

- 在这里，众多知名企业面试官将为你撕开神秘的求职面纱
- 在这里，多位求职达人将现身说法为你揭开求职谜团
- 在这里，各种类型的企业招聘细节都会被展露无疑
- 在这里，我们将为你抽丝剥茧，还原IT行业的真相
- 在这里，我们将为你指点迷津，告诉你职场上的风风雨雨
- 在这里，我们将为你点石成金，成为那盏指引你前进的灯塔

# Java 程序员 面试笔试

何昊 薛鹏 叶向阳 编著

# 宝典



本书覆盖了历年来各大IT名企95%以上的面试笔试题，当你细细品读完本书的知识后，各类企业的offer将任由你挑选。本书将带你走进神奇的求职之旅。



机械工业出版社  
CHINA MACHINE PRESS

# Java 程序员面试笔试宝典

何昊 薛鹏 叶向阳 编著

机械工业出版社

本书引入了一批来自于名牌高校、就职于明星企业的职场达人的真实求职案例,通过他们的求职经验与教训,将整个求职过程生动形象地展示在读者面前,进而对求职者起到一定的指引作用。同时,为了更具说服力,本书特邀多位IT名企的面试官现身说法,以独特的视角对面试过程中求职者存在的各类问题进行了深度剖析。为了能够让读者对即将投身的工作有一些更加清楚的认识,能够更加有针对性地进行求职准备,本书对各种类型的IT企业的招聘环节进行了庖丁解牛式的分析。

技术性知识的考查是程序员求职中最重要的内容,鉴于此,本书除了对传统的计算机相关知识(Java语言基础知识、Web基础知识、数据结构与算法、数据库、设计模式等)以及面试笔试真题进行分析与解答外,还根据当前计算机技术的发展潮流,对面试笔试中常见的海量数据处理问题进行了详细的分析。

本书是一本适合计算机相关专业毕业生阅读的求职指导用书,同时也适合期望在计算机软硬件行业大显身手的计算机爱好者阅读。

## 图书在版编目(CIP)数据

Java程序员面试笔试宝典/何昊,薛鹏,叶向阳编著. —北京:机械工业出版社,2014.6

ISBN 978-7-111-47746-4

I. ①J… II. ①何…②薛…③叶… III. ①JAVA语言—程序设计  
IV. ①TP312

中国版本图书馆CIP数据核字(2014)第192692号

机械工业出版社(北京市百万庄大街22号 邮政编码100037)

策划编辑:时静 责任编辑:时静 吴晋瑜

责任校对:张艳霞 责任印制:乔宇

北京机工印刷厂印刷(三河市南杨庄国丰装订厂装订)

2014年9月第1版·第1次印刷

184mm×260mm·22印张·544千字

0 001—3 500册

标准书号:ISBN 978-7-111-47746-4

定价:48.80元



凡购本书,如有缺页、倒页、脱页,由本社发行部调换

电话服务

网络服务

社服务中心:(010)88361066 教材网:<http://www.cmpedu.com>

销售一部:(010)68326294 机工官网:<http://www.cmpbook.com>

销售二部:(010)88379649 机工官博:<http://weibo.com/cmp1952>

读者购书热线:(010)88379203 封面无防伪标均为盗版

# 前 言

很多朋友问我为什么编写《Java 程序员面试笔试宝典》，它与我先前编写的《程序员面试笔试宝典》究竟有何不同？其实，我之所以创作本书，就是希望通过本书能够为主攻 Java 方向的求职者提供一些帮助。《程序员面试笔试宝典》一书面市后，在学生群中反响不错。很多读者反映，通过阅读本书，他（她）们找到了自己比较满意的工作，但不足之处在于该书偏重于 C/C++ 语言，而很多招聘岗位考查的却是 Java 语言的相关内容。此外，根据前程无忧、智联招聘和中华英才三大专业招聘网站的就业数据调查分析可以看出，在目前的就业市场上，Java 工程师的市场需求量非常大，丝毫不亚于 C/C++ 工程师，所以，他（她）们希望能够看到一本专门针对 Java 语言面试笔试的书。

于是，我萌发了编写一本有关 Java 程序员面试笔试之书的想法。近些年来，无论是传统的互联网应用，还是当前发展迅猛的云计算、海量数据处理以及移动互联网，都离不开 Java 语言。Java 语言始终在信息技术浪潮中扮演着极其重要的角色。从历次编程语言排行榜不难看出，Java 语言的应用与 C 语言不分伯仲，它们都可以称得上是计算机历史上重要的编程语言。鉴于此，很多 IT 企业（例如门户网站、即时通信、电子商务、搜索引擎等）、手机应用开发企业等都使用 Java 语言作为开发语言，因此自然希望求职者熟悉 Java 语言的基本原理，并能够熟练使用 Java 语言从事实际的项目研发，所以，也会将 Java 作为面试笔试环节的主要考查内容。

作为《程序员面试笔试宝典》的姊妹篇，本书一方面延续并继承了其中诸如面试官箴言、面试心得交流、企业面试笔试攻略、海量数据处理等深受读者好评的经典内容；另一方面，编者结合当前的实际情况，对面试笔试中常涉及的 Java 语言基础知识进行了庖丁解牛式的深度剖析。此外，本书还针对各大 IT 名企的笔试考题，总结出了适用于应试的方法与思路，可以帮助读者轻松应对 Java 面试笔试中的各类问题。

本书由何昊、薛鹏、叶向阳共同编著。在本书的编写过程中，董西成、邵帅、王震、伍文明、李超、曹润涛、郭晶晶、阎贝、林方超、廖兰新、李志强、厉孙德、褚艳利、丁志浩、卢山、梁敏、回永利等为本书的编写提供了非常宝贵的材料。软件工程中心武方方主任、张向虎主任、鲁昊鹏研究员、张剑研究员、张玉博高工、屈华敏研究员、苏媚高工、杨黎高工、李靖高工、张敏高工、赵亮高工、辛航高工、徐建军高工、郑小宁高工等领导对我的工作给予了无微不至的帮助。何四为律师为本书提供了一些有关版权的法律援助。机械工业出版社计算机分社的时静编辑给了我大力的支持与通力的配合。除此之外，我的父母、亲人、同事、朋友、同学，无论我遇到了多大的挫折与困难，他们都一如既往地支持与帮助我，使我能够开开心心地度过每一天。在此对以上所有人一并致以最衷心的感谢。

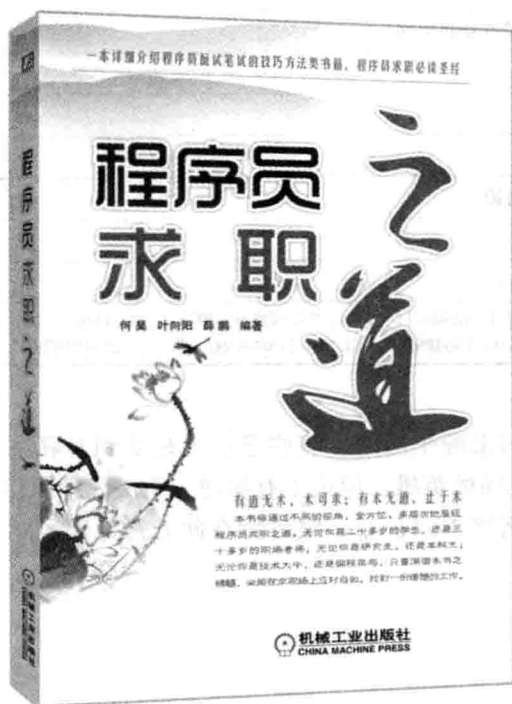
创作的过程是一个自我斗争、自我救赎的过程。无数个节假日，无数个深夜，当其他人沉浸在欢歌笑语中时，我需要安静地坐在计算机前，对所编写的内容仔细推敲，力求简单明了；将实例代码一一验证，力求准确无误。尽管孤独，但我觉得只要自己所做的事情、所付出的辛苦能为读者们增添几分求职成功的把握，我就心满意足了。

由于本人水平有限，书中不足之处在所难免，还望读者见谅。读者如果发现问题或是有此方面的困惑，可以通过邮箱 [xdhehao@foxmail.com](mailto:xdhehao@foxmail.com) 或者 [xdxuepeng@foxmail.com](mailto:xdxuepeng@foxmail.com) 与我们联系。

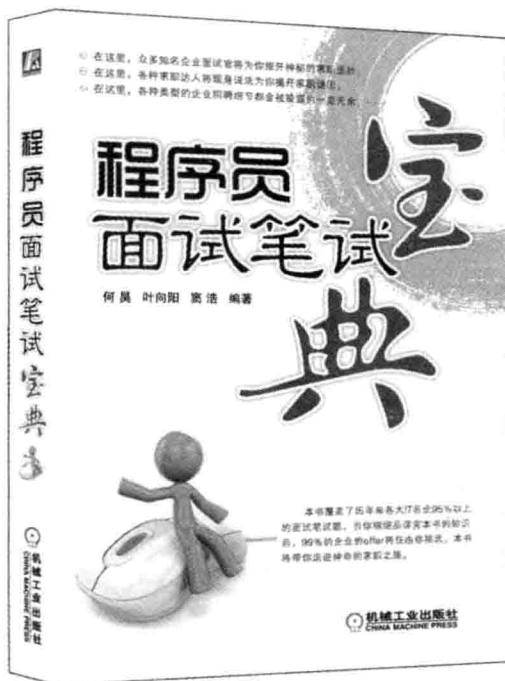
何 昊

于古都西安

## 优秀图书推荐



作者：何昊等编著  
出版社：机械工业出版社  
出版时间：2014-3-1  
ISBN：9787111457466  
定价：29.80 元



作者：何昊等编著  
出版社：机械工业出版社  
出版时间：2012-10-1  
ISBN：9787111398790  
定价：59.80 元

# 目 录

前言	main (String[] args) 这个方法	48
上篇 面试笔试经验技巧篇	4.1.4 如何实现在 main() 方法执行前输出 “Hello World”	49
第1章 面试官箴言	4.1.5 Java 程序初始化的顺序是怎样的	50
1.1 有道无术，术可求；有术无道， 止于术	4.1.6 Java 中的作用域有哪些	51
1.2 求精不求全	4.1.7 一个 Java 文件中是否可以定义 多个类	52
1.3 脚踏实地，培养多种技能	4.1.8 什么是构造函数	53
1.4 保持空杯心态	4.1.9 为什么 Java 中有些接口没有 任何方法	54
1.5 职场是能者的舞台	4.1.10 Java 中的 clone 方法有什么作用	56
1.6 学会“纸上谈兵”	4.1.11 什么是反射机制	60
1.7 小结	4.1.12 package 有什么作用	61
第2章 面试心得交流	4.1.13 如何实现类似于 C 语言中函数 指针的功能	62
2.1 心态决定一切	4.2 面向对象技术	64
2.2 假话全不说，真话不全说	4.2.1 面向对象与面向过程有什么区别	64
2.3 走自己的路，让别人去说吧	4.2.2 面向对象有哪些特征	65
2.4 夯实基础谋出路	4.2.3 面向对象的开发方式有什么优点	65
2.5 书中自有编程法	4.2.4 什么是继承	66
2.6 笔试成绩好，不会被鄙视	4.2.5 组合和继承有什么区别	66
2.7 不要一厢情愿做公司的 “备胎”	4.2.6 多态的实现机制是什么	67
2.8 小结	4.2.7 重载和覆盖有什么区别	69
第3章 企业面试笔试攻略	4.2.8 抽象类 (abstract class) 与接口 (interface) 有什么异同	70
3.1 互联网企业	4.2.9 内部类有哪些	71
3.2 网络设备提供商	4.2.10 如何获取父类的类名	73
3.3 外企	4.2.11 this 与 super 有什么区别	74
3.4 国企	4.3 关键字	76
3.5 研究所	4.3.1 变量命名有哪些规则	76
3.6 创业型企业	4.3.2 break、continue 以及 return 有 什么区别	76
3.7 如何抉择	4.3.3 final、finally 和 finalize 有什么 区别	77
下篇 面试笔试技术攻克篇	4.3.4 assert 有什么作用	79
第4章 Java 基础知识	4.3.5 static 关键字有哪些作用	80
4.1 基本概念	4.3.6 使用 switch 时有哪些注意事项	83
4.1.1 Java 语言有哪些优点		
4.1.2 Java 与 C/C++ 有什么异同		
4.1.3 为什么需要 public static void		

4.3.7	volatile 有什么作用	85	4.8.1	为什么说 Java 是平台独立性语言	125
4.3.8	instanceof 有什么作用	86	4.8.2	Java 平台与其他语言平台有哪些区别	126
4.3.9	strictfp 有什么作用	86	4.8.3	JVM 加载 class 文件的原理机制是什么	126
4.4	基本类型与运算	87	4.8.4	什么是 GC	127
4.4.1	Java 提供了哪些基本数据类型	87	4.8.5	Java 是否存在内存泄露问题	130
4.4.2	什么是不可变类	89	4.8.6	Java 中的堆和栈有什么区别	131
4.4.3	值传递与引用传递有哪些区别	91	4.9	容器	133
4.4.4	不同数据类型的转换有哪些规则	94	4.9.1	Java Collections 框架是什么	133
4.4.5	强制类型转换的注意事项有哪些	95	4.9.2	什么是迭代器	134
4.4.6	运算符优先级是什么?	95	4.9.3	ArrayList、Vector 和 LinkedList 有什么区别	135
4.4.7	Math 类中 round、ceil 和 floor 方法的功能各是什么	96	4.9.4	HashMap、HashTable、TreeMap 和 WeakHashMap 有哪些区别	136
4.4.8	++i 与 i++ 有什么区别	97	4.9.5	用自定义作为 HashMap 或 HashTable 的 key 需要注意哪些问题	137
4.4.9	如何实现无符号数的右移操作	98	4.9.6	Collection 和 Collections 有什么区别	142
4.4.10	char 型变量中是否可以存储一个中文汉字	100	4.10	多线程	142
4.5	字符串与数组	102	4.10.1	什么是线程? 它与进程有什么区别? 为什么要使用多线程	142
4.5.1	字符串创建与存储的机制是什么	102	4.10.2	同步和异步有什么区别	143
4.5.2	"=="、equals 和 hashCode 有什么区别	103	4.10.3	如何实现 Java 多线程	144
4.5.3	String、StringBuffer、StringBuilder 和 StringTokenizer 有什么区别	105	4.10.4	run()方法与 start()方法有什么区别	146
4.5.4	Java 中数组是不是对象	107	4.10.5	多线程同步的实现方法有哪些	148
4.5.5	数组的初始化方式有哪几种	107	4.10.6	sleep()方法与 wait()方法有什么区别	149
4.5.6	length 属性与 length()方法有什么区别	109	4.10.7	终止线程的方法有哪些	150
4.6	异常处理	110	4.10.8	synchronized 与 Lock 有什么异同	152
4.6.1	finally 块中的代码什么时候被执行	110	4.10.9	什么是守护线程	156
4.6.2	异常处理的原理是什么	113	4.10.10	join()方法的作用是什么	157
4.6.3	运行时异常和普通异常有什么区别	114	4.11	Java 数据库操作	158
4.7	输入输出流	116	4.11.1	如何通过 JDBC 访问数据库	158
4.7.1	Java IO 流的实现机制是什么	116	4.11.2	JDBC 处理事务采用什么方法	160
4.7.2	管理文件和目录的类是什么	118	4.11.3	Class.forName 的作用是什么	161
4.7.3	Java Socket 是什么	119	4.11.4	Statement、PreparedStatement 和 CallableStatement 有什么区别	161
4.7.4	Java NIO 是什么	120			
4.7.5	什么是 Java 序列化	121			
4.7.6	System.out.println()方法使用需要注意哪些问题	124			
4.8	Java 平台与内存管理	125			

- 4.11.5 getString()方法与 getObject()  
方法有什么区别 ..... 163
- 4.11.6 使用 JDBC 时需要注意  
哪些问题 ..... 164
- 4.11.7 什么是 JDO ..... 164
- 4.11.8 JDBC 与 Hibernate 有什么区别 ... 164
- 第 5 章 Java Web ..... 165**
  - 5.1 Servlet 与 JSP ..... 165
    - 5.1.1 页面请求的工作流程是怎样的 ... 165
    - 5.1.2 HTTP 中 GET 与 POST 方法  
有什么区别 ..... 166
    - 5.1.3 什么是 Servlet ..... 166
    - 5.1.4 doPost()方法与 doGet()  
方法怎么选 ..... 168
    - 5.1.5 什么是 Servlet 的生命周期 ..... 168
    - 5.1.6 JSP 有哪些优点 ..... 169
    - 5.1.7 JSP 与 Servlet 有何异同 ..... 169
    - 5.1.8 如何使用 JSP 与 Servlet 实现  
MVC 模型 ..... 170
    - 5.1.9 Servlet 中 forward 和 redirect  
有什么区别 ..... 172
    - 5.1.10 JSP 的内置对象有哪些 ..... 173
    - 5.1.11 request 对象主要有哪些方法 ..... 174
    - 5.1.12 JSP 有哪些动作 ..... 175
    - 5.1.13 JSP 中 include 指令和 include 动作  
有什么区别 ..... 176
    - 5.1.14 会话跟踪技术有哪些 ..... 176
    - 5.1.15 Web 开发中如何指定  
字符串的编码 ..... 177
    - 5.1.16 什么是 Ajax ..... 178
    - 5.1.17 cookie 和 session 有什么区别 ..... 178
  - 5.2 J2EE 与 EJB ..... 179
    - 5.2.1 什么是 J2EE ..... 179
    - 5.2.2 J2EE 中常用的术语有哪些 ..... 179
    - 5.2.3 EJB 有哪些不同的类别 ..... 181
    - 5.2.4 EJB 与 JavaBean 有什么异同 ..... 182
    - 5.2.5 EJB 有哪些生命周期 ..... 182
    - 5.2.6 EJB 的角色有哪几种 ..... 184
    - 5.2.7 EJB 的开发流程是怎样的 ..... 185
    - 5.2.8 EJB 3.0 与 EJB 2.0 有哪些  
不同之处 ..... 187
    - 5.2.9 EJB 容器有哪些作用 ..... 187
    - 5.2.10 EJB 规范规定 EJB 中禁止的  
操作有哪些 ..... 187
    - 5.2.11 Web 服务器与 Web 应用服务器  
有什么区别 ..... 188
    - 5.2.12 什么是 Web Service ..... 188
    - 5.2.13 SOAP 与 REST 有什么区别 ..... 189
    - 5.2.14 什么是 XML ..... 190
    - 5.2.15 数据库连接池的工作机制  
是怎样的 ..... 193
    - 5.2.16 J2EE 开发有哪些调优的方法 ..... 193
  - 5.3 框架 ..... 194
    - 5.3.1 什么是 Struts 框架 ..... 194
    - 5.3.2 Struts 框架响应客户请求的  
工作流程是什么 ..... 196
    - 5.3.3 Struts 框架的数据验证可分为  
几种类型 ..... 197
    - 5.3.4 Form Bean 的表单验证流程  
是什么 ..... 197
    - 5.3.5 在 Struts 配置文件中, <action >  
元素包含哪些属性和子元素 ..... 197
    - 5.3.6 ActionForm Bean 的作用有哪些 ... 198
    - 5.3.7 ActionForm 的执行步骤有哪些 ..... 198
    - 5.3.8 forward 与 global-forward  
有什么区别 ..... 198
    - 5.3.9 Struts 如何实现国际化 ..... 199
    - 5.3.10 Struts 1 与 Struts 2 有哪些区别 ... 199
    - 5.3.11 什么是 IoC ..... 201
    - 5.3.12 什么是 AOP ..... 203
    - 5.3.13 什么是 Spring 框架 ..... 205
    - 5.3.14 什么是 Hibernate ..... 206
    - 5.3.15 什么是 Hibernate 的二级缓存 ..... 211
    - 5.3.16 Hibernate 中 session 的 update() 和  
saveOrUpdate()、load() 和 get()  
有什么区别 ..... 212
    - 5.3.17 Hibernate 有哪些主键  
生成策略 ..... 213
    - 5.3.18 如何实现分页机制 ..... 214
    - 5.3.19 什么是 SSH ..... 214
- 第 6 章 数据库原理 ..... 216**
  - 6.1 SQL 语言的功能有哪些 ..... 216
  - 6.2 内连接与外连接有什么区别 ..... 218
  - 6.3 什么是事务 ..... 219



6.4 什么是存储过程? 它与函数 有什么区别与联系 .....	220	8.3.7 如何进行堆排序 .....	255
6.5 各种范式有什么区别 .....	221	8.3.8 各种排序算法有什么优劣 .....	257
6.6 什么是触发器 .....	222	8.4 位运算 .....	257
6.7 什么是游标 .....	223	8.4.1 如何用移位操作实现乘法运算 .....	257
6.8 如果数据库日志满了, 会出现 什么情况 .....	224	8.4.2 如何判断一个数是否 为 2 的 n 次方 .....	258
6.9 union 和 union all 有什么区别 .....	224	8.4.3 如何求二进制数中 1 的个数 .....	259
6.10 什么是视图 .....	225	8.5 数组 .....	260
<b>第 7 章 设计模式</b> .....	226	8.5.1 如何寻找数组中的最小值与 最大值 .....	260
7.1 什么是单例模式 .....	226	8.5.2 如何找出数组中第二大的数 .....	261
7.2 什么是工厂模式 .....	227	8.5.3 如何求最大子数组之和 .....	262
7.3 什么是适配器模式 .....	229	8.5.4 如何找出数组中重复元素 最多的数 .....	266
7.4 什么是观察者模式 .....	229	8.5.5 如何求数组中两两相加等于 20 的 组合种数 .....	267
<b>第 8 章 数据结构与算法</b> .....	231	8.5.6 如何把一个数组循环右移 k 位 .....	268
8.1 链表 .....	231	8.5.7 如何找出数组中第 k 个 最小的数 .....	269
8.1.1 如何实现单链表的增删操作 .....	231	8.5.8 如何找出数组中只出现 一次的数字 .....	270
8.1.2 如何从链表中删除重复数据 .....	234	8.5.9 如何找出数组中唯一的 重复元素 .....	272
8.1.3 如何找出单链表中的倒数 第 k 个元素 .....	235	8.5.10 如何用递归方法求一个整数 数组的最大元素 .....	276
8.1.4 如何实现链表的反转 .....	236	8.5.11 如何求数对之差的 最大值 .....	276
8.1.5 如何从尾到头输出单链表 .....	237	8.5.12 如何求绝对值最小的数 .....	280
8.1.6 如何寻找单链表的中间结点 .....	237	8.5.13 如何求数组中两个元素的 最小距离 .....	282
8.1.7 如何检测一个链表是否有环 .....	238	8.5.14 如何求指定数字在数组中第一次 出现的位置 .....	283
8.1.8 如何在不知道头指针的情况下 删除指定结点 .....	239	8.5.15 如何对数组的两个子有序段 进行合并 .....	284
8.1.9 如何判断两个链表是否相交 .....	239	8.5.16 如何计算两个有序整型 数组的交集 .....	285
8.2 栈与队列 .....	241	8.5.17 如何判断一个数组中数值 是否连续相邻 .....	287
8.2.1 栈与队列有哪些区别 .....	241	8.5.18 如何求解数组中反序对的 个数 .....	288
8.2.2 如何实现栈 .....	241	8.5.19 如何求解最小三元组距离 .....	289
8.2.3 如何用 $O(1)$ 的时间复杂度 求栈中最小元素 .....	243	8.6 字符串 .....	292
8.2.4 如何实现队列 .....	244	8.6.1 如何实现字符串的 反转 .....	292
8.2.5 如何用两个栈模拟队列操作 .....	246		
8.3 排序 .....	247		
8.3.1 如何进行选择排序 .....	247		
8.3.2 如何进行插入排序 .....	248		
8.3.3 如何进行冒泡排序 .....	249		
8.3.4 如何进行归并排序 .....	250		
8.3.5 如何进行快速排序 .....	252		
8.3.6 如何进行希尔排序 .....	254		

8.6.2	如何判断两个字符串是否由相同的字符组成	293	8.8.1	如何消除嵌套的括号	309
8.6.3	如何删除字符串中重复的字符	295	8.8.2	如何不使用比较运算就可以求出两个数的最大值与最小值	311
8.6.4	如何统计一行字符中有多少个单词	297	<b>第9章 海量数据处理</b>		<b>312</b>
8.6.5	如何按要求打印数组的排列情况	298	9.1	问题分析	312
8.6.6	如何输出字符串的所有组合	300	9.2	基本方法	312
<b>8.7 二叉树</b>		<b>301</b>	9.3	经典实例分析	325
8.7.1	二叉树基本概念	301	9.3.1	top K 问题	325
8.7.2	如何实现二叉排序树	303	9.3.2	重复问题	328
8.7.3	如何层序遍历二叉树	305	9.3.3	排序问题	330
8.7.4	已知先序遍历和中序遍历, 如何求后序遍历	306	<b>附录</b>		<b>331</b>
8.7.5	如何求二叉树中结点的最大距离	308	附录 A	软件企业 Java 笔试真题 1	331
<b>8.8 其他</b>		<b>309</b>	附录 B	软件企业 Java 笔试真题 2	336
			附录 C	软件企业 Java 笔试真题 3	339
			附录 D	求职有用网站及 QQ 群一览表	342

上篇

# 面试笔试经验技巧篇

第 1 章 面试官箴言

第 2 章 面试心得交流

第 3 章 企业面试笔试攻略

# 第1章

## 面试官箴言

什么样的求职者能够获得面试官的青睐？求职者需要准备哪些内容来面对形形色色的面试官？什么样的企业适合自己发展？在新的工作岗位上，如何努力才能从人才济济的企业中脱颖而出？本章中，几位资深软件工程师将现身说法，为您一一解答上述问题。

### 1.1 有道无术，术可求；有术无道，止于术

丁志浩，男，硕士，某知名芯片公司软件工程师。

以下这些内容是写给即将成为职业人的在校学生的，希望能够对他们的求职与以后的工作有一定的参考作用。

在介绍求职之前，我想先说一些与具体技术无关但却比技术更加重要的东西，主要有以下两个方面的内容：第一点，认清自我；第二点，保持强烈的求知欲。之所以提及这两点，并且认为它们是最重要的东西，是结合我的亲身经历，我认为一个人最重要的是认清自我，只有认清了自我，你才会知道自己想要做什么、适合做什么、能做什么。在某种程度上来说，这比所学的知识、技术更加重要。只有方向正确了，才会有前进的动力；只有有了前进的动力，才会为目标不断努力；只有朝着正确方向不断努力了，才可能会有收获。其次，要有强烈的求知欲，随着年龄的增大、个人阅历的增长，生活、家庭、工作会慢慢消磨掉你的雄心壮志，而能保持强烈的求知欲实属难能可贵。世界上很少有学不会的东西，就看你是否用心去做了，是否愿意花时间、动脑筋、投入精力去做，万事就怕认真，只要你认真做了，通常是可以学会的。

切入正题，作为一名以程序员为职业目标的求职者，关注的领域主要还是以技术为主，IT企业在面试的时候主要关注求职者什么方面的内容呢？以我这些年的工作经历来看，大企业看道，小企业看术。有道无术，术可求；有术无道，止于术。具体来说，大企业更加看重的是求职者的基础知识以及解决问题的能力。一般而言，大企业都会有比较完备的培训机制，它可以在较短时间内把一个什么都不会的员工塑造成一个它想要的人；而小企业则不然，他们更加注重求职者的实用性，求职者当前会什么，能给企业带来什么。这种思维方式的不同其实也是由企业的性质决定的，其本身没有对错之分。当然这也无可厚非，所以个人建议求职者最好夯实计算机基础知识，操作系统、编译原理、算法等这些基础知识就是重中之重了，需要重点掌握。万变不离其宗，当你到达了一定程度，对你而言只是个形式上的差异而已。

对于求职者需要如何准备才能更好地获得面试官的青睐，我觉得，IT企业一般需要的大

多数都是技术型人才，所以具有以下3个优点的人，一般更能受到面试官的亲睐：①基本功扎实的人，基础扎实了，后劲就足，发展前景就更好；②具有强烈的求知欲、对未知领域比较感兴趣、能够接受新事物的人；③在某个领域有比较深入的研究的人。因为如果求职者已经在某个方面有了比较深入的研究，有了良好的基础，对于将来的发展肯定会更好，例如，当前好多企业都在搞云计算，如果求职者对 Hadoop 这种架构有比较深入的理解，当然就比不懂 Hadoop 的求职者成功率更高。

有了录用通知（offer）以后，在挑选 offer 时，求职者往往也很纠结，其实我在这里也不是告诉求职者是该选择互联网还是芯片公司，或是其他类型企业，因为对这个问题，仁者见仁智者见智，每个人考虑的侧重点也不一样，所以在此我不说到底该选什么企业，以免误导大家，但我可以给求职者一个建议：往大的方面讲，首先是选择行业，然后选择企业，最后是选择职业。最好能够综合自己的兴趣爱好，因为兴趣是最好的老师。

入职之后，如何才能适应新的工作岗位，完成从学生到职业人的华丽转变呢？一般而言，刚毕业时，新人都是雄心壮志、意气风发，想在新的工作岗位上大展拳脚、有所作为，这虽然是一件非常好的事情，但是现代企业分工很明确，尤其是对于企业的新员工，刚工作时，很有可能接触的东西都是些缺乏技术含量或是相对边缘化的东西，只是充当企业的一颗“小螺丝钉”而已。所以在此，我建议求职者在刚入职时，最好能够放低姿态，当将军的人，都是从小兵一步步做起的。刚毕业态度最重要，切忌整天怨天尤人，否则会给人一种浮躁的感觉，对将来的发展肯定是不利的。

## 1.2 求精不求全

褚艳利，女，硕士，某知名电子商务公司软件工程师。

时光荏苒，我已经成为 IT 业一名所谓的“老鸟”了，但我也曾是一名普通的求职者，也曾在求职的路上历经风雨。希望我的一些经历和感悟，能为朋友们提供些许帮助。

对于应届生求职，我觉得每一场面试都是从“闻味儿”开始的。看似是一场简单的聊天，但其实求职者的各方面已经在被面试官考查了，例如在沟通过程中，从求职者的谈吐、穿着、眼神、沟通过程中，或多或少就“闻”出很多层“味道”了（求职者的性格、处事态度、表达能力、沟通能力、团队合作能力）。经常会听到求职者说：“面试官今天一道技术题都没问我。”这多是面试官对求职者综合素质的一种肯定（前提是成绩单不能太难看）。如果是应聘技术类职位，那么求职者的技术水平还是要积累的。

对于技术的积累，我觉得是“求精不求全”，如今的高等院校通常都会开设“C 语言”、“C++”、“Java”、“网络”、“数据库”、“编译原理”、“软件工程”等课程，但由于精力有限，毕竟不是每个人都可以做到门门精、样样通，所以我建议从兴趣出发，深入学习几门课程（当然，其他课程也要学，毕竟是在技术领域，一些概念和基本原理不知晓是不行的），例如我个人比较钟爱数据结构、算法、C 语言、操作系统等专业知识，对这些下足功夫做足功课，也正是这些基础，让我打赢了很多场艰难“战役”。当然，在面试他人的过程中，我也会问到一些可能他们不太擅长的知识，例如设计模式。其实我并不是为难他，只要他能讲出自己的理解，并直言自己这方面知识的欠缺，我也觉得无可厚非，这种坦白比不懂装懂来得更真实、更有力量。所以，作为一名“过来人”，我觉得大部分面试官在面试时，会更加侧重于考查求职者擅长的方面，从这点能看到求职者未来发展和潜力。

作为一名职场新手，在准备过程中，求职者应该根据职位要求略作筹备。虽然说万变不离其宗，但根据职位要求，有针对性地准备一下，效果会更好。例如面试数据库开发的，DB（数据库）知识就需要好好准备一下，这样不至于因什么也答不出来而弄得气氛太尴尬，也可获得后续的面试机会。对于普通的软件开发类职位，我认为求职者应该必备以下知识：数据结构、某类编程语言、操作系统和基本 DB 知识。

我觉得要想获得面试官们的青睐，求职者需要注意以下几个方面的问题：

- 1) 衣着妆扮。对于技术类职位，衣着妆扮虽然不做要求，但也不能过于邈邈。女性求职者画一点淡妆更好。
- 2) 眼神交流。记住，你对面坐着的是面试官，不是墙壁，你需要跟他有眼神交流。不要怕，试着抬起头来，面试官的笑容多半可以缓解求职者的紧张情绪，以及答不上题的尴尬气氛。之所以害怕，其实是自己吓倒了自己。
- 3) 气氛把握。语速不要太快，太快就容易将自己置于紧张的状态之中，回答问题无论会与不会，都要放慢节奏，松缓身心，因为你的状态会直接影响面试官的感受以及判断。
- 4) 背景了解。如果你参加一家公司的面试，最好是你真心喜欢的，并且对公司多少应该有些了解。例如公司理念、制度、规划，谈谈你喜欢的、你认为可以改善的（这一点上要注意“度”），如果你用了心，面试官往往会给予更多机会的。
- 5) 轻松话题。如果谈得比较愉快，求职者可以自己制造些轻松话题，例如旅游、业界话题等。

很多师弟、师妹们问我，如何挑选 offer，需要权衡哪些内容。我不是一名职业规划师，所以不能告诉他们如何做选择，我只能告诉他们，当初我在选择 offer 的时候，考虑了哪些内容，以供他们参考。但总的来说，我觉得应该参考以下 5 点内容：

- 1) 兴趣点。兴趣是最好的老师，如果没有兴趣，你很难在工作岗位上有所作为。
- 2) 公司未来发展空间和路线。很多时候不能只盯住眼前的利益，要从长远看，一个企业的发展空间和路线、对未来市场的认知与把握都会决定你未来的发展方向，所以，最好能够对企业的未来发展空间与路线有一个较清醒的认识。
- 3) 薪酬福利。“钱不是万能的，没有钱是万万不能的”。一个企业再好，如果不给工资，同样没人会去，因为人要吃饭穿衣，所以必须仔细考虑薪酬福利。
- 4) 个人成长点。每个企业对人才的定位都不一样，所以在选择 offer 时，尽量选择一些企业的核心研发部门，在这样的部门里面，个人成长、个人机会都会非常好。
- 5) 城市。什么样的城市是自己希望的，是政治中心北京，还是东方明珠上海；是人间天堂杭州，还是千年古都西安；是天府之国成都，还是千湖之城武汉。各个城市有各个城市的优劣，所以没有谁能够告诉你哪个城市好哪个城市不好，关键需要你自己拿主意。

其实，选完了 offer 之后，就面临着一个从学生到职业人身份的转换了，如何转换角色，我个人觉得新人入职之初，最重要的就是练就基本功，这个阶段犹如蚕蜕，是痛苦但也美丽的变身。例如，我们做的是线上一级系统，承载着每秒数万笔交易的创建及支付，那么系统的架构、稳定性、容量、可扩展性以及各种底层技术实现，方方面面要学的太多，任务紧、压力大、面对着无数个不可能，这个过程看似痛苦但却会让我们成长得非常快。尤其是当项目真正上线运转起来时，那些你原先认为不可能做的事情都做到了，还做得非常漂亮，那种成就感真的是无以言表。而且做每件事情的时候，一定要把姿态放下来、心态静下来、自信提上去，与你的团队一起合作，把不可能当作历史，把可能写在今天。经历一段时间的锻炼之后，你应该仔细思考一下，问一问自己是否可以独当一面，是否在业界，至少在公司部门内，可以听到你的“声音”、可以看到你的建议，如果可以，那么恭喜你，你应该可以升职了。

## 1.3 脚踏实地，培养多种技能

廖兰新，男，硕士，某创新型企业高级研发工程师、开发经理

作为一名一线的技术研发人员，我结合自己这么多年来在工作中的经历，分享一些经验给即将走入职场的毕业生，希望能帮助他们在求职的路上少走一些弯路。

### (1) 行业选择

我个人觉得应届毕业生择业时，选择适合自己的行业是非常重要的。对于计算机类专业的毕业生，可供选择的行业很多，例如商业银行类、国企、央企、传统的软件公司、新兴的互联网公司等等。而这些行业又各有各的特点，对求职者能力的要求迥异。例如国企的工作相对轻松、薪资一般（体制内）、福利很好，对技术要求不是太高，对项目进度的要求一般不紧迫；互联网公司工作一般比较辛苦，对项目进度要求非常紧迫，技术研发能力也要求高，而企业文化一般较为自由，其薪资待遇也相对较高。所以，求职者应该根据自己的兴趣爱好以及能力特点选择合适的行业。

### (2) 技术领域选择

随着现代化管理技术的不断发展，IT企业中的技术分工也越来越明显。俗话说：“隔行如隔山”同样是计算机科学技术，不同技术领域的人在技术上也是非常迥异的，例如互联网企业与芯片企业关注的重心就不一样。毕业生一般也很难做到“通才”。所以，在求职时，求职者应尽量选择自己喜欢或擅长的专业领域，这些会决定你今后职业生涯的主要工作内容，而且一般也不会轻易更换。

### (3) 雇主选择

不同的雇主对求职者要求也不一样。以大型科技公司与创业型科技公司为例加以比较。创业型公司一般研发人员相对较少，每个研发人员都需要能够独当一面，对整个产品的核心代码都了如指掌，上至前端开发、Web界面，下至后台底层实现、操作系统，所以这对于个人成长是非常好的锻炼机会，但同样，创业公司也有其自身的局限性，由于工作的需要，员工一般身兼数职，经常加班，而且在专业技能上都不够规范，相比大型科技公司完善的团队、严格的规章制度等，相对欠缺。

但总的来说，创业型公司更能全方位地激发个人潜能，多角度地发展个人能力，而大型科技公司可以集中锻炼某项专业技能。当然，上述说法也不是绝对的，比如某些小型高科技公司也聚集了业内的人才，完全具备大企业的“高精尖”特点，而一些大公司的某些部门在初创阶段可能也会像创业公司一样艰苦。如果难于抉择，那你就尽量去一家已步入正轨的大公司吧。

### (4) 求职建议

因为企业需要，我曾经担任过一段时间的面试官，帮助企业招聘新人。我们确实非常希望招聘到优秀的人才，但在招聘过程遇到了很多令人遗憾的事情。例如有的人在面试时因为紧张或是其他原因，发挥不出真实水平；有的人水平一般，却夸夸其谈，不脚踏实地，真正让他设计算法时，一头雾水。在此，我想说明一点，企业在招聘新人时，需要这样的人才：对人对事有信心、掌握多项技能、基础扎实、有冲劲、愿拼搏。所以，我建议毕业生在平时的学习中，一定要脚踏实地地学好专业知识，适当地扩展专业技能。

### (5) 能力培养

进入工作岗位之后，很多毕业生颇感迷茫，很难快速从学生的角色向职业人的角色转变。我觉得IT行业的职业人应注重培养自身的3种能力：技术能力、管理能力和领导能力。职场

新人往往要靠技术能力进入职场，最初的晋升也主要依靠技术能力，它可以让你成为一名优秀的单兵或一名称职的经理，但很难让你成为优秀的经理人，因为它的杠杆效应非常有限，这时就需要第二种能力：管理能力。管理其实是对资源的管理和利用，以有效、可靠地生产产品或提供服务。管理能力一般可以通过学习得到，教育、经验、培训都是提高管理能力的手段。当然，个人的悟性也很重要，能够从表面现象中分析出规律，对管理能力来说很重要。管理能力主要是释放物的能力，它可给你一定的杠杆力量，能让你在小范围内有所贡献，但不会让你“走”很远。这时就需要第三种能力：领导能力。领导能力是释放别人的能力，再通过别人来释放个人或物的能力。领导能力作用可谓巨大，因为它有二级杠杆的效用。就领导而言，技术能力的重要性非常有限，管理能力次之，领导能力最为重要。职业道路不是单行道，而是可以从技术职位向管理职位过渡，再由管理向领导职位过渡的。

## 1.4 保持空杯心态

王震，男，硕士，某知名互联网企业研发工程师。

好友何昊拜托我一件事情，就是给当前程序员写一些关于求职的意见与建议，这着实有些为难我，并非我不愿意去做这样一件事情，而是因为本人入行虽然比较早，但入职却不太久，与一些资深的 IT 精英们相比，只能算是初出茅庐，所以不敢妄自尊大。不过，我非常愿意分享一下个人这些年来的几点粗浅体会，以起到抛砖引玉的效果。

对于个人的发展，扎实的基本功将更有利于你在行业里站稳脚跟，走得更远。“术业有专攻”，所谓专业，在于求深而不在于求广。当然，话无绝对，更广的知识面可以帮助你对整个大行业背景有一个比较清晰的认识，知道自己在产业链中处于一个什么样的位置，能够做出多大的成就，有多大的发展空间。结合我自己的经历，以软件类研发为例，具体而言，后台开发方向，系统、网络的底层，比如操作系统事件机制（例如 Windows 消息机制、Linux epoll 等），TCP/IP 协议栈，C/C++ STL 等，这些是服务器开发的主战场，对这里每项技术需要了解的程度应如同战场上士兵对手中所握兵器需要熟悉的程度一样。也许对小规模服务器程序开发而言，谈论这些内容可能有些夸大其辞、危言耸听，但确实存在很多需要如此考虑的情况，例如当前很多网上订票系统很难满足实际应用的需要，引起了用户的极大反感。而在前端方面，由于技术更迭较快，快速学习能力就显得尤为重要，程序员应紧跟时代潮流就要看准当前的形式，了解站在时代前沿的人有哪些，他们做了什么以及他们的研究成果有哪些。

至于经典的数据结构、算法，无论是前端研发还是后台研发都会有所涉及，即便是更深入的掌握一般也只在较专业的算法密集型领域，比如搜索、GIS 等。而对于你、对于面试官更注重什么，则看你们更侧重哪方面的内容了。

如果是已经入行的程序员应聘新的企业，经验及能力通常是面试官考查的重头戏。说得更直白一点，作为利益链条上的一环，你具备什么资本，能为公司创造什么价值，才是面试官关注的焦点所在，这也是你需要真正搞清楚并且为之准备的内容。做过什么项目，取得了什么样的成就，既说明了你的过往表现，也能将你的潜在价值表露一二。

进入工作岗位，我相信，不管是刚入职的毕业生还是已打拼多年的程序员，以“空杯心态”去融入当前企业文化，绝对不是件坏事。只有认可了企业的文化，工作时，你才能积极主动，才能上进、才能得到提升。就职业发展而言，一般公司都会有量化的绩效指标，完成这个指标即是一种自我提升，而在任务指标之外，结合自身情况制定出半年或全年个人发展规划，可以说是对



自己短期能力提升的督促和目标实现的指引，有助于自己向着更明确的方向发展。

以上愚见称不上是成功的经验，只是我对程序员这个行业一点浅薄的理解而已。

## 1.5 职场是能者的舞台

林方超，学士，北京某上市公司软件工程师。

关于毕业生如何求职这个问题，老实说，我的“经验”并不是很多，若干年以前，因为应聘前准备得比较充分，所以命中率比较高，虽然也拿到了几个不错的 offer，但最终还是选择了现在这家企业。这么多年过去了，回过头来想想，也是感悟颇深。

我认为一个非常有针对性的准备工作，包括心理准备与知识准备，这对计算机相关专业的毕业生求职非常有用。

首先，求职者应当找准自己的位置，即通常所说的职位。一个对职位有着准确预期、对自己有着准确定位的人，在个人简历、面试中都能够表达出更准确、更吸引人的信息，而不至于投递完简历之后就没了下文。而找准一个方向、找准一个行业或是锁定一个企业，不仅可以缩小求职的范围，而且还可以让你在有限的精力、有限的时间内将准备的内容进一步深入、细化。如果做到了这一点，不管是大企业的招聘还是小企业的招聘，也不管是在笔试还是面试，你很快就能发现，真正能够与你竞争的人、能够把你 PK 掉的人真的是屈指可数，此时你就成了求职大军中“笑到最后的人”（插入一个感悟：时下往往被人普遍提及的流行技术，反而是陈词滥调，只有真正理解其中思想者才能脱颖而出，如果没有十成的把握，我宁可绝口不提）。

通过一些有针对性的准备工作后，笔试一般就不会存在问题了。而紧接着需要面对的就是面试这一关，每一次求职机会都很宝贵，每一次面试机会也很难得，而成功随时会降临。作为求职者，不应当将机会随意浪费掉，将成功拒之门外。所以，不要总以为自己运气好，可以“裸装上阵”赌一把。因为作为求职者，在与企业的博弈中，我们是弱勢的，因此，你需要对所应聘的企业以及岗位有一定的认识与了解，当然，你通常在此之前对其可能一无所知，如果此时稀里糊涂地去了，自然也是稀里糊涂地回来。其实，只要提前做好功课，这些都不是问题，因为如今的企业一般都会有自己的宣传网站，里面会详细地介绍企业的发展历程和现状，此外，不少网站在校园招聘时也会列出详细的招聘信息，这些内容都可以好好看看。至于对这些内容需要了解到什么程度，就要看这家企业在你心目中的地位了。想象一下，在面试时，当你谈及企业的一些信息时，面试官会想要给你介绍更多，甚至愿意带你去实地参观一下，那么接下来基本就可以直接谈待遇、谈签约了。

介绍再多的理论和方法，也只是“纸上谈兵”，是否可行还需要用行动来验证，只有行动了才能体会到其中的价值。如果成功拿到 offer，那是最理想的；如果没成功，最好要让面试官给你些建议，遇到他说不出来或闪烁其词的情况，说明面试官是凭个人喜好作出的判断，大可不必理会；而一针见血的评价以及善意的建议都会对你未来的求职、成长有很大的帮助，所以，不能被“一根绳子绊倒两次”，无论是成功了还是失败了，都会有所启发，成功可以收获经验，失败同样可以得到教训。

挑选 offer 也是一件比较艰难的事情。个人建议求职者最好按照自己的职业规划进行比较，但如果确实没有很明确的职业规划，或是从来没有想过职业规划这个问题，你可以优先挑选有发展潜力的工作，这样的工作会给你带来许多意外的收获，最终帮助你确定自己的职业路线，构建你的职业规划。

最后，我想说的是，职场是能者的舞台，真正比拼的是各种能力。技术是一种能力、交际