

菜鸟入门：Java语言学习六大要点 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/645/2021_2022__E8_8F_9C_E9_B8_9F_E5_85_A5_E9_c104_645233.htm

1. 学习的目的是为了将来进行应用程序的开发，而不是进行语言理论研究 2. 将来的应用开发是在成熟的平台上展开，而不是自己从底层开发平台

一、掌握静态方法和属性 静态方法和属性用于描述某一类对象群体的特征，而不是单个对象的特征。Java中大量应用了静态方法和属性，这是一个通常的技巧。但是这种技巧在很多语言中不被频繁地使用。理解静态方法和属性对于理解类与对象的关系是十分有帮助的，在大量的Java规范中，静态方法和属性被频繁使用。因此学习者应该理解静态方法和属性。Java在方法和属性的调用上是一致的，区别只表现在声明的时候，这和c是不同的。

二、重视接口 在面向对象早期的应用中大量使用了类继承。随着软件工程理论不断发展，人们开始意识到了继承的众多缺点，开始努力用聚合代替继承。软件工程解决扩展性的重要原则就是抽象描述，直接使用的工具就是接口。接口近年来逐渐成为Java编程方法的核心。另一方面，就应用而言，大部分开发是建立在规范基础之上的，不需要自己建立复杂的继承关系和庞大的类。因此读懂规范和用好规范已经成为应用程序开发人员的首要任务，Java各项规范的主要描述手段就是接口。

三、学好集合框架 Java描述复杂数据结构的主要方式是集合框架。Java没有指针，而是通过强大的集合框架描述数组、对象数组等复杂的数据结构。学好这些数据结构的描述方法对于应用程序编写，特别是涉及到服务器方、3层结构编程至关重要。程序员在

这个时候不能再用诸如数据库结果集之类的结构描述数据了。由于很多语言没有这么强大的集合框架体系，很多初学者不知所措，更不知道拿来做什么用，因此应该引起足够的重视。

四、例外捕捉 Java对例外捕捉的强调是空前的，它强迫程序员用显著的与逻辑方法完全不同的方式描述例外捕捉，对于程序描述的完整性和严谨性有很大的意义。c也有类似的机制，但是我们看到很多c程序员并不习惯使用这些机制。Java的初学者应该充分学习好这种例外捕捉机制，养成良好的编程习惯。

五、多线程需要理解机理 很多Java程序员热衷于多线程程序编写，认为是对逻辑能力的挑战。其实在大量应用中根本就不需要编写多线程程序，或者说大多数编写应用程序的程序员不会去写多线程程序。这是因为多线程机制都内置到基础平台当中了。程序员应该了解的是多线程原理和多线程安全，这对于今后准确地把握程序是至关重要的。例如JSP中编写到不同的位置对于多个用户环境的安全影响完全不同，又如著名的Super Servlet是每一个访问作为一个进程，但是每一个页面是一个线程，和Servlet正好相反，对程序的性能和安全的影响有天壤之别。

六、了解网络编程 Java号称是最强的网络编程语言，但是大多数应用程序开发人员是从来不会自己开发什么底层的网络程序的。需要做只是了解原理就够了。网络机制的实现是靠平台实现的，除非自己开发平台，否则是不需要知道socket怎么实现，怎么监听访问的。因此在这方面花太多的功夫就偏离了“将来的应用开发是在成熟的平台上展开，而不是自己从底层开发平台”这一假设。

100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com