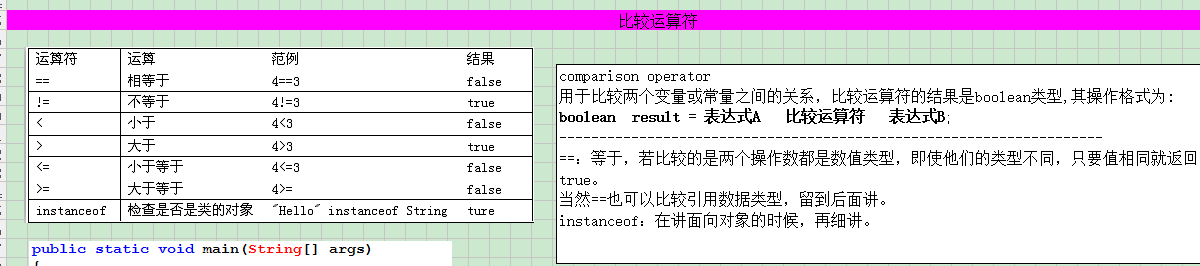
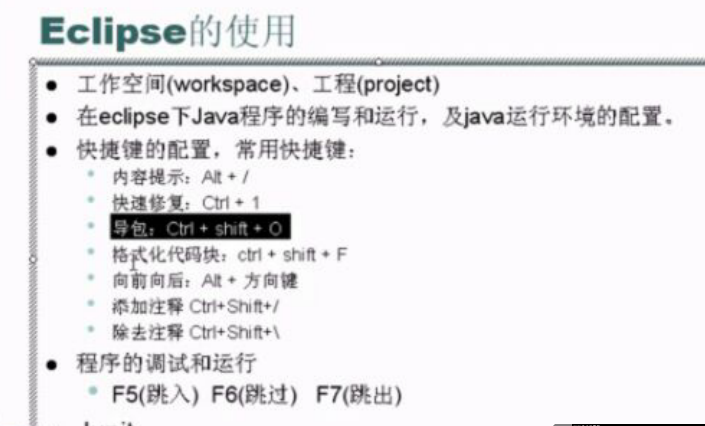
InstanceOf 检查是否是类的对象

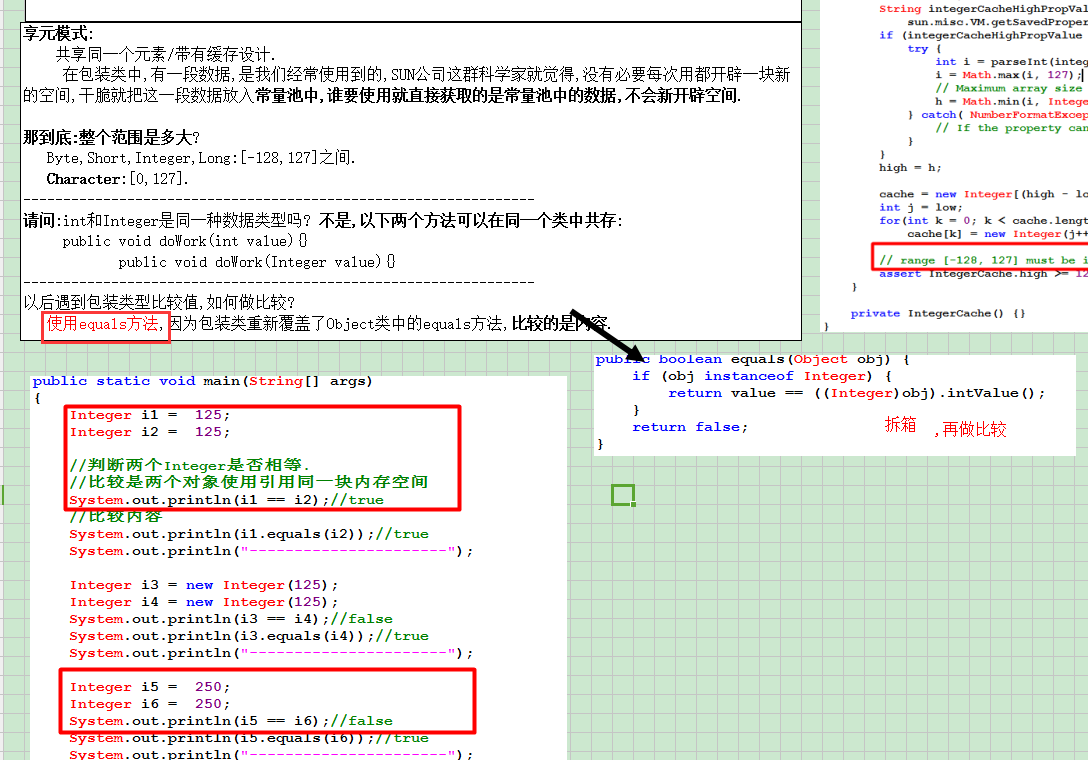


Eclipse常用快捷键

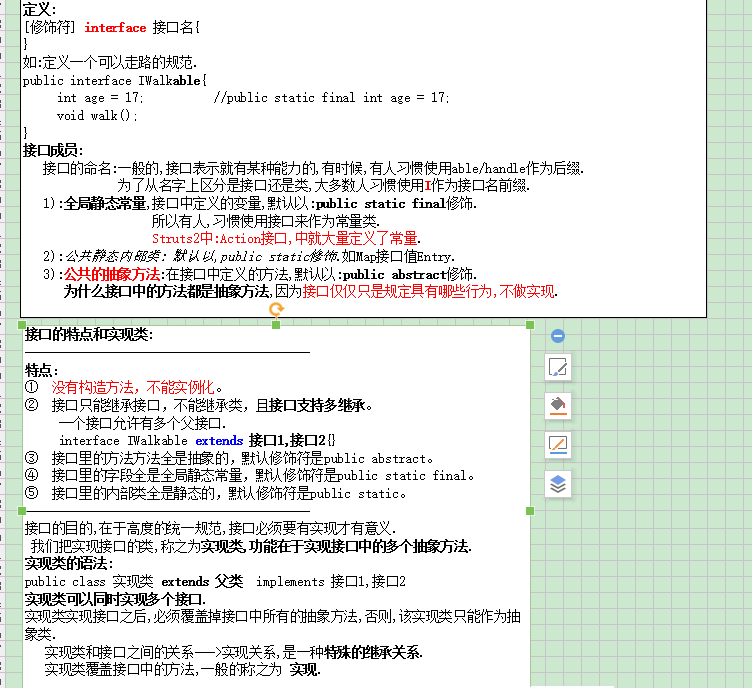


Crtl + o 查看文件内的方法

Equals 比较内容



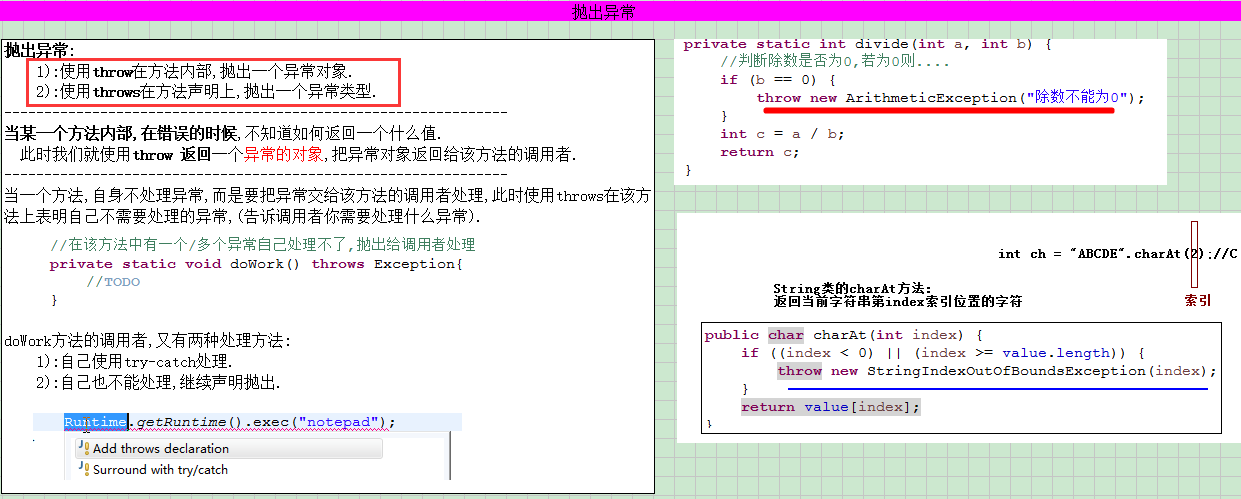
Interface



Throw 与 throws 抛出异常:

1):使用throw在方法内部,抛出一个异常对象.

2):使用throws在方法声明上,抛出一个异常类型.



并发与并行的概念

并发和并行是即相似又有区别：(微观)

并行：指两个或多个事件在同一时刻发生； 强调的是时间点.

并发：指两个或多个事件在同一时间段内发生； 强调的是时间段.

-----------------------------------------------------------------------------------

在操作系统中，在多道程序环境下，并发性是指在一段时间内有多个程序在同时运行，但在单CPU系统中，每一时刻却仅能有一道程序执行（时间片），故微观上这些程序只能是分时地交替执行。倘若计算机系统中有多个CPU，则这些可以并发执行的程序便可被分配到多个处理机上，实现并行执行，即利用每个处理机来处理一个可并发执行的程序，这样，多个程序便可以同时执行，因为是微观的，所以大家在使用电脑的时候感觉就是多个程序是同时执行的。

所以，大家买电脑的时候喜欢买“核”多的，其原因就是“多核处理器”电脑可以同时并行地处理多个程序，从而提高了电脑的运行效率。

单核处理器的计算机肯定是不能并行的处理多个任务的，只能是多个任务在单个CPU上并发运行,CPU调度。

同理线程也是一样的，从宏观角度上理解线程是并行运行的，但是从微观角度上分析却是串行运行的，即一个线程一个线程的去运行，当系统只有一个CPU时，线程会以某种顺序执行多个线程，我们把这种情况称之为线程调度。

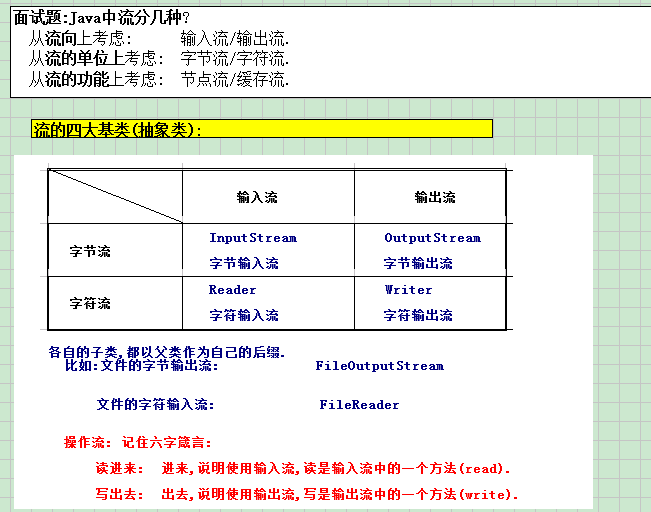
面试题:进程与线程的区别：

进程：有独立的内存空间，进程中的数据存放空间（堆空间和栈空间）是独立的，至少有一个线程。

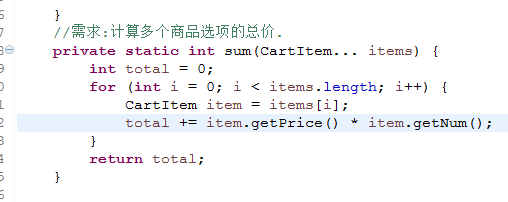
线程：堆空间是共享的，栈空间是独立的，线程消耗的资源也比进程小，相互之间可以影响的，又称为轻型进程或进程元。

因为一个进程中的多个线程是并发运行的，那么从微观角度上考虑也是有先后顺序的，那么哪个线程执行完全取决于CPU调度器，程序员是控制不了的。我们可以把多线程并发性看作是多个线程在瞬间抢CPU资源，谁抢到资源谁就运行，这也造就了多线程的随机性。

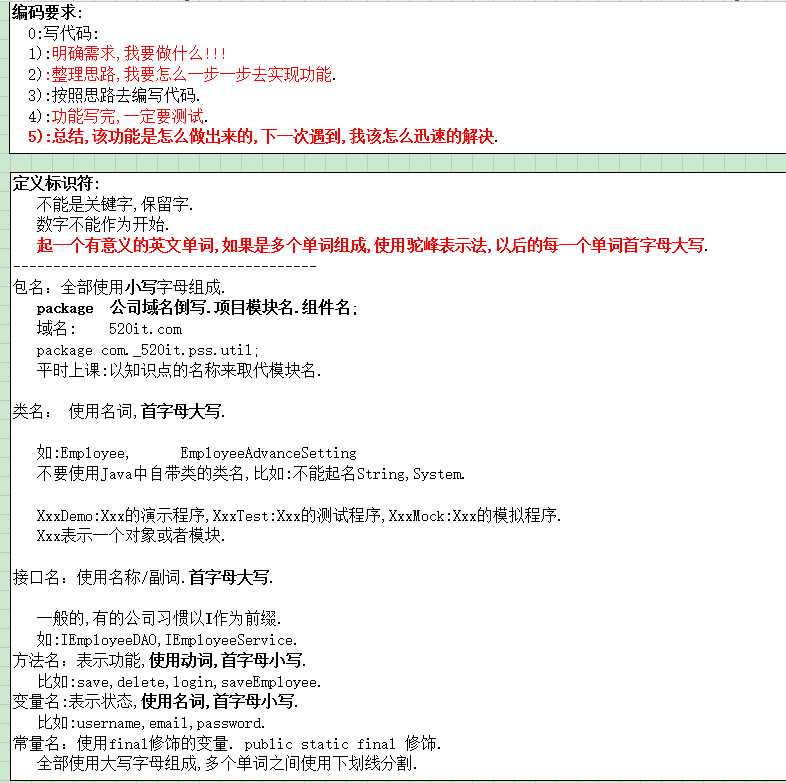
I/O 操作



可变参数



编码规范



Java 常用框架

