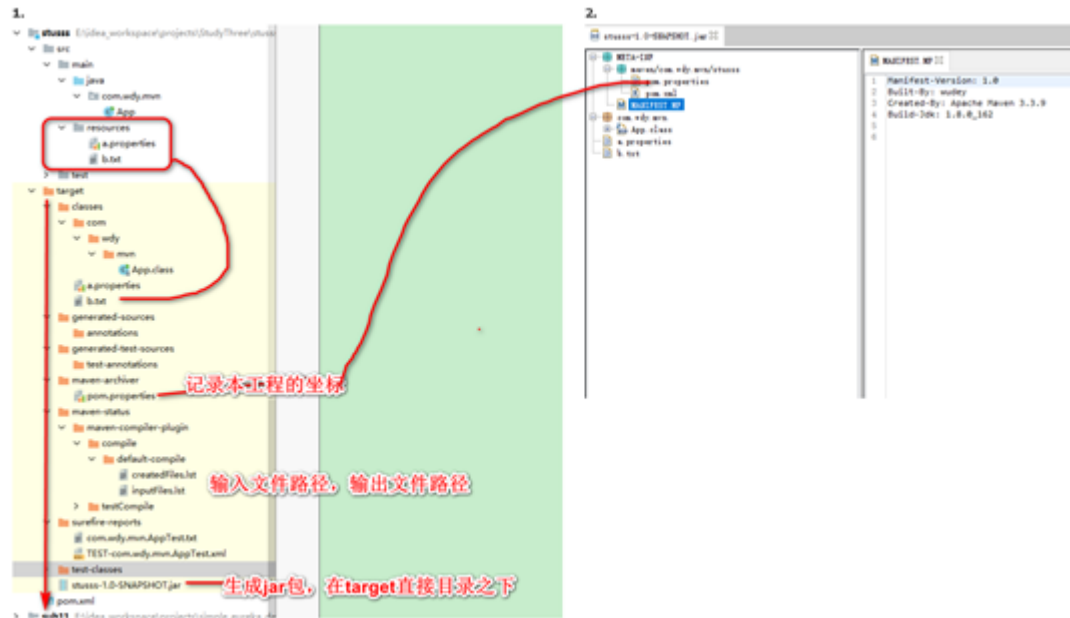


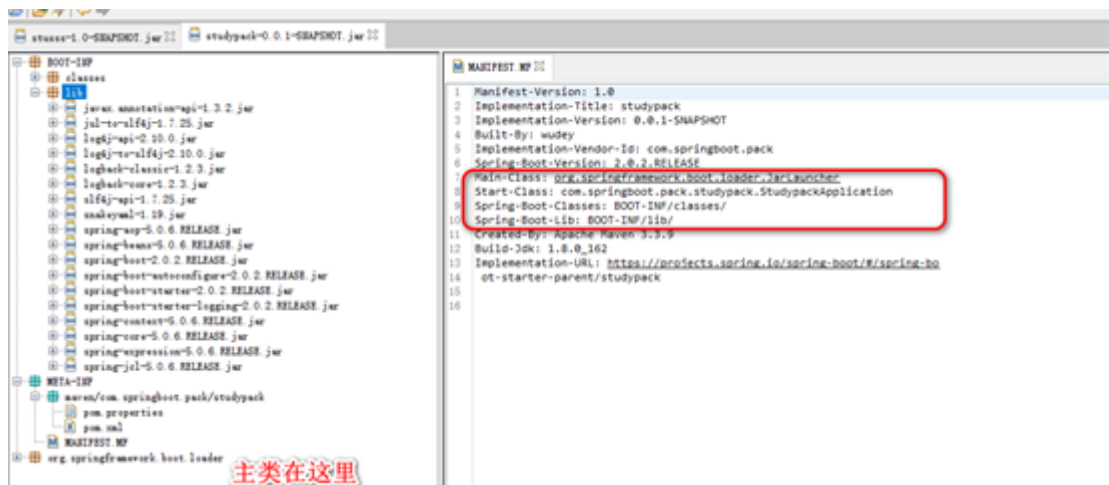
1. 普通maven工程

普通的maven工程，依赖lib在打包的时候，并不会加进来，只是将pom加了进来，要正常运行必须要有maven环境

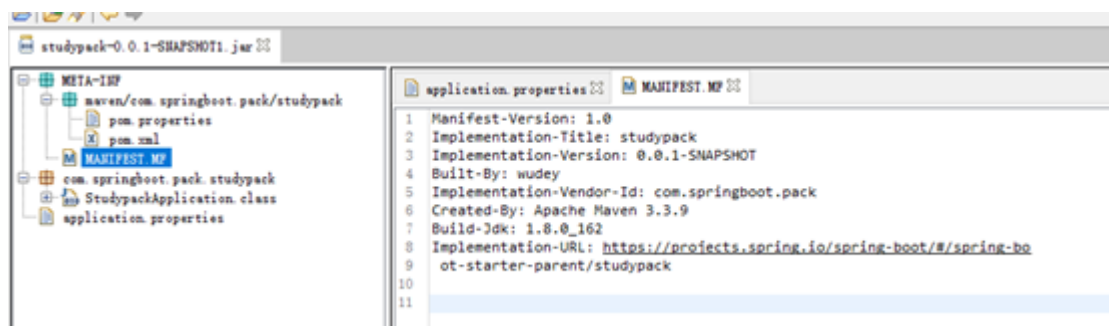


2. spring boot项目

Spring boot maven插件打的包，除了scope为provided的依赖全部进来

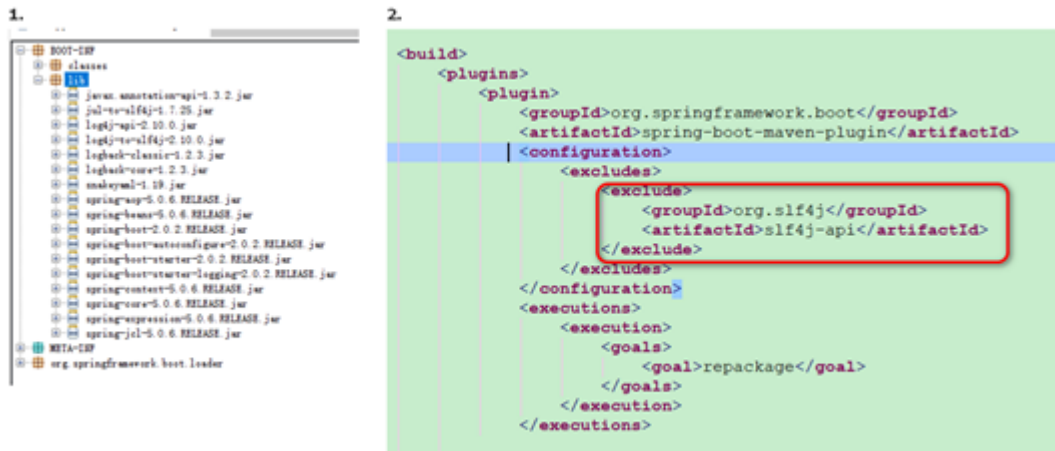


它的原始包，跟普通的maven包是一样的：



3. spring boot打包排除某个包

Spring boot打包时排除某个包，除了scope为provided的依赖全部进来



- 用artifactId排除



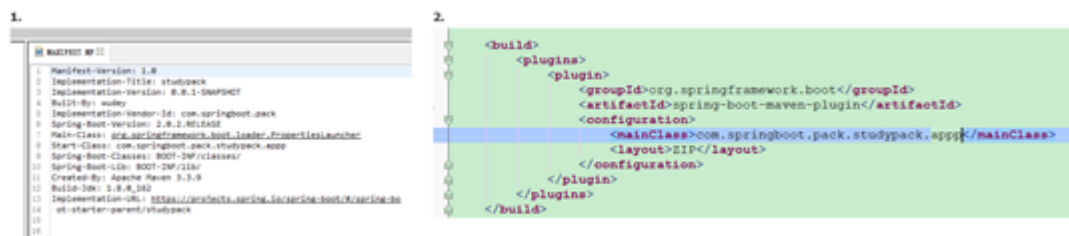
- 用groupId排除

```

<build>
    ...
    <plugins>
        ...
        <plugin>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-maven-plugin</artifactId>
            <version>1.5.6.RELEASE</version>
            <configuration>
                <excludeGroupIds>com.foo</excludeGroupIds>
            </configuration>
        </plugin>
        ...
    </plugins>

```

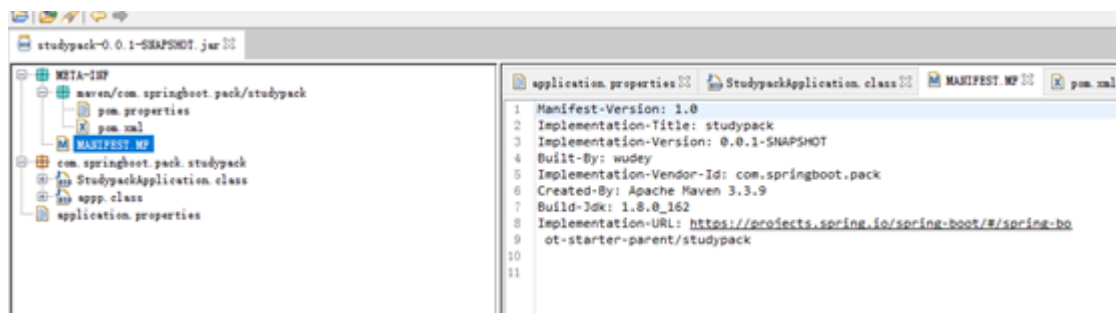
4. 重写manifest



包的布局:

- JAR: regular executable JAR layout.
- WAR: executable WAR layout. provided dependencies are placed in WEB-INF/lib-provided to avoid any clash when the war is deployed in a servlet container.
- ZIP (alias to DIR): similar to the JAR layout using PropertiesLauncher.
- MODULE: Bundle dependencies (excluding those with provided scope) and project resources. Does not bundle a bootstrap loader.
- NONE: Bundle all dependencies and project resources. Does not bundle a bootstrap loader.

只打工程包，依赖不要



这个跟直接用maven插件差不多。

5. 打包全部:

```

<build>
  <plugins>
    <plugin>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-maven-plugin</artifactId>
      <executions>
        <execution>
          <goals>
            <goal>repackage</goal>
          </goals>
        </execution>
      </executions>
    </plugin>
  </plugins>
</build>

```

6. maven-resources-plugin插件打包

这里面包含了该插件的两个目标resource和copy-resources

```

<build>
  # 指定资源文件位置
  <resources>
    <resource>
      <directory>src/main/resources</directory>
      <filtering>true</filtering> #开启正则匹配
      <includes>
        <include>**/*.properties</include>
      </includes>
    </resource>
    <resource>
      <directory>src/main/resources</directory>
      <filtering>false</filtering> #不开启正则匹配
      <includes>
        <include>**/*.xml</include>
      </includes>
    </resource>
  </resources>
  #引入插件
  <plugins>
    <plugin>
      <artifactId>maven-resources-plugin</artifactId>
      <version>3.1.0</version>
      #执行copy-resources目标
      <executions>
        <execution>
          <id>copy-resources</id>
          <!-- here the phase you need -->
          <phase>validate</phase>
          <goals>
            <goal>copy-resources</goal>
          </goals>
          <configuration>
            #copy-resources输出目录
            <outputDirectory>${basedir}/target/extra-resources</outputDirectory>
          </configuration>
        </execution>
      </executions>
    </plugin>
  </plugins>
</build>

```

```
        #copy-resources输入目录
    <resources>
        <resource>
            <directory>src/non-packaged-resources</directory>
            <filtering>true</filtering>
        </resource>
    </resources>
</configuration>
</execution>
</executions>
</plugin>
</plugins>
</build>
```