

21世纪高等教育计算机规划教材



微课示例

采用「引导式上机实践」教学方法  
主要实验程序均配有微课视频  
通过综合项目培养程序设计思维与能力  
配套《Java程序设计（微课版）》


# Java程序设计

## 习题与实践 | 微课版

田春瑾 / 主编

普运伟 王樱子 / 副主编

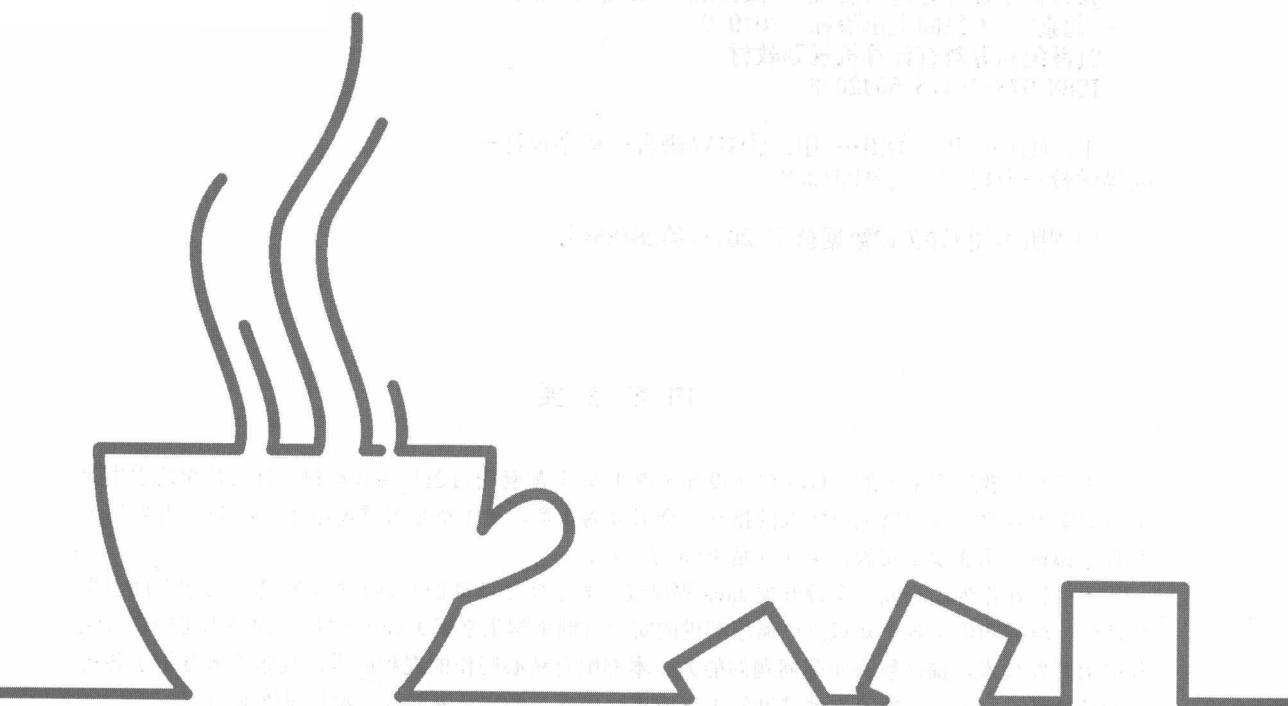
胡钰 柳翠寅 刘领兵 付湘琼 姜迪 / 参编

 中国工信出版集团

 人民邮电出版社  
POSTS & TELECOM PRESS

计算机规划教材

清华大学出版社



# Java程序设计

## 习题与实践 | 微课版

田春瑾 / 主编

普运伟 王樱子 / 副主编

胡钰 柳翠寅 刘领兵 付湘琼 姜迪 / 参编

人民邮电出版社

北京



## 图书在版编目 (CIP) 数据

Java程序设计习题与实践：微课版 / 田春瑾主编

— 北京：人民邮电出版社，2019.2

21世纪高等教育计算机规划教材

ISBN 978-7-115-50420-3

I. ①J… II. ①田… III. ①JAVA语言—程序设计—  
高等学校—教材 IV. ①TP312.8

中国版本图书馆CIP数据核字(2018)第281851号

## 内 容 提 要

本书是与普运伟主编的《Java 程序设计（微课版）》配套的习题与实践教材，目的是帮助学生课后巩固学习并给学生提供上机实践的指导。全书分为4章，第1章是习题，第2章是Java开发环境及程序调试，第3章是实验，第4章是Java综合设计。

本书旨在作为适合高等学校开展Java程序设计教学实践活动的基础教程。精选的习题有助于学生理解、巩固和消化各个知识点；循序渐进的实验有助于学生掌握Java程序设计的基本思想、方法与应用开发技术，提高解决实际问题的能力。本书配有精心制作的视频解析，视频深入分析了各实验的解决思路和方法，能有效地帮助学生完成实验内容，提高学生的程序设计思维能力。

本书可作为高等学校学生学习Java程序设计课程的辅导和实践教材，也可以作为广大Java爱好者和编程人员学习Java程序设计的参考书。

- 
- ◆ 主 编 田春瑾
  - 副 主 编 普运伟 王樱子
  - 责任编辑 刘海漂
  - 责任印制 焦志炜
  
  - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号  
邮编 100164 电子邮件 315@ptpress.com.cn  
网址 <http://www.ptpress.com.cn>  
大厂聚鑫印刷有限责任公司印刷
  
  - ◆ 开本：787×1092 1/16  
印张：12.25 2019年2月第1版  
字数：291千字 2019年2月河北第1次印刷
- 

定价：32.80元

读者服务热线：(010)81055256 印装质量热线：(010)81055316

反盗版热线：(010)81055315

广告经营许可证：京东工商广登字 20170147号

# 前言

PREFACE

本书是与普运伟主编的《Java 程序设计（微课版）》配套的习题与实践教材，也可单独用作上机实践教材。全书分为 4 章：第 1 章是与配套主教材各章对应的习题；第 2 章介绍了 Java 开发环境的准备、配置，常用开发工具的使用及程序调试的一般方法；第 3 章精心设计了 16 个实验，每个实验分为“实验目的”“实验内容”“实验拓展与思考”3 个部分；第 4 章是 Java 综合设计，侧重介绍设计要求及综合设计实例。

书中第 1 章的习题紧扣配套主教材各章内容的要点，以掌握 Java 语言的应用为目的，提供了知识要点回顾视频，有效地帮助学生理解基本概念，掌握相关知识。第 2 章扫清学生对环境配置的学习障碍。第 3 章的上机实验是学习计算机程序设计语言的重要环节，是对学生进行基本编程能力训练、增强学生实践能力的关键环节，是培养学生程序设计思维能力的重要手段。该章一般采取“关键语句填空、改错与调试、重要方法编写、程序片断补充”等形式，并配有视频解析资源，以引导式方式帮助学生厘清思路，深入分析，寻找解决问题的方法，最后给出实验结果。同时，每个实验最后的拓展与思考紧紧围绕实验和教学内容，进一步启迪学生思维，培养学生的知识迁移与应用能力。第 4 章的 Java 综合设计由两个具有较高综合性、涉及全书大部分内容的编程实例组成，有助于激发学生的编程积极性，进一步培养学生的综合问题解决能力。

本书第 1 章的 1.1 节、1.5 节由普运伟编写，1.2 节由胡钰编写，1.3 节、1.9 节由田春瑾编写，1.4 节、1.8 节由王樱子编写，1.6 节由柳翠寅编写，1.7 节由刘领兵编写，1.10 节、1.11 节由付湘琼编写；第 2 章由普运伟编写；第 3 章的实验 1、实验 8、实验 9 由普运伟编写；实验 2、实验 3 由胡钰编写，实验 4、实验 5、实验 14、实验 15 由田春瑾编写；实验 6、实验 7、实验 13 由王樱子编写；实验 10、实验 11 由柳翠寅编写；



实验 12 由刘领兵编写；实验 16 由付湘琼编写；第 4 章由姜迪编写。由田春瑾、普运伟、王樱子负责全书的统稿、修改和定稿。

本书得到昆明理工大学特色精品系列教材建设项目立项支持。在编写过程中，本书得到昆明理工大学教务处及许多领导、老师和同学的支持与帮助，在此对他们表示感谢，同时对本书所参考的图书的作者表示衷心的感谢。

编者

2018 年 11 月

# 目 录

## CONTENTS

### 第1章 习 题

---

1.1 Java 语言概述 .....	1
1.2 Java 语法基础 .....	10
1.3 程序流程控制 .....	21
1.4 数组 .....	34
1.5 Java 面向对象编程 .....	43
1.6 Java 实用类库 .....	57
1.7 异常与断言 .....	64
1.8 Java 文件操作 .....	69
1.9 Swing 程序设计 .....	77
1.10 Applet 程序设计 .....	84
1.11 多线程程序设计 .....	88

### 第2章 Java开发环境及程序 调试

---

2.1 JDK 的下载、安装和配置 ...	93
2.1.1 JDK 的下载 .....	93
2.1.2 JDK 的安装 .....	94
2.1.3 JDK 的配置 .....	94
2.2 UltraEdit 的安装、配置及 使用 .....	96

2.2.1 UltraEdit 的安装 .....	96
2.2.2 UltraEdit 的配置 .....	96
2.2.3 UltraEdit 的使用 .....	98

### 2.3 NetBeans IDE 的下载、安装 和使用 .....

2.3.1 NetBeans 的下载和 安装 .....	101
2.3.2 使用 NetBeans 开发 Java 程序 .....	102
2.3.3 运行带命令行参数的 Java 程序 .....	104

### 2.4 程序调试 .....

2.4.1 程序错误 .....	105
2.4.2 NetBeans 的程序调试 功能 .....	105
2.4.3 单步调试 .....	106
2.4.4 断点调试 .....	108

### 第3章 实 验

---

实验 1 Java 程序的编辑、编译和 运行 .....	109
实验 2 数据类型与运算符 .....	111
实验 3 Java 输入与输出 .....	113

实验 4	选择结构	116
实验 5	循环结构	119
实验 6	一维数组	121
实验 7	二维数组	124
实验 8	类和对象及类的继承	127
实验 9	类的多态及其接口	131
实验 10	Java 包和字符串类	135
实验 11	Java 集合框架类的 应用	138
实验 12	异常与断言	142
实验 13	Java 文件操作	145
实验 14	窗口布局及事件处理	148
实验 15	常用 Swing 组件的 应用	151
实验 16	Applet 及多线程程序 设计	155

## 第4章 Java综合设计

---

4.1	综合设计的目的与要求	160
4.2	综合设计报告格式	162
4.3	综合设计范例	163
4.3.1	范例 1——对象的文件 存取操作	163
4.3.2	范例 2——简易计算器 设计	172
4.4	综合设计参考题目	177
4.4.1	数据结构及文件类	177
4.4.2	Swing 应用	178

## 附录 部分习题参考答案

181

---

参考文献 190

---



# 第1章 习 题

习题是根据配套主教材所涉及的教学内容精心组织和编写的,题型为判断题、单选题、填空题和编程题4种,有很好的针对性,能帮助学生更好地理解Java语言的基本概念、基本知识、语法规则和常用类库的用法,熟悉一些常用算法和标准编程模块,掌握Java编程的基本方法和实用技巧,达到巩固所学知识、训练程序设计思维能力、提高学习效率和学习效果的目的。

## 1.1 Java语言概述

### 学习效果检验

- Java语言是如何发展的?Java语言有哪些特点?
- JVM、JRE和JDK有何关系?Java程序是怎样运行的?
- 开发一个Java程序要经历哪些过程?
- Java程序的基本结构如何?编程中要注意哪些语法规范?
- Java标识符遵循哪些规则?
- 为Java程序输入数据的常见方法有哪些?Java程序如何输出信息?



学习要点

### 一、判断题

1. Java既是开发环境,又是应用环境,它代表了一种新的计算模式。( )
2. Java是一种严格的面向对象语言,编写的所有代码都限定在类内完成。( )
3. JVM的代码格式为压缩的字节码,因而效率较高。( )
4. 字节码是不依赖于具体硬件平台的二进制代码。( )
5. 如果Java文件中包含一个公有类,则该文件的文件名必须与该公有类一致。( )
6. 在一个源文件中最多只能定义一个公有类和一个非公有类。( )
7. Java的结构中立特性正好适合开发运行于不同计算机平台上的各种网络软件。( )
8. Java并不是一门单独的程序设计语言,而是包括Java语言、Java运行环境、Java Server Page等众多技术于一身的业界解决问题的平台。( )
9. Java EE是各类Java应用平台的基础和核心。( )
10. Java源文件是一种文本文件,而编译得到的字节码文件是一种特殊的二进制文件。( )
11. class文件和exe文件都是可执行文件。( )
12. 由于JVM的独特作用,字节码文件可以独立于具体的机器平台。( )

13. JRE 是 JVM 的基础和核心。( )
14. Path 变量用于设置 JVM 搜索 class 文件的默认搜索路径,也即告诉 Java 虚拟机默认情况下到什么地方去寻找 \*.class 文件。( )
15. 在 Java 源程序中,每行的缩进量可以根据代码的清晰性随意设置,但必须标明行号。( )
16. 若 Java 源程序中定义了类 HelloWorld,则文件名必须取名为 HelloWorld.java。( )
17. 对于源程序 Test.java,编译通过后将得到字节码文件 Test.class。( )
18. 使用解释程序将字节码文件翻译成本地代码时,不需要指明文件的扩展名 class。( )
19. 注释有助于提高程序的可阅读性,但注释太多将影响程序的执行效率。( )
20. Java 的任何程序都是从定义一个类开始的。( )
21. main 方法的标准定义写法为 public static void main(String[] args),其中的 args 具有特殊含义,不能修改为其他名称。( )
22. Java 的类定义和方法定义都是以“{”标记开始,以“}”标记结束。( )
23. Java 语句以分号结束,同一行上可以写多条语句,也可以将一条语句写在多行上。( )
24. “+”号除了表示加法运算外,还可以用于字符串连接操作。( )
25. main 方法是 Java 程序的执行起点,因此必须首先定义包含 main 方法的类,且 main 方法必须在类中最先定义。( )
26. Java 是严格区分大小写的语言,sum 和 Sum 是不同的标识符。( )
27. Java 程序的注释分为单行注释、多行注释和文档注释 3 种。( )
28. 要为 Java 程序输入一个实数,可以使用命令行参数配合封装类的方法。( )
29. System.in 相当于 Java 的标准输入设备,即键盘。( )
30. Java 并没有直接输入数据的方法,而是将 System.in 输入的内容当作数据流进行统一处理。( )

## 二、单选题

1. Java 语言正式发布于\_\_\_\_\_年。  
A. 1991                      B. 1995                      C. 1996                      D. 1998
2. 下列有关 Java 的说法中,错误的是\_\_\_\_\_。  
A. Java 是简单的、面向对象的语言                      B. Java 具有与平台无关的特点  
C. Java 程序只能解释执行                      D. Java 是一种面向网络的语言
3. 用于将 Java 程序编译成字节码文件的工具是\_\_\_\_\_。  
A. javac.exe                      B. java.exe                      C. appletviewer.exe                      D. javadoc.exe
4. 在 JDK 中,负责解释执行字节码文件的程序是\_\_\_\_\_。  
A. javac.exe                      B. java.exe                      C. appletviewer.exe                      D. javadoc.exe
5. 在 JDK 中,用来解释已经转换成字节码的 Applet 程序的工具是\_\_\_\_\_。  
A. javac.exe                      B. java.exe                      C. appletviewer.exe                      D. javadoc.exe
6. Java 程序运行的入口点是\_\_\_\_\_。

- A. main 方法      B. start 方法      C. init 方法      D. 构造函数
7. 不能用于修饰 main 方法的关键字是\_\_\_\_\_。
- A. public      B. static      C. void      D. private
8. 如果一个方法为 void 类型, 这表明\_\_\_\_\_。
- A. 该方法为公有方法, 可被所有其他方法访问  
B. 该方法不需要创建任何类的实例即可访问  
C. 该方法可被子类继承  
D. 该方法不会返回任何值
9. Java 语言的诞生源于 Sun 公司于 1990 年启动的\_\_\_\_\_项目。
- A. Green      B. Blue      C. Red      D. Orange
10. 下面有关 Java 语言特点的描述中, 错误的是\_\_\_\_\_。
- A. Java 通过自动垃圾回收机制实现对内存的自动管理  
B. Java 具有面向对象编程的封装性、继承性和多态性特征  
C. Java 和 C++ 语言一样, 采用多重继承机制提高代码的复用效率  
D. Java 内置了多种网络协议, 适于网络编程
11. 下列有关 Java 运行机制的说法中, 错误的是\_\_\_\_\_。
- A. JVM 可以将同一个字节码文件正确解释成本机上可执行的二进制代码  
B. 字节码文件相当于 JVM 的可执行文件  
C. 字节码文件并不直接面对具体的机器平台, 而是面对 JVM  
D. Windows 平台和 Linux 平台上安装的 JVM 是一样的
12. 下列有关 JVM、JRE 和 JDK 的说法中, 错误的是\_\_\_\_\_。
- A. JVM 是 Java 程序实现“一次编写, 到处运行”的基础  
B. 要在一个平台上运行 Java 程序, 必不可少的是 JVM 而不是 JRE  
C. JDK 是开发 Java 程序的必备工具包  
D. 运行 Java 程序离不开 JRE 的支持, 开发 Java 程序则需要 JDK 工具包
13. 在 JDK 的 CLASSPATH 变量的设置中, “.” 表示\_\_\_\_\_。
- A. 当前目录      B. Java 安装目录      C. JRE 所在目录      D. 任意目录
14. Java 源文件的扩展名为\_\_\_\_\_。
- A. java      B. class      C. exe      D. obj
15. 以下工具软件中, 不能用于开发 Java 程序的是\_\_\_\_\_。
- A. UltraEdit      B. NetBeans      C. Microsoft Word      D. Eclipse
16. 源程序 Test.java 中定义了类 Test, 则下面说法中错误的是\_\_\_\_\_。
- A. Test 类可以用 public 修饰, 也可以不用  
B. 编译通过后, 必定产生类文件 Test.class  
C. 编译通过后, 只能产生字节码文件 Test.class  
D. 要执行该程序, 执行命令不一定是 java Test
17. 下面有关 Java 程序结构的说法中, 错误的是\_\_\_\_\_。



- A. 可以包含多个类, 但只能包含一个 public 类
  - B. 只能包含一个类, 该类可以定义为 public 类, 但文件名必须和类名相同
  - C. 当包含 public 类时, 文件名必须和该 public 类相同
  - D. 可以包含多个类, 这些类都不用 public 修饰
18. 下面有关 Java 程序的 main 方法的说法中, 正确的是\_\_\_\_\_。
- A. main 方法是程序运行的起点, 每个类中必须包含一个 main 方法
  - B. main 方法定义在主类中, 任何 Java 程序有且仅有一个主类
  - C. main 方法通常用 public、static 和 void 进行修饰, 也可以不用
  - D. 一个 Java 程序必须包含一个 main 方法
19. 当一个 Java 程序中要使用多个类时, 错误的做法是\_\_\_\_\_。
- A. 可以将多个类放在同一个文件中, 但只能有一个类定义为 public
  - B. 可以将每个类放在单独的文件中, 访问权限不定
  - C. 可以将多个类放在同一个文件中, 访问权限不定
  - D. 每个文件中的类个数不定, 但每个文件只能有一个 public 类
20. 要在不创建对象的情况下访问某个类的成员方法, 需用\_\_\_\_\_关键字修饰该方法。
- A. public
  - B. static
  - C. void
  - D. class
21. 在下面的标识符中, 合法的是\_\_\_\_\_。
- A. \$username
  - B. 2sum
  - C. int
  - D. your name
22. 在下面的标识符中, 不合法的是\_\_\_\_\_。
- A. \$username
  - B. username
  - C. \_int
  - D. your name
23. 关于 Java 注释, 错误的说法是\_\_\_\_\_。
- A. 注释的作用主要是提高程序的可读性
  - B. Java 编译器将忽略单行注释和多行注释, 但不会忽略文档注释
  - C. 所有注释都不会影响 Java 字节码文件的执行效率
  - D. 文档注释常用于对类、类成员进行说明
24. 下列有关 Java 程序基本结构的说法中, 错误的是\_\_\_\_\_。
- A. Java 源文件可以包含多个类, 但只能有一个 public 类
  - B. 每一个 Java 类既可以包含成员变量, 也可以包含成员方法
  - C. main 方法是 Java 程序的执行起点, 只能在一个类中进行定义
  - D. Java 的类名首字母必须为大写字母
25. 下列有关 Java 程序基本编程规范的说法中, 错误的是\_\_\_\_\_。
- A. Java 类名一般采用名词形式, 且每个单词的首字母习惯使用大写字母
  - B. Java 中的变量名一般采用小写的名词, 或首字符小写而后面各单词首字母大写的形式
  - C. 保持合理的代码缩进格式可增强程序的可读性
  - D. Java 注释应合理使用, 以不影响程序执行效率为准
26. 某 Java 程序执行时输入“java 类名 a1 a2”, 则 main 方法的参数 args[1] 中的内容为\_\_\_\_\_。

- A. a1                      B. a2                      C. "a1"                      D. "a2"

27. 要在程序中使用某个“包”内的类，需使用\_\_\_\_\_语句。

- A. System.out      B. package                      C. import                      D. use

28. 以下程序的输出结果是\_\_\_\_\_。

```
public class Test {  
    public static void main(String[] args) {  
        int a,b,c,d;  
        a=3;b=4;c=5;d=6;  
        System.out.println("a+b+c+d="+a+b+c+d);  
    }  
}
```

- A. a+b+c+d=18                      B. a+b+c+d=3456  
C. a+b+c+d=abcd                      D. 编译出错

29. 关于下面的程序，正确的说法是\_\_\_\_\_。

```
01 public class Test {  
02     public static void main(String[] args) {  
03         int x=3;y=5;  
04         System.out.print("x="+x+",");  
05         System.out.print("y="+y);  
06     }  
07 }
```

- A. 程序第 03 行有错                      B. 程序第 04 行有错  
C. 程序输出: x=3,y=5                      D. 程序输出: x=3,  
y=5

30. 关于下面的程序，正确的说法是\_\_\_\_\_。

```
01 public class Test {  
02     public static void main(String[] a) {  
03         double pi=3.14;  
04         double r=2;  
05         double area=pi*r*r;  
06         System.out.println(area);  
07     }  
08 }
```

- A. 第 02 行有错，参数部分 a 应修改为 args  
B. 第 03 行有错，应修改为 double PI=3.14;  
C. 第 04 行有错，应修改为 double r=2.0;  
D. 程序没有任何错误，可以输出正确的结果

31. 以下程序的输出结果是\_\_\_\_\_。

```
public class Test {  
    public static void main(String[] args) {  
        System.out.print("123");  
    }  
}
```

```
        System.out.println("456");  
        System.out.print("789");  
    }  
}
```

- A. 123456789      B. 123456789      C. 123456789      D. 123789456

32. 以下程序的输出结果是\_\_\_\_\_。

```
public class Test {  
    public static void main(String[] args) {  
        int x=5,y=3;  
        System.out.println("result="+x+y);  
    }  
}
```

- A. result=8      B. result=53      C. result=35      D. 编译出错

33. 关于下面的程序, 正确的说法是\_\_\_\_\_。

```
01 public class Cylinder {  
02     public static void main(String[] args) {  
03         double r,h,v;  
04         r=1;  
05         h=2.0;  
06         v=Math.PI*r*r*h;  
07         System.out.println("result="+v);  
08     }  
09 }
```

- A. 第 04 行有错, 应修改为 r=1.0;  
B. 第 05 行有错, 应修改为 h=2;  
C. 第 06 行有错, 应修改为 v=3.14\*r\*r\*h;  
D. 程序没有任何错误, 可以输出正确的结果

34. 假设变量 a 和 b 的值分别为 3 和 5, 若希望输出结果为 “a=3, b=5”, 则程序中的 System.out.println 方法的调用格式为\_\_\_\_\_。

- A. System.out.println("a=", a, "b=", b);  
B. System.out.println("a="+a, "b="+b);  
C. System.out.println("a="+a+"b="+b);  
D. System.out.println("a="+a+",b="+b);

35. 假设 Java 源文件中已定义类 A 和类 B。在类 A 中已有方法定义 “public static void fun(int a)”, 在类 B 中已定义整型变量 result, 则在类 B 中正确调用 fun 方法的语句是\_\_\_\_\_。

- A. result=fun(2);      B. result=A.fun(2);      C. fun(2);      D. A.fun(2);

36. 以下程序的输出结果是\_\_\_\_\_。

```
public class Test1 {  
    public static void main(String[] args) {
```



```
double result;  
result=1+1.0/2;  
System.out.println("result="+result);  
}  
}
```

- A. result=1                      B. result=1.0                      C. result=1.5                      D. result=2.0

37. 若已定义变量 `radius` 为 `float` 类型, 程序拟通过 “`area=Math.PI*radius*radius;`” 计算圆的面积, 则用于保存圆面积的 `area` 变量应定义为\_\_\_\_\_类型。

- A. `int`                              B. `float`                              C. `double`                              D. `boolean`

38. 程序采用命令行参数方式为某 Java 程序输入一个双精度实数, 则可正确获得该双精度实数的形式是\_\_\_\_\_。

- A. `Integer.parseInt(args[0])`                              B. `Integer.parseInt(args[1])`  
C. `Double.parseDouble(args[0])`                              D. `Double.parseDouble(args[1])`

39. 下面的方法中, 不能实现为 Java 程序输入数据的是\_\_\_\_\_。

- A. 使用命令行参数  
B. 将标准输入流对象 `System.in` 封装成 `BufferedReader` 对象  
C. 将标准输入流对象 `System.in` 封装成 `Scanner` 对象  
D. 直接使用 `System.in` 对象的各种方法

40. `Scanner` 对象中用于读取一个整数的方法是\_\_\_\_\_。

- A. `next`                              B. `nextInt`                              C. `nextDouble`                              D. `readLine`

### 三、填空题

1. Java 编译器可以将 Java 源程序编译成与机器无关的二进制代码文件, 即字节码文件, 它的扩展名是\_\_\_\_\_。

2. 若一个程序中包含一个名为 `Sam1` 的公有类和一个名为 `Sam2` 的非公有类, 则该文件必须命名为\_\_\_\_\_。

3. 目前, Java 被划分为 3 个主要的技术分支, 即: \_\_\_\_\_、Java EE 和 Java ME。

4. Java 运行环境主要包括: \_\_\_\_\_、Java API 库和软件部署技术等。

5. 类加载路径是通过设置 \_\_\_\_\_ 变量来完成的。

6. 对于源程序 `Test.java`, 对其进行编译的命令为\_\_\_\_\_。

7. 要执行 `TestDemo.class` 字节码文件, 正确的命令是\_\_\_\_\_。

8. 要在屏幕上输出 “Hello, Java!”, 但输出后并不换行, 应使用的语句是\_\_\_\_\_。

9. 如果一个方法没有返回值, 应使用 \_\_\_\_\_ 关键词进行修饰。

10. 如果一个方法希望在不创建对象的时候就可使用, 则必须用 \_\_\_\_\_ 关键词修饰。

11. Java 语言的文档注释以 \_\_\_\_\_ 开始, 以 `*/` 结束。

12. 若要将一个字符串转换为整型数据, 应使用 `Integer` 类的 \_\_\_\_\_ 方法。

13. 要定义一个 Java 类, 需使用的关键字是\_\_\_\_\_。

14. Java 程序常见的输入方式有 2 种: \_\_\_\_\_ 和使用 `System.in` 进行数据流处理。

15. 要在程序中使用一个整型变量, 应使用关键字\_\_\_\_\_对该变量进行定义。
16. Java 的跨平台特性主要是通过独特的\_\_\_\_\_运行机制实现的。
17. Java 应用程序的执行起点是\_\_\_\_\_方法。
18. Java 程序的输入和输出主要是通过 Java API 中的\_\_\_\_\_类实现的。
19. 在 System.out.println 方法调用中, 要实现两个字符串的连接操作, 需使用的运算符是\_\_\_\_\_。
20. 若方法 fun 带有两个整型参数 a 和 b, 且该方法为“公有访问, 无返回值”, 则方法头部应定义为\_\_\_\_\_。
21. 若方法 fun 需返回整数 result, 则返回语句应写为\_\_\_\_\_。
22. Scanner 对象中用于读取一个字符串的方法是\_\_\_\_\_。
23. 若 main 方法被定义为“public static void main(String a[])”, 则对应的第 1 个命令行参数可表示为\_\_\_\_\_。
24. 若类 A 中已定义方法“public static void fun(int x)”, 则在类 B 中要以实际参数 3 调用 fun 方法的语句是\_\_\_\_\_。
25. 当用\_\_\_\_\_命令对 Java 源文件进行处理时, Java 的文档注释部分可生成类似 Java API 帮助文档的 HTML 说明文件。
26. 下述程序的运行结果为\_\_\_\_\_。

```
public class Test {  
    public static void main(String[] args) {  
        System.out.print("123");  
        /* System.out.println("456"); */  
        System.out.print("789");  
    }  
}
```

27. 以下程序的输出结果是\_\_\_\_\_。

```
public class Test {  
    public static void main(String[] args) {  
        System.out.print("123"+456);  
    }  
}
```

28. 以下程序的输出结果是\_\_\_\_\_。

```
public class Test {  
    public static void main(String[] args) {  
        int a=5,b=10;  
        System.out.print("a/b="+a/b);  
    }  
}
```

29. 下面程序的功能是采用命令行参数形式输入圆的半径, 然后计算圆的面积。请将横线处的程序代码补充完整。

```
public class Test {  
    public static void main(String[] args) {  
        double radius,area;  
        radius=_____;  
        area=Math.PI*radius*radius;  
        System.out.print("area="+area);  
    }  
}
```

30. 下面程序的功能是采用 Scanner 类从键盘输入一个整数和一个实数,并计算它们的乘积。请将横线处的程序代码补充完整。

```
import java.io.InputStream;  
import java.util.Scanner;  
public class Test {  
    public static void main(String[] args) {  
        int a;  
        double b;  
        InputStream is = System.in;  
        Scanner scan = new Scanner(is);  
        System.out.println("请输入一个整数和一个实数:");  
        a=scan.nextInt();  
        b=_____;  
        System.out.println("a*b=" +a*b);  
    }  
}
```

#### 四、编程题

1. 编写程序,在屏幕上输出下面的图案。

```
    *  
   ***  
  *****  
 *****  
*****
```

2. 已知圆的半径为 2.5,编写程序计算圆的周长和面积。
3. 已知圆柱体的半径为 2,高为 5,编写程序求该圆柱体的体积。
4. 试编写一个 getMax 方法,用于求两个数之中的最大数。再编写 main 方法,测试 getMax 方法的正确性。
5. 使用命令行参数方式,为 Java 程序输入两个实数,并求这两个实数的乘积。
6. 使用命令行参数方式,从键盘输入梯形的上底、下底和高,计算该梯形的面积。
7. 使用 System.in 为 Java 程序输入一个实数,并给出最靠近该实数的最大整数(不大于该实数)。
8. 使用 Scanner 类,从键盘输入圆柱体的半径和高,计算该圆柱体的体积。



## 1.2 Java 语法基础



学习要点

### 学习效果检验

- Java 提供了哪几种基本数据类型?
- 各种基本数据类型的变量在程序中是如何赋值和使用的?
- 如何理解运算符的优先级和结合性?
- 表达式中变量或常量的类型不一致时, 如何处理类型的转换?
- Math 类和 Random 类的常用方法有哪些? 如何使用这些方法?

### 一、判断题

1. 基本数据类型的变量在声明时, 系统会给它们分配相应的存储空间。( )
2. 引用型数据类型的变量在声明时, 系统不会为它们分配相应的存储空间。( )
3. Java 程序与平台和操作系统有关。( )
4. 在 Java 语言中只能使用 ASCII 码。( )
5. 在 Java 语言中字符编码只能使用 Unicode 码。( )
6. 在 Java 语言中可以同时使用 ASCII 码和 Unicode 码。( )
7. 在 Java 语言中, 基本的数据类型有 4 种, String 属于基本的数据类型。( )
8. Java 语言的基本数据类型有 4 种: 整型、浮点型、布尔型和字符型。( )
9. 布尔型变量只有两个值 true 和 false, 且它们不对应于任何整数值。( )
10. 在 Java 中, 0 表示 false, 非 0 表示 true。( )
11. 布尔型常量可以自动转换成为短整型常量。( )
12. 在 Java 语言中要声明一个标识符并且定义为常量, 只能用关键字 final 来实现。( )
13. 语句“char a = 78;”可通过编译, 因为编译器会隐含地将 78 强制转换为 char 型。( )
14. 在 Java 语言中, 1 和 1.0 是两个相同类型的值, 只是写法不同。( )
15. 在 Java 语言中, 小数会被默认为 double 类型的值。( )
16. 在 Java 语言中 'a' 和 "a" 是两个相同类型的字符, 只是写法不同。( )
17. 语句“b=a++;”相当于两条语句的作用。( )
18. Java 程序中的变量均需先声明, 然后才能使用。( )
19. char 型变量的默认值是 '\u0000'。( )
20. 字符型变量中只存放一个字符。( )
21. 语句“y\*=x+2;”等价于“y=y\*x+2;”。( )
22. Java 对逻辑与 (&&) 和逻辑或 (||) 提供了短路操作功能。( )
23. 从占用字节多的数据类型转换到占用字节少的数据类型, 必须使用强制转换数据类型。  
( )
24. 数据由高类型转换到低类型时, 采用强制转换, 数据精度不会受到损失。( )
25. 所有逻辑运算符的优先级别都高于算术运算符。( )
26. Java 语言使用的是 Unicode 字符集, 每个字符在内存中占 8 位。( )