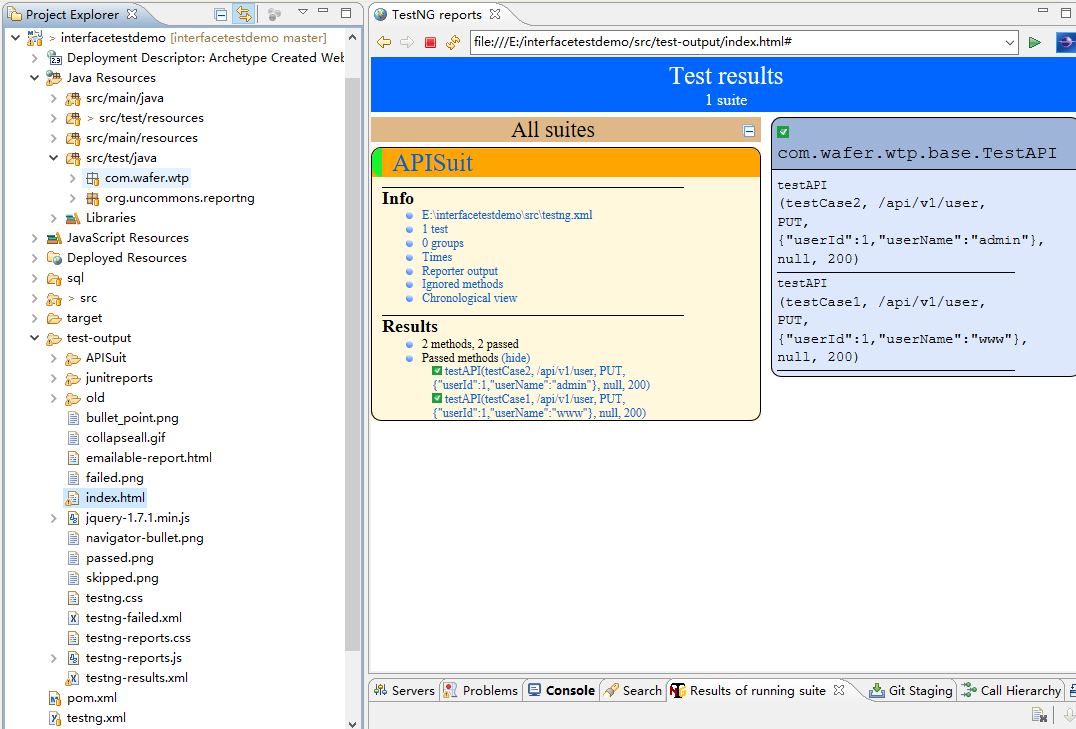
注意:所有Excel数据文件必须以TestAPI开头，以. xls结尾。

### 3．用例执行

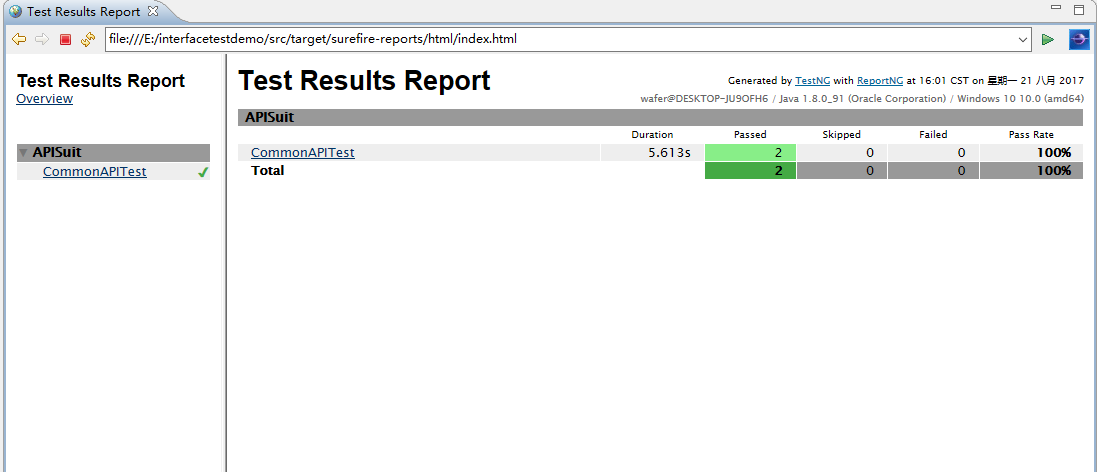
执行方式一：在根目录下有testng.xml。右键testng.xml 🡪 Run As(Debug As) 🡪 TestNG Suit执行测试数据。



测试执行完成之后，会生成相应的测试报告，报告在test-output目录下的index.html。



执行方式二：在根目录下执行mvn test命令(需要执行初始化脚本时添加 –Dinit=true参数)执行测试。执行完成之后会在target（target/ surefire-reports/ html/index.html）目录下生成测试报告。



# SpringMVC环境集成TestNG

## 准备工作

开发环境Eclipse安装TestNG插件和Maven插件。

## 代码集成

1. 拷贝src/test/resources和src/test/java下的文件到项目对应的目录。
2. [修改src/test/java下com.wafer.wtp.base.TestAPI类](mailto:修改src/test/java下com.wafer.wtp.base.TestAPI的类注解@SpringBootTest)的 @ ContextConfiguration注解，配置项目Spring环境，如: @ContextConfiguration("classpath\*:/applicationContext.xml")。
3. 修改src/test/java下com.wafer.wtp.base.AbstractTestNGTest类中变量值。

DP\_DATA\_TYPE:接口测试数据源类型(rest,json,excel)

DP\_REST\_PATH:数据类型为rest时，数据来源地址(http://localhost:8000/api/v1/dss/1)

1. 拷贝testng.xml到项目根路径下。
2. 修改pom.xml文件，添加对应依赖和Maven配置。

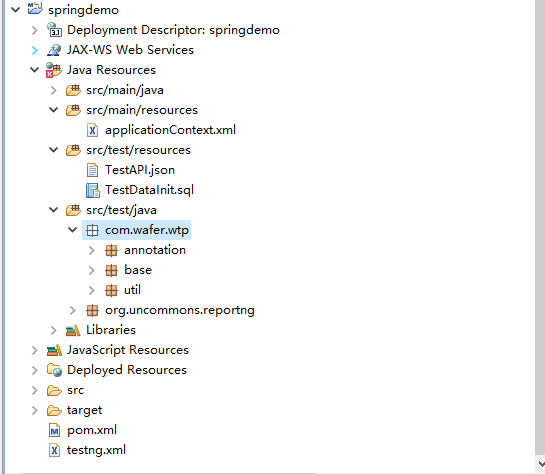
添加依赖如下：

|  |
| --- |
| <!-- TestNG相关依赖 -->  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-test</artifactId>  <version>4.3.10.RELEASE</version>  <scope>test</scope>  </dependency>    <dependency>  <groupId>net.sf.json-lib</groupId>  <artifactId>json-lib</artifactId>  <version>2.4</version>  <classifier>jdk15</classifier>  </dependency>  <dependency>  <groupId>javax.transaction</groupId>  <artifactId>javax.transaction-api</artifactId>  <version>1.2</version>  </dependency>  <dependency>  <groupId>junit</groupId>  <artifactId>junit</artifactId>  <version>4.12</version>  <type>jar</type>  <scope>test</scope>  </dependency>  <dependency>  <groupId>org.mockito</groupId>  <artifactId>mockito-core</artifactId>  <version>1.10.19</version>  <scope>test</scope>  </dependency>  <dependency>  <groupId>org.hamcrest</groupId>  <artifactId>hamcrest-core</artifactId>  <version>1.3</version>  <scope>test</scope>  </dependency>  <dependency>  <groupId>org.hamcrest</groupId>  <artifactId>hamcrest-library</artifactId>  <version>1.3</version>  <scope>test</scope>  </dependency>  <dependency>  <groupId>com.fasterxml.jackson.core</groupId>  <artifactId>jackson-core</artifactId>  <version>2.8.9</version>  </dependency>    <dependency>  <groupId>com.fasterxml.jackson.core</groupId>  <artifactId>jackson-databind</artifactId>  <version>2.8.9</version>  </dependency>    <dependency>  <groupId>com.fasterxml.jackson.core</groupId>  <artifactId>jackson-annotations</artifactId>  <version>2.8.9</version>  </dependency>    <dependency>  <groupId>org.skyscreamer</groupId>  <artifactId>jsonassert</artifactId>  <version>1.4.0</version>  <scope>test</scope>  </dependency>    <!-- 添加testNG依赖 -->  <dependency>  <groupId>org.testng</groupId>  <artifactId>testng</artifactId>  <version>6.9.10</version>  <scope>test</scope>  </dependency>  <!-- 依赖reportNg 关联testNg -->  <dependency>  <groupId>org.uncommons</groupId>  <artifactId>reportng</artifactId>  <version>1.1.4</version>  <scope>test</scope>  <exclusions>  <exclusion>  <groupId>org.testng</groupId>  <artifactId>testng</artifactId>  </exclusion>  </exclusions>  </dependency>  <!-- 依赖Guice -->  <dependency>  <groupId>com.google.inject</groupId>  <artifactId>guice</artifactId>  <version>4.0</version>  <scope>test</scope>  </dependency>  <!-- 依赖velocity -->  <dependency>  <groupId>velocity</groupId>  <artifactId>velocity-dep</artifactId>  <version>1.4</version>  </dependency>  <!-- 依赖poi读取Excel数据 -->  <dependency>  <groupId>org.apache.poi</groupId>  <artifactId>poi</artifactId>  <version>3.14</version>  </dependency> |

添加Maven配置如下：

|  |
| --- |
| <build>  <plugins>  <plugin>  <groupId>org.apache.maven.plugins</groupId>  <artifactId>maven-compiler-plugin</artifactId>  <configuration>  <source>1.8</source>  <target>1.8</target>  </configuration>  </plugin>  <!-- 运行项目中src/test/Java文件夹下的testcase -->  <plugin>  <groupId>org.apache.maven.plugins</groupId>  <artifactId>maven-surefire-plugin</artifactId>  <configuration>  <!-- 不对xls进行转码 -->  <properties>  <property>  <name>usedefaultlisteners</name>  <value>false</value>  </property>  <property>  <name>listener</name>  <value>org.uncommons.reportng.HTMLReporter, org.uncommons.reportng.JUnitXMLReporter</value>  </property>  </properties>  <suiteXmlFiles>  <suiteXmlFile>testng.xml</suiteXmlFile>  </suiteXmlFiles>  <!-- 系统参数默认不初始化数据 mvn test -DinitData=true -->  <systemPropertyVariables>  <init>false</init>  <filePath>classpath:TestDataInit.sql</filePath>  </systemPropertyVariables>  </configuration>  </plugin>  </plugins>  <testResources>  <testResource>  <directory>src/test/java</directory>  <includes>  <include>\*\*/\*.xls</include>  <include>\*\*/\*.json</include>  </includes>  <filtering>false</filtering>  </testResource>  <testResource>  <directory>src/test/resources</directory>  <includes>  <include>\*\*/\*.xls</include>  <include>\*\*/\*.json</include>  </includes>  <filtering>false</filtering>  </testResource>  </testResources>  </build> |

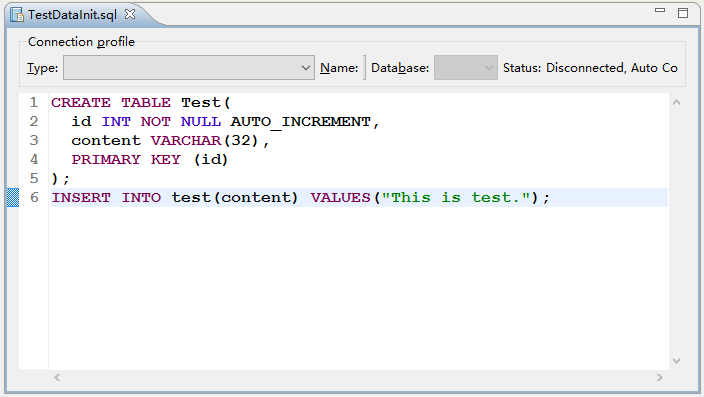
代码集成之后的工程目录如下所示:



## 接口测试

### 1．测试数据初始化

在src/test/resources目录中的TestDataInit.sql文件中维护数据库初始化sql脚本，如果需要执行初始化脚本，在mvn test命令后添加 -Dinit=true 参数即可执行。

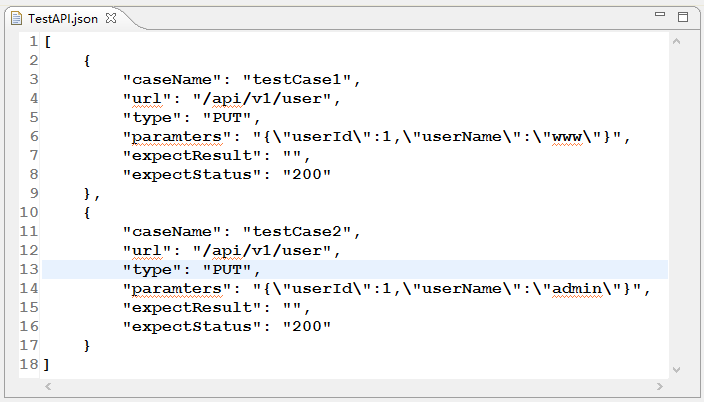


### 2．测试接口数据准备

测试接口数据支持json文件和Excel文件两种方式，两种方式只能选择一种。

#### JSON数据维护

在src/test/resources目录中的TestAPI.json维护接口测试数据。



1. caseName：测试用例的名称
2. url：接口请求地址
3. type：请求类型（GET,POST,PUT,DELETE）
4. paramters：请求需要的参数，如果请求没有参数，则不填
5. exceptResult：请求期望返回的结果，如果不需要校验返回结果，则不填
6. exceptStatus：期望请求执行的状态，如果不需要校验请求状态，则不填

为方便维护数据，接口数据文件可以根据模块划分。例如：

用户管理接口测试数据可以维护在TestAPI\_UserManage.json；

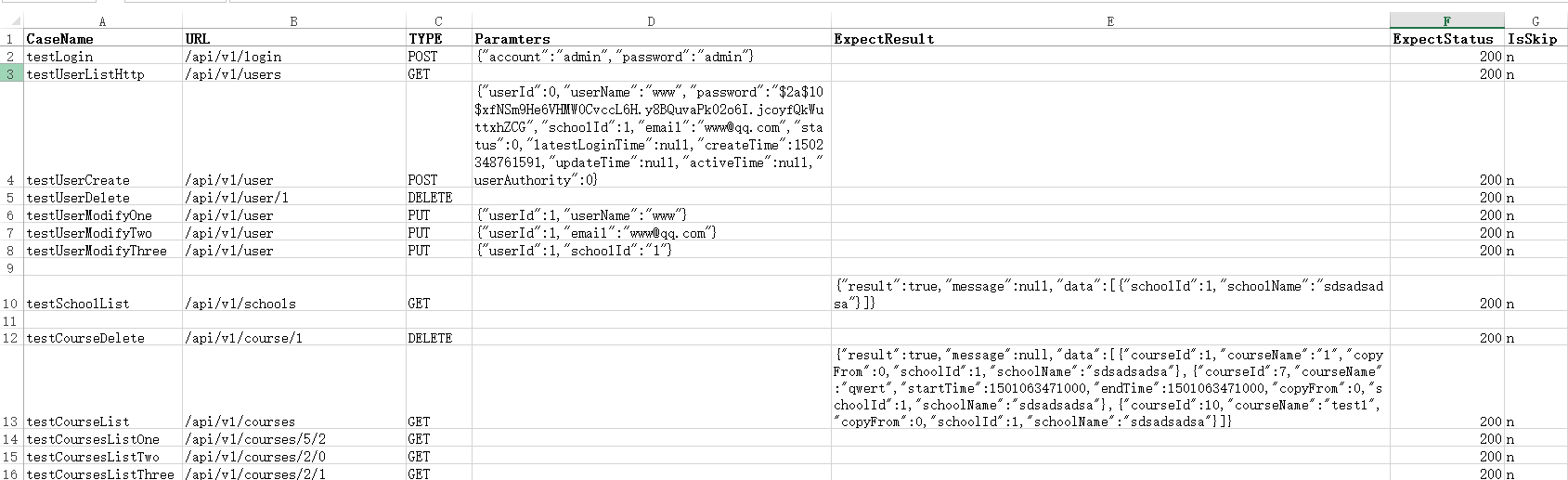
项目管理接口测试数据可以维护在TestAPI\_ProjectManage.json；

……

注意:所有jon数据文件必须以TestAPI开头，以.json结尾。

#### Excel数据维护

在src/test/resources目录中的TestAPI.xls维护接口测试数据。



1. CaseName：测试用例的名称
2. URL：接口请求地址
3. TYPE：请求类型（GET,POST,PUT,DELETE）
4. Paramters：请求需要的参数，如果请求没有参数，则不填
5. ExceptResult：请求期望返回的结果，如果不需要校验返回结果，则不填
6. ExceptStatus：期望请求执行的状态，如果不需要校验请求状态，则不填
7. IsSkip:标记这条Case是否跳过(不执行)（y,n）

为方便维护数据，接口数据文件可以根据模块划分。例如：

用户管理接口测试数据可以维护在TestAPI\_UserManage.xls；

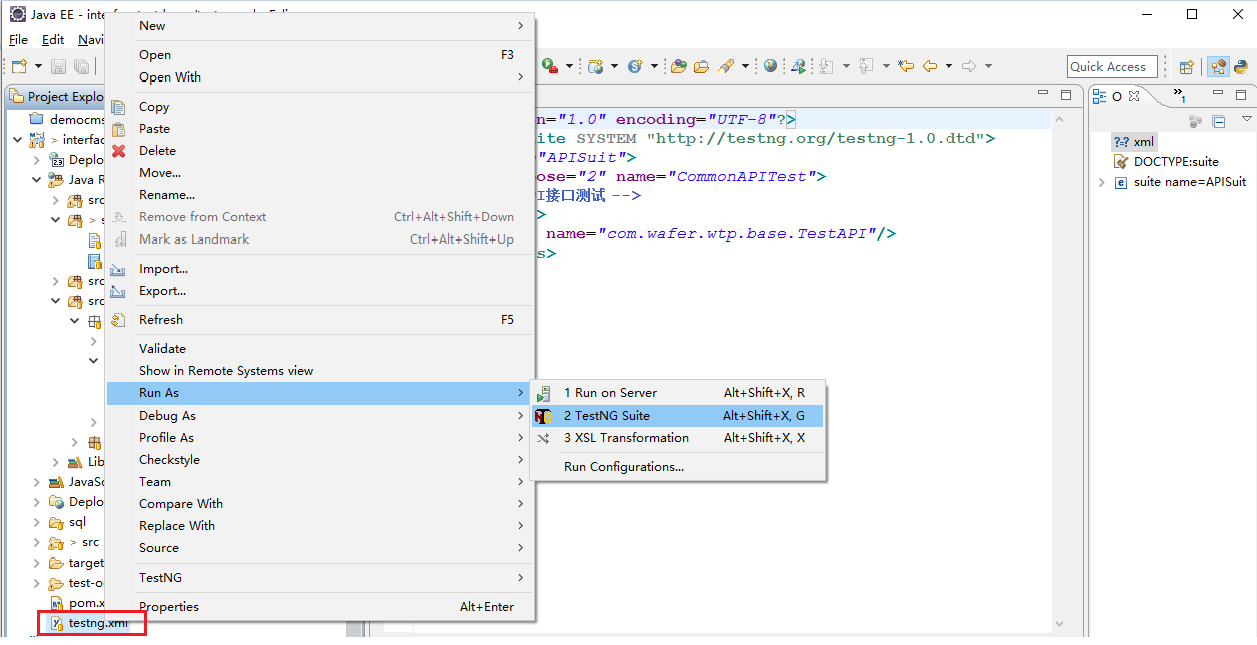
项目管理接口测试数据可以维护在TestAPI\_ProjectManage.xls；

……

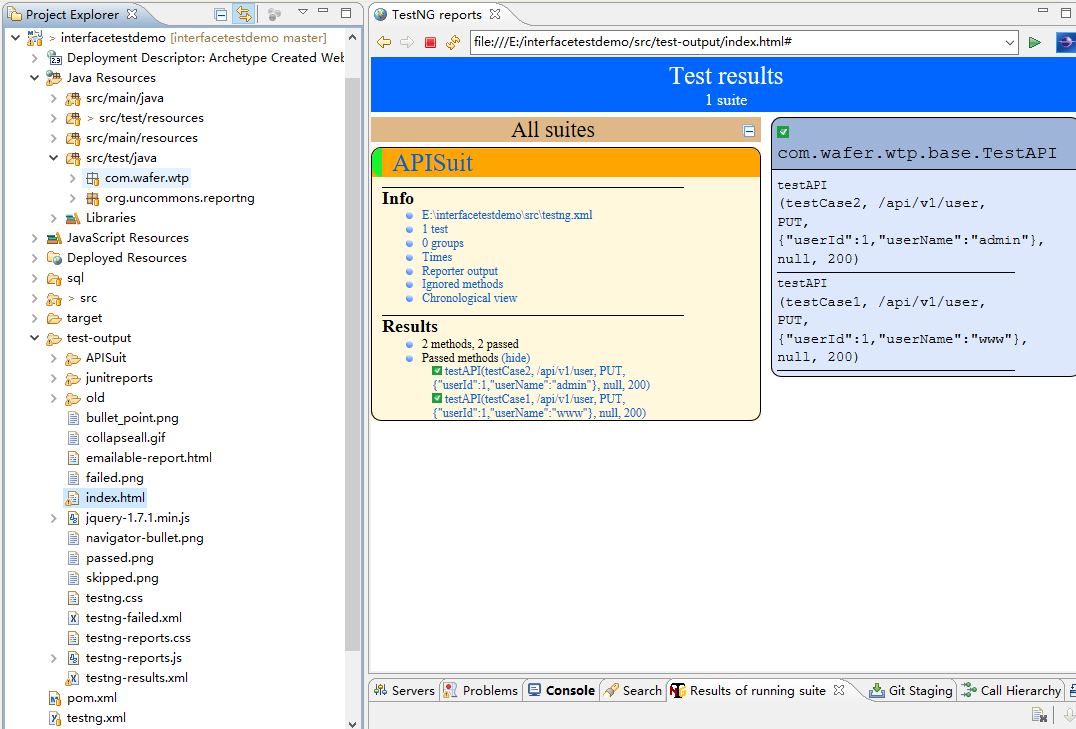
注意:所有Excel数据文件必须以TestAPI开头，以. xls结尾。

### 3．用例执行

执行方式一：在根目录下有testng.xml。右键testng.xml 🡪 Run As(Debug As) 🡪 TestNG Suit执行测试数据。



测试执行完成之后，会生成相应的测试报告，报告在test-output目录下的index.html。



执行方式二：在根目录下执行mvn test命令(需要执行初始化脚本时添加 –Dinit=true参数)执行测试。执行完成之后会在target（target/ surefire-reports/ html/index.html）目录下生成测试报告。

