第 1 章

初识 Java

Chapter 1

Java 是当今较流行的编程语言之一，Java SE 是 Java EE 和安卓开发的基础。1995 年，美国 Sun Microsystems 公司正式向 IT 业界推出了 Java 语言，该语言具有安全、跨平台、面向对象、简单、适用于网络等显著特点。当时以 Web 为主要形式的互联网正在迅猛发展，Java 语言的出现迅速引起所有程序员和软件公司的极大关注，程序员们纷纷尝试用 Java 语言编写网络应用程序，并利用网络把程序发布到世界各地进行运行。如今，Java 经过 20 多年的发展，已经广泛应用于企业级 Web 应用开发和移动应用开发中。

技能目标

- 理解 JVM、JRE、JDK 之间的关系。
- 理解源程序、字节码文件的关系。
- 理解 Java 的可移植性。

1.1 Java 开发环境配置与搭建

任务描述

“工欲善其事，必先利其器”，学习 Java 之前，必须先了解 Java 的开发运行环境和环境的配置，那么，如何配置 Java 的开发环境呢？

任务分析

首先到 Java 官网下载 JDK 安装包，安装完成后配置 Path 环境变量，配置完成后才可以使用 javac 命令。

相关知识与实施步骤

软件安装

Java 最早是 Sun 公司的产品，后来 Sun 公司被 Oracle 公司收购，因此，要下载 Java 需

需要到 Oracle 官网下载。官网下载地址为 https://www.java.com/zh_CN/。

进入官网后可以由官网选择适合你的操作系统与机器的 Java 版本进行下载，Java 的下载界面如图 1-1 所示。

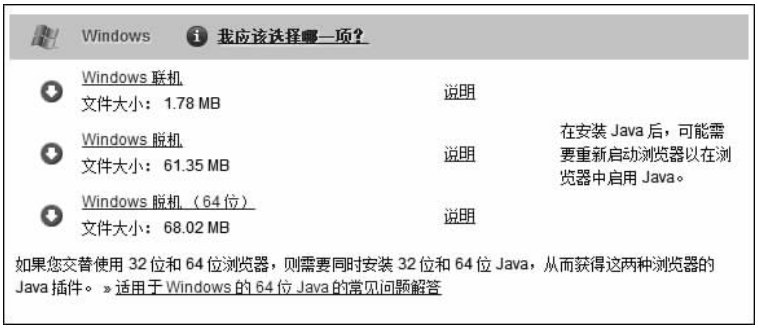

图 1-1　Java 的下载界面

根据自己的操作系统选择 32 位的还是 64 位的，Java 编者只能安装 32 位的 1.8 版本，所以本书按照 1.8 的版本进行讲解。脱机下载后得到如下软件(见图 1-2)。

图 1-2　下载到本地的 JDK

具体的安装步骤如下。

(1) 双击下载的软件包，开始 Java 的安装。编者将它安装在 D:\Program Files\Java\jdk1.8.0_144 文件夹。安装完成后，打开命令提示符窗口，执行 java -version 命令。如果看到如图 1-3 所示信息，说明安装完成。

图 1-3　测试 JDK 是否成功安装

(2) 执行 javac 命令，提示“'javac' 不是内部或外部命令，也不是可运行的程序或批处理文件。”如图 1-4 所示。

图 1-4 javac 命令不可用

(3) 配置环境变量。右击“计算机”图标，选择“属性”命令，打开“系统”窗口，如图 1-5 所示。

图 1-5 “系统”窗口

(4) 单击“高级系统设置”链接，打开“系统属性”对话框的“高级”选项卡，如图 1-6 所示。

(5) 单击“环境变量”按钮，打开“环境变量”对话框，找到“系统变量”中的 Path，如图 1-7所示。

(6) 双击 Path，打开“编辑系统变量”对话框，如图 1-8 所示。在不改变原有变量值的情况下，在最前面加入 Java 的安装路径。编者的 Java 安装在 D:\Program Files\Java\jdk1.8.0_144 文件夹中，找到 javac 命令所在的 bin 文件夹即 D:\Program Files\Java\

jdk1.8.0_144\bin，将该路径复制到 Path 变量值的最前面并加上英文分号“;”，如图 1-9 所示。接着单击所有的“确定”按钮。

图 1-6 “高级”选项卡

图 1-7 “环境变量”对话框

图 1-8 “编辑系统变量”对话框

图 1-9 添加 javac 命令的路径

（7）重新打开一个命令提示符窗口（必须重新打开），执行“javac -version”命令，显示当前 javac 的版本号，如图 1-10 所示，说明 Path 变量配置成功。

图 1-10 环境配置完成

至此，Java 的环境已经安装配置完成，可以进行 Java 程序的开发工作了。

1.2 编写第一个 Java 程序

任务描述

编写第一个 Java 程序，输出“Hello World!”。

任务分析

编写第一个 Java 程序需要注意它的程序结构。

相关知识与实施步骤

1. 编写第一个 Java 程序

（1）编程。在 D:\文件夹下新建一个文件名为 HelloWorld.txt(注意大小写敏感)的文本文档，然后将文件扩展名改为.java(见图 1-11)。

图 1-11 新建一个 Java 文件

用记事本打开 HelloWorld.java 文件，并输入以下内容，必须区分大小写(见图 1-12)。

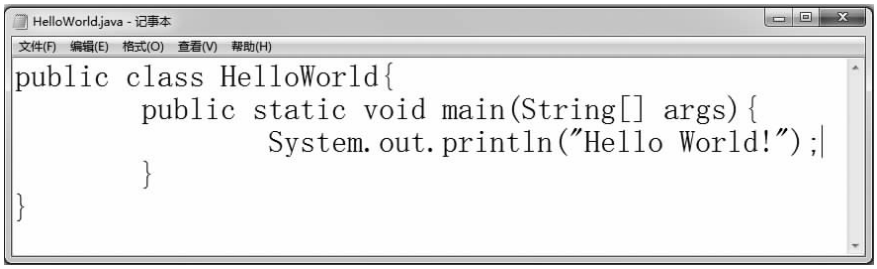

图 1-12 编写 Java 文件

下面介绍 Java 程序的结构。

```
1 public class HelloWorld{
2     public static void main(String[] args){
3         System.out.println("Hello World!");
4     }
5 }
```

在编写 Java 程序的过程中需要严格区分大小写。第 1 行中 HelloWorld 是类的名称，也是文件的名称，大小写一定要保持一致；第 2 行中的 main 方法是程序的入口，其中的关键字 public static void main(String[] args)缺一不可；第 3 行是 Java 的输出语句，目的是在控制台上输出“Hello World!”，该语句结束后需要用“;”结束；最后，Java 的程序体包含在一对大括号之间，大括号成对使用，并且开始的大括号在一行的结束位置。

编写 Java 程序时，一定按 Java 规范准确编写每一行代码，这是程序员最基本的技能标准之一。

（2）编译。打开命令提示符窗口，进入 HelloWorld.java 文件所在位置，本文件在 D:\，如图 1-13 所示。

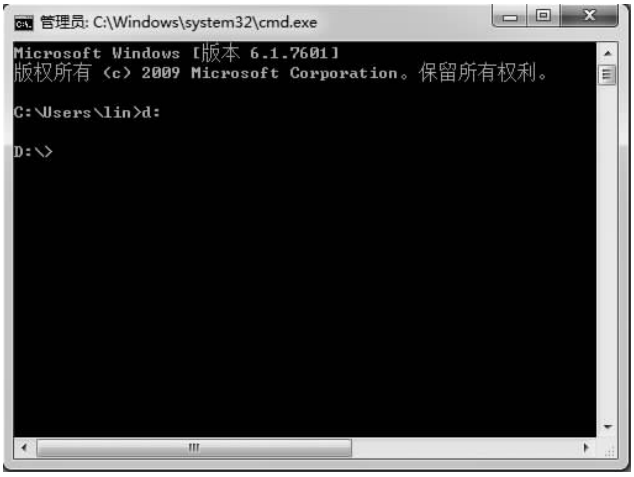

图 1-13　进入 Java 文件所在的目录

执行 javac HelloWorld.java 命令（注意带扩展名），如图 1-14 所示。

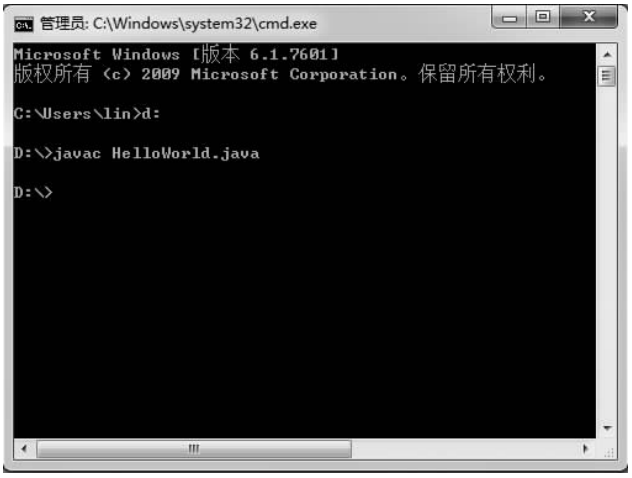

图 1-14　成功编译 Java 文件

编译成功后，将在 HelloWorld.java 文件所在文件夹中生成一个同名的.class 文件（见图 1-15）。

（3）运行。执行 java HelloWorld 命令（注意不带扩展名），输出效果如图 1-16 所示。这样一个最简单的 Java 程序从编写到运行的过程就完成了。

图 1-15 成功编译 Java 文件后生成.class 文件

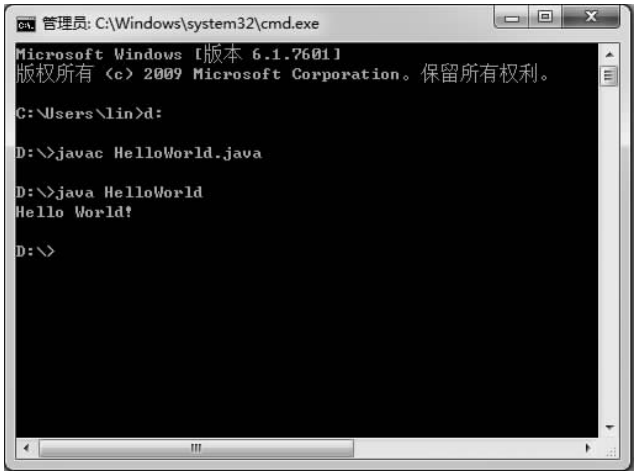

图 1-16 成功运行.class 文件

总结：Java 的运行过程：①编写源程序.java 文件；②通过编译器将.java 文件编译成.class 字节码文件；③Java 运行平台的 JVM 对字节码文件进行解释执行并输出，如图 1-17 所示。

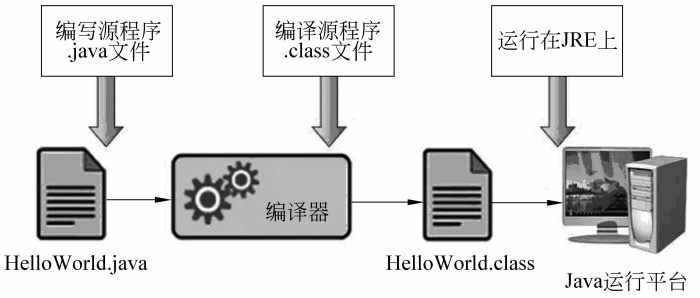

图 1-17 Java 的开发和运行过程

Java 之所以能够跨平台应用，是因为不同的平台下有不同的 Java 安装包，安装包安装完成后产生 Java 虚拟机 JVM(Java Virtual Machine)，.class 文件实际上是运行在 JVM 下，因此运行结果都一样。

2. JDK、JRE、JVM 之间的关系

JDK(Java Development Kit)是针对 Java 开发员的产品，是整个 Java 的核心，包括 Java 运行环境 JRE、Java 工具和 Java 基础类库。Java Runtime Environment(JRE)是运行 Java 程序所必需的环境的集合，包含 JVM 标准实现及 Java 核心类库。JVM 是整个 Java 实现跨平台的最核心的部分，能够运行以 Java 语言编写的程序，JDK、JRE、JVM 之间的关系如图 1-18 所示。

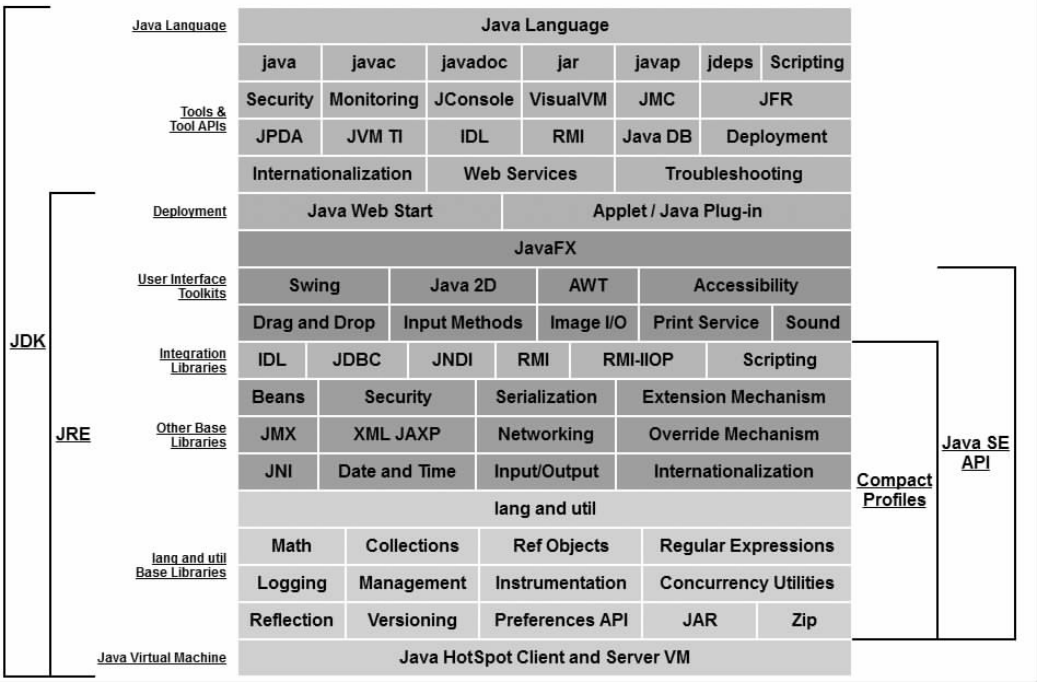

图 1-18　JDK、JRE、JVM 之间的关系

JDK 中包含 JRE，在 JDK 的安装目录下有一个名为 jre 的文件夹，里面有两个文件夹 bin 和 lib，在这里可以认为 bin 里的就是 JVM，lib 中则是 JVM 工作所需要的类库，而 JVM 和类库合起来就称为 jre。

JDK 是整个 Java 的核心，包括了 Java 运行环境 JRE、Java 工具(javac/java/jdb 等)和 Java 的基础类库(即 Java API 包括 rt.jar)。

JRE 是 Java 程序不可缺少的运行环境，也是通过它，Java 的开发者才得以将自己开发的程序发布到用户手中，让用户使用。

JRE 中包含了 JVM，运行时类库 RCL 和 Java 应用启动器，这些是运行 Java 程序的必要组件。

与 JDK 不同，JRE 是 Java 运行环境，并不是一个开发环境，所以没有包含任何开发

工具(如没有 javac 命令),只是针对使用 Java 程序的用户。

JVM 是整个 Java 实现跨平台的最核心的部分,所有的 Java 程序会首先被编译为 .class 文件,这种类文件可以在虚拟机上执行。也就是说 .class 文件并不直接与机器的操作系统相对应,而是经过虚拟机间接与操作系统交互,由虚拟机将程序解释给本地系统执行。

仅有 JVM 还不能执行 .class 文件,因为在解释 .class 文件时 JVM 需要调用解释所需要的类库,而 JRE 包含类库。

JVM 屏蔽了与具体操作系统平台相关的信息,使得 Java 程序只须生成在 Java 虚拟机上运行的目标代码(字节码),就可以在多种平台上不加修改地运行。这就是 Java 的跨平台性。

1.3 使用集成开发环境

任务描述

实际开发中,为了提高效率,都会采用集成开发环境。

任务分析

1.2 节中编写 Java 程序用的是记事本,比较烦琐还容易出错,所以实际开发中,均采用集成开发环境来完成。当今比较流行的集成开发环境包括 Eclipse 和 MyEclipse。

相关知识与实施步骤

1. 集成开发环境简介

MyEclipse 是免费的产品,可到其官网下载,MyEclipse 开发简单的 Java 程序很优秀,但后续因为涉及开发框架程序和大型网站,就需要安装插件,因此比较烦琐。MyEclipse 几乎包含了所有 Java 的开发要件,因此编者采用 MyEclipse 来进行。

首先到 MyEclipse 官网下载,地址为 http://www.myeclipsecn.com/。安装后打开,其界面如图 1-19 所示。

2. 在集成开发环境中开发 Java 程序

(1) 新建 Java 项目工程。选择图 1-19 中的 File→New→Java Project 命令,打开 New Java Project(新建 Java 项目)对话框,输入项目名称,单击 Finish 按钮,如图 1-20 所示。

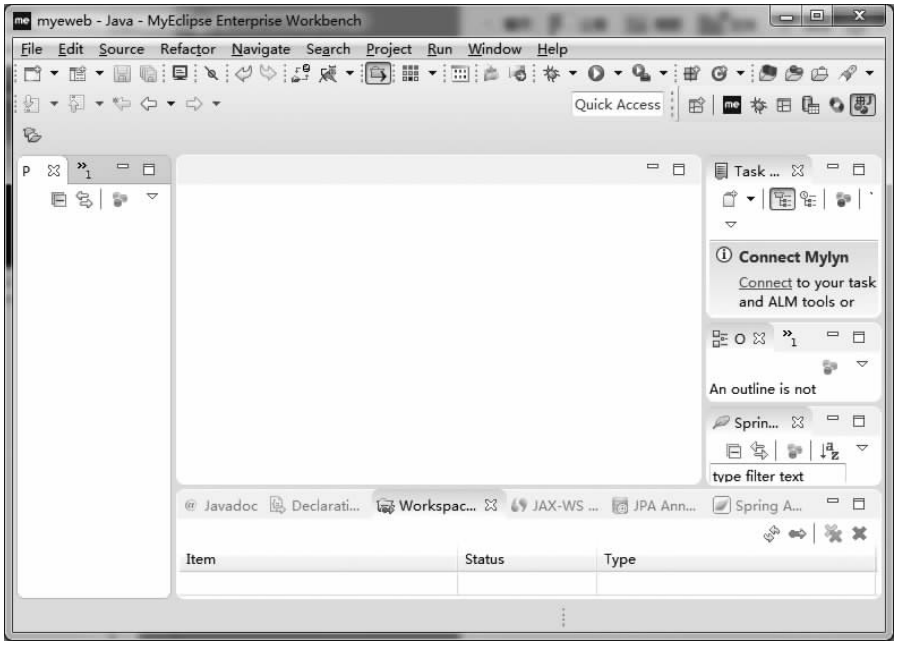

图 1-19　MyEclipse 界面

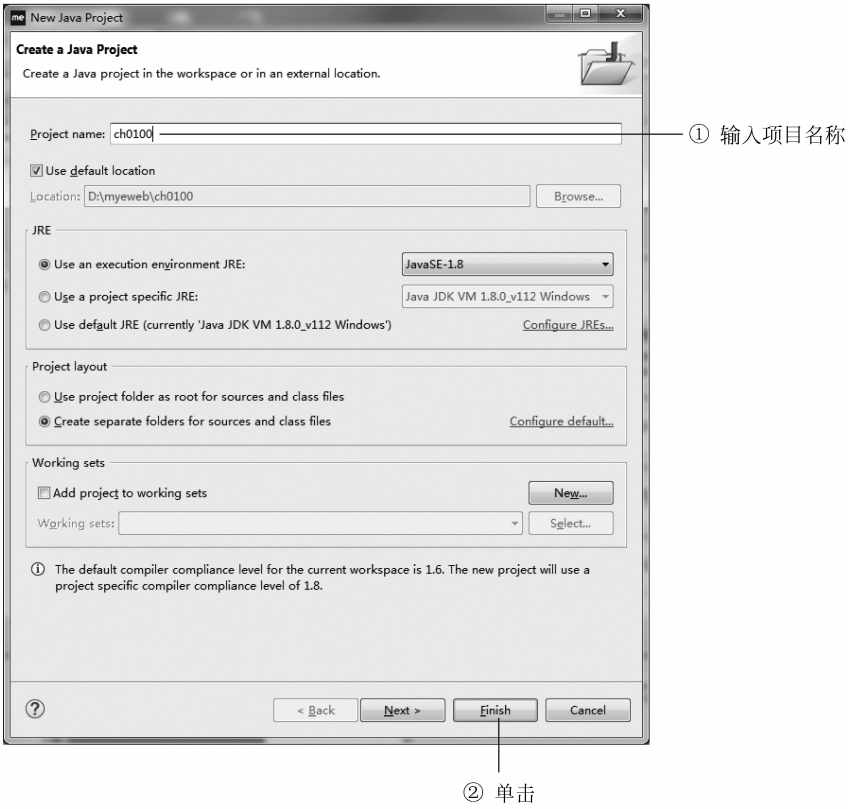

图 1-20　新建项目