

第 1 章 Java 程序设计语言概述



Java 是由 Sun 公司开发的一种应用于分布式网络环境的程序设计语言,Java 语言拥有跨平台的特性,它编译的程序能够运行在多种操作系统平台上,可以实现“一次编写,到处运行”。本章将介绍 Java 语言的特点、目标、开发环境的搭建、运行原理以及开发工具的使用。



- 了解 Java 语言的版本及应用范围;
- 了解 Java 语言的特点;
- 了解 Java 语言的设计目标;
- 掌握 Java 开发环境的搭建;
- 了解 Java 程序运行的原理;
- 掌握 Java 开发工具 Eclipse 的使用。

任务 1 Java 简介



本任务简要介绍 Java 语言的各个版本及其应用领域。

一般的初学者都认为 Java 是一种编程语言,实际上,Java 不仅是一种语言,它更是一个平台。它还提供了开发类库、运行环境、部署环境等一系列支持。

根据应用范围的不同,Java 分为 3 个版本:Java SE、Java EE 和 Java ME。

Java SE(Java Standard Edition)包含了标准的 JDK、开发工具、运行时环境和类库,适合

开发桌面应用程序和底层应用程序。同时它也是 Java EE 的基础平台。

Java EE(Java Enterprise Edition)采用标准化的模块组件,为企业级应用提供了标准平台,简化了复杂的企业级编程。现在 Java EE 已经成为了一种软件架构和企业级开发的设计思想。

Java ME(Java Micro Edition)包含高度优化精简的 Java 运行时环境,主要用于开发具有有限的连接、内存和用户界面能力的设备应用程序。例如移动电话(手机)、PDA(电子商务)、能够接入电缆服务的机顶盒或者各种终端和其他消费电子产品。

今天,无论是银行管理还是手机消费,从科学研究的巨型计算机到笔记本电脑,Java 的身影无处不在,可见 Java 已经成为行业内最流行最时髦的编程技术。

任务2 Java语言的特点



本任务主要介绍 Java 语言的特点。

Java 语言具有简单、面向对象、分布式、解释型、健壮、安全、与体系结构无关、可移植、高性能、多线程、动态执行等语言特性。另外还提供了丰富的类库,方便用户进行自定义操作。

1. 简单

Java 在设计上与 C++ 十分相近。Java 去掉了 C++ 中许多极少被使用、难以理解和容易混淆的特性,例如运算符重载、多重继承等。同时增加了自动内存回收机制,程序员不必费心管理内存。因此程序设计更加简单。

2. 面向对象

Java 语言以面向对象为基础。在 Java 语言中,不能在类外面定义单独的数据和函数,所有的对象都要派生于同一个基类,并共享它所有的功能。也就是说,Java 语言最外部的数据类型是对象,所有的元素都要通过类和对象来访问。

3. 分布式

Java 中内置了 TCP/IP、HTTP、FTP 等协议。因此,Java 应用程序可以通过 URL 地址访问网络上的对象,访问方式与访问本地文件系统几乎完全相同。

4. 解释器通用性

Java 解释器能直接对 Java 字节码进行解释执行。经过编译生成的字节码可以在提供 Java 虚拟机的任何一个系统上解释运行,不需要额外存储。

5. 健壮

Java 能够检查程序在编译和运行时的错误。类型检查能帮助用户检查出许多在开发早期出现的错误。同时许多集成开发环境 (IDE) 的出现使编译和运行 Java 程序更加容易。

6. 安全

Java 的设计目标是提供使用于网络/分布式运算环境,所以安全性问题自然是不容忽视的。Java 的验证技术是以公钥加密法为基础的。

7. 可移植性

Java 程序具有与体系结构无关的特性。这一特性使 Java 程序可以方便地移植到网络上的不同计算机。同时,Java 的类库中也实现了针对不同平台的接口,使这些类库可以移植。

8. 高效能

虽然 Java 字节码是解释运行的,但经过仔细设计的字节码可以通过 JIT 技术转换成高效能的本地代码。

9. 多线程

Java 支持多线程编程,多线程机制使程序代码能够并行执行,充分发挥了 CPU 的运行效率。程序设计者可以用不同的线程完成不同的子功能,极大地扩展了 Java 语言的功能。

任务3 Java 的设计目标



本任务简要介绍 Java 语言最初的设计目标。

Internet 迅猛发展,使 Java 迅速成为了最流行的网络编程语言。最初设计 Java 有以下几个目标。

- (1) 不依赖于特定的平台,一次编写到处运行;
- (2) 完全的面向对象;
- (3) 内置对计算机网络的支持;
- (4) 借鉴 C++ 优点,尽量简单易用。

任务4 Java 开发环境的搭建



本任务主要介绍 Java 开发环境的搭建与配置。

阶段1 JDK 的下载安装

JDK(Java Development Kit)是 Java 的开发工具包,是 Java 开发者必须安装的软件环境。JDK 包含了 JRE 和开发 Java 程序所需的工具,如编译器、调试器、反编译器和文档生成器等。

JRE(Java Runtime Environment)是 Java 运行时环境,包含了类库和 JVM(Java 虚拟机),是 Java 程序运行的必要环境。如果只运行 Java 程序,没有必要安装 JDK,只要安装 JRE 就可以了。

从 Sun 公司网站下载 JDK1.6 的具体网址为:<http://java.sun.com/javase/downloads/index.html>。

需要注意,Java 是跨平台的开发语言,根据平台的不同要选择不同的 JDK。本书选择 Windows 平台。JDK 又分为在线安装包和离线安装包两种,这里选择离线安装方式。

下载的 JDK1.6 安装包保存到硬盘上,文件名为 `jdk-6u2-windows-i586-p.exe`,执行该文件,按照向导安装。关闭所有正在运行的程序,接受许可协议,设置 JDK 的安装路径并选择要安装的组件,如图 1-1 所示。

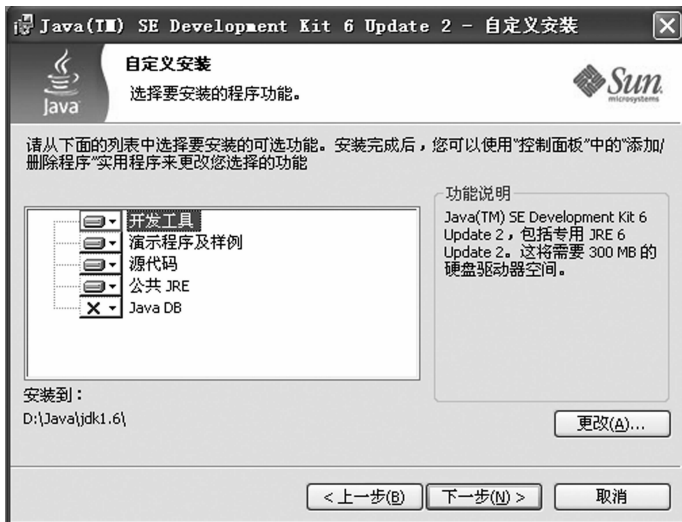


图 1-1 设置 JDK 的安装路径及选择安装的组件

更改安装路径到“D:\Java\jdk1.6”,选择要安装的组件。在安装过程中定义 JRE 安装路径到“D:\Java\jre1.6”。在弹出的提示安装完成的对话框中,取消选中“显示自述文件”复选框,单击“完成”按钮,即可完成 JDK 的安装。

安装完成后的 JDK1.6 的目录如图 1-2 所示。



图 1-2 JDK 安装目录

主要目录和文件简介如下。

bin 目录:开发工具,包括开发、运行、调试和文档生成的工具,主要是 *.exe 文件。

demo 目录:演示文件,附源代码的 Java 文件,演示了 Java 的一些功能。

include 目录:C 语言头文件,支持 Java 本地方法调用的必要文件。

jre 目录:运行时环境,包括 Java 虚拟机、类库、辅助运行的支持文件。

lib 目录:类库,开发时需要的一些类库和文件。

src.zip 文件:Java 核心类源文件,感兴趣的学习者可以解压后研究。

其中,bin 目录中的两个文件最重要,编程中经常使用。

javac.exe ——Java 编译器。

java.exe ——Java 解释器,调用 Java 虚拟机执行 Java 程序。

选择“开始”→“运行”命令,在弹出的“运行”对话框中输入“cmd”,如图 1-3 所示。



图 1-3 输入“cmd”

进入 DOS 命令行,输入“java -version”并按回车键,出现如图 1-4 所示画面,即为安装成功。

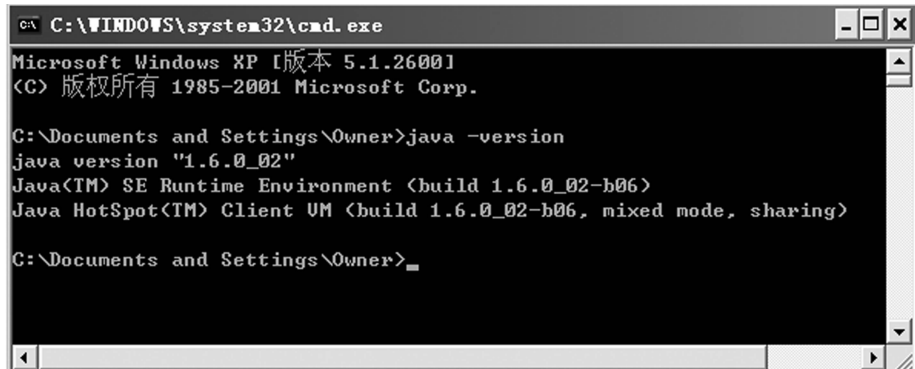


图 1-4 测试 JDK 是否安装成功

阶段 2 Java 开发环境配置

安装完 JDK 后,需要设置环境变量及测试 JDK 配置是否成功,具体操作步骤如下。

(1)在“我的电脑”图标上右击,在弹出的快捷菜单中选择“属性”命令。在弹出的“系统属性”对话框中打开“高级”选项卡,单击“环境变量”按钮,打开“环境变量”对话框,单击“系统变量”选项区域中的“新建”按钮。

(2)弹出“新建系统变量”对话框,在“变量名”文本框中输入“JAVA_HOME”,在“变量值”文本框中输入 JDK 的安装路径,单击“确定”按钮,如图 1-5 所示,完成环境变量 JAVA_HOME 的配置。



图 1-5 JAVA_HOME 环境变量的配置

(3)在“系统变量”选项区域中查看 PATH 变量,如果不存在,则新建变量 PATH,否则选中该变量,单击“编辑”按钮,在“变量值”文本框的起始位置添加“% JAVA_HOME% \bin”,单击“确定”按钮,注意不要漏掉最后的“;”符号。

(4)在“系统变量”选项区域中查看 CLASSPATH 变量,如果不存在,则单击“新建”按钮,在弹出的“新建系统变量”对话框的“变量名”文本框中输入“CLASSPATH”,在“变量值”文本框中输入“% JAVA_HOME% \lib\dt.jar;% JAVA_HOME% \lib\tools.jar”。

(5)测试 JDK 是否能够在机器上运行,在 DOS 命令行窗口输入“javac”,输出帮助信息即为配置正确。

任务5 Java 程序运行的原理



本任务介绍 Java 程序运行的原理。

编写 Java 程序可以使用简单的工具,下面用 Windows 记事本程序编写一个简单的 Java 文件,如图 1-6 所示。

A screenshot showing the development process of a Java program. The top window is a Notepad window titled 'HelloWorld.java - 记事本' containing the following code:

```
public class HelloWorld{
    public static void main(String[] args){
        System.out.println("hello world!!!");
    }
}
```

The bottom window is a Windows Command Prompt titled 'C:\WINDOWS\system32\cmd.exe' showing the following commands and output:

```
Microsoft Windows XP [版本 5.1.2600]
(C) 版权所有 1985-2001 Microsoft Corp.

C:\Documents and Settings\Owner>d:

D:\>javac HelloWorld.java

D:\>java HelloWorld
hello world!!!

D:\>
```

图 1-6 Java 的开发过程

把代码保存到 D 盘,命名为 HelloWorld.java,在 DOS 命令行编译源代码: javac HelloWorld.java,编译正确生成 HelloWorld.class 文件,Java 解释器解释执行 class 文件: java HelloWorld。如图 1-6 所示。

由运行过程可以了解 Java 的运行原理,如图 1-7 所示。

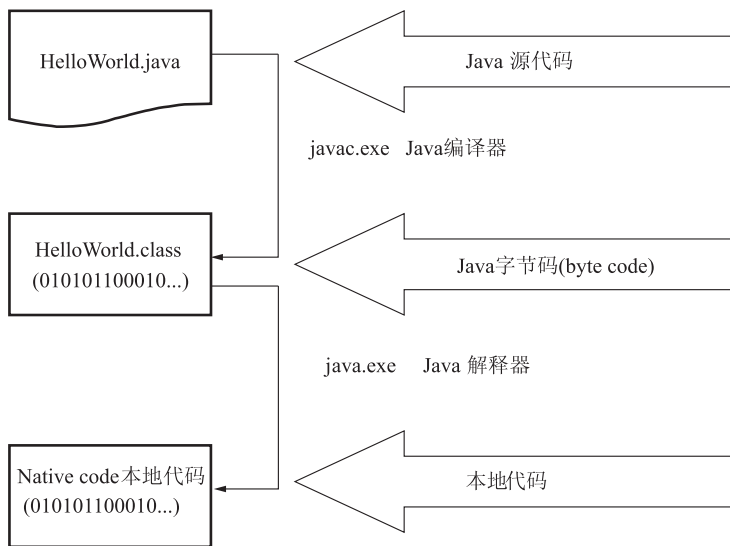


图 1-7 Java 程序运行原理

任务 6 Java 开发工具 Eclipse



本任务介绍 Java 开发工具 Eclipse 的相关知识,包括 Eclipse 的安装和使用,Eclipse 编写 Java 程序的流程。

Eclipse 是一个基于 Java 的、开放源码的、可扩展的应用开发平台,它为编程人员提供了一流的 Java 集成开发环境。Eclipse 是利用 Java 语言写成的,因此 Eclipse 是可以支持跨平台操作的,是一个成熟的可扩展的体系结构。它的价值还体现在为创建可扩展的开发环境提供了一个开发源代码的平台。这个平台允许任何人构建与环境或其他工具无缝集成的工具,而工具与 Eclipse 无缝集成的关键是插件。通过不断地集成各种插件,Eclipse 的功能也在不断地扩展,以便支持各种不同的应用。

阶段1 Eclipse 的安装与启动

安装 Eclipse 前需要先安装 JDK,关于 JDK 的安装和配置参见任务 4 中的内容。可以从 Eclipse 的官方网站(<http://www.eclipse.org>)下载最新版本的 Eclipse。本书使用的 Eclipse 版本是 3.5。

Eclipse 下载完成后,解压,即完成了 Eclipse 的安装。

在 Eclipse 初次启动时,需要设置工作空间,本书中将 Eclipse 安装到 D 盘根目录下,将工作空间设置在“D:\eclipse\workspace”中,如图 1-8 所示。

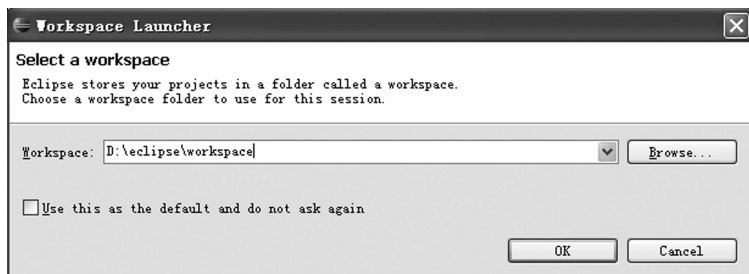


图 1-8 设置工作空间

每次启动 Eclipse 时,都会出现设置工作空间的对话框,如果不需要每次启动都出现该对话框,可以选中 Use this as the default and do not ask again 复选框将该对话框屏蔽。

单击 OK 按钮,进入到 Eclipse 工作台,如图 1-9 所示。

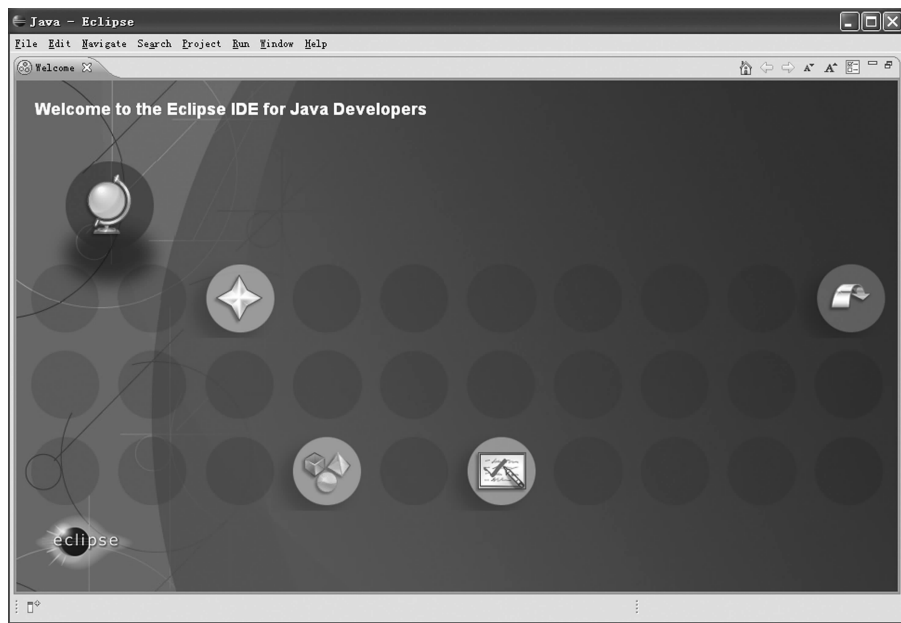


图 1-9 Eclipse 的欢迎界面

Eclipse 工作台是一个 IDE 开发环境。主要由以下几部分组成,如图 1-10 所示。

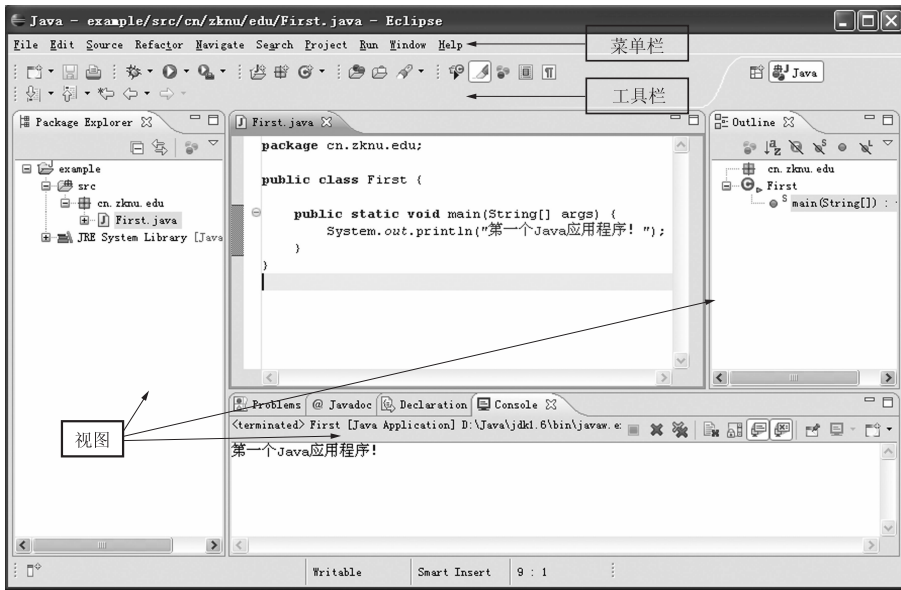


图 1-10 Eclipse 工作台

阶段 2 Eclipse 编写 Java 程序的流程

Eclipse 编写 Java 程序必须经过新建 Java 项目、新建 Java 类、编写 Java 代码和运行程序 4 个步骤,下面分别介绍。

1. 新建 Java 项目

(1) 在 Eclipse 中选择“File”→“New”→“Java Project”命令,如图 1-11 所示。

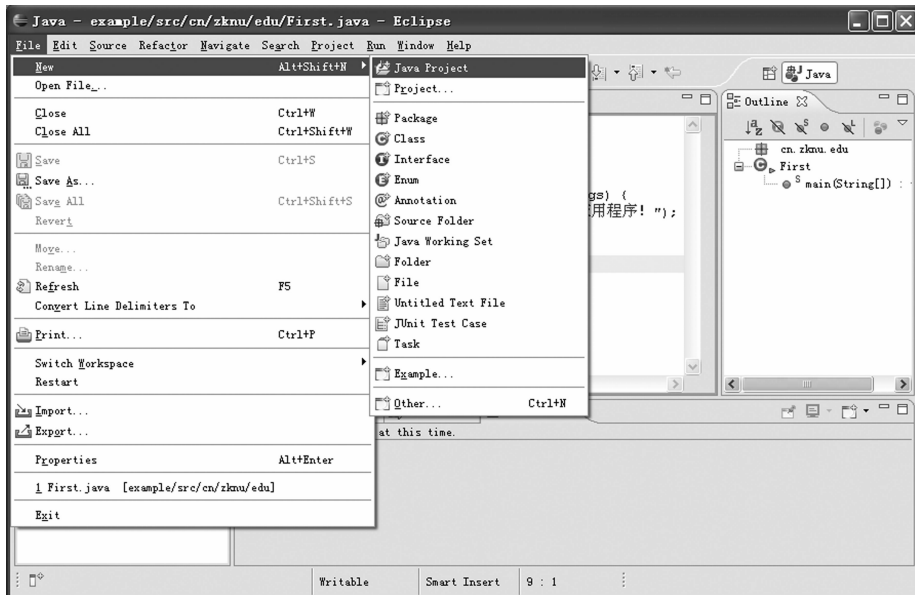


图 1-11 选择 Java Project 命令

(2) 打开 Create a Java Project(新建项目)对话框,如图 1-12 所示。



图 1-12 Create a Java Project(新建项目)对话框

(3) 单击“Next”按钮,进入到 Java Settings (Java 构建设置)对话框,配置 Java 的构建路径,如图 1-13 所示。



图 1-13 Java Settings (Java 构建设置)对话框