

云服务器 ECS

SDK 参考



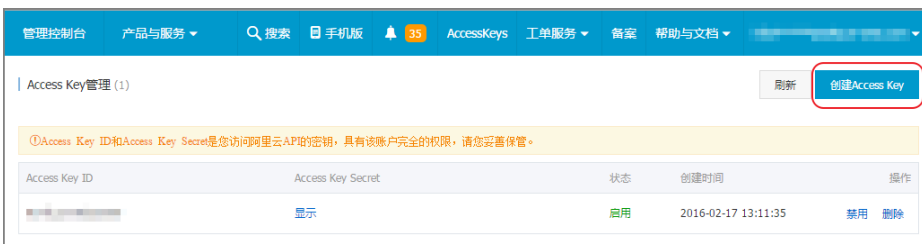
SDK 参考

创建 Access Key

登录 [云服务器管理控制台](#)。

单击页面顶部的 **Access Keys**，或直接单击 **Access Keys 管理控制台**。

创建、管理您的 Access Key。



SDK

目前阿里云官方提供的 SDK 分为以下语言版本：

- Java
- Python
- PHP
- C#

更多语言版本的 SDK，您可以前往[第三方 SDK 服务](#)中进行选择。

相较于老版 JAVA SDK 直接开放下载，新版的阿里云 JAVA SDK 采用了Maven 的方式分发，阿里云旗下所有的 JAVA SDK 都统一在了一个Maven库中，方便统一管理。

下载 SDK 的步骤如下（以windows7 64位+Eclipse Luna为例）：

首先在Maven官方下载页面 (<http://maven.apache.org/download.cgi>) 上下载对应操作系统的Maven软件，后方的checksum可供您校验下载文件是否正确无误：

Files

Maven is distributed in several formats for your convenience. Simply pick a ready-made binary distribution archive and follow the installation instructions. Use a source archive if you intend to build Maven yourself.

In order to guard against corrupted downloads/installations, it is highly recommended to verify the signature of the release bundles against the public KEYS used by the Apache Maven developers.

	Link	Checksum	Signature
Binary tar.gz archive	apache-maven-3.3.3-bin.tar.gz	apache-maven-3.3.3-bin.tar.gz.md5	apache-maven-3.3.3-bin.tar.gz.asc
Binary zip archive	apache-maven-3.3.3-bin.zip	apache-maven-3.3.3-bin.zip.md5	apache-maven-3.3.3-bin.zip.asc
Source tar.gz archive	apache-maven-3.3.3-src.tar.gz	apache-maven-3.3.3-src.tar.gz.md5	apache-maven-3.3.3-src.tar.gz.asc
Source zip archive	apache-maven-3.3.3-src.zip	apache-maven-3.3.3-src.zip.md5	apache-maven-3.3.3-src.zip.asc

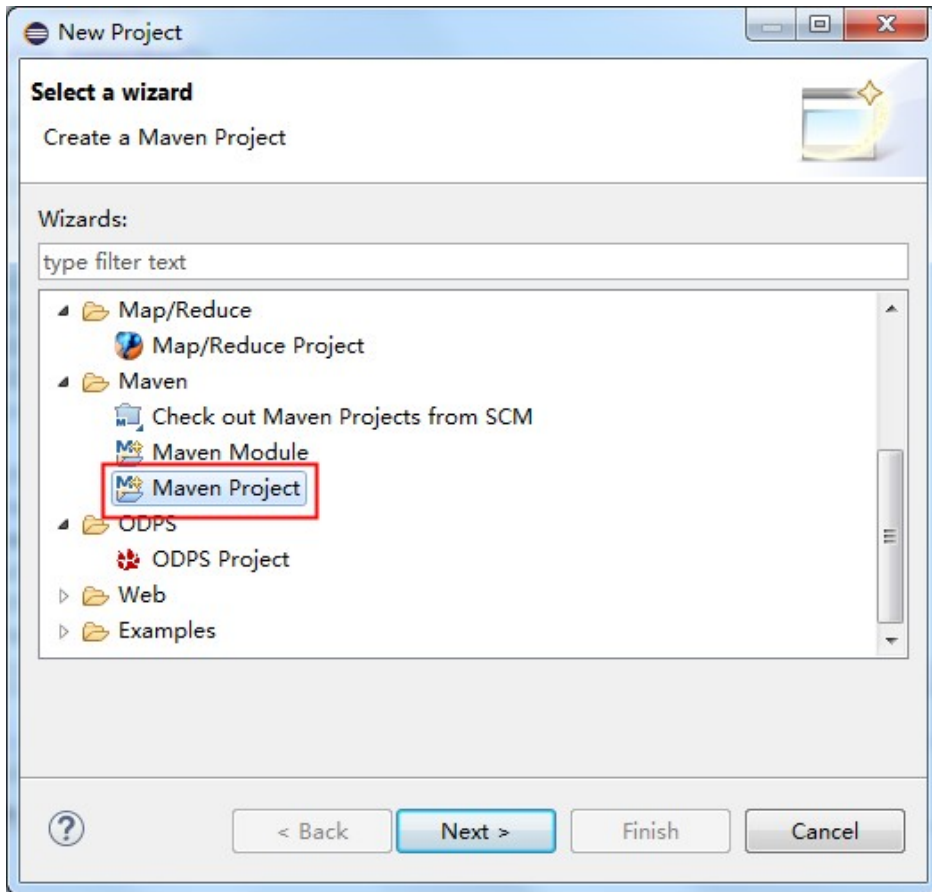
打开JAVA SDK的下载页面：<http://develop.aliyun.com/sdk/java>，将阿里云的SDK存放的Maven库加入到Maven软件中。找到之前下载的Maven压缩包，解压，将阿里云的maven库信息添加到conf文件夹下的settings.xml中：

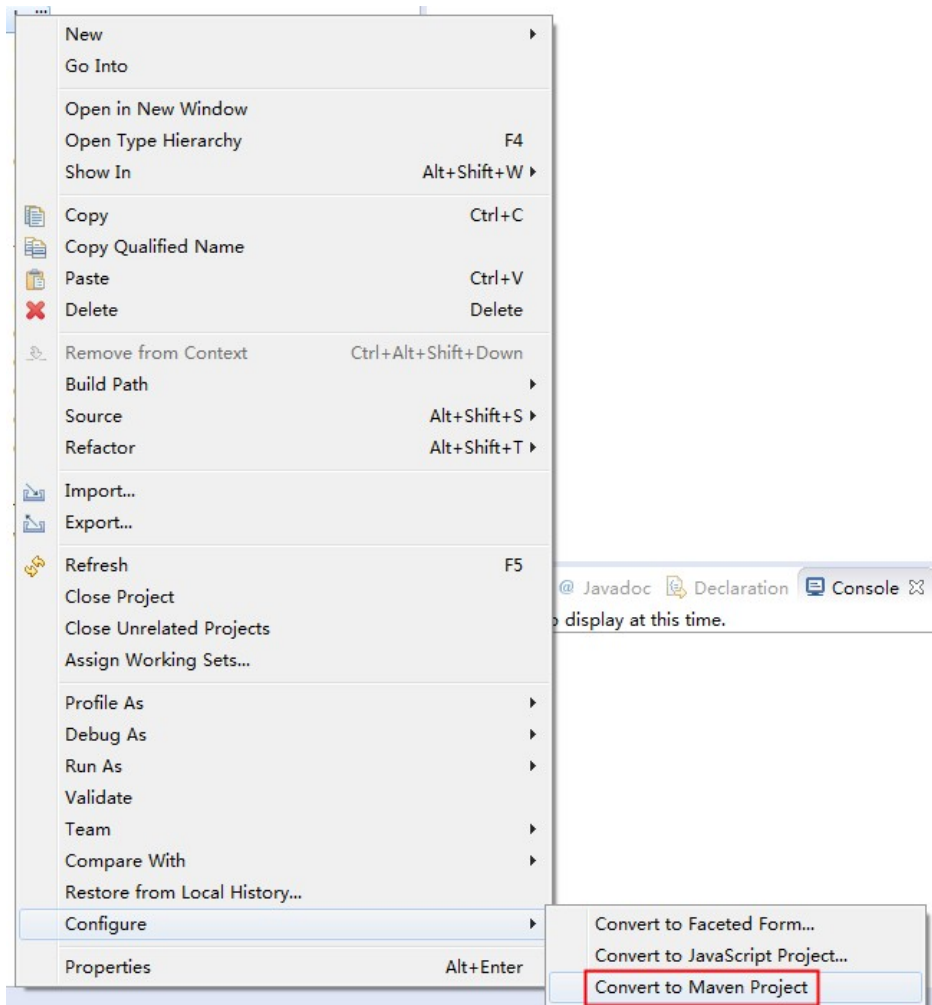
```

237 | | <tomcatLocation> ${tomcatPath} </tomcatLocation>
238 | | </configuration>
239 | | </plugin>
240 | | ...
241 | |
242 | | NOTE: If you just wanted to inject this configuration whenever someone set 'target-env' to
243 | | anything, you could just leave off the <value/> inside the activation-property.
244 | |
245 | | -->
246 | | <profile>
247 | |   <id>nexus</id>
248 | |
249 | |   <repositories>
250 | |     <repository>
251 | |       <id>sonatype-nexus-staging</id>
252 | |       <name>Sonatype Nexus Staging</name>
253 | |       <url>https://oss.sonatype.org/service/local/staging/deploy/maven2/</url>
254 | |       <releases>
255 | |         <enabled>true</enabled>
256 | |       </releases>
257 | |       <snapshots>
258 | |         <enabled>true</enabled>
259 | |       </snapshots>
260 | |     </repository>
261 | |   </repositories>
262 | |

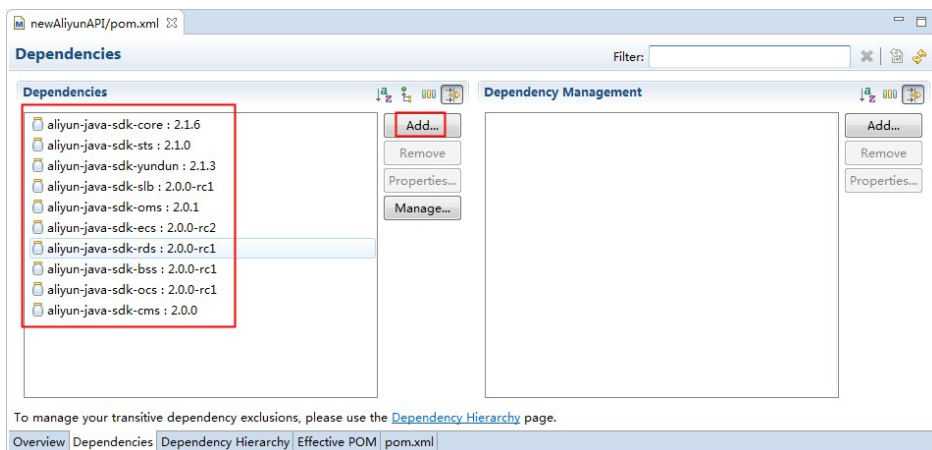
```

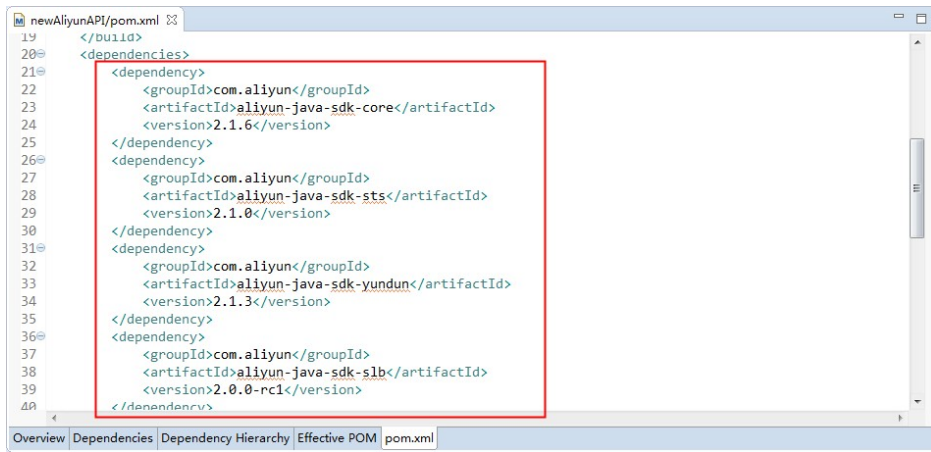
在Eclipse添加一个Maven项目或将已有的项目的项目转换为Maven项目：





打开项目下的pom.xml文件，将Maven的dependency加入到其中，可以通过图形化界面也可以直接编辑pom.xml文件进行添加：

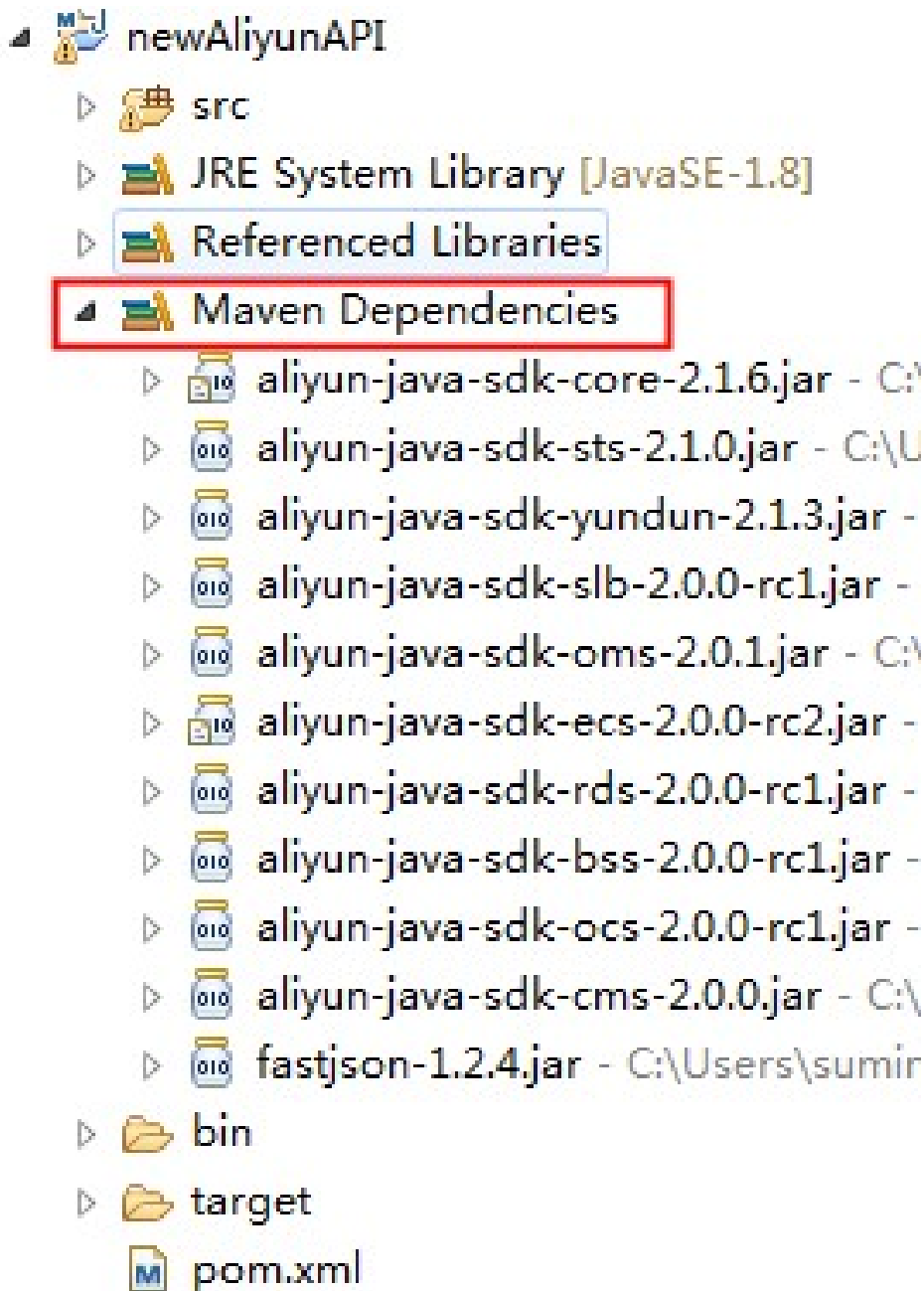




```

newAliyunAPI/pom.xml
19 </build>
20 <dependencies>
21 <dependency>
22 <groupId>com.aliyun</groupId>
23 <artifactId>aliyun-java-sdk-core</artifactId>
24 <version>2.1.6</version>
25 </dependency>
26 <dependency>
27 <groupId>com.aliyun</groupId>
28 <artifactId>aliyun-java-sdk-sts</artifactId>
29 <version>2.1.0</version>
30 </dependency>
31 <dependency>
32 <groupId>com.aliyun</groupId>
33 <artifactId>aliyun-java-sdk-yundun</artifactId>
34 <version>2.1.3</version>
35 </dependency>
36 <dependency>
37 <groupId>com.aliyun</groupId>
38 <artifactId>aliyun-java-sdk-slb</artifactId>
39 <version>2.0.0-rc1</version>
40 </dependencies>
    
```

保存之后就可以看到在项目下的Maven Dependencies中自动下载并加入了阿里云的SDK jar包了



快速判断老版和新版SDK的方法

可以通过下表中的SDK参数快速识别所使用的是老版还是新版SDK。

分辨方法：

对比项	旧版SDK	新版SDK
提交操作请求的方法	execute()	getAcsResponse()
存放Access Key和Access Key Secret的类	AliyunClient	IClientProfile
生成存放凭据对象的方法	new DefaultAliyunClient(APIUrl,	DefaultProfile.getProfile(Regi onId, Access Key, Access Key

	Access Key, Access Key Secret)	Secret)
包名前缀	com.aliyun.api	com.aliyuncs

通过以上几点可以快速的分辨程序中调用的是老版还是新版的SDK。若您使用的是老版SDK，建议您尽快切换到新版的SDK上，以便获得最新功能。