

Java 学习之路：不走弯路，就是捷径

0. 引言

软件开发之路是充满荆棘与挑战之路,也是充满希望之路。Java 学习也是如此,没有捷径可走。梦想像《天龙八部》中虚竹一样被无崖子醍醐灌顶而轻松获得一甲子功力,是很不现实的。每天仰天大叫"天神啊,请赐给我一本葵花宝典吧",殊不知即使你获得了葵花宝典,除了受自宫其身之苦外,你也不一定成得了"东方不败",倒是成"西方失败"的几率高一点。

"不走弯路,就是捷径",佛经说的不无道理。

1. 如何学习程序设计?

Java 是一种平台,也是一种程序设计语言,如何学好程序设计不仅仅适用于 Java,对 C++等其他程序设计语言也一样管用。有编程高手认为,Java 也好 C 也好没什么分别,拿来就用。为什么他们能达到如此境界?我想是因为编程语言之间有共通之处,领会了编程的精髓,自然能够做到一通百通。如何学习程序设计理所当然也有许多共通的地方。

1.1 培养兴趣

兴趣是能够让你坚持下去的动力。如果只是把写程序作为谋生的手段的话,你会活的很累,也太对不起自己了。多关心一些行业趣事,多想想盖茨。不是提倡天天做白日梦,但人要是没有了梦想,你觉得有味道吗?可能像许多深圳本地农民一样,打打麻将,喝喝功夫茶,拜拜财神爷;每个月就有几万十几万甚至更多的进帐,凭空多出个"食利阶层"。你认为,这样有味道吗?有空多到一些程序员论坛转转,你会发现,他们其实很乐观幽默,时不时会冒出智慧的火花。

1.2 慎选程序设计语言

男怕入错行,女怕嫁错郎。初学者选择程序设计语言需要谨慎对待。软件开发不仅仅是掌握一门编程语言了事,它还需要其他很多方面的背景知识。软件开发也不仅仅局限于某几个领域,而是已经渗透到了各行各业几乎每一个角落。

如果你对硬件比较感兴趣,你可以学习 C 语言/汇编语言,进入硬件开发领域。如果你对电信的行业知识及网络比较熟悉,你可以在 C/C++等之上多花时间,以期进入电信软件开发领域。如果你对操作系统比较熟悉,你可以学习 C/Linux 等等,为 Linux 内核开发/驱动程序开发/嵌入式开发打基础。

如果你想介入到应用范围最广泛的应用软件开发(包括电子商务电子政务系统)的话,你可以选择 J2EE 或.NET,甚至 LAMP 组合。每个领域要求的背景知识不一样。做应用软件需要对数据库等很熟悉。总之,你需要根据自己的特点来选择合适你的编程语言。

1.3 要脚踏实地,快餐式的学习不可取

先分享一个故事。

有一个小朋友,他很喜欢研究生物学,很想知道那些蝴蝶如何从蛹壳里出来,变成蝴蝶便会飞。有一次,他走到草原上面看见一个蛹,便取了回家,然后看着,过了几天以后,这个蛹出了一条裂痕,看见里面的蝴蝶开始挣扎,想抓破蛹壳飞出来。这个过程达数小时之久,蝴蝶在蛹里面很辛苦地拼命挣扎,怎么也没法子走出来。这个小孩看着看着不忍心,就想不如让我帮帮它吧,便随手拿起剪刀在蛹上剪开,使蝴蝶破蛹而出。但蝴蝶出来以后,因为翅膀不够力,变得很臃肿,飞不起来。

这个故事给我们的启示是:欲速则不达。

浮躁是现代人最普遍的心态,能怪谁?也许是贫穷落后了这么多年的缘故,就像当年的大跃进一样,都想大步跨入***主义社会。现在的软件公司、客户、政府、学校、培训机构等等到处弥漫着浮躁之气。就拿我比较熟悉的大连大工 IT 职业培训来说吧,居然打广告宣称"20 多年的计算机职业教育,辽宁省十佳学校",殊不知中国计算机发展才几年,软件发展才几年,居然去报名的学生不少,简直是藐视天下程序员。培训出来的"程序员"大多不知道 OO,OP 为何物?社会环境如是,我们不能改变,只能改变自己,闹市中的安宁,弥足珍贵。许多初学者 C++/Java 没开始学,立马使用 VC/JBuilder,会使用 VC/JBuilder 开发一个 Hello World 程序,就忙不迭的向世界宣告,"我会软件开发了",简历上也大言不惭地写上"精通 VC/Java"。结果到软件

公司面试时要么被三两下打发走了，要么被驳的体无完肤，无地自容。到处碰壁之后才知道捧起《C++编程思想》《Java 编程思想》仔细钻研，早知如此何必当初呀。

"你现在讲究简单方便，你以后的路就长了"，好象也是佛经中的劝戒。

1.4 多实践,快实践

彭端淑的《为学一首示子侄》中有穷和尚与富和尚的故事。

从前，四川边境有两个和尚，一个贫穷，一个有钱。一天，穷和尚对富和尚说："我打算去南海朝圣，你看怎么样？"富和尚说："这里离南海有几千里远，你靠什么去呢？"穷和尚说："我只要一个水钵，一个饭碗就够了。"富和尚为难地说："几年前我就打算买条船去南海，可至今没去成，你还是别去吧！"一年以后，富和尚还在为租赁船只筹钱，穷和尚却已经从南海朝圣回来了。

这个故事可解读为：任何事情，一旦考虑好了，就要马上上路，不要等到准备周全之后，再去干事情。假如事情准备考虑周全了再上路的话，别人恐怕捷足先登了。软件开发是一门工程学科，注重的就是实践，"君子动口不动手"对软件开发人员来讲根本就是错误的，他们提倡"动手至上"，但别害怕，他们大多温文尔雅，没有暴力倾向，虽然有时候蓬头垢面的一副"比尔盖茨"样。有前辈高人认为，学习编程的秘诀是：编程、编程、再编程，笔者深表赞同。不仅要多实践，而且要快实践。我们在看书的时候，不要等到你完全理解了才动手敲代码，而是应该在看书的同时敲代码，程序运行的各种情况可以让你更快更牢固的掌握知识点。

1.5 多参考程序代码

程序代码是软件开发最重要的成果之一，其中渗透了程序员的思想与灵魂。许多人被《仙剑奇侠传》中凄美的爱情故事感动,悲剧的结局更有一种缺憾美。为什么要以悲剧结尾？据说是因为写《仙剑奇侠传》的程序员失恋而安排了这样的结局，他把自己的感觉融入到游戏中，却让众多的仙剑迷扼腕叹息。

多多参考代码例子，对 Java 而言有参考文献[4.3],有 API 类的源代码(JDK 安装目录下的 src.zip 文件)，也可以研究一些开源的软件或框架。

1.6 加强英文阅读能力

对学习编程来说，不要求英语，但不能一点不会，。最起码像 Java API 文档(参考文献[4.4])这些东西还是要能看懂的,连猜带懵都可以；旁边再开启一个"金山词霸"。看多了就会越来越熟练。在学 Java 的同时学习英文，一箭双雕多好。另外好多软件需要到英文网站下载，你要能够找到它们，这些是最基本的要求。英语好对你学习有很大的帮助。口语好的话更有机会进入管理层，进而可以成为剥削程序员的"周扒皮"。

1.7 万不得已才请教别人

笔者在 Martix 与 Java 论坛的在线辅导系统中解决学生问题时发现，大部分的问题学生稍做思考就可以解决。请教别人之前，你应该先回答如下几个问题。

你是否在 google 中搜索了问题的解决办法？

你是否查看了 Java API 文档？

你是否查找过相关书籍？

你是否写代码测试过？

如果回答都是"是"的话，而且还没有找到解决办法，再问别人不迟。要知道独立思考的能力对你很重要。要知道程序员的时间是很宝贵的。

1.8 多读好书

书中自有颜如玉。比尔盖茨是一个饱读群书的人。虽然没有读完大学，但九岁的时候比尔盖茨就已经读完了所有的百科全书，所以他精通天文、历史、地理等等各类学科，可以说比尔·盖茨不仅是当今世界上金钱的首富，而且也可以称得上是知识的巨富。

笔者在给学生们上课的时候经常会给他们推荐书籍，到后来学生实在忍无可忍开始抱怨，"天呐，这么多书到什么时候才能看完了"，"学软件开发，感觉上了贼船"。这时候，我的回答一般是，"别着急，什么时候带你们去看看我的书房，到现在每月花在技术书籍上的钱 400 元，这在软件开发人员之中还只能够算是中等的"，学生当场晕倒。（注：这一部分学生是刚学软件开发的）

1.9 使用合适的工具

工欲善其事必先利其器。软件开发包含各种各样的活动，需求收集分析、建立用例模型、建立分析设计模型、编程实现、调试程序、自动化测试、持续集成等等，没有工具帮忙可以说是寸步难行。工具可以提高开发效率，使软件的质量更高 BUG 更少。组合称手的武器。到飞花摘叶皆可伤人的境界就很高了，无招胜有招，手中无剑心中有剑这样的境界几乎不可企及。

2. 软件开发学习路线

两千多年的儒家思想孔孟之道，中庸的思想透入骨髓，既不冒进也不保守并非中庸之道，而是找寻学习软件开发的正确路线与规律。

从软件开发人员的生涯规划来讲，我们可以大致分为三个阶段，软件工程师→软件设计师→架构设计师或项目管理师。不想当元帅的士兵不是好士兵，不想当架构设计师或项目管理师的程序员也不是好的程序员。我们应该努力往上走。让我们先整理一下开发应用软件需要学习的主要技术。

- A. 基础理论知识，如操作系统、编译原理、数据结构与算法、计算机原理等，它们并非不重要。如不想成为计算机科学家的话，可以采取"用到的时候再来学"的原则。
- B. 一门编程语言，现在基本上都是面向对象的语言，Java/C++/C#等等。如果做 WEB 开发的话还要学习 HTML/JavaScript 等等。
- C. 一种方法学或者说思想，现在基本都是面向对象思想（OOA/OOD/设计模式）。由此而衍生的基于组件开发 CBD/面向方面编程 AOP 等等。
- D. 一种关系型数据库，ORACLE/SqlServer/DB2/MySQL 等等
- E. 一种提高生产率的 IDE 集成开发环境 JBuilder/Eclipse/VS.NET 等。
- F. 一种 UML 建模工具，用 ROSE/MISIO/钢笔进行建模。
- G. 一种软件过程，RUP/XP/CMM 等等，通过软件过程来组织软件开发的众多活动，使开发流程专业化规范化。当然还有其他的一些软件工程知识。
- H. 项目管理、体系结构、框架知识。

正确的路线应该是：B→C→E→F→G→H。

还需要补充几点：

- 1). 对于 A 与 C 要补充的是，我们应该在实践中逐步领悟编程理论与编程思想。新技术虽然不断涌现，更新速度令人眼花缭乱雾里看花；但万变不离其宗，编程理论与编程思想的变化却很慢。掌握了编程理论与编程思想你就会有拨云见日之感。面向对象的思想在目前来讲是相当关键的，是强势技术之一，在上面需要多投入时间，给你的回报也会让你惊喜。
- 2). 对于数据库来说是独立学习的，这个时机就由你来决定吧。
- 3). 编程语言作为学习软件开发的主线，而其余的作为辅线。
- 4). 软件工程师着重于 B、C、E、D；软件设计师着重于 B、C、E、D、F；架构设计师着重于 C、F、H。

3. 如何学习 Java?

3.1 Java 学习路线

3.1.1 基础语法及 Java 原理

基础语法和 Java 原理是地基，地基不牢靠，犹如沙地上建摩天大厦，是相当危险的。学习 Java 也是如此，必须要有扎实的基础，你才能在 J2EE、J2ME 领域游刃有余。参加 SCJP（SUN 公司认证的 Java 程序员）考试不失为一个好方法，原因之一是为了对得起你交的 1200 大洋考试费，你会更努力学习，原因之二是 SCJP 考试能够让你把基础打得很牢靠，它要求你跟 JDK 一样熟悉 Java 基础知识；但是你千万不要认为考过了 SCJP 就有多了不起，就能够获得软件公司的青睐，就能够获取高薪，这样的想法也是很危险的。获得"真正"的 SCJP 只能证明你的基础还过得去，但离实际开发还有很长的一段路要走。

3.1.2 OO 思想的领悟

掌握了基础语法和 Java 程序运行原理后,我们就可以用 Java 语言实现面向对象的思想了。面向对象,是一种方法学;是独立于语言之外的编程思想;是 CBD 基于组件开发的基础;属于强势技术之一。当以后因工作需要转到别的面向对象语言的时候,你会感到特别的熟悉亲切,学起来像喝凉水这么简单。

使用面向对象的思想进行开发的基本过程是:

- 调查收集需求。
- 建立用例模型。
- 从用例模型中识别分析类及类与类之间的静态动态关系,从而建立分析模型。
- 细化分析模型到设计模型。
- 用具体的技术去实现。
- 测试、部署、总结。

3.1.3 基本 API 的学习

进行软件开发的时候,并不是什么功能都需要我们去实现,也就是经典名言所说的"不需要重新发明轮子"。我们可以利用现成的类、组件、框架来搭建我们的应用,如 SUN 公司编写好了众多类实现一些底层功能,以及我们下载过来的 JAR 文件中包含的类,我们可以调用类中的方法来完成某些功能或继承它。那么这些类中究竟提供了哪些方法给我们使用?方法的参数个数及类型是?类的构造器需不需要参数?总不可能 SUN 公司的工程师打国际长途甚至飘洋过海来告诉你他编写的类该如何使用吧。他们只能提供文档给我们查看,Java DOC 文档(参考文献 4.4)就是这样的文档,它可以说是程序员与程序员交流的文档。

基本 API 指的是实现了一些底层功能的类,通用性较强的 API,如字符串处理/输入输出等等。我们又把它成为类库。熟悉 API 的方法一是多查 Java DOC 文档(参考文献 4.4),二是使用 JBuilder/Eclipse 等 IDE 的代码提示功能。

3.1.4 特定 API 的学习

Java 介入的领域很广泛,不同的领域有不同的 API,没有人熟悉所有的 API,对一般人而言只是熟悉工作中要用到的 API。如果你做界面开发,那么你需要学习 Swing/AWT/SWT 等 API;如果你进行网络游戏开发,你需要深入了解网络 API/多媒体 API/2D3D 等;如果你做 WEB 开发,就需要熟悉 Servlet 等 API 啦。总之,需要根据工作的需要或你的兴趣发展方向去选择学习特定的 API。

3.1.5 开发工具的用法

在学习基础语法与基本的面向对象概念时,从锻炼语言熟练程度的角度考虑,我们推荐使用的工具是 Editplus/JCreator+JDK,这时候不要急于上手 JBuilder/Eclipse 等集成开发环境,以免过于关注 IDE 的强大功能而分散对 Java 技术本身的注意力。过了这一阶段你就可以开始熟悉 IDE 了。

程序员日常工作包括很多活动,编辑、编译及构建、调试、单元测试、版本控制、维持模型与代码同步、文档的更新等等,几乎每一项活动都有专门的工具,如果独立使用这些工具的话,你将会很痛苦,你需要在堆满工具的任务栏上不断的切换,效率很低下,也很容易出错。在 JBuilder、Eclipse 等 IDE 中已经自动集成编辑器、编译器、调试器、单元测试工具 JUnit、自动构建工具 ANT、版本控制工具 CVS、DOC 文档生成与更新等等,甚至可以把 UML 建模工具也集成进去,又提供了丰富的向导帮助生成框架代码,让我们的开发变得更轻松。应该说 IDE 发展的趋势就是集成软件开发中要用到的几乎所有工具。

从开发效率的角度考虑,使用 IDE 是必经之路,也是从一个学生到一个职业程序员转变的里程碑。

Java 开发使用的 IDE 主要有 Eclipse、JBuilder、JDeveloper、NetBeans 等几种;而 Eclipse、JBuilder 占有的市场份额是最大的。JBuilder 在近几年来一直是 Java 集成开发环境中的霸主,它是由备受程序员尊敬的 Borland 公司开发,在硝烟弥漫的 Java IDE 大战中,以其快速的版本更新击败 IBM 的 Visual Age for Java 等而成就一番伟业。IBM 在 Visual Age for Java 上已经无利可图之下,干脆将之贡献给开源社区,成为 Eclipse 的前身,真所谓"柳暗花明又一村"。浴火重生的 Eclipse 以其开放式的插件扩展机制、免费开源获得广大程序员(包括几乎所有的骨灰级程序员)的青睐,极具发展潜力。

3.1.6 学习软件工程

对小型项目而言，你可能认为软件工程没太大的必要。随着项目的复杂性越来越高，软件工程的必要性才会体现出来。参见"软件开发学习路线"小节。

3.2 学习要点

确立的学习路线之后，我们还需要总结一下 Java 的学习要点，这些要点在前文多多少少提到过，只是笔者觉得这些地方特别要注意才对它们进行汇总，不要嫌我婆婆妈妈啊。

3.2.1 勤查 API 文档

当程序员编写好某些类，觉得很有成就感,想把它贡献给各位苦难的同行。这时候你要使用"Javadoc"工具（包含在 JDK 中）生成标准的 Java DOC 文档,供同行使用。J2SE/J2EE/J2ME 的 DOC 文档是程序员与程序员交流的工具，几乎人手一份，除了菜鸟之外。J2SE DOC 文档官方下载地址：

<http://Java.sun.com/j2se/1.5.0/download.jsp>, 你可以到 google 搜索 CHM 版本下载。也可以在线查看：
<http://Java.sun.com/j2se/1.5.0/docs/api/index.html>。

对待 DOC 文档要像毛主席语录，早上起床念一遍，吃饭睡觉前念一遍。

当需要某项功能的时候，你应该先查相应的 DOC 文档看看有没有现成的实现，有的话就不必劳神费心了直接用就可以了，找不到的时候才考虑自己实现。使用步骤一般如下：

- 找特定的包，包一般根据功能组织。
- 找需要使用类，类命名规范的话我们由类的名字可猜出一二。
- 选择构造器，大多数使用类的方式是创建对象。
- 选择你需要的方法。

3.2.2 查书/google->写代码测试->查看源代码->请教别人

当我们遇到问题的时候该如何解决？

这时候不要急着问别人，太简单的问题，没经过思考的问题，别人会因此而瞧不起你。可以先找找书，到 google 中搜一下看看，绝大部分问题基本就解决了。而像"某些类/方法如何使用的问题"，DOC 文档就是答案。对某些知识点有疑惑是，写代码测试一下，会给你留下深刻的印象。而有的问题，你可能需要直接看 API 的源代码验证你的想法。万不得已才去请教别人。

3.2.3 学习开源软件的设计思想

Java 领域有许多源代码开放的工具、组件、框架，JUnit、ANT、Tomcat、Struts、Spring、Jive 论坛、PetStore 宠物店等等多如牛毛。这些可是前辈给我们留下的瑰宝呀。入宝山而空手归，你心甘吗？对这些工具、框架进行分析，领会其中的设计思想，有朝一日说不定你也能写一个 XXX 框架什么的，风光一把。分析开源软件其实是你提高技术、提高实战能力的便捷方法。

3.2.4 规范的重要性

没有规矩，不成方圆。这里的规范有两层含义。第一层含义是技术规范，多到 <http://www.jcp.org> 下载 JSRXXX 规范，多读规范，这是最权威准确最新的教材。第二层含义是编程规范，如果你使用了大量的独特算法，富有个性化的变量及方法的命名方式；同时，没给程序作注释，以显示你的编程功底是多么的深厚。这样的代码别人看起来像天书，要理解谈何容易，更不用说维护了，必然会被无情地扫入垃圾堆。Java 编码规范到此查看或下载 <http://Java.sun.com/docs/codeconv/>，中文的也有，啊，还要问我在这，请参考 3.2.2 节。

3.2.5 不局限于 Java

很不幸，很幸运，要学习的东西还有很多。不幸的是因为要学的东西太多且多变，没时间陪老婆家人或女朋友，导致身心疲惫，严重者甚至导致抑郁症。幸运的是别人要抢你饭碗绝非易事，他们或她们需要付出很多才能达成心愿。

Java 不要孤立地去学习，需要综合学习数据结构、OOP、软件工程、UML、网络编程、数据库技术等知识，用横向纵向的比较联想的方式去学习会更有效。如学习 Java 集合的时候找数据结构的书看看；学 JDBC

的时候复习数据库技术；采取的依然是"需要的时候再学"的原则。

4. 结束语

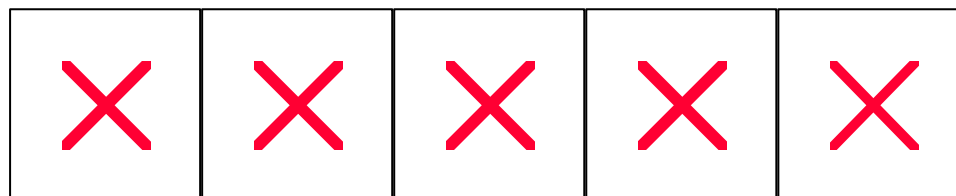
需要强调的是，学习软件开发确实有一定的难度，也很辛苦，需要付出很多努力，但千万不要半途而废。本文如果能对一直徘徊在 Java 神殿之外的朋友有所帮助的话，笔者也欣慰了。哈哈，怎么听起来老气横秋呀？没办法，在电脑的长期辐射之下，都快变成小老头了。最后奉劝各位程序员尤其是 MM 程序员，完成工作后赶快远离电脑，据《胡播乱报》报道，电脑辐射会在白皙的皮肤上面点缀一些小黑点，看起来鲜艳无比…… (文章提到的大连大工 IT 培训。本人没有任何攻击的意思)

参考资料：这篇文章转自 CSDN: <http://tb.blog.csdn.net/TrackBack.aspx?PostId=742557>

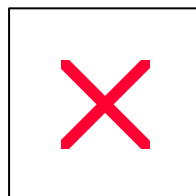
回答者: [ziyixia](#) - [助理](#) 三级 12-26 11:10

评价已经被关闭

目前有 2 个人评价



不好
0%
(0)



好

100% (2)

对最佳答案的评论

我学了些 JAVA 的基础，感觉还可以，还要深入

评论者: [wuahan_123](#) - [试用期](#) 一级

其他回答

共 2 条

从基本看看（语法不是问题了）

主要了解对象（JAVA 的精髓吧）

然后就可以记忆 API 了

回答者: [xpnui](#) - [江湖新秀](#) 五级 12-26 10:58

楼上说的很好！

但依我个人观点，

只要还是看兴趣！

有兴趣，没有干不成的事！

我想学 Java 请问看些什么 书最有帮助 最适合

悬赏分: 5 - 解决时间: 2006-7-13 08:21

提问者: [Mephtos](#) - [试用期](#) 一级



最佳答案

学习程序的最好方式是参照源代码，先仿照，后创新，慢慢有积累了，就可以收发自由了。没关系的，慢慢来，不可能一步到位的。**java** 确实很难学。

学习 **Java** 其实应该上升到如何学习程序设计这种境界，其实学习程序设计又是接受一种编程思想。每一种语言的程序设计思想大同小异，只是一些由语言特性的而带来的细微差别比如 **Java** 中的 **Interface**，你几乎在以前的学习中没有碰到过。以下我仔细给你说几点：

1. 我们必须明确一个大方向，也就是说现在面向对象的编程范畴。尽管人工智能曾经有所浪潮（看看 **Borland** 为什么有 **Turbo Prolog**），但未来 5-10 年工业界广泛承认并接受的将是面向对象式的编程。

2. 工业界目前最流行的面向对象编程语言就是 **C++**和 **Java**。所以基本上锁定这两个方向就可以了。而且完全可以同时掌握。

3. 掌握 **Java** 的精华特性而且一定要知道为什么。比如，**Interface** 和 **multi-thread**。用 **interface** 是更好的多继承的模型，而多线程则是设计到语言一级的重要特性。要完全理解 **interface** 是为什么，用多线程又有几种常用的编程模型。

4. 理解了语言的特性是为什么了之后，就可以试着上升到设计这个层次，毕竟学习语言是要用的。目前比较好的开发模式是采用自定向下的面向对象的设计，加上 **MVC** 的模式（你可以看一下我介绍的关于 **MVC** 的内容）。首先要找出最顶层的对象（这往往是最难的），然后一层一层往下递归，记住每次应符合 **7+/-2** 的原则，因为我们人的短记忆就是这样。一般有图形用户界面的 应从界面开始设计。

5. 有了基本设计模型后，可以学一些设计模式（**Design Pattern**）。这是目前证明很有效的。比如体系结构模式（**Layering** 分层，**Pipe/Filter** 管道或过滤器），设计模式（有很多，比如对象池 **Object Pool**、缓冲池 **Cache** 等），编程模式（比如 **Copy-on-Write**）。懂了这些模式之后，就会对 系统的整体结构有很好的把握，而学术上也有倾向一个系统完全可以由各种模式组合而成。前面提到的 **MT** 实际上就有好几种模式，掌握后就不用自己花很多时间去试了。另外一个 很重要的领域就是并行 和分布式计算领域，大概有 20 种左右。

6. 接下来就不能纸上谈兵了，最好的方法其实是实践。一般教科书上的例子并不能算是实践，只能算是让你掌握语言特性用的。而提倡做实际的 **Project** 也不是太好，因为你还没有熟练的能力去综合各种技术，这样只能是你自己越来越迷糊。我认为比较好的方法是找一些比较经典的例子，每个例子比较集中一种编程思想而设计的，比如在我的实践当中，我曾经学习过一个很经典的例子就是用 **Java** 实现的 **HotDraw**（源自 **SmallTalk**），你可以用 **rolemodel** 或 **hotdraw** 在搜索引擎上找一下，我记不大清楚了。好象 **rolemodel.com** 是个网站，上面有原代码和一些基本设计的文档。另一个来源可以到 www.umlchina.com 是个不错的文档基地。从 **HotDraw** 上我学到了什么是 **Framework**，以及如何用 **rolemodel** 的方式来构造，这样我就可以应用到其他的地方。顺便说一句，这个例子你绝对不会觉得小，只会觉得大，并且他还是真正的商用的 **Framework**。

7. 结合前面学到的设计模式你就可以很好的理解这些经典的例子。并且自己可以用他来实现一些简单的系统。如果可以对他进行进一步的修改，找出你觉得可以提高性能的地方，加上自己的设计，那就更上一个层次了，也就会真正地感到有所收获。

8. 好象以上谈的跟 **Java** 没什么关系，其实我们早就应该从单纯的学习语言到真正的学习好编程的领域。学习技术是没有止境的，你学习第一种语言可能要半年时间，以后每种语言都不应该超过两个月，否则你会觉得学习语言是包袱，是痛苦。

9. 学习是为了用的，是为了让你的程序产生价值，把握住这个原则会比较轻松点。

我带了 **JAVA** 班以后，来学习的同学很多都是初学者，一部分是急着找工作的，一部分是很感兴趣的。他们都想在短短一两个星期内掌握，这是不切实际的。而且这样做很容易让自己心浮气燥，难以静下心来思考。

JAVA 是比较热门，但我们自己不能头脑发热。我觉得所有初学者要学好 **JAVA** 或 **C** 或 **UNIX** 等任何一门计算机技术，一定要做到：

动手，动脑，动嘴，静心。

其中动手是最为重要的了。很多人学计算机技术就是看书、听课，当时是明白了，过后马上就忘记。如果不愿意敲键盘，就不如不学。即使有时候所敲的代码你不明白，可以先运行，看看结果，增加感性认识，然后上升到理性的认识。特别是在大家做考题的时候，很多东西，概念，你只要随便敲一个程序做做实验就能很深刻的理解，但很多人都是不愿意动手，这样的学习效率能高吗。

动脑，动嘴，就是要多想问题，多思考，有不懂的问题就大声说出来。很多人一是不动脑筋想问题，也就提不出问题。还有一些人怕自己提出的问题傻，怕人笑话。其实 **IT** 行业技术面很广，工作了 **5、6** 年的人说出一些傻话来也是正常的。

然后静心很重要。现在社会很浮躁，**IT** 行业更浮躁，这种心态是学不好东西的。学会了基本的东西，高级的内容才容易掌握，否则就是本末倒置。当然有些人情况特殊，急着工作，那去学很多人 **JAVA** 做了几年了，连多态，线程都没搞明白。

JAVA 中，有些概念不好理解怎么办，有学生说我就是不理解线程是怎么回事。我觉得理解是需要时间的，一下不能理解，多看看，甚至多背几遍就理解了。读书千遍，其义自见，现在也管用。而且我觉得初学时候要不求甚解，多做，多敲代码。等你做的足够多的时候，见多识广的时候，概念，理论的东西就很容易理解了。

JAVA 的精华就在面向对象思想，好比指针是 **C** 语言的精华一样，多花点心思就能掌握

很多网友问我学习 **Java** 有没有什么捷径,我说“无他，唯手熟尔”。但是我却很愿意将？

自己学习的一些经验写出来，以便后来者少走弯路,帮助别人是最大的快乐嘛！？

要想学好 **Java**，首先要知道 **Java** 的大致分类。我们知道，自从 **Sun** 推出 **Java** 以来，就力图？

使之无所不包，所以 Java 发展到现在，按应用来分主要分为三大块：J2SE、J2ME 和 J2EE，

这也就是 Sun 的 ONE(Open Net Environment)体系。J2SE 就是 Java2 的标准版，主要用于桌

面应用程序的编程；J2ME 主要应用于嵌入式系统开发，如手机和 PDA 的编程；J2EE 是 Java

2 的企业版，主要用于分布式的网络程序的开发，如电子商务网站和 ERP 系统。J2SE 开发

桌面应用软件比起 VC、VB、Delphi 这些传统开发语言来说，优势好象并不明显。J2ME 对于初

学者来说，好象又有点深奥，而且一般开发者很难有开发环境。所以现在应用最广泛又

最好学的就是 J2EE 了。J2EE 又包括许多组件，如 Jsp, Servlet, JavaBean, EJB, JDBC, Java

Mail 等。要学习起来可不是一两天的事。

那么又该如何学习 J2EE 呢？当然 Java 语法得先看一看的，I/O 包，Util 包，Lang 包你都熟

悉了吗？然后再从 JSP 学起。如果你学过 HTML，那么事情要好办的多，如果没有，那你快

去补一补 HTML 基础吧。其实 JSP 中的 Java 语法也不多，它更象一个脚本语言，有点象 ASP

。然后你就该学一学 Servlet 了。Servlet 就是服务器端小程序，他负责生成发送给客户

端的 HTML 文件。JSP 在执行时，也是先转换成 Servlet 再运行的。虽说 JSP 理论上可以完全

取代 Servlet，这也是 SUN 推出 JSP 的本意，可是 Servlet 用来控制流程跳转还是挺方便的？

，也令程序更清晰。接下来你应该学习一下 JavaBean 了，可能你早就看不管 JSP 在 HTML 中

嵌 Java 代码的混乱方式了，这种方式跟 ASP 又有什么区别呢？还好，SUN 提供了 JavaBean

可以把你的 JSP 中的 Java 代码封装起来，便于调用也便于重用。接着就是 EJB 了，EJB 就是

Enterprise JavaBean，看名字好象它是 JavaBean，可是它和 JavaBean 还是有区别的。它

是一个体系结构，你可以搭建更安全、更稳定的企业应用。它的大量代码已由中间件（

也就是我们常听到的 Weblogic, Websphere 这些 J2EE 服务器）完成了，所以我们要做的程

序代码量很少，大部分工作都在设计和配置中间件上。至于 JDBC，就不用我多说了，你

如果用 Java 编过存取数据库的程序，就应该很熟悉。还有，如果你要用 Java 编发送电子

邮件的程序，你就得看看 Javamail 了。？

好了，对 Java 和 J2EE 有了一些基本概念之后，你就应该编一些程序了，千万不要纸上谈？

兵哦。最好找一些有实例且带光盘的书来看，这样看到好的程序就可以直接 Ctrl+C 再 Ct？

rl+V，也不用劳您老大架再亲自把它？再输一遍吧，再说直接复制还不用怕出错，何乐而？

不为呢！还有就是要经常上一些好的 Java 编程文章，有好的文章要 Cut 下来，有问题尽管？

问，只要问题不是太傻，一般高手都会回答你的。下面介绍几个好的 Java 方面的编程网？

站：？

CSDN 论坛？<http://www.csdn.net/>中国最有名的技术论坛，《程序员》杂志就是他们出？

版的，你可以在上面提出问题，马上就有人回答你，如果你觉得好，你可以给那人加分？

；？

Java 研究组织 <http://www.javaresearch.org/>上面有很多原创文章，高手还是挺多的；？

？

Java 开发者？<http://www.chinajavaworld.com/>那里 Java 资料比较全；？

java.com.cn？<http://www.java.com.cn/>看这域名就知道有多牛，注册用户快接近一万？

了，同时在线人数也在一千左右，人气很旺的；？

IBM 的开发者网络 <http://www-900.ibm.com/developerWorks/cn/java/index.shtml?IBM?>

永远的蓝色巨人；？

那么我书也看了，程序也做了，别人问我的问题我都能解决了，是不是就成为高手了呢？

？当然没那么简单，这只是万里长征走完了第一步。不信？那你出去接一个项目，你知？

道怎么下手吗，你知道怎么设计吗，你知道怎么组织人员进行开发吗？你现在脑子里除？

了一些散乱的代码之外，可能再没有别的东西了吧！你现在最缺的是实际的工作经验，？

而不是书本上那些凭空想出来的程序。所以你快去找一份 Java 的编程工作来做吧（如果？

是在校学生可以去做兼职啊），在实践中提高自己，那才是最快的。不过你得祈祷在公

司里碰到一个高手，而且他还愿意不厌其烦地教你，这样好象有点难哦！？

还有一个办法就是读开放源码的程序了。我们知道开放源码大都出自高手，他们设计合

理，考虑周到，再加上有广大的程序员参与，代码的价值自然是字字珠玑，铿锵有力（？

对不起，偶最近《金装四大才子》看多了）。学 Java 必读的两个开源程序就是 Jive 和 Pe

t?Store。Jive 是国外一个非常著名的 BBS 程序，完全开放源码。论坛的设计采用了很多？

先进的技术，如 Cache、用户认证、Filter、XML 等，而且论坛完全屏蔽了对数据库的访？

问，可以很轻易的在不同数据库中移植。论坛还有方便的安装和管理程序，这是我们平？

时编程时容易忽略的一部份（中国程序员一般只注重编程的技术含量，却完全不考虑用？

户的感受，这就是我们与国外软件的差距所在）。Jive 的资料在很多网站上都有，大家？

可以找来研究一下。相信你读完代码后，会有脱胎换骨的感觉。遗憾的是 Jive 从 2.5 以后？

就不再无条件的开放源代码，同时有 licence 限制。不过幸好还有中国一流的 Java 程序员？

关注它，外国人不开源了，中国人就不能开源吗？这里向大家推荐一个汉化的 Jive 版本？

——J 道。Jive(J 道版)是由中国 Java 界大名鼎鼎的 banq 在 Jive?2.1 版本基础上改编而成，？

全中文，增加了一些实用功能，如贴图，用户头像和用户资料查询等，而且有一个开发？

团队在不断升级。你可以访问 banq 的网站 <http://www.jdon.com/?> 去下载。Pet?Store（？

宠物店）是 SUN 公司为了演示其 J2EE 编程规范而推出的开放源码的程序，应该很具有权威？

性，想学 J2EE 和 EJB 的朋友不要错过了。有意思的是，Pet?Store 还经常作为各大 J2EE 厂？

商的产品性能的比较标准，就连微软也曾作过一个 Pet?Store 试图证明.NET 性能比 J2EE 优？

越，可见 Pet?Store 的例子有多么深入人心。？

你一定会高兴地说，哈哈，原来成为 Java 高手就这么简单啊!记得 Tomjava 也曾碰到过一？

个项目经理，号称 Java 很简单，只要三个月就可以学会。其实说这种话的人就如当年小？

日本号称“三个月拿下中国”一样大言不惭。不是 Tomjava 泼你冷水，你现在只是学到了？

Java 的骨架，却还没有学到 Java 的精髓。接下来你得研究设计模式了。设计模式是高级？

程序员真正掌握面向对象核心思想的必修课。设计模式并不是一种具体"技术",它讲述的？

是思想，它不仅仅展示了接口或抽象类在实际案例中的灵活应用和智慧，让你能够真正？

掌握接口或抽象类的应用，从而在原来的 Java 语言基础上跃进一步，更重要的是，设计？

模式反复向你强调一个宗旨：要让你的程序尽可能的可重用。关于设计模式的资料，还？

是向大家推荐 banq 的网站 <http://www.jdon.com/?>，他把 GOF 的 23 种模式以通俗易懂的方？

式诠释出来，纯 Java 描述，真是经典中的经典。有时间再研究一下 MVC 结构（把 Model-V？

iew-Control 分离开的设计思想）吧，现在很流行的 Struts 就是它的一种实现方式，不？

过 Struts 用起来实在是繁，我们只要学习其精髓即可，我们完全可以设计自己的 MVC？

结构。然后你再研究一下软件 Refactoring(重整)和极限 XP 编程，相信你又会上一个台阶？

。？

做完这些，你不如整理一下你的 Java 代码，把那些经典的程序和常见的应用整理出来，？

再精心打造一番，提高其重用性和可扩展性。你再找几个志同道合的朋友成立一个工作？

室吧，你可以去承接一些项目做了，一开始可能有些困难，可是你有技术积累，又考虑？

周全，接下项目来可以迅速作完，相信大家以后都会来找你的，所以 Money 就哗啦啦的来？

了。。。。。。？

当然你也可以参加一些开源项目，一方面可以提高自己，另一方面也是为中国软件事业？

做贡献嘛！开发者在互联网上用 CVS 合作开发，用 QQ,MSN,E-mail 讨论联系，天南海北的？

程序员分散在各地却同时开发同一个软件，是不是很有意思呢？？

下面介绍两个好的开源项目网站：？

湖北省软件公共开发平台 <http://gro.clinux.org/?>

共创联盟 <http://cosoft.org.cn/?>

哇，好高兴哦，我终于成为高手了！非也，非也。古人云：“识时务者为俊杰”。你知？

道计算机界现在的发展形势吗？你知道微软的.NET 蓝图和 SUN?ONE 计划之间的明争暗斗吗？

？你知道计算机技术将向何处发展吗？其实从各大计算机厂商最近的动作,都可以看出来“？

Web 服务将是下一代互联网应用的制高点”，而微软的.NET 蓝图和 SUN?ONE 计划的斗争焦？

点，也就是 Web 服务。Web 服务就是一个崭新的分布式计算模型，它是一系列标准的综合？

（XML,SOAP,UDDI,WSDL 和 WSFL 等）。它使得不同语言编写的软件能够轻易的集成起来，？

使网络资源和 Web 站点变成一种服务而不是混乱的垃圾场。不远的将来，我们就可以在家？

里点击一下鼠标，就可以完成出门旅游的全部准备工作，包括定飞机票，定旅游线路，？

定好房间等。请注意，这所有的一切都是 Web 站点间自动完成的，再也不用象现在一样，？

表面上是电子商务，实际上很多环节都是人工操作。也许你会觉得这是天方夜谈，不过？

就近的说，你也很有可能承接一个项目，要集成两个企业的 ERP 系统。很有可能上游企业？

的系统是用 Dephi 编的，而下游企业的系统是用 Java 编的。你说你是 Java 高手，大家都看？

者你怎么做呢。所以啊，你还得学习新技术，如 Web 服务，而且你 Dephi 也要懂一点吧（？

Dephi6 现在已经提供 Web 服务的控件了）。？你编的 Java 系统，可能要和.NET 集成，所以？

你.NET 要懂一点吧？到最后，你可能发现你已经成为 Java 高手了，但很多时间却在搞别？

的技术。太极张三丰里说，最厉害的招式就是没有招式，可能就是这个道理吧！？

因为刚刚兴起，所以网上 Web 服务的资料不是很多，我还是给大家推荐几个网站吧：？

中国 UDDI 技术联盟 <http://www.uddi-china.org/?>

CSDN 的柴晓路专栏 <http://www.csdn.net/develop/author/ColumnAuthor/fennivel/?>（？

注：柴晓路也是互联网上一个有名的人物，他发表过很多关于 Web 服务的文章，还出了一？

书，应该称的上是中国 Web 服务技术的先行者）？

IBM 的开发者网络的 XML&Web?Service 专栏:?

<http://www-900.ibm.com/developerWorks/cn/xml/index.shtml?csdn?IBM> 可是 Web 服务?

的力推者?

呜...你费劲千心万苦,总算成为 Java 高手了,怎叫人不由喜极而泣呢!是啊,真不容易?

,真不简单,真叫人感动啊!那么打个电话告诉我吧,什么?我的电话号码是多少?昏?

到,你打电话问不就知道了吗,真是的.....?

回答者: [fireflyzgb](#) - [高级经理](#) [六级](#) 10-21 17:28

对最佳答案的评论

学习 java 我推荐给你一本<<java 大学简明教程>>即使没有接触过编程也可以很容易学会

评论者: [yinfengnong](#) - [初学者](#) [一级](#)

还没看完就不想学了!

评论者: [水色胭脂 2007](#) - [试用期](#) [一级](#)

兄弟我真服你了佩服

评论者: [xuqingze](#) - [试用期](#) [一级](#)

其他回答

共 1 条

不是的。只能说如果你学过 C++的话,学习 JAVA 会比较容易上手。但不是说没学过 C++就学不好 JAVA。

JAVA 和 C++都是面向对象的编程语言,从语法规则上和编程思想上都有很大的相似之处。学习 JAVA,我想给你一个我的拙见,找本基础一点并较全面的书,先把它的语法规则搞懂,之后找些实例或者习题,进行上机实践。当然,这种方法是在你已经习过某个计算机编程语言的情况下的一种好方法。

但不管怎么说,上机实践是非常重要的部分,如果只是看书,哪怕你有再好的记忆力,也学不会的。

希望这些能对你有帮助。祝你好运

我是新手,学习 JAVA 应该走哪些步骤!请各位大哥不吝赐教!谢谢!

 悬赏分: 100 - 解决时间: 2007-5-17 06:56

我刚刚接触 JAVA,可是我决定今后就从这方面发展,将来成为一名杰出的 JAVA 程序员!

可是我不知道怎么去学习它,在网上看了很多的资料,看到它包括那么多那么多的类,我不知道该去学习哪些类,最先学什么,然后接着学什么!

还有,我现在也下了 45 集 C 语言视频教程,是从基础到了解的课程内容.我需不需要先学完这些 C 语言视频教程来打 JAVA 基础了,还是直接学习 JAVA 语言.如果不需要用 C 语言打基础的话,我就省了很多时间了,但是如果用 C 语言打基础再学 JAVA 容易上手些的话,我就觉得那个时间发得也值了!

我现在还徘徊在这样的境地里,不知所措,请求各位有经验的大哥们不吝赐教.给小弟指点迷津.
小弟在此谢谢了!...

问题补充: 各位热心的前辈们,谢谢你们的不吝赐教!

我现在对 C 语言没有什么基础的,那我还要不要先学习 C 语言再去学 Java 了!还是直接学 Java!
谢谢了!...

提问者: [lihaibi](#) - [经理](#) [四级](#)



最佳答案

一、入门

《Java 2 从入门到精通》— 推荐

《Thinking in Java》— 强烈推荐*

O'reilly 的 Java 编程基础系列 — 参考*

二、进阶

《Java Cook Book》— 非常推荐* (包含了 Java 编程的 Tips, 适合当做手册来查阅)

《O'reilly-Java IO》— 推荐* (包含 Java IO 编程的各个方面)

《O'reilly-Database Programming with JDBC》— 推荐* (JDBC 编程)

《O'reilly-Java Programming with Oracle JDBC》— 参考*

三、Java Web 编程

《O'reilly-Java Server Pages》— 强烈推荐*

《O'reilly-Java Servlet Programming》— 非常推荐*

《O'reilly-Jakarta Struts》— 推荐* (Java Web 编程的一个 MVC 实现框架 Struts 的书)

四、EJB 编程

《J2EE 应用与 BEA Weblogic Server》— 强烈推荐

《Mastering EJB 2.0》— 非常推荐*

《Enterprise Java Bean》— 推荐*

五、Java XML 编程

《O'reilly-Java and XML》— 推荐*

《O'reilly-Java and SOAP》— 参考* (Java 的 SOAP 编程)

六、设计模式

《Core J2EE Patterns》— 强烈推荐* (J2EE 设计模式, 设计企业应用软件必备参考书)

《EJB Design Patterns》— 推荐*

七、其它

《O'reilly Ant - The Definitive Guide》— 推荐* (Ant 是一种功能非常强大的 Java 工具)

Note:

强烈推荐书籍: 建议购买, 重点学习

非常推荐书籍: 建议花时间学习

推荐书籍: 在学有余力的情况下, 建议学习

参考书籍：有兴趣的情况下学习

标明*号的书籍有电子版本

可以去 <http://www.china-pub.com> 看看

推荐以上书刊

回答者：[feelaaa](#) - [助理 三级](#) 3-3 01:05

其他回答

共 15 条

我的有个 QQ 群，关于 JAVA 技术探讨的群，
JAVA 爱好群号码是 10774901

回答者：[lovenertbaby](#) - [见习魔法师 二级](#) 3-3 01:05

楼上说的不实际 Thinking in java 根本不适合初学者看

学了 c 对学其他的语言有很大的帮助~但是不是学习其他语言就一定要先学 C
当然如果有很多时间的话还是建议学学 c 很有用的

至于 java 的话 确实对初学者感觉很乱~~什么 j2ee j2se j2me 太多了 呵呵
实际可以简单的理解 j2se 就是做软件 j2ee 就是 web 开发也就是门户网站之类的 j2me 是手机开发
实际无论 j2 什么也好都是万变不离其中~基本语法大体一致，所以如果想学不要考虑什么方向，直接找本初级 java 书 开始看
等你学得差不多自然知道方向了 方向是别人不能给你定的

我觉得学 java 最主要的是学和作
学一点 做一点 在学一点在做一点~~实践中才能提高

顺道鄙视一下二楼 复制一堆不知所云

回答者：[你猜我说的对么](#) - [门吏 二级](#) 3-3 11:01

楼主， 你有 c 基础的话， 学习 java 是很快的。重要的是自己动手去做，参与实际项目的锻炼， 这才会提高得很快。掌握 Java 的语言规则和养成良好的编程习惯，其次才谈到对类库使用的熟练程度，再其次是 OO 思想的掌握与应用，再其次就是 Design Pattern 的问题了。

下面文章转自 javanb 网：

<http://www.javanb.com/java/1/17261.html>

我可以结合自己的经验大致给你说一说，希望对你有帮助，少走些弯路。
学习 Java 其实应该上升到如何学习程序设计这种境界，其实学习程序设计又是接受一种编程思想。每一种语言的程序设计思想大同小异，只是一些由语言特性的而带来的细微差别，

比如 **Java** 中的 **Interface**，你几乎在以前的学习中没有碰到过。以下我仔细给你说几点：

1. 我们必须明确一个大方向，也就是说现在面向对象的编程范畴。尽管人工智能曾经有所浪潮（看看 **Borland** 为什么有 **Turbo Prolog**），但未来 5-10 年工业界广泛承认并接受的将是面向对象式的编程。

2. 工业界目前最流行的面向对象编程语言就是 **C++** 和 **Java**。所以基本上锁定这两个方向就可以了。而且完全可以同时掌握。

3. 掌握 **Java** 的精华特性而且一定要知道为什么。比如，**Interface** 和 **multi-thread**。用 **interface** 是更好的多继承的模型，而多线程则是设计到语言一级的重要特性。要完全理解 **interface** 是为什么，用多线程又有几种常用的编程模型。

4. 理解了语言的特性是为什么了之后，就可以试着上升到设计这个层次，毕竟学习语言是要用的。目前比较好的开发模式是采用自定向下的面向对象的设计，加上 **MVC** 的模式（你可以看一下我介绍的关于 **MVC** 的内容）。首先要找出最顶层的对象（这往往是最难的），然后一层一层往下递归，记住每次应符合 **7+/-2** 的原则，因为我们人的短记忆就是这样。一般有图形用户界面的应从界面开始设计。

5. 有了基本设计模型后，可以学一些设计模式（**Design Pattern**）。这是目前证明很有效的。比如体系结构模式（**Layering** 分层，**Pipe/Filter** 管道或过滤器），设计模式（有很多，比如对象池 **Object Pool**、缓冲池 **Cache** 等），编程模式（比如 **Copy-on-Write**）。懂了这些模式之后，就会对系统的整体结构有很好的把握，而学术上也有倾向一个系统完全可以由各种模式组合而成。前面提到的 **MT** 实际上就有好几种模式，掌握后就不用自己花很多时间去试了。另外一个很重要的领域就是并行和分布式计算领域，大概有 20 种左右。

6. 接下来就不能纸上谈兵了，最好的方法其实是实践。一般教科书上的例子并不能算是实践，只能算是让你掌握语言特性用的。而提倡做实际的 **Project** 也不是太好，因为你还没有熟练的能力去综合各种技术，这样只能是你自己越来越迷糊。我认为比较好的方法是找一些比较经典的例子，每个例子比较集中一种编程思想而设计的，比如在我的实践当中，我曾经学习过一个很经典的例子就是用 **Java** 实现的 **HotDraw**（源自 **SmallTalk**），你可以用 **rolemodel** 或 **hotdraw** 在搜索引擎上找一下，我记不大清楚了。好象 **rolemodel.com** 是个网站，上面有原代码和一些基本设计的文档。另一个来源可以到 www.umlchina.com 是个不错的文档基地。从 **HotDraw** 上我学到了什么是 **Framework**，以及如何用 **rolemodel** 的方式来构造，这样我就可以应用到其他的地方。顺便说一句，这个例子你绝对不会觉得小，只会觉得大，并且他还是真正的商用的 **Framework**。

7. 结合前面学到的设计模式你就可以很好的理解这些经典的例子。并且自己可以用他来实现一些简单的系统。如果可以对他进行进一步的修改，找出你觉得可以提高性能的地方，加上自己的设计，那就更上一个层次了，也就会真正地感到有所收获。

8. 好象以上谈的跟 **Java** 没什么关系，其实我们早就应该从单纯的学习语言到真正的学习

好编程的领域。学习技术是没有止境的，你学习第一种语言可能要半年时间，以后每种语言都不应该超过两个月，否则你会觉得学习语言是包袱，是痛苦。

9. 学习是为了用的，是为了让你的程序产生价值，把握住这个原则会比较轻松点。

没有第 10 点了，因为没有东西是十全十美的，哈哈~~。

<http://www.javanb.com/java/1/17261.html>

参考资料: <http://www.javanb.com/java/1/17261.html>

回答者: [张三的哥](#) - [助理 二级](#) 3-3 14:48

不用学 C 如果学了 C 再学 Java 会更容易

建议看《Thinking in Java》说得非常详细

适合初学者,这是我 Java 的入门教材

<http://soft.chinaw3c.com/soft/code/034/JAVA3.rar>

(用迅雷下载这本书,用超星图书馆看)

看这个就够了

看完之后你就知道接下来要做什么了

二楼列出这么多书 让人不知道该做什么

回答者: [dovegg](#) - [魔法学徒 一级](#) 3-4 10:07

"张三的哥"的回答很不错，应该对楼主有帮助。<http://www.javanb.com> 这个网站的内容还真丰富

回答者: [imlaoshang](#) - [试用期 一级](#) 3-4 17:35

最好是有钱去培训一下，多做题目对你肯定有好处，java 理解要比 C 语言深，入门书记配合孙卫琴的《java 面向对象编程》还是有助于提高的，还有，想好学什么版的 java，！ C 语言最好先学，这样学起来轻松些

给你这个网站，希望对你有帮助 <http://www.java-cn.com/>

回答者: [tokoyo1](#) - [试用期 一级](#) 3-4 22:27

欢迎大家一起讨论学习 JAVA,也欢迎初学者 由于是新群人少较少请大家一起帮忙宣传.营造良好的学习气氛

在次寻求 JAVA 高手入驻.JAVA 技术讨论群 19064380

回答者: [dumingqing](#) - [试用期 一级](#) 3-5 13:47

www.javabc.com

JAVA 学习库

回答者: [linxuwei86](#) - [助理 二级](#) 3-5 22:11

很多网友问我学习 Java 有没有什么捷径,我说“无他，唯手熟尔”。但是我却很愿意将？

自己学习的一些经验写出来，以便后来者少走弯路,帮助别人是最大的快乐嘛！？

要想学好 Java，首先要知道 Java 的大致分类。我们知道，自从 Sun 推出 Java 以来，就力

图？

使之无所不包，所以 **Java** 发展到现在，按应用来分主要分为三大块：**J2SE**、**J2ME** 和 **J2EE**，

这也就是 **Sun ONE (Open Net Environment)** 体系。**J2SE** 就是 **Java2** 的标准版，主要用于桌

面应用程序的编程；**J2ME** 主要应用于嵌入式系统开发，如手机和 **PDA** 的编程；**J2EE** 是 **Jav**

a2 的企业版，主要用于分布式的网络程序的开发，如电子商务网站和 **ERP** 系统。**J2SE** 开

桌面应用软件比起 **VC**、**VB**、**DEPHI** 这些传统开发语言来说，优势好象并不明显。**J2ME** 对于初

学者来说，好象又有点深奥，而且一般开发者很难有开发环境。所以现在应用最广泛又？

最好学的就是 **J2EE** 了。**J2EE** 又包括许多组件，如 **Jsp**、**Servlet**、**JavaBean**、**EJB**、**JDBC**、**Java**？

Mail 等。要学习起来可不是一两天的事。？

那么又该如何学习 **J2EE** 呢？当然 **Java** 语法得先看一看的，**I/O** 包，**Util** 包，**Lang** 包你都熟？

悉了吗？然后再从 **JSP** 学起。如果你学过 **HTML**，那么事情要好办的多，如果没有，那你快？

去补一补 **HTML** 基础吧。其实 **JSP** 中的 **Java** 语法也不多，它更象一个脚本语言，有点象 **ASP**？

。然后你就该学一学 **Servlet** 了。**Servlet** 就是服务器端小程序，他负责生成发送给客户？

端的 **HTML** 文件。**JSP** 在执行时，也是先转换成 **Servlet** 再运行的。虽说 **JSP** 理论上可以完全？

取代 **Servlet**，这也是 **SUN** 推出 **JSP** 的本意，可是 **Servlet** 用来控制流程跳转还是挺方便的？

，也令程序更清晰。接下来你应该学习一下 **JavaBean** 了，可能你早就看不管 **JSP** 在 **HTML** 中？

嵌 **Java** 代码的混乱方式了，这种方式跟 **ASP** 又有什么区别呢？还好，**SUN** 提供了 **JavaBean**？

可以把你的 JSP 中的 Java 代码封装起来，便于调用也便于重用。接着就是 EJB 了，EJB 就是？

Enterprise?JavaBean,看名字好象它是 Javabeen，可是它和 Javabeen 还是有区别的。它？

是一个体系结构，你可以搭建更安全、更稳定的企业应用。它的大量代码已由中间件（？

也就是我们常听到的 Weblogic,Websphere 这些 J2EE 服务器）完成了，所以我们要做的程？

序代码量很少，大部分工作都在设计和配置中间件上。至于 JDBC，就不用我多说了，你？

如果用 java 编过存取数据库的程序，就应该很熟悉。还有，如果你要用 Java 编发送电子？

邮件的程序，你就得看看 Javamail 了。？

好了，对 Java 和 J2EE 有了一些基本概念之后，你就应该编一些程序了，千万不要纸上谈？

兵哦。最好找一些有实例且带光盘的书来看，这样看到好的程序就可以直接 Ctrl+C 再 Ct？

rl+V，也不用劳您老大架再亲自把它？再输一遍吧，再说直接复制还不用怕出错，何乐而？

不为呢！还有就是要经常上一些好的 Java 编程文章，有好的文章要 Cut 下来，有问题尽管？

问，只要问题不是太傻，一般高手都会回答你的。下面介绍几个好的 Java 方面的编程网？

站：？

CSDN 论坛？<http://www.csdn.net/>中国最有名的技术论坛，《程序员》杂志就是他们出？

版的，你可以在上面提出问题，马上就有人回答你，如果你觉得好，你可以给那人加分？

；？

Java 研究组织 <http://www.javaresearch.org/>上面有很多原创文章，高手还是挺多的；？

？

Java 开发者？<http://www.chinajavaworld.com/>那里 Java 资料比较全；？

java.com.cn？<http://www.java.com.cn/>看这域名就知道有多牛，注册用户快接近一万？

了，同时在线人数也在一千左右，人气很旺的；？

IBM 的开发者网络 <http://www-900.ibm.com/developerWorks/cn/java/index.shtml?IBM?>

永远的蓝色巨人；？

那么我书也看了，程序也做了，别人问我的问题我都能解决了，是不是就成为高手了呢？

？当然没那么简单，这只是万里长征走完了第一步。不信？那你出去接一个项目，你知？

道怎么下手吗，你知道怎么设计吗，你知道怎么组织人员进行开发吗？你现在脑子里除？

了一些散乱的代码之外，可能再没有别的东西了吧！你现在最缺的是实际的工作经验，？

而不是书本上那些凭空想出来的程序。所以你快去找一份 **Java** 的编程工作来做吧（如果？

是在校学生可以去做兼职啊），在实践中提高自己，那才是最快的。不过你得祈祷在公？

司里碰到一个高手，而且他还愿意不厌其烦地教你，这样好象有点难哦！？

还有一个办法就是读开放源码的程序了。我们知道开放源码大都出自高手，他们设计合？

理，考虑周到，再加上有广大的程序员参与，代码的价值自然是字字珠玑，铿锵有力（？

对不起，偶最近《金装四大才子》看多了）。学 **Java** 必读的两个开源程序就是 **Jive** 和 **Pe？**

t?Store。**Jive** 是国外一个非常著名的 BBS 程序，完全开放源码。论坛的设计采用了很多？

先进的技术，如 **Cache**、用户认证、**Filter**、**XML** 等，而且论坛完全屏蔽了对数据库的访？

问，可以很轻易的在不同数据库中移植。论坛还有方便的安装和管理程序，这是我们平？

时编程时容易忽略的一部份（中国程序员一般只注重编程的技术含量，却完全不考虑用？

户的感受，这就是我们与国外软件的差距所在）。**Jive** 的资料在很多网站上都有，大家？

可以找来研究一下。相信你读完代码后，会有脱胎换骨的感觉。遗憾的是 **Jive** 从 2.5 以后？

就不再无条件的开放源代码，同时有 **licence** 限制。不过幸好还有中国一流的 **Java** 程序员？

关注它，外国人不开源了，中国人就不能开源吗？这里向大家推荐一个汉化的 **Jive** 版本？

—J 道。Jive(J 道版)是由中国 Java 界大名鼎鼎的 banq 在 Jive?2.1 版本基础上改编而成,?
全中文,增加了一些实用功能,如贴图,用户头像和用户资料查询等,而且有一个开发?
团队在不断升级。你可以访问 banq 的网站 <http://www.jdon.com/?> 去下载。Pet?Store (?
宠物店)是 SUN 公司为了演示其 J2EE 编程规范而推出的开放源码的程序,应该很具有权威?
性,想学 J2EE 和 EJB 的朋友不要错过了。有意思的是, Pet?Store 还经常作为各大 J2EE
厂?
商的产品性能的比较标准,就连微软也曾作过一个 Pet?Store 试图证明.NET 性能比 J2EE
优?
越,可见 Pet?Store 的例子有多么深入人心。?
你一定会高兴地说,哈哈,原来成为 Java 高手就这么简单啊!记得 Tomjava 也曾碰到过一?
个项目经理,号称 Java 很简单,只要三个月就可以学会。其实说这种话的人就如当年小?
日本号称“三个月拿下中国”一样大言不惭。不是 Tomjava 泼你冷水,你现在只是学到了?
Java 的骨架,却还没有学到 Java 的精髓。接下来你得研究设计模式了。设计模式是高级?
程序员真正掌握面向对象核心思想的必修课。设计模式并不是一种具体"技术",它讲述的?
是思想,它不仅仅展示了接口或抽象类在实际案例中的灵活应用和智慧,让你能够真正?
掌握接口或抽象类的应用,从而在原来的 Java 语言基础上跃进一步,更重要的是,设计?
模式反复向你强调一个宗旨:要让你的程序尽可能的可重用。关于设计模式的资料,还?
是向大家推荐 banq 的网站 <http://www.jdon.com/?>,他把 GOF 的 23 种模式以通俗易懂的方?
式诠释出来,纯 Java 描述,真是经典中的经典。有时间再研究一下 MVC 结构(把 Model-V?
iew-Control 分离开的设计思想)吧,现在很流行的 Struts 就是它的一种实现方式,不?
过 Struts 用起来实在是很繁,我们只要学习其精髓即可,我们完全可以设计自己的 MVC?

结构。然后你再研究一下软件 **Refactoring**(重整)和极限 **XP** 编程，相信你又会上一个台阶？

。？

做完这些，你不如整理一下你的 **Java** 代码，把那些经典的程序和常见的应用整理出来，？

再精心打造一番，提高其重用性和可扩展性。你再找几个志同道合的朋友成立一个工作？

室吧，你可以去承接一些项目做了，一开始可能有些困难，可是你有技术积累，又考虑？

周全，接下项目来可以迅速作完，相信大家以后都会来找你的，所以 **Money** 就哗啦啦的来？

了。。。。。。？

当然你也可以参加一些开源项目，一方面可以提高自己，另一方面也是为中国软件事业？

做贡献嘛！开发者在互联网上用 **CVS** 合作开发，用 **QQ,MSN,E-mail** 讨论联系，天南海北的？

程序员分散在各地却同时开发同一个软件，是不是很有意思呢？？

下面介绍两个好的开源项目网站：？

湖北省软件公共开发平台 <http://gro.clinux.org/?>

共创联盟 <http://cosoft.org.cn/?>

哇，好高兴哦，我终于成为高手了！非也，非也。古人云：“识时务者为俊杰”。你知？

道计算机界现在的发展形势吗？你知道微软的.NET 蓝图和 **SUN?ONE** 计划之间的明争暗斗吗？

？你知道计算机技术将向何处发展吗？其实从各大计算机厂商最近的动作,都可以看出来“？

Web 服务将是下一代互联网应用的制高点”，而微软的.NET 蓝图和 **SUN?ONE** 计划的斗争焦点？

点，也就是 **Web** 服务。**Web** 服务就是一个崭新的分布式计算模型，它是一系列标准的综合？

（**XML,SOAP,UDDI,WSDL** 和 **WSFL** 等）。它使得不同语言编写的软件能够轻易的集成起来，？

使网络资源和 Web 站点变成一种服务而不是混乱的垃圾场。不久的将来，我们就可以在家？

里点击一下鼠标，就可以完成出门旅游的全部准备工作，包括定飞机票，定旅游线路，？

定好房间等。请注意，这所有的一切都是 Web 站点间自动完成的，再也不用象现在一样，？

表面上是电子商务，实际上很多环节都是人工操作。也许你会觉得这是天方夜谈，不过？

就近的说，你也很有可能承接一个项目，要集成两个企业的 ERP 系统。很有可能上游企业？

的系统是用 Dephi 编的，而下游企业的系统是用 Java 编的。你说你是 Java 高手，大家都看？

者你怎么做呢。所以啊，你还得学习新技术，如 Web 服务，而且你 Dephi 也要懂一点吧（？

Dephi6 现在已经提供 Web 服务的控件了）。？你编的 Java 系统，可能要和 .NET 集成，所以？

你 .NET 要懂一点吧？到最后，你可能发现你已经成为 Java 高手了，但很多时间却在搞别？

的技术。太极张三丰里说，最厉害的招式就是没有招式，可能就是这个道理吧！？

因为刚刚兴起，所以网上 Web 服务的资料不是很多，我还是给大家推荐几个网站吧：？

中国 UDDI 技术联盟 <http://www.uddi-china.org/>？

CSDN 的柴晓路专栏 <http://www.csdn.net/develop/author/ColumnAuthor/fennivel/>（？

注：柴晓路也是互联网上一个有名的人物，他发表过很多关于 Web 服务的文章，还出了一？

书，应该称的上是中国 Web 服务技术的先行者）？

IBM 的开发者网络的 XML&Web?Service 专栏：？

<http://www-900.ibm.com/developerWorks/cn/xml/index.shtml?csdn?IBM> 可是 Web 服务？

的力推者？

呜...你费劲千心万苦，总算成为 Java 高手了，怎叫人不由喜极而泣呢！是啊，真不容易？

，真不简单，真叫人感动啊！那么打个电话告诉我吧，什么？我的电话号码是多少？昏？

到，你打电话问不就知道了吗，真是的.....？

回答者: [fireflyzqb](#) - 高级经理 七级 3-7 16:05

楼上几位仁兄辛苦了,按你们的方法培养出来的一定是人才

本人没有那么多高谈阔论,只在这里说说我个人对你的问题的一点看法

首先是要先学 C 还是先学 JAVA,这个要看你的兴趣,如果你喜欢 JAVA,只学 JAVA 就可以了,完全可以把 C 抛开

接下来说说你对于 JAVA 的看法,不要觉得 JAVA 有很多很多的类,这点不应该是你关系的,JAVA 中是有很多的包,很多的类,而且这些类在 JAVA API(一个小说明文档)中都可以找到,只要你大致了解这些类是做什么用的,有一个印象,在使用时直接去查就可以了,完全不必去死记,我认为,对于初学这来说,你最主要的问题是弄清楚什么是方法以及方法之间的调用,再就是类的一些特行,比如说多态\继承\等等,还有就是接口的使用,上溯造形的意义,内部类什么的,学这些的时候也一样,除了几个特例之外,都可以在使用时现查 API

好了,要上课了,说得有点乱,总结一下,就是说学习 JAVA 重在学习如何去使用其中的东西,如类\方法\借口什么的,而不重在记住多少,还有就是看书不要杂,对于初学者来说,选一本书从头看就可以了,现在的书抄来的比较多,能明白吧?

真要上课了,如果觉得我说的有点道理,还有什么问题的话,可以加群 3624801 大家一起讨论,好运!

回答者: [yh9871377](#) - 试用期 一级 3-9 15:17

Core Java 不错

但是 Tinking in Java

现在看还早了 ,会看的很郁闷的 .

///////补充

没学过编程的话,,先学下 C

了解下编程思想 ,

回答者: [4709274](#) - 见习魔法师 二级 3-13 16:40

你把我 设为最“好”我会漫漫的给你讲 为你提供资料

参考资料:“福兴韵”贴吧

回答者: [福兴韵](#) - 见习魔法师 二级 3-14 02:57

很简单去下中国黑客 / 红客网那里有很多这样的资料让你看的

回答者: [wangjiaxue101](#) - 试用期 一级 3-15 18:12

努力啊

回答者: [303837647](#) - 兵卒 一级 3-16 22:25

直接学 java,找本最基础的书,最好清华大学出版社的,看完后把里面的代码全部敲一遍,之后再学 java 编程思想第 3 版,接着再详细学习四个重要的包,接着做一些 scjp 的题,接着找工具学习,再学习 jsp,swing 等等等,不说了呵呵入门资料请看我的 bokee

<http://auuppp.bokee.com/5630891.html>

直接学 java 啊,c 不用学的啊

从 JDK 下载到 HELLOWORLD 调试成功

<http://auuppp.bokee.com/5630891.html>

看完之后再加分也不迟.

呵呵

我想学 java 和 JSP (从零学起)。

 悬赏分: 60 - 解决时间: 2006-12-31 13:44

我想学编程,我比较想学 java 和 JSP,我的电脑配置: CPU1200MHz, 内存 256M, 硬盘 5G, 显卡也比较落后。我的问题如下:

- 1.这样的电脑配置能否满足我学 java 和 JSP?
- 2.我学 java 和 JSP 应该如何入门? 需要先学习什么?
- 3.学习 java 和 JSP 需要安装什么软件? 如何获得这些软件? 是从网上免费下载还是到电脑城购买?
- 4.我这样的电脑配置能否安装 windows2000 和红帽子 linux 的双系统? java 和 JSP 都能在这两种系统中运行吗?

望各位高手赐教,我会追加分的。

问题补充: 我的硬盘确实是 5G, 装个 windows2000 应该还是可以的吧? 我还有一个问题: java 与 java2 是什么关系?

提问者: [zx936_96](#) - 试用期 一级

 最佳答案

- 1.学 JAVA 够了 JSP 就有点不行了, 建议加个内存
- 2.先看《JAVA 2 学习指南》, 里面完全是最基本的语法(这本书是考 SCJP 的, 语法都全了), JSP 么你先学 HTML, 把基本标签看懂, 然后就去买本《JSP 开发大全》照着打里面的例子, 多打打就会了, JSP 其实也不难, 难的是其他东西。。。
- 3.jdk, 必须的,
eclipse, 这东西功能超强。。。必用。。JAVA 编程用的
myeclipse, 其实也就是个 ECLIPSE 的插件, 可以用于做 JSP 等关于 WEB 的东西
tomcat, 学 JSP 必须的, 是个容器共你打开 JSP 页面用的。。
数据库, 这个也是必须的, 还要学个 SQL 语言, 很容易的
- 4.双系统有困难, 2 个都能运行
- 5.java2 增加了更多的类库, 可以更容易的使用

上面说的东西自己 BAIDU 下就能找到下载的地方~

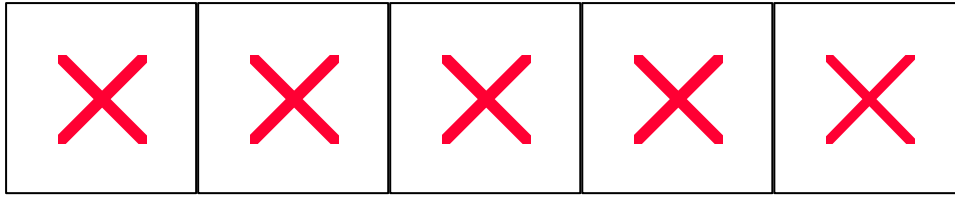
回答者: [jamesluka](#) - 助理 二级 12-23 15:54

提问者对于答案的评价:

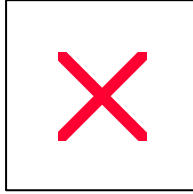
谢谢大家的回答, 小弟感激不尽。

评价已经被关闭

目前有 1 人评价



不好
0%
(0)



好

100% (1)

其他回答

共 4 条

1.足够运行了.

2.买书吧,推荐<JAVA 从入门到精通>

多做练习,多找一些习题来看.

推荐你几个网站.

<http://rjx.cs1g.cn:888/>

<http://www.softhouse.com.cn/news/category/94/>编程语言/Java/63.html

<http://www.gzu521.com/>

还可以下载一些视频教程,BT 下载有.

3.当然是 JAVA 的 JDK 你可以到中文官方免费下载 <http://cn.sun.com/>

4.不知道你在那里搞到的 5G 硬盘(从来没看到过 5G 的硬盘)这块硬盘不要希望安装 98 以上的操作系统了.希望你是打错了!

回答者: [xlceo](#) - [见习魔法师 二级](#) 12-21 03:35

我见过 4G 的- =~!

跟我那 80G 硬盘一样大小(体积)

我大学是学 B 的,考国家二级没考下来,可是明年就取消 VB 了郁闷呀

想看看 C++ 和 JAVA 哪个更好(就业方面)重新学过

回答者: [Hihapy](#) - [助理 二级](#) 12-26 00:52

Take a look at www.threes.cn, it could be some help for you

回答者: [jinhuiyu1982](#) - [童生 一级](#) 12-28 12:56

想学到精通 java 3 年也学不完!

jsp 主要掌握常用标签(最好是先掌握 HTML 标签,要是会 javascrip 就更好了)

java 会点基础,懂一些熟语(比如'类', '继承', '方法')

建议你再学点 SQL 文,这些都是一体的。

所以你随便看点基础书就行,要是想用电脑学的话,买一些学习光盘即可

楼上的 eclipse 的确是好东西,它是实际做项目的 web 框架需要很多配置,还有插件,配置

不好根本就运行不了，对于初学者要求过高。

当具备了一定的基础知识后，进入项目有人带一带，马上你的水平就会提升，学好 java 是要通过实践的！

初学 JAVA 语言面临的几个问题

 悬赏分：50 - 解决时间：2006-6-19 20:25

我是一名 JAVA 语言的初学者；

并且之前对 C,C++,VB 等等语言都没做过比较系统的学习；

我想问的问题是：

一.我应该先从哪方面做起？

二.学习任何一门语言都要边学边动手试验.初学 JAVA 语言时候应该安装什么软件？

三.如果想偏向互联网方面的应用,应该重点学习什么？

大虾们可以讲讲自己的见解,多谢指教！

提问者： [yanyingke](#) - [秀才 二级](#)

 最佳答案

1,我不同意楼上说法。**Thinking in Java** 对于有一定语言基础的人来说算是一本提高的书，但是对于新手来说这绝对不算是入门的书。如果想要入门的话

我建议楼主能找到英语原版教材，如果英语不行就要找一本介绍 JAVA 进本语法的书，我倒是有几个电子版的不错，楼主要的话给我联系方式。如果有机会就找一个系统的学过的人当老师辅导你。也可以去下载视频教程，不过比较难下

因为实在是太大了。并且他们都是注重就业，并不是深入地讲解。

2，你说的很对，要边学边实践，对于初学者，还是去安装 JDK 吧，网上随便搜搜就能找到，因为这个用起来比较简单。要是想用大的 IDE 就用 JBuilder 吧，这个是图形界面的，下载地址不是很多哦，因为它是收费的。还可以用 Eclipse，这个也不错，现在很流行的，它的官方网站就有下载。但是这些大的 IDE 需要你额外学习他们的使用方法。

3，偏向互联网的话肯定就是 JSP, Javascript 这些东西了，不过说实话，这些东西的潜力不大，社会需求也不多，还是去学习 J2EE（企业开发），J2ME（手机游戏开发）好，薪水很高，需求量大。

如果学习过程中有什么疑问欢迎交流，联系方式见我资料。

回答者：  [江水寒](#) - [高级经理 六级](#) 6-3 22:52

提问者对于答案的评价：

谢谢

对最佳答案的评论

回答 JAVA 问题没有一个是好的？都是半桶水叮当响

评论者： [pongyoufa](#) - [秀才 二级](#)

其他回答

共 6 条

1.推荐一本好书 Think in java

去 verycd 下载一些好的视频教程 比如孙鑫的

2.安装 j2sdk 1.4 或者 Jbuilder 或者 Netbean 等 IDE 工具

3.既然偏重互联网就学 JSP 或者 ASP.NET 编程

去下载张孝祥的 JSP

去下载北大青鸟的 ASP.NET

下载网址都是 verycd.com

不错用的是 **emule** 速度比较慢

慢慢的下吧

回答者: [xiaoqiang2005](#) - 举人 四级 6-3 22:22

1.从 **helloworld** 入手!学习一个程序的基本语法结构,然后是基本数据类型,运算符和表达式,然后是控制语句,这些差不多的时候可以开始对象,方法,类,包,接口,接着可以开始学习简单的数据结构,如数组了!呵呵!再往后线程,图形,jdbc 等等!

2.就装一个 **jdk**,学习的时候少用集成开发环境!因为你要学习!

3.不知道你的互联网方式是指 **Applet** 还是 **Socket**????呵呵!学习重点就是他们两,就看你喜欢什么了!

没有其他语言基础学习可能会困难点!我对语言的态度就是一点:语法过了,剩下就是编码的问题!只要有支持文档!编程就是体力劳动!

回答者: [wangtk1982](#) - 举人 四级 6-3 22:32

请问楼上的, **j2me** 的开发电子书有没有,可否上传一个

回答者: [lzjseed](#) - 见习魔法师 二级 6-4 00:20

Think in java 一开始的时候最好不要看,那个东西不是对于初学者而言的。初学 **JAVA** 的时候可以不用买书,网上有很多 **JAVA** 入门的资料。刚开始学习 **JAVA** 可以安装 **jdk1.4**, 因为现在对于 **JDK1.5** 的入门资料可能比较少, 很多特性不同于 1.4 的, 然后编写代码可以使用 **EditPlus** (不推荐 **notepad**, 那个东西比较恶心, 除非你能忍受所有的字都是黑色的)。偏向于互联网方面应该学习 **JSP**, 应用在工作方面的话还因该学习目前比较流行的框架, 例如 **Struts** 以及 **Hibernate**、**Spring** 等, 还应该了解主流的服务器, 例如 **WebLogic** (电子政务用的比较多)、**WebSphere** (IBM 很强的 IDE 以及 **Application Server**)、**Tomcat** (中小型应用、个人开发环境) 等。

回答者: [wz_starcatcher](#) - 举人 四级 6-4 00:28

1、先想办法弄好 **java** 编译的环境。

2、在整出个“**Hello World**”。

3、学会弄出一些按钮和文本框之类的东西

4、什么循环呀! 继承呀! 父类, 子类呀, 跟着学!

.....

回答者: [webstar16](#) - 助理 二级 6-5 09:55

我也是 **java** 初学者,我在北大青鸟上学,感觉 **java** 很简单的,我没有买什么参考书,网上的资源很丰富,还有必须要说明的一点就是 **java** 语言不同于其他编程语言,他不是像 **c** 语言那样,拼命的作题就能提高的,**java** 的概念性东西非常多,主要就是类和对象的关系,还有 **java** 的三大特性--封装性,继承性和多态性,针对这三大特性,深入理解其概念,然后就看 **API** 就行,官方下载一个 **jdk**,现在最近版本是 1.5,编辑工具建议使用 **jCrator**,不知拼没拼错呀,还有就是多看看

别人的代码,刚开始不要担心自己看不懂,我就是从不懂到懂走过来的,最后说明一点,网上的资源很丰富,不是很有钱,不要去买厚厚的书,电子版不是很好吗?