

1.JAVA环境搭建

1.1 JAVA介绍

Java是由Sun公司于1995年5月推出的Java程序设计语言（以下简称Java语言）和Java平台的总称。

Java是一个完整的平台，不仅提供了优秀的编程语言，而且还提供了大量的可重用代码以及一个能提供安全性、可移植性、自动垃圾回收等系统服务的执行环境。



1992年Java的前身Oak问世，Oak最初的目的是面向家用电器的智能化。其开发团队的主要成员Bill Joy、Patrik Naughton、Mike Sheridan 和 James Gosling（Java之父）等。因为投标失败，Oak几乎无家可归，处境相当不乐观。后来在1994年该团队发现Oak非常适合于Internet，并且在得到Sun公司的支持后该团队决定进军Internet。因为Oak已被其它公司注册，这种新的语言的名字便改为Java。到1995年Java技术正式诞生，1996年第一个正式版本发布。



1998年12月，Java 1.2发布，随后产品的名称更改为“Java 2标准版软件开发工具箱1.2版”

Java编程语言的主要目标是：

提供一种解释环境，提高开发速度；

使用户能运行不止一个活动线程；

在程序运行时，能动态支持程序改变，提供更好的安全性。

更重要的是Java技术从诞生开始是一项免费技术。

Java语言是Sun公司推出的新一代面向对象程序设计语言，适合于平台无关性应用程序开发。今天的Java由于自身所特有的性能已经广泛的被使用2009年04月20日oracle以74亿美元收购Sun。

<https://www.oracle.com>

1.2 JDK,JRE,JVM介绍

JDK, Java SDK最早叫Java Software Develop Kit, 后来改名为JDK, 即Java Develop Kit。JDK作为Java开发工具包, 主要用于构建在Java平台上运行的应用程序、Applet 和组件等。包括: 类库、编译器、调试器、Java 运行时环境 (JRE)

JRE(Java Runtime Environment) 、Java运行环境, 用于解释执行Java的字节码文件。

JVM(Java Virtual Mechinal), Java虚拟机, 是JRE的一部分。它是整个java实现跨平台的最核心的部分, 负责解释执行字节码文件, 是可运行java字节码文件的虚拟计算机。所有平台的上的JVM向编译器提供相同的接口, 而编译器只需要面向虚拟机, 生成虚拟机能识别的代码, 然后由虚拟机来解释执行。



普通用户而只需要安装 JRE (Java Runtime Environment) 来运行 Java 程序。而程序开发者必须安装JDK来编译、调试程序。

1.3 JDK环境搭建

(1)安装jdk并复制jdk的目录:

C:\Program Files\Java\jdk1.8.0_181\bin

(2)配置计算机系统变量:

计算机->右键属性->高级->环境变量->系统变量

(3)系统变量的配置:

a:配置path:将jdk的路径放在该变量值的最前面, 然后加分号

注: 这里的路径有bin

b:在系统变量中新建变量:

变量名: JAVA_HOME

变量值: C:\Program Files\Java\jdk1.8.0_181

注: 这里没有bin目录

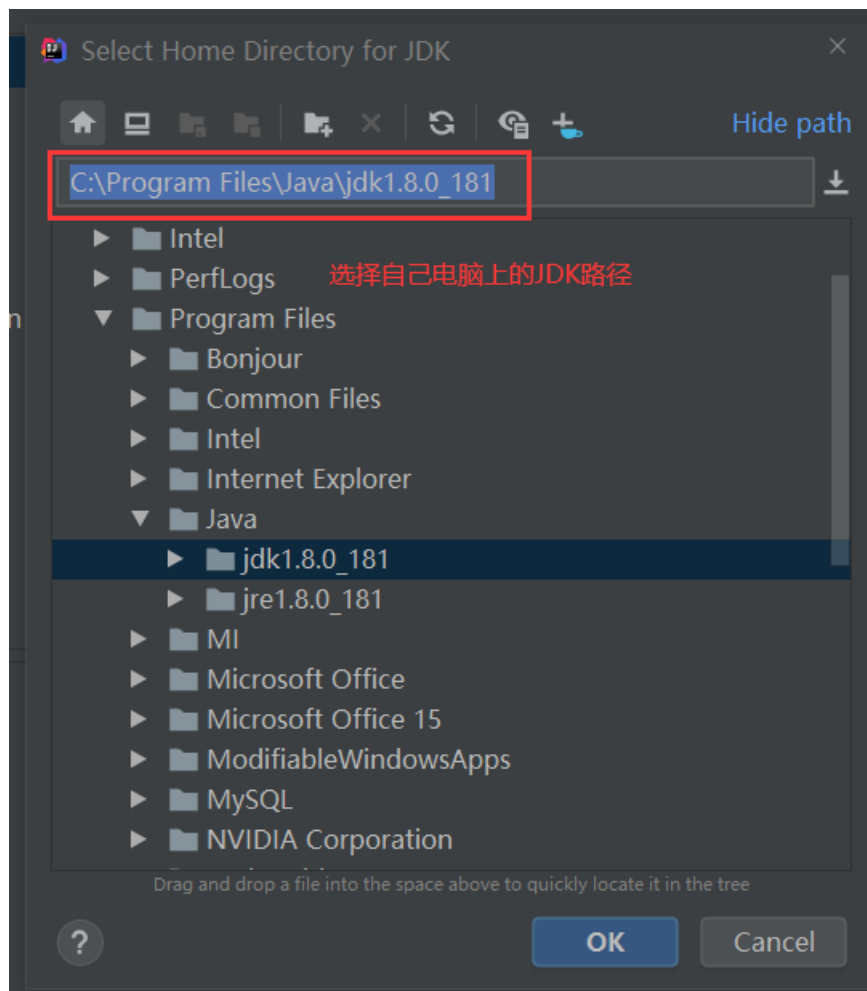
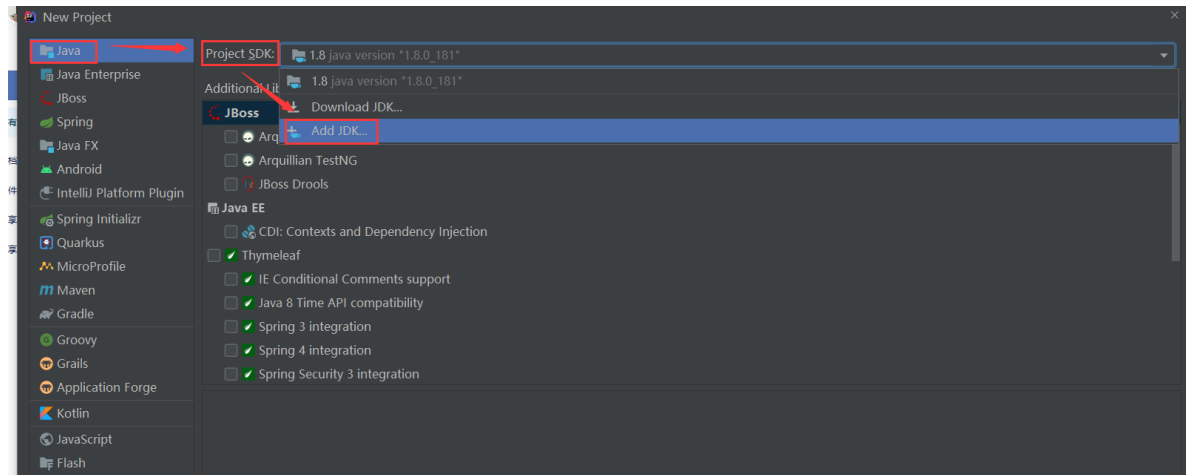
(4)测试:

打开cmd(开始->搜索框或 win+R->运行:cmd), 分别输入java -version进行测试

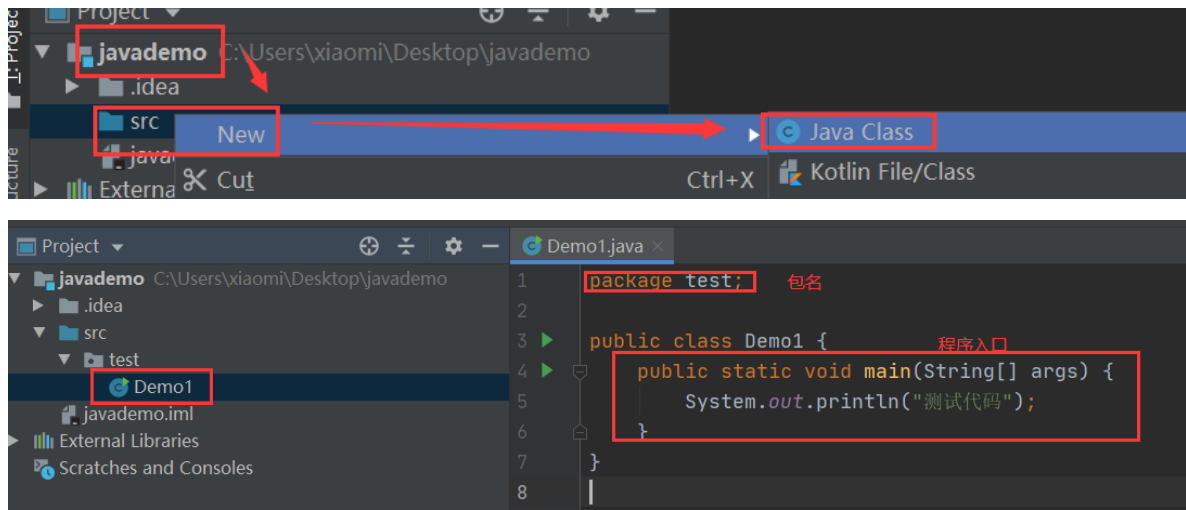
```
C:\Users\xiaomi>java -version
java version "1.8.0_181"
Java(TM) SE Runtime Environment (build 1.8.0_181-b13)
Java HotSpot(TM) 64-Bit Server VM (build 25.181-b13, mixed mode)
```

1.4 开发工具介绍与操作：IDEA

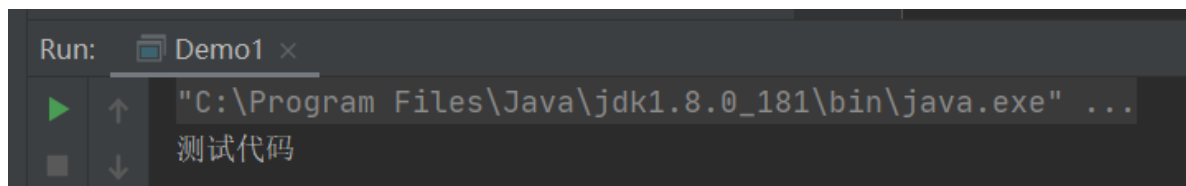
1.4.1 关联JDK



1.4.2 创建java项目，与测试类



测试效果:



1.5 变量定义

1.5.1 使用java代码实现自我介绍

```
public class Demo1 {  
    public static void main(String[] args) {  
        System.out.println("我叫袁海鹏");  
        System.out.println("我的工作是JAVA讲师");  
        System.out.println("我的爱好是:敲代码");  
    }  
}
```

1.5.2 变量的使用

需求:将上述效果变成动态数据

方案:使用数据类型-String 表示字符串, 非数字类型的都可以归类到字符串

语法:String 变量名="变量值" //变量名:数字, 字母, 下划线, \$,不能使用数字开头即可

代码增强:

```
public class Demo1 {  
    public static void main(String[] args) {  
        String name="袁海鹏";  
        String title="JAVA讲师";  
        String love="敲代码";  
        System.out.println("我叫"+name);  
        System.out.println("我的工作是"+title);  
        System.out.println("我的爱好是:"+love);  
    }  
}
```

补充:这里的加号表示拼接效果, 在变量与非变量之间存在

1.6 OOP

1.6.1 类与对象

(1)万物皆对象

(2) 对比

生活中	外在描述	功能性描述(能做的事情)
java	属性	方法

(3)类：将对象中公用的属性和方法提取出来，放在类中

类和对象的关系：

类是抽象的，对象是具体的，类是对象的集合。(类是具有相同属性或方法的对象的集合)

示例：学生--类，张三--对象

注：描述类时，不能定位到具体事物。

描述对象时，要定位到具体事物。

1.6.2 代码实现

(1)创建类

```
public class 类名{
    //1.1 定义属性，属性的定义语法和变量一样，只是称呼不同
    public 数据类型 属性名;
    //1.2 定义方法
    public void 方法名(参数){
        //方法体(要做的事情)
    }
}
```

(2)创建类的对象，**通过对象调用类中的属性和方法**

```
创建对象：类名 对象名=new 类名(); // 类名()->构造方法
调用属性：赋值：对象名.属性名=值；    取值：对象名.属性名
调用方法：对象名.方法名();
```

1.6.3 需求:使用面向对象的思想实现自我介绍

User.java

```
public class User {  
    public String name;  
    public String title;  
    public String love;  
    public void show(){  
        System.out.println("我叫"+name);  
        System.out.println("我的工作是"+title);  
        System.out.println("我的爱好是:"+love);  
    }  
}
```

测试类:




```
public class Demo1 {  
    public static void main(String[] args) {  
        User user = new User();  
        user.name="袁海鹏";  
        user.title="JAVA讲师";  
        user.love="敲代码";  
        user.show();  
    }  
}
```

2.jar包管理-Maven

2.1 jar包介绍

在软件领域，JAR文件（Java归档，英语：Java Archive）为一种软件包文件格式，通常用于聚合大量的Java类文件、相关的元数据和资源（文本、图片等）文件到一个文件，以便开发Java平台应用程序或库。

JAR文件为一种归档文件，以ZIP格式构建，以.jar为文件扩展名。用户可以使用JDK自带的jar命令创建或提取JAR文件。

 commons-pool.jar
 druid-1.1.21.jar
 mysql-connector-java-5.1.37-bin.jar

2.2 Maven介绍

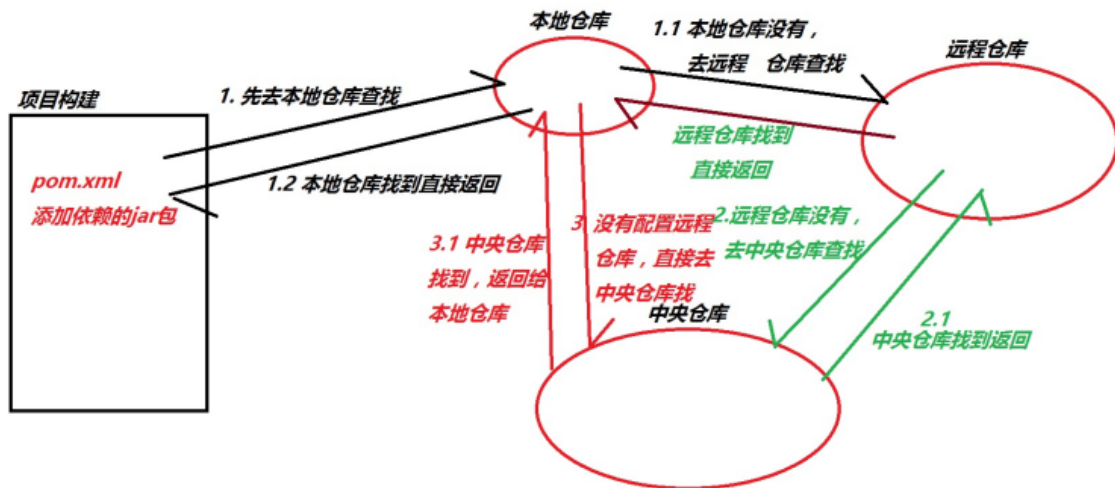
Maven官网: <http://maven.apache.org/>



Maven 是 Apache 下的一个纯 Java 开发的开源项目。基于项目对象模型（缩写：POM）概念，Maven 利用一个中央信息片段能管理一个项目的构建、报告和文档等步骤。

Maven 是一个项目管理工具，可以对 Java 项目进行构建、依赖管理。

Maven的jar包管理模式



2.3 Maven安装与配置

Maven下载: <http://maven.apache.org/download.cgi>

(1) 首先要确定jdk 已经安装和配置

把maven解压到放到D盘

(2) 配置maven环境变量: M2_HOME

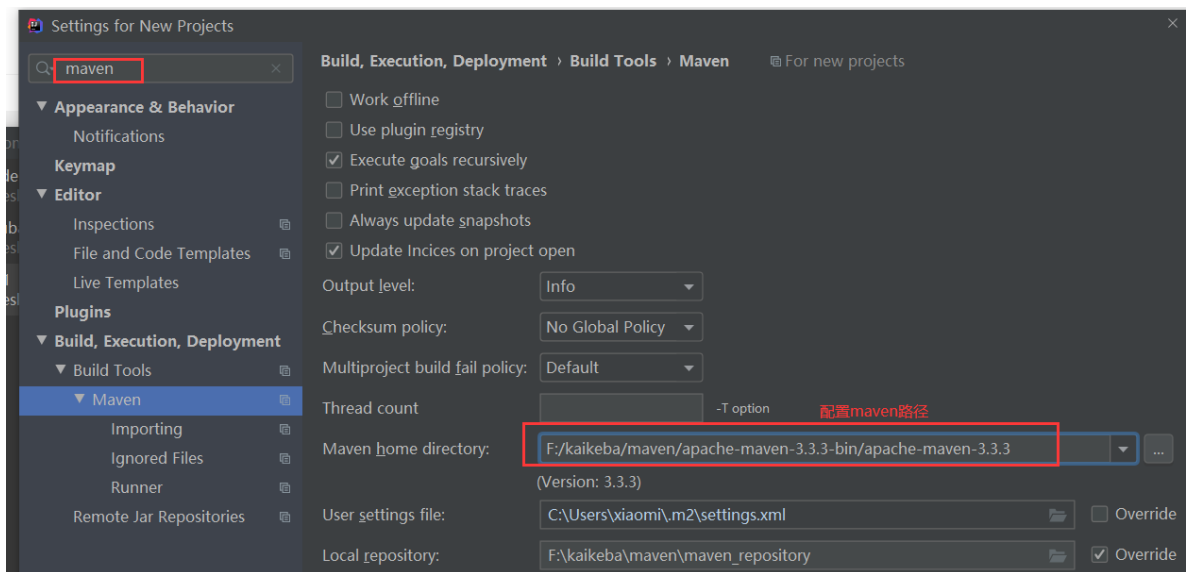
D:\maven\apache-maven-3.3.3

(3) 同时配置path:%M2_HOME%\bin

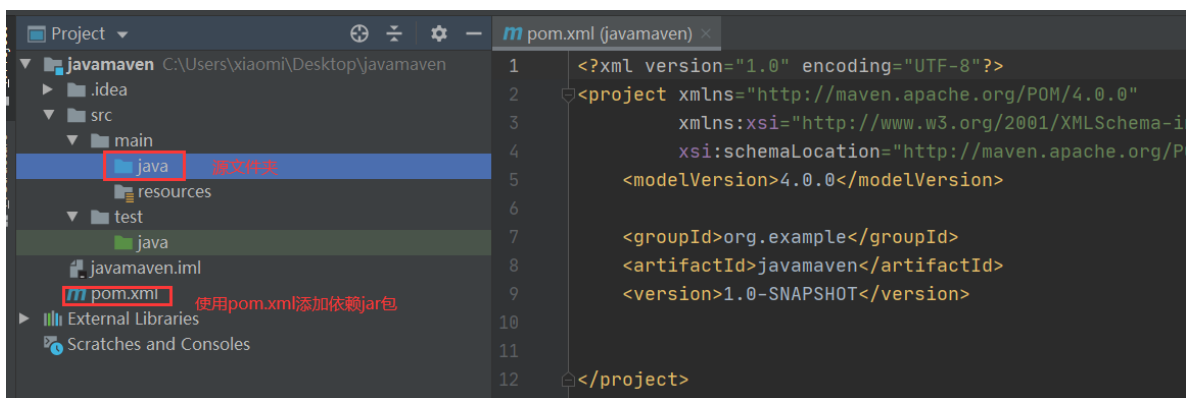
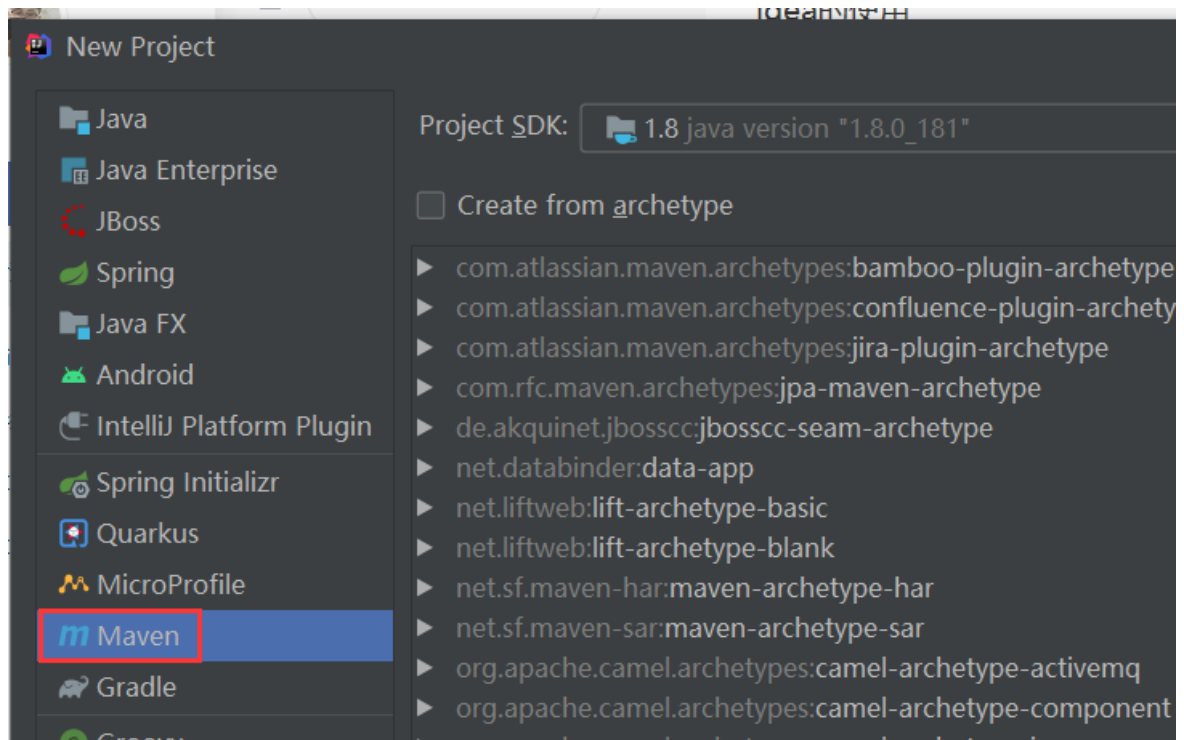
(4) 测试: mvn -version

注:maven-3.3.3兼容的jdk版本是1.7以上版本

2.4 IDEA关联Maven



2.5 创建Maven类型项目



仓库地址:<https://mvnrepository.com/>

3.JSP与html页面标签

3.1 html的基本结构

```
<html>
  <head>
    <title>标题</title>
  </head>
  <body>
    页面内容
  </body>
</html>
```

注：之后的文字,图片，视频，音乐等都添加到body标签中

3.2 表单

作用：将用户填写的数据提交给服务器(mysql)

语法：

```
<form method="提交方式" action="数据提交的地址">
  //表单元素
</form>
```

注：method的取值有两个：get和post，action是数据提交的路径名

3.3 表单元素

语法：

(1)文本框：

用于接收用户输入的信息

(2)提交按钮：

3.4 jsp介绍

JSP是可以嵌套JAVA代码的html页面

页面后缀: *.jsp