

零壹快学程序设计系列丛书

书网合一

小白实战手册

零壹快学

零基础Java 从入门到精通

立体化教学模式 零基础快速入门 零壹快学 编著

真人教学，手把手教你学Java

- 丰富示例，贴近场景——丰富的代码示例，编程场景化
- 视频教学，动手操作——扫码即可学习配套视频，通俗易懂
- 线上问答，及时解惑——在线论坛，解答学习中遇到的疑问
- 海量题库，随时学习——大量题目练习，加快理解

SPM 零基础快学系列 广东人民出版社



零基础Java 从入门到精通

零壹快学 编著

SPM 南方出版传媒 广东人民出版社

·广州·

图书在版编目 (CIP) 数据

零基础Java从入门到精通 / 零壹快学编著. —广州: 广东人民出版社, 2019.8
ISBN 978-7-218-13614-1


I. ①零… II. ①零… III. ①JAVA语言—程序设计 IV. ①TP312.8

中国版本图书馆CIP数据核字 (2019) 第111490号

Ling Jichu Java Cong Rumen Dao Jingtong

零 基 础 J a v a 从 入 门 到 精 通

零壹快学 编著

 版权所有 翻印必究

出 版 人：肖风华

统筹策划：李婧玮

责任编辑：严耀峰

封面设计：画画鸭工作室

内文设计：奔流文化

责任技编：周 杰 吴彦斌

出版发行：广东人民出版社

地 址：广州市海珠区新港西路204号2号楼（邮政编码：510300）

电 话：（020）85716809（总编室）

传 真：（020）85716872

网 址：<http://www.gdpph.com>

印 刷：佛山市浩文彩色印刷有限公司

开 本：787毫米×1092毫米 1/16

印 张：35.5 字 数：400千

版 次：2019年8月第1版 2019年8月第1次印刷

定 价：85.00元

如发现印装质量问题，影响阅读，请与出版社（020-32449105）联系调换。

售书热线：020-32449123

零壹快学

《零基础 Java 从入门到精通》编委会

主 编：吕鉴倬

副 主 编：宋宏宇 楚朋志 焦智超

编委会成员：景悦诚 李延杰 蔡宇轩 郑显军

前言

历经七十多年的发展，无论是对于国内数以十万计的学习者而言，还是在有着多年培训经验的编者看来，学习编程语言，仍存在不小的难度，甚至有不少学习者因编程语言的复杂多变、难度太大而选择了中途放弃。实际上，只要掌握了其变化规律，即使再晦涩难懂的计算机专业词汇也无法阻挡学习者的脚步。对于初学者来说，若有一本能看得懂，甚至可以用于自学的编程入门书是十分难得的。为初学者提供这样一本书，正是我们编写本套丛书的初衷。

零壹快学以“零基础，一起学”为主旨，针对零基础编程学习者的需求和学习特点，由专业团队量身打造了本套计算机编程入门教程。本套丛书的作者都从事编程教育和培训工作多年，拥有丰富的一线教学经验，对于学习者常遇到的问题十分熟悉，在编写过程中针对这些问题花费了大量的时间和精力来加以阐释，对书中的每个示例反复推敲，加以取舍，按照学习者的接受程度雕琢示例涉及的技术点，力求成就一套真正适合初学者的编程书籍。

本套丛书涵盖了Java、PHP、Python、JavaScript、HTML、CSS、Linux、iOS、C语言、C++、C#等计算机语言，同时借助大数据和云计算等技术，为广大编程学习者提供计算机各学科的视频课程、在线题库、测评系统、互动社区等学习资源。

◆ 课程全面，聚焦实战

本套丛书涵盖多门计算机语言，内容全面、示例丰富、图文并茂，通过通俗易懂的语言讲解相关计算机语言的特性，以点带面，突出开发技能的培养，既方便学习者了解基础知识点，也能帮助他们快速掌握开发技能，为编程开发设计积累实战经验。

◆ 专业团队，紧贴前沿

本套丛书作者由一线互联网公司高级工程师、知名高校教师和研究所技术人员等组成，线上线下同步进行专业讲解及点评分析，为学习者扫除学习障碍。与此同时，团队

在内容研发方向上紧跟当前技术领域热点，及时更新，直击痛点和难点。

- ◆ **全网覆盖，应用面广**

本套丛书已全网覆盖Web、APP和微信小程序等客户端，为广大学习者提供包括计算机编程、人工智能、大数据、云计算、区块链、计算机等级考试等在内的多门视频课程，配有相关测评系统和技术交流社区，互动即时性强，可实现在线教育随时随地轻松学。

Java是全球最流行的编程语言之一，被全世界各大软件公司广泛使用。与其他编程语言相比，Java语言几乎是所有联网应用的基础，因此Java拥有庞大且成熟的生态环境，从PC端到移动端，都可以看到Java语言的身影。

本书基于最新的Java 8编写而成，摒弃了一些低版本中不常用的知识，循序渐进地对Java进行讲解。读者在学习完本书之后，可以根据教程搭建知名的Spring框架，对本书的学习做一个总结。对于零基础或者基础比较薄弱的读者，本书可以作为Java的快速入门教材。我们衷心希望本书能为各位读者提供切实的帮助。

• 本书内容

- ◆ **基础知识**：第1~3章，主要介绍了Java的概况、安装以及常见配置、Java的基础语法，帮助读者打好基础，快速进入Java的学习之中。

- ◆ **核心技术**：第4~12章，主要介绍Java的核心应用，包括函数方法、流程控制和语言结构、字符串、数组、正则表达式、面向对象编程、Java常用类、Java集合类和反射与注解，帮助读者掌握Java的核心操作。

- ◆ **高级应用**：第13~20章，主要介绍Java日期和时间、Java I/O、异常处理、多线程与并发、MySQL数据库、JDBC操作、加密技术和Spring框架，帮助读者向更高层次的Java应用操作迈进。

• 本书特点

- ◆ **由浅入深，过程完善**。本书结合实际学习经验，对Windows、Linux和Mac三种环境下的Java安装都进行了详细的讲解，先介绍基本操作，帮助读者尽快入门，再逐步展开Java的每个知识点，配合完整的操作步骤，提升读者的学习成就感。

- ◆ **示例丰富，贴近场景**。本书提供了丰富的操作示例，而且每个操作均有文字解释，便于读者清晰理解操作命令的含义。这些示例大多选自工作中的各类场景，让读者可以感受日常工作中Java的运用，提高分析排查问题的能力，增加实战操作经验。

- ◆ **视频教学，动手操作。**本书每一章都配有教学视频，直观展示了Java编程的运行效果，并配有通俗易懂的解释。

- ◆ **知识拓展，难度提升。**本书非常全面地讲解了Java实用的知识点，同时在每一章末尾设有“小结”和“知识拓展”。通过在“知识拓展”部分中列举一些重要或有一定难度的知识点，为有能力的读者提供更多的拓展类学习内容，多维度强化自身的学习，加深对Java的理解。

- ◆ **线上问答，及时解惑。**本书为确保广大读者的学习能够顺利进行，提供了在线答疑服务，希望通过这种方式及时解决读者在学习Java的过程中所遇到的困难和疑惑。

• 本书配套资源（可扫下方二维码获取）

- ◆ **大量的代码示例。**通过运行这些代码，读者可以进一步巩固所学的知识。
- ◆ **零壹快学官方视频教程。**力求让读者学以致用，加强实战能力。
- ◆ **在线答疑。**为读者解惑，帮助读者解决学习中的困难，快速掌握要点。

• 本书适用对象

- ◆ 编程的初学者、爱好者与自学者
- ◆ 高等院校和培训学校的师生
- ◆ 职场新人
- ◆ 准备进入互联网行业的再就业人群
- ◆ “菜鸟”程序员
- ◆ 初、中级程序开发人员



零壹快学微信公众号

《零基础Java从入门到精通》从初学者角度出发，详细讲述了Java编程语言所有的基础知识点和开发实战中需要的必备技能。全书内容通俗易懂，示例丰富，步骤清晰，图文并茂，可以使读者轻松掌握Java的精髓，活学活用，是一本实用的Java入门书，也是在开发实战中必备的Java参考手册。

编者

2019年7月

目 录

CONTENTS

第 1 章 走进 Java	1	2.4 小结	26
1.1 Java 编程语言概述	1	2.5 知识拓展	26
1.1.1 Java 的历史	1	第 3 章 Java 基础语法	29
1.1.2 Java 的发展历程	2	3.1 Java 主类结构	29
1.1.3 使用场景和优势	3	3.1.1 Java 包	30
1.1.4 Java 6 和 Java 8	4	3.1.2 类的成员变量和局部变量	31
1.2 如何学好 Java	5	3.1.3 访问权限修饰词	32
1.2.1 Java 语言特性	5	3.1.4 编写主方法	32
1.2.2 第一个 Java 程序	6	3.2 注释及使用场景	33
1.2.3 学好 Java 的建议	7	3.2.1 注释文档	33
1.3 Java API 文档	8	3.2.2 嵌入 HTML 语言和标签	34
1.4 Web 项目介绍	9	3.3 基本数据类型	35
1.5 网站开发基本流程	10	3.3.1 整数类型	35
1.6 小结	11	3.3.2 浮点类型	41
1.7 知识拓展	11	3.3.3 字符类型	44
1.7.1 常用软件资源	11	3.3.4 布尔类型	47
1.7.2 Java 开发社区	12	3.3.5 引用类型对象	48
1.7.3 Java 10	13	3.3.6 特殊值 null	49
第 2 章 Java 配置安装和 IDE 介绍 ..	14	3.4 数据类型之间的转换	49
2.1 Windows 下搭建 Java 环境	14	3.4.1 自动转换	49
2.1.1 JDK 下载与安装	14	3.4.2 强制转换	51
2.1.2 配置 JDK 环境	16	3.5 常量与变量	53
2.2 Mac 下搭建 Java 环境	19	3.5.1 常量	53
2.3 Java IDE——Eclipse	19	3.5.2 变量	56
2.3.1 Eclipse 下载与安装	20	3.6 操作运算符	58
2.3.2 Eclipse 使用	23	3.6.1 算术运算符	58

3.6.2	比较运算符	59	5.3.4	return 语句	116
3.6.3	赋值运算符	60	5.4	小结	117
3.6.4	递增运算符和递减运算符	62	5.5	知识拓展	117
3.6.5	逻辑运算符	63	第 6 章 字符串	119	
3.6.6	三元运算符	64	6.1	字符串 String 类	119
3.6.7	位运算符	65	6.1.1	创建字符串	119
3.6.8	instanceof 运算符	66	6.1.2	初始化字符串	121
3.6.9	运算符优先级	66	6.2	字符串常见操作	127
3.7	表达式	67	6.2.1	字符串连接	127
3.8	小结	68	6.2.2	字符串长度	130
3.9	知识拓展	68	6.2.3	查找字符串	131
3.9.1	编码规范的建议	68	6.2.4	字符串替换	135
3.9.2	Java 关键字	69	6.2.5	字符串截取	139
第 4 章 Java 方法	72		6.2.6	字符串分割	140
4.1	方法的概念	72	6.2.7	字符串首尾内容判断	142
4.2	方法定义和使用	72	6.2.8	字符串首尾去空格	143
4.2.1	方法参数	74	6.2.9	字符串大小写转换	144
4.2.2	方法返回值	78	6.2.10	字符串比较	146
4.2.3	方法类型声明	79	6.2.11	字符串格式化输出	149
4.2.4	命令行参数使用	80	6.2.12	其他字符串操作	154
4.3	可变参数方法	81	6.3	StringBuilder 类与 StringBuffer 类	158
4.4	小结	83	6.3.1	StringBuilder 类	158
4.5	知识拓展	83	6.3.2	StringBuffer 类	165
4.5.1	Java 内置类和内置方法介绍	83	6.4	小结	165
4.5.2	有趣的方法自身调用	85	6.5	知识拓展	166
第 5 章 流程控制和语言结构	88		第 7 章 数组	168	
5.1	条件控制语句	88	7.1	数组介绍	168
5.1.1	if 和 else 语句	88	7.1.1	什么是数组	168
5.1.2	switch 语句	98	7.1.2	数组的构成	168
5.2	循环控制语句	103	7.2	数组创建	169
5.2.1	for 循环语句	103	7.2.1	数组创建方法	169
5.2.2	while 循环语句	108	7.2.2	数组索引	171
5.3	跳转语句	110	7.2.3	多维数组	171
5.3.1	continue 语句	110	7.3	数组的遍历与输出	174
5.3.2	break 语句	113	7.3.1	foreach 遍历	174
5.3.3	goto 语句	116			

7.3.2	for 遍历	175	9.1.1	对象	212
7.3.3	Arrays 类中的 toString 静态方法	176	9.1.2	类	214
7.4	数组常见操作	177	9.2	Java 与面向对象	215
7.4.1	数组长度	177	9.2.1	类的声明	215
7.4.2	向数组添加元素	179	9.2.2	类的实例化	217
7.4.3	删除数组元素	181	9.2.3	成员属性	218
7.4.4	删除重复数据	182	9.2.4	成员方法	219
7.4.5	数组查找	184	9.2.5	访问成员的属性和方法	221
7.4.6	数组排序	186	9.2.6	变量作用域	223
7.4.7	数组复制	188	9.2.7	对象的应用	226
7.4.8	数组比较	189	9.2.8	修饰符关键字	228
7.5	小结	190	9.2.9	静态常量	228
7.6	知识拓展	191	9.2.10	静态变量	229
			9.2.11	静态方法	231
			9.2.12	静态代码块	232
第 8 章	正则表达式	193	9.3	构造方法	233
8.1	正则表达式介绍	193	9.4	类的继承和多态	234
8.2	正则表达式语法	194	9.4.1	继承	234
8.2.1	普通字符	194	9.4.2	多态	244
8.2.2	字符转义	195	9.5	高级特性	250
8.2.3	元字符	196	9.5.1	final 的使用	250
8.2.4	限定符	197	9.5.2	对象克隆	254
8.2.5	定位符	198	9.6	抽象类与接口	266
8.2.6	分组构造	199	9.6.1	抽象类	266
8.2.7	匹配模式	199	9.6.2	接口	270
8.3	Java 处理正则	200	9.7	小结	273
8.3.1	java.util.regex 包介绍	200	9.8	知识拓展	273
8.3.2	Pattern 类	201	9.8.1	MVC 设计模式	273
8.3.3	Matcher 类	202	9.8.2	单例设计模式	277
8.3.4	PatternSyntaxException 类	203			
8.4	小结	204	第 10 章	Java 常用类	280
8.5	知识拓展	204	10.1	包装类	280
8.5.1	贪婪与非贪婪匹配	204	10.1.1	Integer 整型类	282
8.5.2	零宽断言	206	10.1.2	Double 浮点型类	286
8.5.3	常用正则表达式参考	207	10.1.3	Boolean 布尔型类	289
			10.1.4	Character 字符型类	292
			10.1.5	高精度数字类	295
第 9 章	面向对象编程	210			
9.1	面向对象介绍	210			

10.1.6	Number 数字类	296	12.1.4	获取方法的信息	342
10.1.7	Void 类	296	12.2 注解	346	
10.2 Math 类	297		12.2.1	元注解	346
10.2.1	Math 类中的常量	297	12.2.2	内置注解介绍	347
10.2.2	Math 类中的常见方法	297	12.2.3	自定义注解	350
10.2.3	随机数	302	12.3 小结	353	
10.3 枚举	304		12.4 知识拓展	353	
10.3.1	枚举定义	305	12.4.1	Spring 注解	353
10.3.2	枚举的常见方法	305	12.4.2	动态代理	354
10.3.3	枚举集合	308	第 13 章 Java 日期和时间	357	
10.4 泛型	308		13.1	概述	357
10.4.1	泛型类	308	13.2	Date 类	359
10.4.2	泛型方法	310	13.3	Calendar 类	361
10.5 小结	311		13.4	DateFormat 类	364
10.6 知识拓展	312		13.5	SimpleDateFormat 类	366
10.6.1	Java 对象生命周期	312	13.6	小结	369
10.6.2	Java 中常用类库介绍	313	13.7 知识拓展	369	
第 11 章 Java 集合类	314		13.7.1	时区划分	369
11.1	什么是集合类	314	13.7.2	Unix 时间戳	369
11.2	Collection 接口	314	13.7.3	Java 和 Unix 时间戳	370
11.3	List 集合	316	第 14 章 Java I/O	372	
11.3.1	ArrayList 类	316	14.1	输入 / 输出流	372
11.3.2	LinkedList 类	319	14.1.1	什么是流	372
11.4 Set 集合	321		14.1.2	输入流	373
11.4.1	HashSet 类	321	14.1.3	输出流	376
11.4.2	TreeSet 类	323	14.1.4	系统预定义流	380
11.5 Map 集合	325		14.2 File 类	381	
11.5.1	HashMap	327	14.2.1	创建 File 文件	381
11.5.2	TreeMap	328	14.2.2	File 文件基本操作	381
11.6 小结	330		14.3 文件输入 / 输出流	386	
11.7 知识拓展	330		14.3.1	FileInputStream 类和 FileOutput- Stream 类	386
第 12 章 Java 反射与注解	335		14.3.2	FileReader 类和 FileWriter 类	388
12.1	Java 反射	335	14.4 缓存输入 / 输出流	391	
12.1.1	java.lang.Class 类介绍	335	14.4.1	BufferedInputStream 类和 Buffered- OutputStream 类	391
12.1.2	获取构造方法的信息	337			
12.1.3	获取成员变量的信息	340			

14.4.2	BufferedReader 类和 BufferedWriter 类	394
14.5	数据输入 / 输出流	398
14.6	Java 序列化	402
14.6.1	序列化概述	402
14.6.2	ObjectOutputStream 序列化	403
14.6.3	ObjectInputStream 反序列化	405
14.7	小结	407
14.8	知识拓展	407
14.8.1	使用 POI 类库处理 Excel 文件	407
14.8.2	使用 GZIP 对文件进行压缩	411
第 15 章	Java 异常处理	413
15.1	什么是异常处理	413
15.1.1	Error 系统异常	414
15.1.2	Exception 抛出异常	416
15.1.3	异常方法	418
15.2	异常处理关键字	418
15.2.1	throw 和 throws 关键字	418
15.2.2	try catch 关键字	421
15.2.3	finally 关键字	423
15.3	常见异常	424
15.3.1	NullPointerException	424
15.3.2	ClassNotFoundException	426
15.3.3	NumberFormatException	426
15.3.4	IllegalArgumentException	427
15.3.5	NoSuchMethodException	428
15.3.6	ClassCastException	429
15.4	自定义异常	430
15.5	小结	431
15.6	知识拓展	431
第 16 章	多线程与并发	436
16.1	Java 与线程	436
16.1.1	线程基本概念	436
16.1.2	Java 线程机制	437
16.2	线程初始化和调用	437
16.2.1	继承 Thread 类	437
16.2.2	实现 Runnable 接口	441
16.2.3	实现 Callable 和 Future 接口	443
16.3	线程生命周期	445
16.4	Java 操作线程	449
16.4.1	加入线程	449
16.4.2	休眠线程	450
16.4.3	中断线程	451
16.5	线程的同步	453
16.5.1	线程安全	453
16.5.2	线程同步机制	454
16.5.3	线程暂停与恢复	457
16.6	小结	458
16.7	知识拓展	458
第 17 章	MySQL 数据库	462
17.1	MySQL 介绍	462
17.2	MySQL 工具介绍	462
17.2.1	MySQL 控制台客户端	462
17.2.2	MySQL Workbench 软件	463
17.3	数据库管理	463
17.3.1	创建数据库	463
17.3.2	选择数据库	464
17.3.3	查看数据库	465
17.3.4	修改数据库	466
17.3.5	删除数据库	466
17.4	字段类型	467
17.4.1	数值类型	467
17.4.2	字符串类型	469
17.4.3	时间类型	469
17.5	数据表操作	470
17.5.1	创建数据表	470
17.5.2	查看数据表	473
17.5.3	修改数据表	474
17.5.4	删除数据表	477
17.6	数据库语句	478

17.6.1	新增数据	478	18.5.2	JDBC 事务回滚	511
17.6.2	查询数据	478	第 19 章 Java 中的加密技术		
17.6.3	修改数据	479	19.1	加密技术概述	514
17.6.4	删除数据	480	19.1.1	加密技术介绍	514
17.6.5	replace 操作	480	19.1.2	对称加密算法	516
17.7	数据表字符集	481	19.1.3	非对称加密算法	516
17.7.1	查看字符集	481	19.1.4	数字签名	517
17.7.2	设置字符集	484	19.2	Java 加密技术	518
17.7.3	处理乱码	485	19.2.1	使用 MD5 加密	518
17.8	数据库索引	485	19.2.2	使用 SHA 加密	523
17.8.1	索引介绍	485	19.2.3	使用 DES 加密	524
17.8.2	唯一索引	486	19.2.4	使用 AES 加密	526
17.8.3	普通索引	486	19.2.5	使用 RSA 加密	529
17.9	小结	487	19.3	加密技术使用场景	535
17.10	知识拓展	487	19.3.1	密码存储	535
第 18 章 JDBC 操作 MySQL 数据库			19.3.2	base64 加密	535
		491	19.4	小结	538
18.1	JDBC 介绍	491	19.5	知识拓展	538
18.2	JDBC 中的常用类	492	19.5.1	密码学之父	538
18.2.1	DriverManager 类	492	19.5.2	万维网的发展	538
18.2.2	Connection 接口	494	19.5.3	Hash 在密码学中的应用	538
18.2.3	Statement 接口	495	19.5.4	加盐算法	539
18.2.4	PreparedStatement 接口	496	第 20 章 Spring 实战		
18.2.5	ResultSet 接口	497	20.1	Spring 概述	540
18.3	JDBC 操作 MySQL	499	20.1.1	Spring 介绍	540
18.3.1	JDBC 创建数据表	499	20.1.2	Spring 模块	541
18.3.2	JDBC 向数据表添加数据	500	20.2	使用 Spring Boot 搭建 RESTful 服务	542
18.3.3	JDBC 修改数据	503	20.3	使用 Spring Data JPA 访问数据 库	545
18.3.4	JDBC 删除数据	505	20.4	小结	551
18.3.5	JDBC 查询数据	508	20.5	知识拓展	551
18.4	小结	508			
18.5	知识拓展	509			
18.5.1	JDBC 批量处理	509			

» 第 1 章

走进Java <<

1.1 Java编程语言概述

当今社会，软件的应用已经渗透到生活的方方面面。我们所使用的在线服务，比如打车、交友、聊天、办公、学习、游戏等，其应用都是通过各种各样的编程语言开发来完成的。

如今每一种被广泛使用的编程语言，都在某一些场景下有不可替代的长处或者突出的优势。比如，C语言在性能方面非常好，R语言擅长大量数据的统计分析，HTML和JavaScript语言在浏览器场景中有不可忽视的优势。而在众多编程语言中，Java是最受欢迎的编程语言之一，同时享有最高的市场占有率，被全世界各大软件公司广泛使用。其原因在于，Java语言几乎是所有类型的联网应用的基础，拥有庞大且成熟的生态环境，从PC端到移动端，都可以看到Java语言的身影。

本章将带你走进Java语言，体会不一样的编程世界。

1.1.1 Java的历史

Java是由Sun Microsystems公司（又称SUN公司）于1995年5月推出的Java程序设计语言和Java平台的总称。它是由詹姆斯·高斯林（James Gosling）和他的同事们共同研发，并在1995年正式推出的。



图1.1.1 Java图标

Java得名于该项目成员组在讨论这个新语言的取名时所喝的Java（爪哇）咖啡，从Java的图标（图1.1.1）也可以窥见其由来。不过也有人声称这一名字是开发人员名字的组合：James Gosling、Arthur Van Hoff（阿瑟·凡·霍夫）、Andy Bechtolsheim（安迪·贝克托克姆），或者是“Just Another Vague Acronym”（意思是“只是另一个含糊的首字母缩略词”）的缩写。

Java语言的前身其实是最早诞生于1991年的Oak语言，是SUN公司为一些消费性电子产品而设计的一个通用环境。而Oak语言最初的目的只是为了开发一种独立于平台的软件技术。这项技术自成一格，但是在网络出现之前，它的优势并没有被人们重视，它甚至差点被SUN公司放弃，直到后来网络技术的普及改变了它的命运。

最开始，Internet上的信息内容大多是乏味死板的HTML文档。这对于那些迷恋于Web浏览的技

术人员来说，意味着低效和重复的工作。对于能在Web中看到一些交互式的内容的市场需求日渐迫切，开发人员开始寻求能够在Web上创建一类无须考虑软硬件平台就可以执行的应用程序，以及性能更加优秀且极大保障程序安全的编程语言。SUN的工程师敏锐地察觉到了这一点，从1994年起，他们开始将Oak技术应用于Web上，并且开发出了HotJava的第一个版本。

当SUN公司于1995年正式以“Java”这个名字推出Java时，几乎所有的Web开发人员都立马折服于Java的魅力：跨平台、动态的Web、Internet计算。从此，Java被广泛接受并推动了Web的迅速发展，常用的浏览器现在均支持Java Applet。另一方面，Java技术也不断更新。1996年1月，SUN公司成立了Java业务集团，专门开发Java技术，并发布了Java历史上第一个开发工具包——JDK 1.0，标志着Java成为一种独立的开发工具。

1.1.2 Java的发展历程

本节列举了Java编程语言这些年的发展历程，感兴趣的读者可以通过拓展资料做深入了解，本书内容不做展开：

1995年5月，Java语言诞生。

1996年1月，JDK 1.0发布。

1996年5月，首届Java One大会在美国圣弗朗西斯科召开。

1996年9月，Java语言已普及约8.3万个网页应用。

1997年2月，JDK 1.1发布，引入JDBC、JavaBeans、内部类等。

1997年9月，Java Developer Connection社区壮大，会员超过10万。

1998年2月，JDK 1.1被下载超过200万次。

1998年12月，企业平台J2EE发布。

1999年6月，SUN公司发布Java的三个版本：标准版（J2SE）、企业版（J2EE）和微型版（J2ME）。

2000年5月，JDK 1.3和1.4版本发布，对Java性能进行了大量优化和增强。

2001年9月，J2EE 1.3版本发布。

2002年2月，J2SE 1.4发布，引入了XML处理、断言、正则表达式等多种特性。

2003年6月，Java One大会上发布JDO（Java对象持久化的新规范）。

2004年9月，研发三年的J2SE 1.5发布，并更名为Java SE 5.0。

2005年6月，Java技术诞生十周年，SUN公司在Java One大会上发布了Java SE 6。J2EE更名为Java EE，J2SE更名为Java SE，J2ME更名为Java ME。

2006年12月，SUN公司发布JRE 6.0。

2009年4月，谷歌公司App引擎支持Java，Oracle公司收购SUN公司，取得Java版权。

2011年7月，Java SE 7正式版发布，增加了简单闭包功能。

2014年3月，Java SE 8正式版发布，引入Lambda表达式。

2017年9月，Java SE 9正式版发布，引入了模块系统、JShell，支持HTTP2。

2018年3月，Java SE 10正式版发布，推出“局部变量类型推断”功能，改进垃圾回收。

2018年9月，Java SE 11正式版发布。

2019年3月，Java SE 12正式版发布。

1.1.3 使用场景和优势

Java是一种跨平台、适合于分布式计算环境的面向对象编程语言。在刚提出之时（1995年），它跟当时的主流语言C和C++相比，有如下优势：

- ◇ 面向对象，提出接口，不支持多重继承，避免歧义。
- ◇ 跨平台。那时C/C++编写的程序只适用于一个操作系统上，为了实现跨平台，需要针对编译器不同、类库不同等进行大量的移植工作。
- ◇ 语言质朴，可以快速被C/C++开发人员掌握。
- ◇ 不支持指针操作、宏等，减少出现bug的机会。
- ◇ 通过实现垃圾自动收集，大大简化了程序设计人员的资源释放管理工作。

除了语言优势，不得不提的是Java推出了一系列API文档和第三方开发包，直接增加了各大厂商自行研发的数据库和消息中间件，大尺度地实现了跨平台语言开发。当时在Windows平台上，没有一项编程语言可以与Java相媲美，实现这种程度的跨平台。

由于这些优势，在Java提出之时，很多应用程序开发人员都开始改用Java，Java同时也获得了众多大厂的支持，比如Oracle、Microsoft、IBM等。

在发展的过程中，Java推出了Servlet，采用线程模型编写Web服务端程序，更是推出了企业分布式开发的技术J2EE。此外，它还有更多的API文档，并开源了大量基于Java的开源项目。

发展到今天，Java的应用场景变得更加多元和庞大。相比于其他各大主流编程语言，Java的优势体现在：比C/C++更简单和高效，适合用于开发操作系统和应用程序；比Python/Ruby更适合上规模的开发，其静态类型本身就可以作为文档，方便快捷，并且运行性能更好；比C#拥有更大的社区和开源软件资源。

目前，Java编程语言在全世界被大范围地使用，应用于各类场景中：

- ◇ 电子商务软件。
- ◇ Web应用系统开发。
- ◇ 互联网业务后台系统。
- ◇ 企业级网站开发。
- ◇ 教育平台。
- ◇ 办公系统。

- ◇ 移动互联网开发等。
- ◇ 大型分布式系统后台开发。

Java已经深入到开发者身边的各种应用场景中，如他们经常使用的开发工具NetBeans、JBuilder等，还有Azureus、CyberDuck、OpenOffice等常用的软件都是通过Java语言编程实现的。

1.1.4 Java 6和Java 8

Java语言自诞生之日起，就一直在不停地更新和升级。目前，Java推出了最新的Java 12版本，并推出了一系列最新的特性，这些特性有的令人欣喜，有的暂时没有办法知道是否实用，有的则需要待时间考证。不过Java生态的强大生命力已经毋庸置疑。

目前应用最广泛且最成熟的版本，依然是Oracle在2014年3月发布的Java 8正式版（又称为JDK 1.8），它的发布时间与Java 7相隔了近三年。在此之前发布时间间隔最长的版本是Java 7，它与Java 6相隔了五年，这是因为在当时两个版本发布中间发生了Oracle收购SUN公司的大事件。因此，Java 6曾是使用率最高的版本。

Java 8在原来Java 6和Java 7的基础上做了大量的改进和提升，新增了不少新特性。比如：

- ◇ Streams：集合（Collections）的改进也是Java 8的一大亮点，而让集合越来越好的核心组件则是“Stream”。它与java.io包里的InputStream和OutputStream是完全不同的全新概念，不要混淆了。

- ◇ 函数式接口：可以在接口里面添加默认方法，并且这些方法可以直接从接口中运行。

- ◇ Lambda：使用它设计的代码会更加简洁。当开发者在编写Lambda表达式时，也会随之被编译成一个函数式接口。

- ◇ Java time：拥有各种各样的时间API，可以处理一些时空连续体方面的特性，比如距离、质量、重量等。

- ◇ Nashorn：Nashorn是Rhino的接替者，该项目的目的是基于Java实现一个轻量级高性能的JavaScript运行环境。

- ◇ Accumulators：通过加法器（Adders）和累加器（Accumulators）基于原来的Java.util.concurrent令该性能得到了进一步的发展。

- ◇ HashMap修复：修复了在String.hashCode()使用中大家熟知的bug，通过采用平衡tree算法来降低复杂度。

Java 8包含了一系列非常实用的特性，本节只是列举了一部分。截至目前为止，Java最新版本为2019年3月发布的Java 12。至于该如何取舍，各位开发者应该根据自己的实际需求去研究和使用的。本书的编程环境将使用Java 8。