

高等教育国家级教学成果二等奖

清华大学计算机基础教育课程系列教材


Java 语言程序设计 (第2版)

学生用书

郑 莉 廖学良 编著



清华大学出版社



高等教育国家级教学成果二等奖

清华大学计算机基础教育课程系列教材

Java 语言程序设计 (第2版)

学生用书

郑 莉 廖学良 编著

清华大学出版社
北京

内 容 简 介

本书是与《Java 语言程序设计(第 2 版)》配套使用的学生用书。书中提供了一个简要的“学习指南”,介绍本套教材的使用方法。各章内容与主教材《Java 语言程序设计(第 2 版)》相对应,每章内容分为三部分:第一部分是“要点导读”,主要为自学者指明学习重点,建议学习方法;第二部分是“实验指导”,每章都有一个精心设计的实验,与《Java 语言程序设计(第 2 版)》相应章节内容配合,使读者在实践中深入理解和熟练掌握主教材的内容;第三部分是习题解答,给出了《Java 语言程序设计(第 2 版)》各章习题的参考答案。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

Java 语言程序设计:第 2 版/郑莉,廖学良编著.--北京:清华大学出版社,2012.8

清华大学计算机基础教育课程系列教材.学生用书

ISBN 978-7-302-28827-5

I. ①J… II. ①郑… ②廖… III. ①JAVA 语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2012)第 103090 号

责任编辑:谢 琛

封面设计:傅瑞学

责任校对:时翠兰

责任印制:杨 艳

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社总机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课件下载: <http://www.tup.com.cn>, 010-62795954

印 装 者:北京嘉实印刷有限公司

经 销:全国新华书店

开 本:185mm×260mm

印 张:10.25

字 数:239 千字

版 次:2012 年 8 月第 1 版

印 次:2012 年 8 月第 1 次印刷

印 数:1~5000

定 价:19.50 元

产品编号:047094-01

序

计算机科学技术的发展不仅极大地促进了整个科学技术的发展,而且明显地加快了经济信息化和社会信息化的进程。因此,计算机教育在各国备受重视,计算机知识与能力已成为 21 世纪人才素质的基本要素之一。

清华大学自 1990 年开始将计算机教学纳入基础课的范畴,作为校重点课程进行建设和管理,并按照“计算机文化基础”、“计算机技术基础”和“计算机应用基础”三个层次的课程体系组织教学:

第一层次“计算机文化基础”的教学目的是培养学生掌握在未来信息化社会里更好地学习、工作和生活所必须具备的计算机基础知识和基本操作技能,并进行计算机文化道德规范教育。

第二层次“计算机技术基础”是讲授计算机软硬件的基础知识、基本技术与方法,从而为学生进一步学习计算机的后续课程,并利用计算机解决本专业及相关领域中的问题打下必要的基础。

第三层次“计算机应用基础”则是讲解计算机应用中带有基础性、普遍性的知识,讲解计算机应用与开发中的基本技术、工具与环境。

以上述课程体系为依据,设计了计算机基础教育系列课程。随着计算机技术的飞速发展,计算机教学的内容与方法也在不断更新。近几年来,清华大学不断丰富和完善教学内容,在有关课程中先后引入了面向对象技术、多媒体技术、Internet 与互联网技术等。与此同时,在教材与 CAI 课件建设、网络化的教学环境建设等方面也正在大力开展工作,并积极探索适应 21 世纪人才培养的教学模式。

为进一步加强计算机基础教学工作,适应高校正在开展的课程体系与教学内容的改革,及时反映清华大学计算机基础教学的成果,加强与兄弟院校的交流,清华大学在原有工作的基础上,重新规划了“清华大学计算机基础教育课程系列教材”。

该系列教材有如下几个特色:

1. 自成体系:该系列教材覆盖了计算机基础教学三个层次的教学内容。其中既包括所有大学生都必须掌握的计算机文化基础,又包括适用于各专业的软、硬件基础知识;既包括基本概念、方法与规范,又包括计算机应用开发的工具与环境。

2. 内容先进:该系列教材注重将计算机技术的最新发展适当地引入教学中来,保持了教学内容的先进性。例如,系列教材中包括了面向对象与可视化编程、多媒体技术与应用、Internet 与互联网技术、大型数据库技术等。

3. 适应面广: 该系列教材照顾了理、工、文等各种类型专业的教学要求。

4. 立体配套: 为适应教学模式、教学方法和手段的改革, 该系列教材中多数都配有习题集和实验指导、多媒体电子教案, 有的还配有 CAI 课件以及相应的网络教学资源。

本系列教材源于清华大学计算机基础教育的教学实践, 凝聚了工作在第一线的任课教师的教学经验与科研成果。我希望本系列教材不断完善, 不断更新, 为我国高校计算机基础教育做出新的贡献。



注: 周远清, 曾任教育部副部长, 原清华大学副校长、计算机专业教授。

前 言

计算机程序设计是一门实践性很强的课程,因此仅通过阅读教科书或听课是不可能完全掌握的,学习程序设计的最重要环节就是实践。对于自学者来说,由于在学习和实践过程中缺乏指导,困难更多一重。

凡是学习程序设计的人,往往有这样的感觉:看书或听课时,对书上写的和老师讲的内容基本上都能够理解,但是,当需要自己编程时却又无从下手。相信每一个讲授程序设计课程的教师也有过这样的经历:有些问题,尽管我们在课堂上再三强调,反复举例,学生还是不能够完全理解,上机时更是错误百出。应该说,这是学习过程中的必然现象。

要想把书本上的知识变为自己所具有的能力,所需要的是实践、实践、再实践。在实践环节中,起主导作用的是学习者自己,旁人是无法代劳的,也不能期望有什么一蹴而就的捷径。但是,由于学生在实践过程中不能随时随地得到指导,因此需要花费时间较多,感觉程序设计课作业负担太重,有的学生甚至因为花四五个小时调不通一个简单的程序而失去了学习兴趣。

本书是与《Java 语言程序设计(第2版)》配套的学生用书,目的就在于为读者的学习提供一些指导,为提高读者的编程能力助一臂之力,使读者在实践过程中少些曲折和彷徨,多些成功的乐趣。本书出版之前已经在清华大学“Java 语言程序设计”课中试用,取得了良好的教学效果。

本书给出一个简要的“学习指南”,各章内容与主教材《Java 语言程序设计(第2版)》相对应,每章内容分为三部分:第一部分是“要点导读”,主要为自学者指明学习重点,建议学习方法;第二部分是“实验指导”,每章都有一个精心设计的实验,与《Java 语言程序设计(第2版)》相应章节内容配合,使读者在实践中深入理解和熟练掌握主教材的内容,每一个实验都包括“实验目的”、“实验任务”、“实验步骤”,实验的完整参考程序可从出版社网站上下载;第三部分是习题解答,给出了《Java 语言程序设计(第2版)》各章习题的参考答案,每个题目可能有多种解法,这里仅给出一种参考解法。

本书的习题解答和实验内容,不仅可以指导读者上机练习,也可以由教师选做例题在课上演示,使教学内容更加丰富。如果读者没有足够的时间一一做完全部习题和实验,可以将剩下的题解作为例题阅读也不失为一种好的选择。

本书主要由郑莉、廖学良编写,参加本书编写工作的还有杜彬、马素霞和王行言。

作 者

2012年1月于清华园

《Java 语言程序设计(第 2 版)》学习指南

《Java 语言程序设计(第 2 版)》是针对初学 Java 语言的读者编写的入门教材,预期的读者主要有这样几类:初学 Java 语言的自学者、以此为 Java 课程教材或参考书的在校学生、Java 课程教师。这里首先给出学习本套教材的总体建议,在本书的后续各章中还会有详细的导读。

1. 主教材《Java 语言程序设计(第 2 版)》的学习方法

对于主教材的学习,读者需要时刻牢记两点:面向对象的编程方法;查阅 Java API 手册,掌握主教材中出现的包、类、方法,并了解 API 中的其他包、类、方法。

具体来说,对于自学读者,在阅读教材时,应该边阅读、边实践。如果对于教材中的某些概念、语法存有疑问,应该立即编写程序以验证。对于教材中的例题,应该先尝试进行编程,之后再去看教材的答案。教材中出现的包、类,都应该通过 Java API 手册了解其主要的功能,并掌握主要方法的功能。所谓熟能生巧,一些 API 在看多了之后,自然而然就知道其功能了。在完全理解了主教材内容以后,再开始做实验和习题。

对于在校学习 Java 课程的学生,应按照教师讲课的进度,提前预习教材。所谓预习,并不是要完全看懂,如果都看懂了,就不必听课了。预习的目的是大致浏览一下新的内容,了解哪些是难点、重点,听课时就比较主动。下课之后要及时复习,然后再写作业。复习时要边看书边看笔记,这时一定要认真阅读书上的内容,并同时查看 Java API 手册,掌握书上用到的类及其方法的使用。教师可能不会在课堂上讲解书中的每个例题,对于教师课上讲的例题和书上的例题,课后复习时都要阅读、上机实践,达到完全理解,并具备自己独立编写例题程序的能力。做到这些以后,再开始写作业。

当然,这只是针对大多数读者的一般的建议,每个人还要根据自己的情况选择适合的方法。

2. 学生用书的使用方法

每学习一章主教材内容,都应该及时通过实验和习题巩固知识、提高实践能力。学生用书中的实验,是针对主教材每一章的重点内容设计的最基本的实践任务,有详细的实验指导,很容易入手,应该首先完成。完成实验之后,可以根据自己的时间和教师的要求,选择部分或全部习题来做。

本书给出了全部习题的答案,是为了方便没有教师指导的自学读者。但是不少读者在没有深入思考之前就急于看答案,这是有害无益的,这样做不仅不能真正提高自己的编程能力,还会扼杀自己的创造性思维能力。特别是对于答案代码较多的习题以及第 6 章之后的习题,因其具有较好的应用性,应该自己先编程,这样才能真正掌握知识。自学读者纷纷来信欢迎习题解答,而大多数教师(包括我自己)都不希望学生看到习题解答。这个矛盾困扰了我很久,始终没有找到两全的解决方案。所以只好在这里给出建议。

当然,有些章的习题较多,如果没有时间全部做完,也可以将一部分习题解答作为例

题来学习。值得强调的是,当看到自己不熟悉的类或者方法时,还是要查阅 API 手册,了解其功能,并能在之后的编程中使用。

3. 本套教材的思路与使用要点

本套教材是针对已经先修过 C/C++ 语言的读者的。前 5 章基本上是讲解 Java 语言的语法以及面向对象编程思想;从第 6 章开始,讲解了一些常用的专题。各专题中用到大量 Java API,读者需要学会自己查阅 API 文档,这也是编写 Java 程序需要具备的基本能力。

目 录

第 1 章 Java 语言基础知识	1
要点导读	1
实验 1 Java 简单程序设计	1
习题解答	4
第 2 章 类与对象的基本概念	9
要点导读	9
实验 2 类与对象的基本概念	9
习题解答	10
第 3 章 类的方法	18
要点导读	18
实验 3 类的方法	18
习题解答	22
第 4 章 类的重用	32
要点导读	32
实验 4 类的重用	33
习题解答	37
第 5 章 接口与多态	45
要点导读	45
实验 5 接口与多态	45
习题解答	49
第 6 章 输入输出流	54
要点导读	54
实验 6 输入输出流和文件	54
习题解答	56
第 7 章 对象群体的组织	67
要点导读	67
实验 7 对象群体的组织	67
习题解答	69
第 8 章 多线程	80
要点导读	80
实验 8 线程	81
习题解答	84

第 9 章 图形用户界面	99
要点导读	99
实验 9 图形用户界面	100
习题解答	102
第 10 章 JDBC 与数据库访问	130
要点导读	130
实验 10 JDBC 与数据库访问	130
习题解答	133
第 11 章 Servlet 程序设计	138
要点导读	138
实验 11 Servlet 程序设计	138
习题解答	139
第 12 章 JSP 程序设计	144
要点导读	144
实验 12 JSP 程序设计	144
习题解答	145

第 1 章

Java 语言基础知识

要点导读

本章介绍 Java 语言的基础知识。Java 是一种面向对象的语言,其主要特点包括面向对象、安全性高、操作平台无关性、内置多线程支持、自动内存管理等。Java 语言提供了强大的类库支持,程序员可以使用丰富的 Java API 完成需要的功能。因此,程序员应该在编程过程中积累对 API 的使用,熟练掌握常用类和常用方法。

开发 Java 程序时,基本的开发工具是 JDK。JDK 中,常用的工具包括 Javac、Java、Jdb 等。另外,还有大量的集成开发环境可以使用。Java 源文件编译成字节码文件后,由运行在特定平台上的 Java 虚拟机解释并执行。Java 虚拟机的多平台性保证了 Java 字节码文件运行时的平台无关性。

Application 和 Applet 是 Java 的两种程序类型。Application 运行在客户端的 Java 虚拟机上,Applet 运行于支持 Java 的 Web 浏览器中,由浏览器的解释器解释 Java 字节码文件并执行。另外,Java 还提供 Servlet、JSP、Bean、脚本运行等技术。

Java 语言也提供了多种基本数据类型,并支持算术、逻辑、赋值、位运算等多种表达式。Java 语言中有较多的关键字,在后续的学习过程中将会相继介绍。

数组是由同类型的数据元素构成的一种数据结构,在编写 Java 程序时会经常使用。多维数组可以理解为数组的数组。

实验 1 Java 简单程序设计

1. 实验目的

- (1) 学会编写简单的 Java 程序。
- (2) 学会使用命令行方式编译、运行 Java 程序。
- (3) 学会使用 NetBeans IDE 5.0 集成开发环境编写 Java 程序。

2. 实验任务

- (1) 用命令行方式编译、运行简单的 Java 应用程序(Application)。
- (2) 使用 NetBeans IDE 集成开发环境和命令行方式编译、运行简单的 Java 应用程序(Application)。

(3) 使用 NetBeans IDE 集成开发环境和命令行方式编译、运行简单的 Java 小程序 (Applet)。

(4) 用命令行方式编译、运行简单的 Java 小程序 (Applet)。

3. 实验步骤

(1) 用命令行编译、运行 Java 应用程序。

在文本编辑工具(例如:记事本)中新建一个文件,保存为“D:/Java/expl/problem1/HelloWorld.java”。编辑文件 HelloWorld.java 的内容为:

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

① 打开 Windows 中的命令行窗口,使用“d:”、“cd *”等命令进入目录“D:/Java/expl/problem1”,使用“dir”命令查看该目录下的文件。

② 使用“javac HelloWorld.java”命令,编译已经写好的 java 文件。使用“dir”命令查看在当前目录下,是否生成了文件“HelloWorld.class”。

③ 使用“java HelloWorld”命令,执行已经生成的“HelloWorld.class”文件,查看输出内容。

(2) Java 应用程序的编译及运行。

创建新的 Java 应用程序——Application

① 选择“文件”→“新建项目”。在“类别”下选择 Java,在“项目”下选择“Java 应用程序”选项,然后单击“下一步”按钮。

② 在“项目名称”下输入 Myapplication,将“项目位置”设置为“D:/Java/expl/problem2”。

③ 选择“创建主类”,并输入 MyClass 作为主类名。

④ 选择“设置为主项目”选项。

⑤ 单击“完成”按钮。在“项目”窗口中将显示 Myapplication 项目,并在源代码编辑器中打开 MyClass.java。

创建和编辑 Java 源代码

⑥ 在源代码编辑器中选择 MyClass.java 标签。

⑦ 将自动生成的代码

```
public class MyClass  
{  
    :  
}
```

替换成以下内容:

```
public class MyClass {
```

```
private int val1, val2;
public void myFun(int x, int y) {
    val1=x;
    val2=y;
    int sum=val1+val2;
    System.out.println("The sum of "+val1+" and "+val2+" is: "+sum);
}
public static void main(String arg[]) {
    MyClass MyObj=new MyClass();
    MyObj.myFun(1, 2);
}
}
```

⑧ 按 Ctrl+S 键保存文件。

编译和运行项目

⑨ 鼠标右键单击 Myapplication→“生成”。

⑩ 鼠标右键单击 Myapplication→“运行”。

(3) Java 小程序的编译及运行。

创建新的 Java 应用程序——Applet

① 选择“文件”→“新建项目”。在“类别”下选择 Java。在“项目”下选择“Java 类库”，单击“下一步”按钮。

② 在“项目名称”下输入 MyApplet。将“项目位置”设置为“D:/Java/exp1/problem3”。

③ 单击“完成”按钮。

④ 在“项目”窗口中将显示 MyApplet 项目。

创建和编辑 Java 源代码

⑤ 右键单击“MyApplet 项目节点”→“新建/其他”。

⑥ 在窗口中类别下选择 Java，在文件类型下选择 Applet，单击“下一步”按钮。

⑦ 在“类名”中填写 MyApplet，单击“完成”按钮。

⑧ 将自动生成的代码替换成以下内容：

```
import java.awt.Graphics;
import java.applet.Applet;
public class MyApplet extends Applet {
    public String s;
    public void init() {
        s=new String("Hello World!");
    }
    public void paint(Graphics g) {
        g.drawString(s, 25, 25);
    }
}
```

⑨ 按 Ctrl+S 键保存文件。

编译和运行项目

⑩ 鼠标右键单击“MyApplet 项目节点”→“生成”。

⑪ 鼠标右键单击“MyApplet.java”→“运行”文件,可以看到,已经启动小程序查看器(Applet Viewer)。

⑫ 查看并学习在 build 文件夹下自动创建 MyApplet.html 文件内容。

(4) 用命令行方式编译、运行 Java 小程序。

① 在文本编辑工具(如记事本)中新建一个文件,保存为“D:/Java/exp1/problem4/MyApplet.java”,编辑文件 MyApplet.java 的内容为:

```
import java.awt.Graphics;
import java.applet.Applet;
public class MyApplet extends Applet {
    public String s;
    public void init() {
        s=new String("Hello World!");
    }
    public void paint(Graphics g){
        g.drawString(s,25,25);
    }
}
```

② 打开 Windows 中的命令行窗口,使用“d:”、“cd *”等命令进入目录“D:/Java/exp1/problem4”。

③ 使用“javac MyApplet.java”命令,编译已经写好的 java 文件。查看在当前目录下,是否生成了文件“MyApplet.class”。

④ 在目录“D:/Java/exp1/problem4”下新建文件 MyApplet.html,编辑其内容为:

```
<HTML>
<HEAD>
    <TITLE>MyApplet</TITLE>
</HEAD>
<BODY>
    <APPLET code="MyApplet.class" width=350 height=200></APPLET>
</BODY>
</HTML>
```

注意: 上述 html 文件中,code = "MyApplet.class" 表示在当前目录下寻找 MyApplet.class。

⑤ 在命令行窗口中执行“appletviewer MyApplet.html”命令,查看结果。

习题解答

1-1 请简述程序设计语言发展的过程以及面向对象语言的基本特征。

解: 程序设计语言从最开始的机器语言发展到汇编语言,再到各种结构化高级语言,

再到目前使用的支持面向对象技术的面向对象语言。

面向对象语言具有如下基本特征：

(1) 抽象和封装。抽象的结果形成类,类中的数据和方法是受保护的,可以根据需要设置不同的访问控制属性。这便保证了数据的安全性,隐藏了方法的实现细节,也方便了使用。

(2) 继承性。可以对已有类增加属性和功能,或进行部分修改来建立新的类,实现代码的重用。

(3) 多态性。在面向对象的程序中,同一个消息被不同的对象接收后可以导致不同的行为。

1-2 在程序设计语言 Fortran、C、C++、Java 中,哪些是面向对象的语言? Java 语言的特点是什么?

解: C++ 和 Java 是面向对象的语言。

Java 语言的特点有:

(1) 面向对象。Java 语言提供了类的机制,在对象中封装了成员变量和方法,实现了数据的封装和信息隐藏;类提供了一类对象的模型,通过继承和多态,实现了代码的复用。

(2) 安全性。Java 不支持指针,一切对内存的访问都必须经过对象的实例变量来实现,防止了以不法手段访问对象的私有成员,同时避免了指针操作中容易产生的错误。Java 的内部安全措施保证 Java 程序在 Java 虚拟机规则下操作,防止未授权的程序访问含有专有信息的系统资源或危及客户机的完整性。

(3) 操作平台无关性。Java 编译器生成与平台无关的字节码指令,只要安装了 Java 运行系统,其程序就可以在任意的处理器上运行。与平台无关的特性使得 Java 程序可以方便地移植到不同的机器。

(4) 多线程。Java 是第一个在语言级提供内置多线程支持的高级语言,这大大简化了多线程程序的编写。

(5) 内存管理。Java 中所有的对象都是通过动态内存分配建立的,Java 对内存自动进行管理并进行垃圾回收。

1-3 Java 5 和 Java 6 分别提供了哪些新的特色?

解: Java 5 提供的新特色有:增强 for 循环、泛型、自动装箱拆箱、枚举类型、可变量长参数、静态 import、注解等。Java 6 提供的新特色有:脚本执行功能,更强的网络服务支持,Java DB,新增桌面 API,处理注解,更强的监控和管理,支持对编译器的访问,对 XML 安全性的支持等。

1-4 Java 开发工具包中用来编译源文件、执行类文件的程序分别是什么?请简单阐述程序编译执行的过程。

解: 编译源文件的程序是 javac,执行类文件的程序是 java。

Java 程序编译执行的过程为:首先把源文件(.java 文件)编译成字节码文件,即类文件(.class);然后由解释器负责解释执行类文件。

1-5 请安装 JDK,并练习设置 path 和 classpath。

解: 在系统的环境变量中设置,具体见教材。

1-6 参照例题,编写一个打印出“The Java World”的 Java Application 程序,并在命令行和 NetBeans 中编译、运行。

解:(1) 命令行方式:

① 新建文件 Test.java,编辑其内容为:

```
public class Test {  
    public static void main(String arg[]) {  
        System.out.println("The Java World");  
    }  
}
```

② 在 Windows 的命令行窗口中,进入 Test.java 所在的目录,使用下列命令编译 Test.java:

```
javac Test.java
```

③ 使用下列命令执行 Test.class:

```
java Test
```

(2) 在 NetBeans 中:

① 选择“文件”→“新建项目”。在“类别”下选择“Java”,在“项目”下选择“Java 应用程序”,然后单击“下一步”按钮。

② 在“项目名称”下输入 Test,并设置“项目位置”。勾选“创建主类”,并输入 Test 作为主类名。勾选“设置为主项目”。单击“完成”按钮。

③ 在源代码编辑器中选择 Test.java 标签。

④ 将 Test.java 文件中自动生成的代码

```
public class Test  
{  
:  
}
```

替换成以下内容:

```
public class Test {  
    public static void main(String arg[]) {  
        System.out.println("The Java World");  
    }  
}
```

⑤ 鼠标右键单击项目名→“生成”。

⑥ 鼠标右键单击项目名→“运行”。

1-7 参照例题,编写一个打印出“The First Applet”的 Java Applet 程序,并编译;然后编写调用此 Applet 类的 HTML 文件,并用 appletviewer.exe 和 Internet Explorer(或者其他浏览器)打开。

解：新建 MyApplet.java 文件，其内容为：

```
import java.awt.Graphics;
import java.applet.Applet;
public class MyApplet extends Applet {
    public String s;
    public void init() {
        s=new String(The First Applet!);
    }
    public void paint(Graphics g) {
        g.drawString(s, 25, 25);
    }
}
```

按照教材编译 MyApplet.java 程序，并使用 appletviewer 和浏览器查看该 Applet。

1-8 基本数据类型有几种？分别是什么？

解：数字型基本数据类型有 6 种，分别为字节型(byte)、短整型(short)、整型(int)、长整型(long)、单精度浮点型(float)和双精度浮点型(double)。关于对这 6 种数字类型的说明见主教材第 17 页表 1-3。

1-9 双目运算符和单目运算符都有哪些？

解：双目运算符有：+（表示加）、-（表示减）、*、/、%，单目运算符有：+（表示正值）、-（表示负值）、++、--。

1-10 “逻辑与”、“逻辑或”、“逻辑非”分别用什么表示？

解：“逻辑与”用 && 表示，“逻辑或”用 || 表示，“逻辑非”用 ! 表示。

1-11 位运算符有哪些？

解：位运算符有：按位反运算符(~)，按位与运算符(&)，按位或运算符(|)，按位异或运算符(^)，左移运算符(<<)，右移运算符(>>)，无符号右移运算符(>>>)。

1-12 int 类型转化为 float 类型时，是否会损失精度？从 float 类型转化为 int 类型呢？

解：从 int 转化为 float 时，不会损失精度。从 float 转化为 int 时，可能会损失精度。

1-13 表示数组元素个数的属性是什么？

解：是 length。

1-14 多维数组的含义是什么？

解：多维数组可以看作数组的数组，即高维数组的每一个元素为一个低维数组。

1-15 编写程序，分别练习声明、创建、初始化和引用一维数组与二维数组。

解：新建 Exe2_15.java 文件，其内容为：

```
public class Exe2_15 {
    public static void main(String[] args) {
        int[] a; //声明一维数组
        float[][] b; //声明二维数组
        int[] c={1, 2, 3}; //创建、初始化一维数组
    }
}
```