

摘要：Java 开发者常常都会想办法如何更快地编写 Java 代码，让编程变得更加轻松。目前，市面上涌现出越来越多的高效编程工具。所以，以下总结了一系列工具列表，其中包含了大多数开发人员已经使用、正在使用或将来一定会用到的高效工具。



Java 开发者常常都会想办法如何更快地编写 Java 代码，让编程变得更加轻松。目前，市面上涌现出越来越多的高效编程工具。所以，以下总结了一系列工具列表，其中包含了大多数开发人员已经使用、正在使用或将来一定会用到的高效工具。这份列表名单包括集成开发环境、集成工具、测试和质量工具等。

## 1、集成开发环境

Eclipse是最有名也最广泛使用的Java集成开发环境（IDE），允许开发者结合语言支持和其他功能到任何的默认包中，而且Eclipse市场有许多定制和扩展的插件。

IntelliJ已经引起了开发者的“追捧”，甚至比Eclipse还受宠，有免费版和专业版。IntelliJ为众多Java EE框架提供了针对框架的编码帮助和提升生产效率的功能，还有大多数应用服务器的部署工具。

NetBeans的也属于IDE类，允许开发者快速、简便地开发桌面应用、移动应用和Web应用。NetBeans是免费开源的，还拥有全球性的用户和开发者社区。

JDeveloper是Oracle提供的一个免费的集成开发环境，它为Oracle融合中间件和融合应用产品提供完整的端到端开发，支持整个开发生命周期。

## 2、测试

JUnit是一个Java语言的单元测试框架，用于编写和运行可重复的测试。它为测试预期结果提供断言，为共享的常用测试数据提供测试装置（test fixtures），以及运行测试的测试者（test runners）。可以在这里看一下JUnit的教程。

TestNG是受到了JUnit的启发开发而成，但同时还具有一些新功能，所以相对而言，TestNG更加强大也更便于使用，如注解（ annotations ）、灵活的测试配置、数据驱动的测试和参数支持，以及强大的执行模型。它涵盖了单元测试、功能、端到端、集成测试，支持各种工具和插件（ Eclipse、IDEA、Maven等 ）。具体内容可以参考[分别使用 Maven Failsafe 与 TestNG 进行单元测试与集成测试的文章](#)。

Mockito是一个mocking框架，帮助开发者编写具有简洁API的精美测试。其测试的可读性很好，并能产生清晰的验证错误。使用Mockito还能在执行后可以对交互进行询问。Mocks是透明的，所以开发者可以专注于测试选定的行为。更多详情可以参考[mockito的相关文章](#)。

Arquillian是一个Java和JVM的测试平台，主要侧重于集成和功能测试。它去掉了单元测试，重点关注真正实时环境下的程序代码的相互作用。Arquillian为测试配备了容器生命周期管理和典型测试精选。

### 3、集成

Apache Maven是一个非常著名的软件项目管理和综合工具。基于项目对象模型（ POM ）的概念，它的信息中心管理着项目的构建、报告和文档。Maven提供了一个统一的构建系统、项目信息和最佳的开发实践指导，并且支持新功能的透明迁移。

Gradle是一个开源的自动化构建系统。Gradle可以自动构建、测试、发布、部署，还有多种软件包或其他类型项目，如生成静态网站。Gradle结合了Ant的能力和灵活性，以及Maven的依赖管理和约定，形成了一种高效的构建方式。

Apache Ant是一个开源的Apache项目，经Apache软件许可发布。它由多种内置任务而成，允许开发者编写、装配、测试和运行java应用程序。它的特点是高度灵活，不会严格限制Java项目的编码惯例或目录布局。

Apache Ivy是一种流行的依赖管理工具，注重灵活性和简单性。它是Apache Ant项目的一个子项目，而Ivy用来解决项目依赖项。外部的XML文件定义项目依赖项，并列出构建项目的所需资源。然后Ivy从artifact 存储库（ artifact repository ）中下载资源，它具有良好的灵活性和可配置性。Apache Ivy 是由Apache Ant集成的，因此将使Apache Maven的依赖管理优势也带到Apache Ant构建文件中，为那些已经使用Apache Ant且不想迁移到Apache Maven的开发者提供了便利。

### 4、应用服务器

Apache Tomcat是一个开源的Web服务器和Servlet容器，由Apache软件基金会（ ASF ）开发。它实现了几个Java EE规范，包括Java Servlet、JavaServer Pages（ JSP ）、Java EL和WebSocket，提供了一个可以运行Java代码的HTTP Web服务器环境。想要了解更多请[点击](#)。

WildFly，其前身是JBoss AS或Jboss，是一个由Jboss开发的应用服务器，现在由红帽在开发维护。它是用Java编写的，并且实现了Java平台和企业版（ Java EE ）规范。WildFly是可以在多个平台上运行的免费开源工具。

GlassFish是由Sun Microsystems的Java EE平台起步，现在由Oracle公司赞助的开源应用服务器项目。支持的版本叫做Oracle GlassFish应用服务器。GlassFish是免费软件，经两个免费的软件许可证批准：通用开发与发布许可（CDDL）和GNU通用公共许可证（GPL）。

Oracle WebLogic 服务器是一个Java EE应用服务器，目前由Oracle公司开发，用于构建和部署企业应用Java EE，支持降低运营成本、提高性能、增强可扩展性和支持Oracle应用组合等的新特性。

## 5、Web应用测试

Apache JMeter是开源的Apache项目，实际上是一个专为负载功能测试和性能测试的Java应用。它最初的设计是用于Web应用，但现在已扩展到其他测试功能。它可以测试静态和动态资源性能，模拟一个或多个服务器负载来测试它的强度或者分析不同负载类型下的整体性能。可以查看关于Jmeter的[更多详情](#)。

SoapUI是一个免费的开源跨平台的功能测试解决方案。它提供了一个图形界面和企业级功能，允许创建和执行自动化功能、回归、符合性和负载测试。SoapUI在一个测试环境下提供了完整的测试覆盖率，而且支持所有的标准协议和技术。

## 6、质量分析

SonarQube是一个开源的平台，现已成为代码质量管理系统的全球领先者，并且因代码质量的持续检查为众人所知。除了Java外，它还支持大多数的语言，而且可以用作Android开发，它集成了大多数的持续集成工具。SonarQube会为重复代码、编码标准、单元测试、代码覆盖、复杂代码、潜在错误、注释、设计和架构而产生报告。

FindBugs也是开源的工具，在Lesser GNU Public license许可下发布。它运行在Java字节码上而不是源代码，可以识别和确定潜在错误的严重程度，如空指针引用、无限递归循环、Java库和死锁的错误使用。同时，它还可以作为Eclipse、NetBeans，IntelliJ IDEA的插件。可以参考[FindBugs Eclipse实例](#)。

PMD是一个源代码分析器。它可以检测到常见的编程错误，如未使用的变量、空的catch块、不必要的对象创建等等。它支持除了包括Java的大多数语言，还包含CPD、复制粘贴探测器。

Checkstyle的是一个开发工具，可以帮助程序员坚持按编码标准来编写Java代码，会自动启动Java代码检查。它的配置性很强，能支持几乎任何的编码标准。

## 7、版本控制

Apache Subversion是众所周知的SVN工具，是一个软件版本和版本控制系统，在一个开源许可下发布。目前，它是由一个Apache软件基金会的项目开发，也是全球性开发者和用户社区的一部分。

Git是一个免费开源的分布式版本控制系统，能够提高大小事务的速度和效率。相对于其他的配置管理工具，Git的特色在于提供廉价的局部分支、便利的暂存区和多个工作流。每个Git工作目录是一个完整的版本

库，有完整历史和完整版本的跟踪能力、独立的网络接入或中央服务器。可以查看[更多Git的文章](#)。

## 8、持续集成

Jenkins是一个屡获殊荣的应用程序，能够监视重复的工作，从建立一个软件项目到由cron运行的作业。它的基本功能是不间断建立和测试软件项目，使得开发者能更容易地将变更集成到项目中，使用户更容易获得新的构建。Jenkins还可以监控外部运行任务的执行，如cron jobs，即使是运行在远程计算机上的任务。

Apache Continuum是一个企业级持续集成服务器。提供的特性包括：自动构建、发布管理、基于角色的权限管理，以及能与其它流行构建工具和源代码管理系统相集成。

Hudson是一个用Java编写的持续集成（CI）工具，在应用服务器上运行。它支持各种SCM工具，如Subversion、Git，可以执行Apache Ant和Apache Maven项目，以及任意shell脚本和Windows批处理命令。它监视重复任务的执行，如建立一个软件项目或由cron运行的作业。

## 9、Java 应用分析

在为分析Java应用开发最创新的智能工具方面，YourKit Java分析器是技术的领头羊。你可以在CPU和内存分析上都使用YourKit解决方案，还可以以最大生产率和零开销来分析大型应用。同时，Java分析器还有几个刚提出的分析创新，为其在各大小型企业专业Java开发者获得了当之无愧的人气。

JProfiler是商业许可的Java分析工具，对于分析性能瓶颈、内存泄漏、CPU负载和解决线程问题等方面非常有效，它支持在同一台机器或远程机器上进行分析。用户可以看到活跃的总字节、实例、线程、类和垃圾回收器活动的可视化表示。它可以作为一个独立应用使用，也可以作为Eclipse软件开发环境的插件。

OneAPM For Java 可以展示业务代码的trace，实时展示缓慢的sql语句和nosql操作，还可以方便地监控JVM。可以智能探知 Java 应用之间的相互调用关系，通过串联复杂的后台组件，动态生成 J2EE 应用整体架构视图。通过拓扑图直观了解分布式或 SOA 架构应用的运行状态，准确定位系统问题。同时监控后台事务和 Web 事务。

## 10、代码覆盖工具

JaCoco是衡量和报告Java代码覆盖率的一个开源工具包。它是由Eclipse公共许可协议许可，是目前唯一能支持Java 8的字节代码覆盖工具。

Clover是一个商业的代码覆盖工具，它提供的指标可以让程序员更好地平衡编程和测试。它在IDE或你的持续集成系统下运行，还包括能加快测试速度的测试优化。

## 11、Bug跟踪

JIRA是帮助团队规划和构建优秀产品的跟踪工具。成千上万的团队选择JIRA来捕捉和组织问题、分配工作、跟进团队活动。它是付费工具，但有一个7天的免费试用期。

Trac是一个针对软件开发项目开发的完善wiki和问题的跟踪系统，以最简洁的方式对Web软件项目进行管理。它还可以帮助开发人员编写优秀的软件，提供了对Subversion和Git（或其他版本控制系统）的接口，集成了wiki和便捷报告程序。Trac允许wiki在问题描述和提交信息中标记，还创建了bug、任务、变更、文件和页面之间的链接和无缝引用。

## 12、文档生成工具

Javadoc是Oracle公司用来从Java源代码中生成HTML格式API文档的生成工具。IDE可以自动生成HTML代码。许多文件编辑器帮助用户生成Javadoc源，并将 Javadoc 信息当作程序员的内部参考。Javadoc也提供了用于创建doclets和taglets的API，让你能够进一步分析 Java 应用的结构。

Document! X是自动化的Java文档工具和完整的文档编写环境的结合体，可以用来为你的Java项目创建准确的专业文档。它会自动将生成的文档集成在Java IDEs中，同时还能给库、命名空间、类、接口、构造函数、方法、属性和字段生成文档页面。

以上就是为Java开发者提供的一系列可以提高编程效率的工具，尽情试用吧！以后，妈妈再也不用担心你的“高效开发”了！

（编译自：<http://www.javacodegeeks.com/2015/02/java-development-tools-programming-boost.html>）