Java各版本特性一览-从Java1.0到Java12

## JDK Version 1.0

1996-01-23 Oak(橡树)  
初代版本，伟大的一个里程碑，但是是纯解释运行，使用外挂JIT，性能比较差，运行速度慢。

## JDK Version 1.1

1997-02-19  
– JDBC(Java DataBase Connectivity);  
– 支持内部类;  
– RMI(Remote Method Invocation) ;  
– 反射;  
– Java Bean;

## JDK Version 1.2

1998-12-08 Playground(操场)  
– 集合框架;  
– JIT(Just In Time)编译器;  
– 对打包的Java文件进行数字签名;  
– JFC(Java Foundation Classes), 包括Swing 1.0, 拖放和Java2D类库;  
– Java插件;  
– JDBC中引入可滚动结果集,BLOB,CLOB,批量更新和用户自定义类型;  
– Applet中添加声音支持

## JDK Version 1.3

2000-05-08 Kestrel(红隼)  
– Java Sound API;  
– jar文件索引;  
– 对Java的各个方面都做了大量优化和增强;

## JDK Version 1.4

2004-02-06 Merlin(隼)  
– XML处理;  
– Java打印服务;  
– Logging API;  
– Java Web Start;  
– JDBC 3.0 API;  
– 断言;  
– Preferences API;  
– 链式异常处理;  
– 支持IPV6;  
– 支持正则表达式;  
– 引入Imgae I/O API

## JAVA 5

2004-09-30 Tiger(老虎)  
– 泛型;  
– 增强循环,可以使用迭代方式;  
– 自动装箱与自动拆箱;  
– 类型安全的枚举;  
– 可变参数;  
– 静态引入;  
– 元数据(注解);  
– Instrumentation;

## JAVA 6

2006-12-11 Mustang(野马)  
– 支持脚本语言;  
– JDBC 4.0API;  
– Java Compiler API;  
– 可插拔注解;  
– 增加对Native PKI(Public Key Infrastructure), Java GSS(Generic Security Service),Kerberos和LDAP(Lightweight Directory Access Protocol)支持;  
– 继承Web Services;

## JAVA 7

- 2011-07-28 Dolphin(海豚)

- switch语句块中允许以字符串作为分支条件;

- 在创建泛型对象时应用类型推断;

- 在一个语句块中捕获多种异常;

- 支持动态语言;

- 支持try-with-resources(在一个语句块中捕获多种异常);

- 引入Java NIO.2开发包;

- 数值类型可以用二进制字符串表示,并且可以在字符串表示中添加下划线;

- 钻石型语法(在创建泛型对象时应用类型推断);

- null值得自动处理;

## JAVA 8

2014-03-18  
– Lambda 表达式 − Lambda允许把函数作为一个方法的参数（函数作为参数传递进方法中。  
– 方法引用 − 方法引用提供了非常有用的语法，可以直接引用已有Java类或对象（实例）的方法或构造器。与lambda联合使用，方法引用可以使语言的构造更紧凑简洁，减少冗余代码。  
– 默认方法 − 默认方法就是一个在接口里面有了一个实现的方法。  
– 新工具 − 新的编译工具，如：Nashorn引擎 jjs、 类依赖分析器jdeps。  
– Stream API −新添加的Stream API（java.util.stream） 把真正的函数式编程风格引入到Java中。  
– Date Time API − 加强对日期与时间的处理。  
– Optional 类 − Optional 类已经成为 Java 8 类库的一部分，用来解决空指针异常。  
– Nashorn, JavaScript 引擎 − Java 8提供了一个新的Nashorn javascript引擎，它允许我们在JVM上运行特定的javascript应用。

## JAVA 9

2017-09-22  
1、模块系统：模块是一个包的容器，Java 9 最大的变化之一是引入了模块系统（Jigsaw 项目）。  
REPL (JShell)：交互式编程环境。  
2、HTTP 2 客户端：HTTP/2标准是HTTP协议的最新版本，新的 HTTPClient API 支持 WebSocket 和 HTTP2 流以及服务器推送特性。  
3、改进的 Javadoc：Javadoc 现在支持在 API 文档中的进行搜索。另外，Javadoc 的输出现在符合兼容 HTML5 标准。  
4、多版本兼容 JAR 包：多版本兼容 JAR 功能能让你创建仅在特定版本的 Java 环境中运行库程序时选择使用的 class 版本。  
5、集合工厂方法：List，Set 和 Map 接口中，新的静态工厂方法可以创建这些集合的不可变实例。  
6、私有接口方法：在接口中使用private私有方法。我们可以使用 private 访问修饰符在接口中编写私有方法。  
7、进程 API: 改进的 API 来控制和管理操作系统进程。引进 java.lang.ProcessHandle 及其嵌套接口 Info 来让开发者逃离时常因为要获取一个本地进程的 PID 而不得不使用本地代码的窘境。  
改进的 Stream API：改进的 Stream API 添加了一些便利的方法，使流处理更容易，并使用收集器编写复杂的查询。  
8、改进 try-with-resources：如果你已经有一个资源是 final 或等效于 final 变量,您可以在 try-with-resources 语句中使用该变量，而无需在 try-with-resources 语句中声明一个新变量。  
9、改进的弃用注解 @Deprecated：注解 @Deprecated 可以标记 Java API 状态，可以表示被标记的 API 将会被移除，或者已经破坏。  
10、改进钻石操作符(Diamond Operator) ：匿名类可以使用钻石操作符(Diamond Operator)。  
改进 Optional 类：java.util.Optional 添加了很多新的有用方法，Optional 可以直接转为 stream。  
11、多分辨率图像 API：定义多分辨率图像API，开发者可以很容易的操作和展示不同分辨率的图像了。  
12、改进的 CompletableFuture API ： CompletableFuture 类的异步机制可以在 ProcessHandle.onExit 方法退出时执行操作。  
13、轻量级的 JSON API：内置了一个轻量级的JSON API  
14、响应式流（Reactive Streams) API: Java 9中引入了新的响应式流 API 来支持 Java 9 中的响应式编程。

## JAVA 10

2018-03-21  
根据官网的公开资料，共有12个重要特性，如下：  
1、JEP286，var 局部变量类型推断。  
2、JEP296，将原来用 Mercurial 管理的众多 JDK 仓库代码，合并到一个仓库中，简化开发和管理过程。  
3、JEP304，统一的垃圾回收接口。  
4、JEP307，G1 垃圾回收器的并行完整垃圾回收，实现并行性来改善最坏情况下的延迟。  
5、JEP310，应用程序类数据 (AppCDS) 共享，通过跨进程共享通用类元数据来减少内存占用空间，和减少启动时间。  
6、JEP312，ThreadLocal 握手交互。在不进入到全局 JVM 安全点 (Safepoint) 的情况下，对线程执行回调。优化可以只停止单个线程，而不是停全部线程或一个都不停。  
7、JEP313，移除 JDK 中附带的 javah 工具。可以使用 javac -h 代替。  
8、JEP314，使用附加的 Unicode 语言标记扩展。  
9、JEP317，能将堆内存占用分配给用户指定的备用内存设备。  
10、JEP317，使用 Graal 基于 Java 的编译器，可以预先把 Java 代码编译成本地代码来提升效能。  
11、JEP318，在 OpenJDK 中提供一组默认的根证书颁发机构证书。开源目前 Oracle 提供的的 Java SE 的根证书，这样 OpenJDK 对开发人员使用起来更方便。  
12、JEP322，基于时间定义的发布版本，即上述提到的发布周期。版本号为\$FEATURE.\$INTERIM.\$UPDATE.\$PATCH，分别是大版本，中间版本，升级包和补丁版本。

## JAVA 11

2018-09-25  
翻译后的新特性有：  
1、181:Nest-Based访问控制  
2、309:动态类文件常量  
3、315:改善Aarch64 intrinsic  
4、318:无操作垃圾收集器  
5、320:消除Java EE和CORBA模块  
6、321:HTTP客户端(标准)  
7、323:局部变量的语法λ参数  
8、324:Curve25519和Curve448关键协议  
9、327:Unicode 10  
10、328:飞行记录器  
11、329:ChaCha20和Poly1305加密算法  
12、330:发射一列纵队源代码程序  
13、331:低开销堆分析  
14、332:传输层安全性(Transport Layer Security,TLS)1.3  
15、333:动作:一个可伸缩的低延迟垃圾收集器 (实验)  
16、335:反对Nashorn JavaScript引擎  
17、336:反对Pack200工具和API

## JAVA 12

2019-03-19  
作为“功能性版本”，JDK 12 总共包含 8 个新的 JEP ，分别为：  
1、189: Shenandoah: A Low-Pause-Time Garbage Collector (Experimental) ：新增一个名为 Shenandoah 的垃圾回收器，它通过在 Java 线程运行的同时进行疏散 (evacuation) 工作来减少停顿时间。  
2、230: Microbenchmark Suite：新增一套微基准测试，使开发者能够基于现有的 Java Microbenchmark Harness（JMH）轻松测试 JDK 的性能，并创建新的基准测试。  
3、325: Switch Expressions (Preview) ：对 switch 语句进行扩展，使其可以用作语句或表达式，简化日常代码。  
4、334: JVM Constants API ：引入一个 API 来对关键类文件 (key class-file) 和运行时工件的名义描述（nominal descriptions）进行建模，特别是那些可从常量池加载的常量。  
5、340: One AArch64 Port, Not Two ：删除与 arm64 端口相关的所有源码，保留 32 位 ARM 移植和 64 位 aarch64 移植。  
6、341: Default CDS Archives ：默认生成类数据共享（CDS）存档。  
7、344: Abortable Mixed Collections for G1 ：当 G1 垃圾回收器的回收超过暂停目标，则能中止垃圾回收过程。  
8、346: Promptly Return Unused Committed Memory from G1 ：改进 G1 垃圾回收器，以便在空闲时自动将 Java 堆内存返回给操作系统。

原文地址：http://openjdk.java.net/projects/jdk/12/

**更多干货笔记关注微信公众号 : 老九学堂**