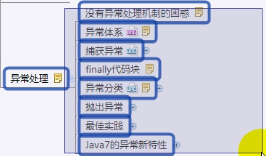
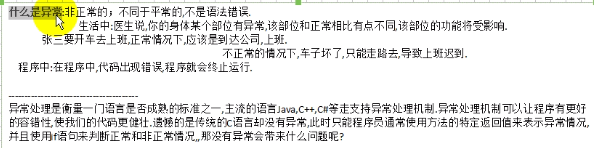
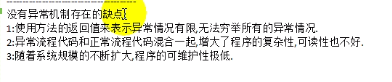
Day13

**01没有异常处理机制的困惑**

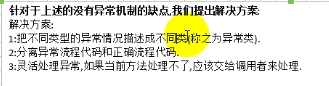


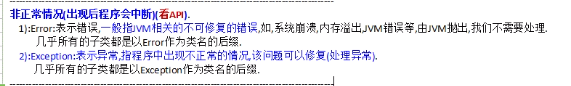


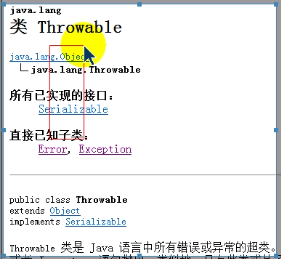


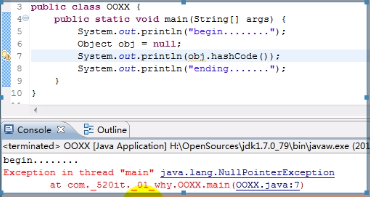


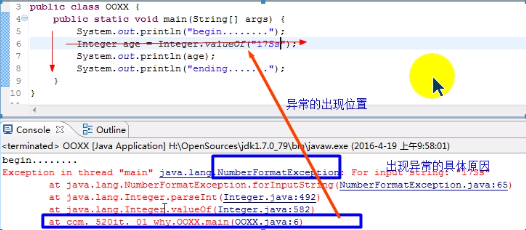
**02异常体系-error和exception**





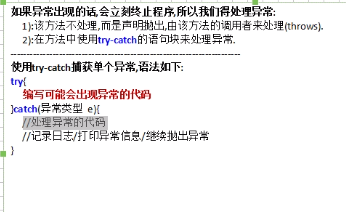




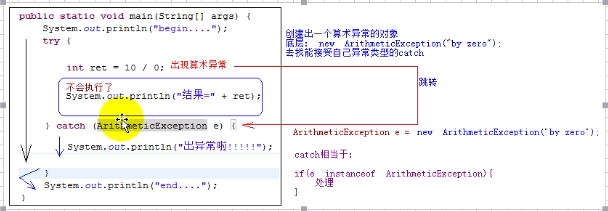


**03try-catch捕获异常**



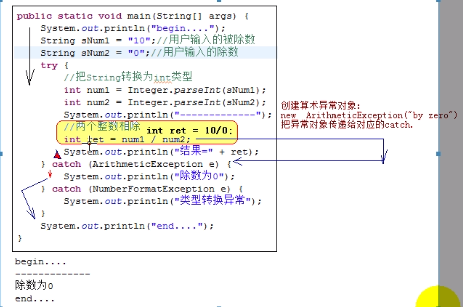


Try-catch都不能单独使用

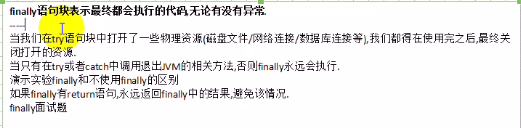


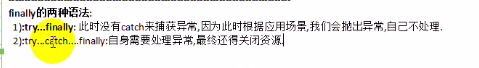
**05使用try-catch捕获多个异常**

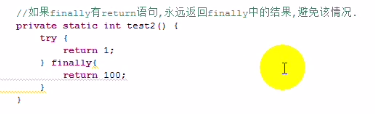
一个try-catch会影响性能，能放在一起则放在一起



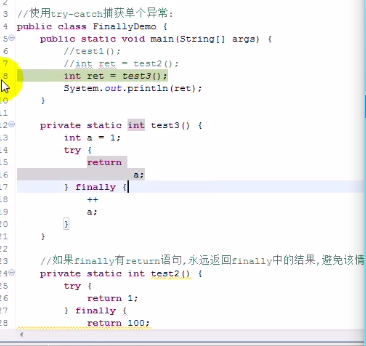
**06保证关闭资源的finally代码块**



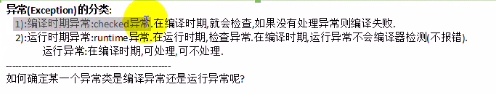


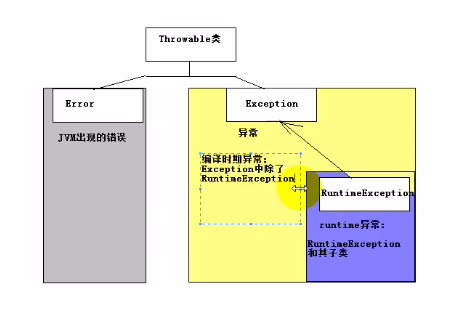


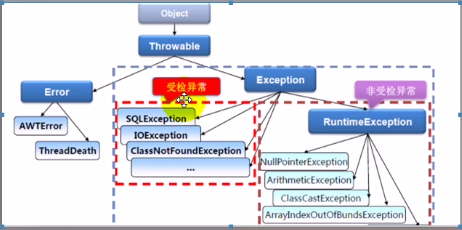
Finally先与return执行



**07编译异常与运行异常**

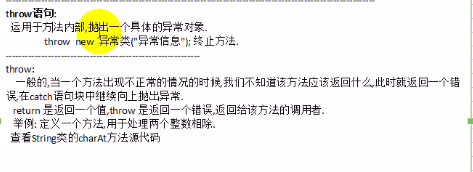






**08返回错误结果-throw语句**

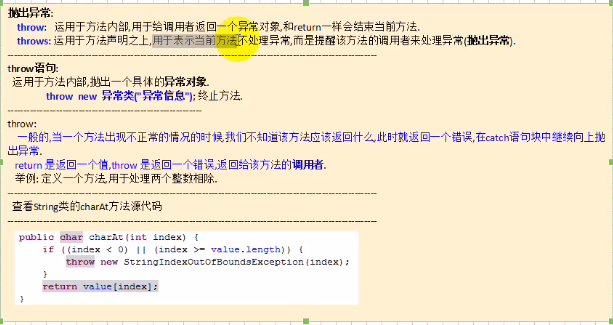
Throw语句



Throw返回一个错误给调用者

**09throws语句**

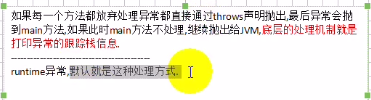
Throws语句



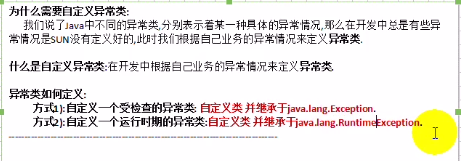
这类异常一共有两种构造方法，一种是无参构造器，一种是参数为字符串的构造器。

Throws用在方法的声明处，告诉方法的调用者需要处理该异常，由调用者在调用之前处理

表示本方法中不处理某种类型的异常，提醒调用者处理该类异常



**10自定义异常类**

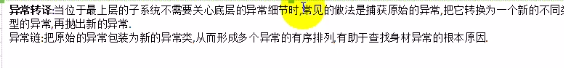


受检查的异常类就是机会使得编译器检查。

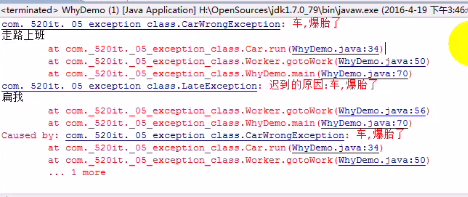
如自定义逻辑异常：logicException

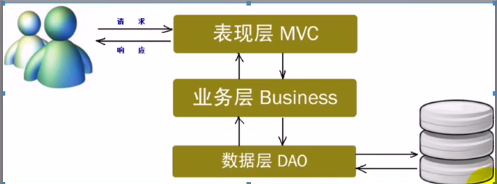
**11异常机制的优势**

**12异常转译和异常链**

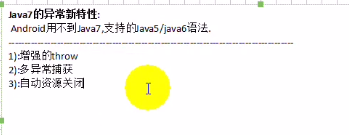


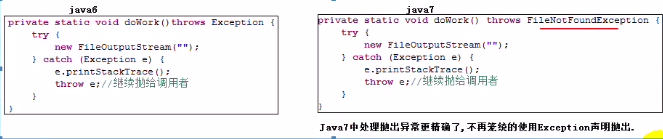


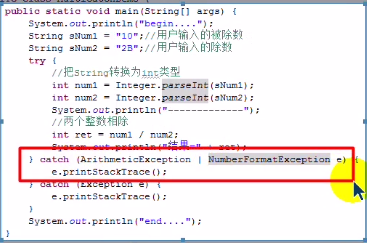


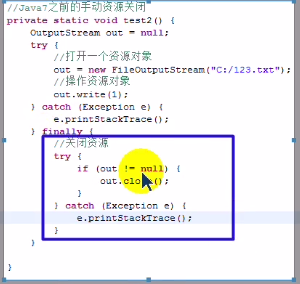


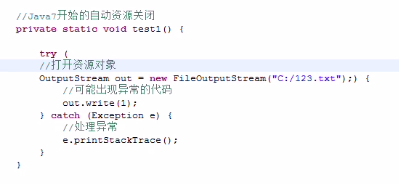
**13java7中的异常的新特性**



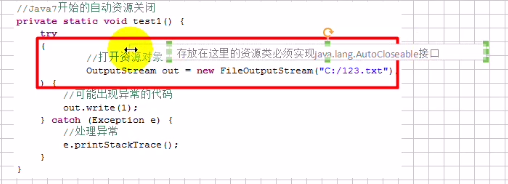




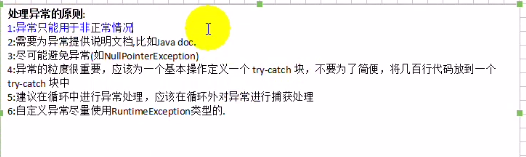




自动关闭的资源必须实现autocloseabe接口



**14处理异常的原则**



Try-catch的存在会影响性能；

需要为异常提供说明文档，记录在文档注释中

**15今日小结**

