Java Protobuf

# 介绍Protocol Buffers

Protocal Buffers(简称protobuf)是谷歌的一项技术，用于结构化的数据序列化、反序列化，常用于RPC 系统（Remote Procedure Call Protocol System）和持续数据存储系统。

其类似于XML生成和解析，但protobuf的效率高于XML，不过protobuf生成的是字节码，可读性比XML差，类似的还有json、Java的Serializable等。

很适合做数据存储或 RPC 数据交换格式。可用于通讯协议、数据存储等领域的语言无关、平台无关、可扩展的序列化结构数据格式。

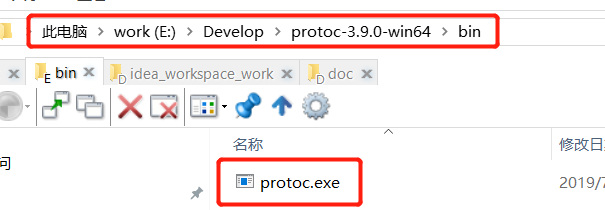
# Protobuf下载

下载地址 https://github.com/protocolbuffers/protobuf/releases/tag/v3.9.0。下载 protobuf-all-3.9.0.zip 和 protoc-3.9.0-win64.zip。



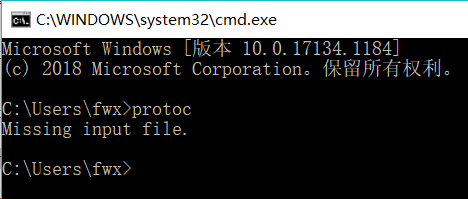
# Protobuf软件安装

1. 解压protoc-3.9.0-win64.zip，找到protoc.exe文件



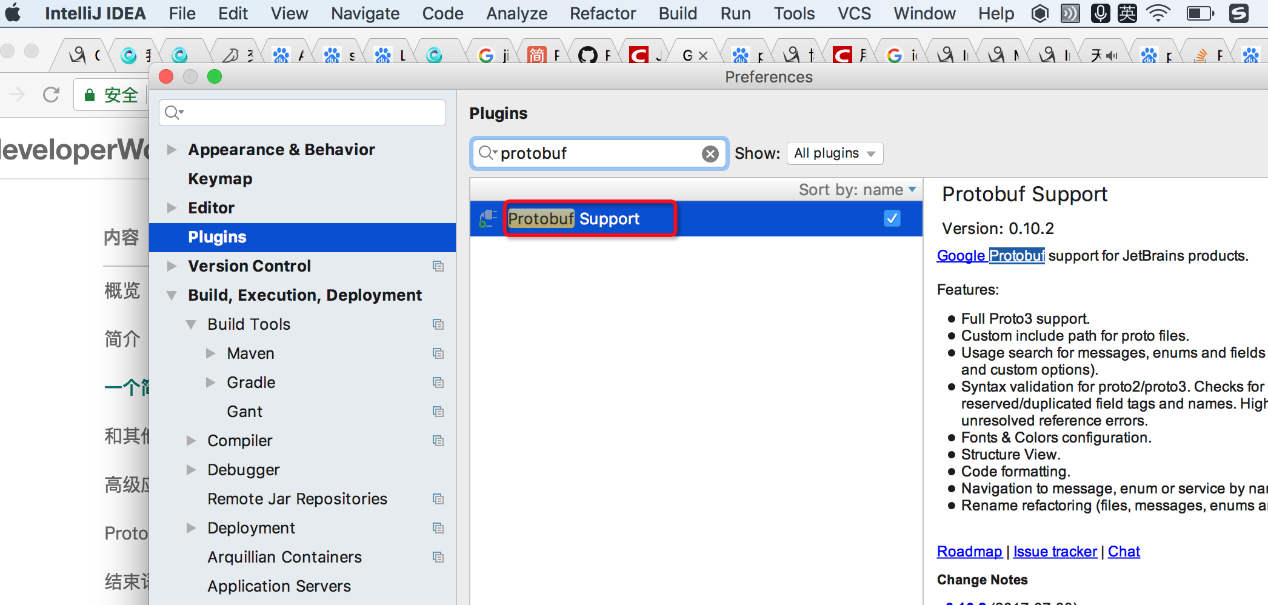
1. 配置protoc环境变量
2. PROTOBUF\_HOME: E:\Develop\protoc-3.9.0-win64
3. Path:%PROTOBUF\_HOME%\bin;
4. Protoc安装检验

在命令行窗口中输入protoc，产生如下结果，代表环境变量配置成功。



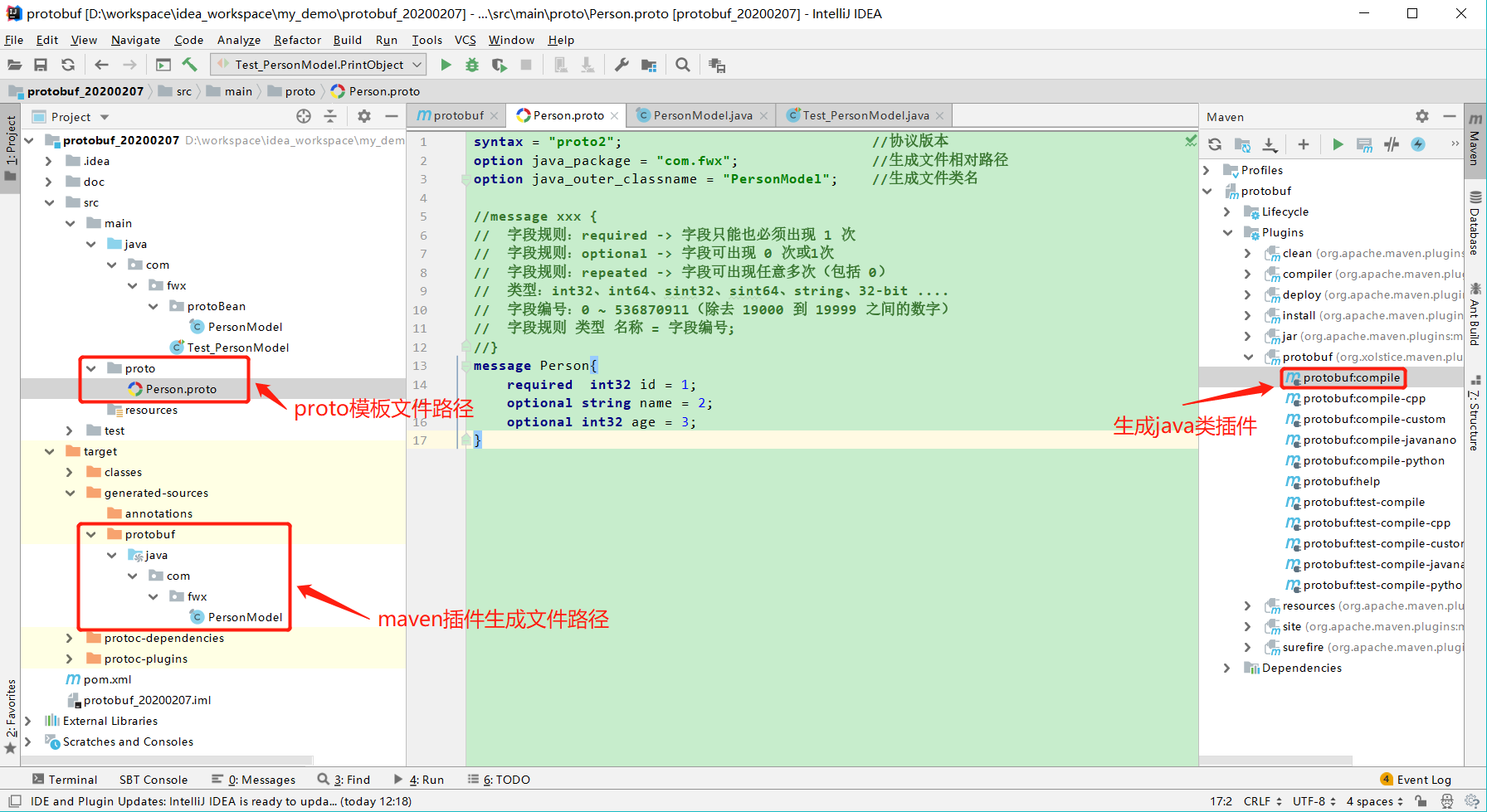
# Idea 安装protobuf插件

安装插件protobuf Support，之后重启



# 创建Maven项目，配置依赖

1. 项目目录



1. pom.xml添加

|  |
| --- |
| *<?***xml version="1.0" encoding="UTF-8"***?>* <**project xmlns="http://maven.apache.org/POM/4.0.0"  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd"**>  <**modelVersion**>4.0.0</**modelVersion**>   <**groupId**>com.fwx</**groupId**>  <**artifactId**>protobuf</**artifactId**>  <**version**>1.0-SNAPSHOT</**version**>   <**dependencies**>  **<!-- https://mvnrepository.com/artifact/com.google.protobuf/protobuf-java -->** <**dependency**>  <**groupId**>com.google.protobuf</**groupId**>  <**artifactId**>protobuf-java</**artifactId**>  <**version**>3.9.0</**version**>  </**dependency**>   <**dependency**>  <**groupId**>junit</**groupId**>  <**artifactId**>junit</**artifactId**>  <**version**>4.12</**version**>  <**scope**>compile</**scope**>  </**dependency**>   **<!-- https://mvnrepository.com/artifact/com.google.protobuf/protobuf-java-util -->  <!-- 可以将protobuf数据转换成json数据 -->** <**dependency**>  <**groupId**>com.google.protobuf</**groupId**>  <**artifactId**>protobuf-java-util</**artifactId**>  <**version**>3.6.1</**version**>  </**dependency**>  </**dependencies**>   <**build**>  <**extensions**>  <**extension**>  <**groupId**>kr.motd.maven</**groupId**>  <**artifactId**>os-maven-plugin</**artifactId**>  <**version**>1.4.1.Final</**version**>  </**extension**>  </**extensions**>   **<!-- protobuf maven插件，可以用来对.proto文件编译生成java文件 -->** <**plugins**>  <**plugin**>  <**groupId**>org.xolstice.maven.plugins</**groupId**>  <**artifactId**>protobuf-maven-plugin</**artifactId**>  <**version**>0.5.0</**version**>  <**configuration**>  <**protocArtifact**>  com.google.protobuf:protoc:3.1.0:exe:${os.detected.classifier}  </**protocArtifact**>  <**pluginId**>grpc-java</**pluginId**>  </**configuration**>  <**executions**>  <**execution**>  <**goals**>  <**goal**>compile</**goal**>  <**goal**>compile-custom</**goal**>  </**goals**>  </**execution**>  </**executions**>  </**plugin**>  </**plugins**>  </**build**> </**project**> |

# 编写.proto模板文件

|  |
| --- |
| syntax = **"proto2"**; **//协议版本** option java\_package = **"com.fwx"**; **//生成文件相对路径** option java\_outer\_classname = **"PersonModel"**; **//生成文件类名  //message xxx { // 字段规则：required -> 字段只能也必须出现 1 次 // 字段规则：optional -> 字段可出现 0 次或1次 // 字段规则：repeated -> 字段可出现任意多次（包括 0） // 类型：int32、int64、sint32、sint64、string、32-bit .... // 字段编号：0 ~ 536870911（除去 19000 到 19999 之间的数字） // 字段规则 类型 名称 = 字段编号; //}** message Person{  required int32 id = 1;  optional string name = 2;  optional int32 age = 3; } |

# 使用protoc对.proto文件生成java类

## 使用cmd命令生成

1. 进入.proto文件所在目录（例：D:\*\rc\main\proto）
2. 使用生成命令：protoc --java\_out=. Person.proto

java\_out：输出java类；

.（“.”）：输出路径为当前路径；

空格；

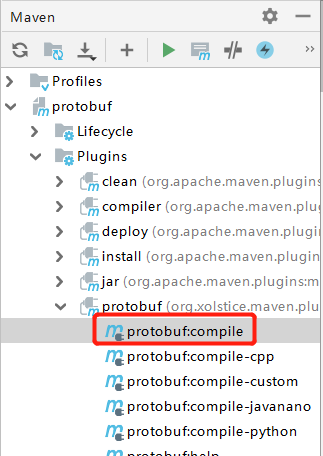
Person.proto：proto模板文件；

1. 检查生成的java文件（路径：D:\*\src\main\proto\com\fwx）

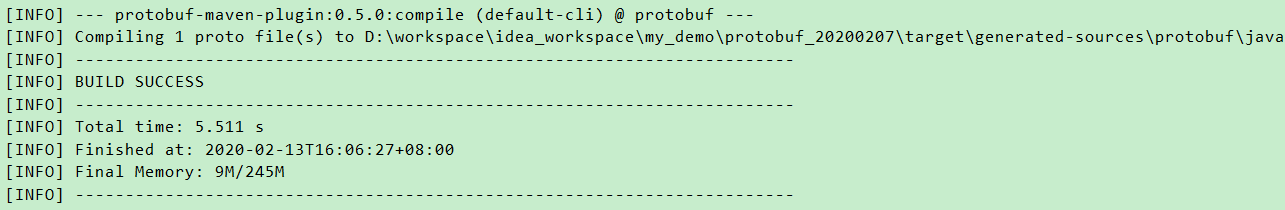


## 使用maven插件生成

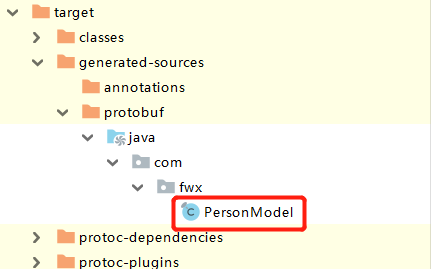
1. 双击protobuf:compile



1. 执行结果信息



1. 查看生成的文件



# 编写测试用例

1. 创建bean路径src\main\java\com\fwx\protoBean
2. 复制PersonModel.java至src\main\java\com\fwx\protoBean
3. 在com.fwx路径下创建测试类：Test\_PersonModel