复习：

1. 正则表达式：Pattern---用于指定规则的，利用这个规则进行判断、筛选等操作

2. 包装类：针对每种基本类型都提供了对应的包装类。---自动封箱/拆箱---字面量的哈希码是不随着环境而改变

## 数学类

Math---最终类，构造方法私有，这个类中提供了大量的操作数据的静态方法---完全的工具类---针对基本类型来提供初等数学运算

BigDecimal---用于精确运算的类，要求参数以字符串形式传入，底层是做的逐位相减

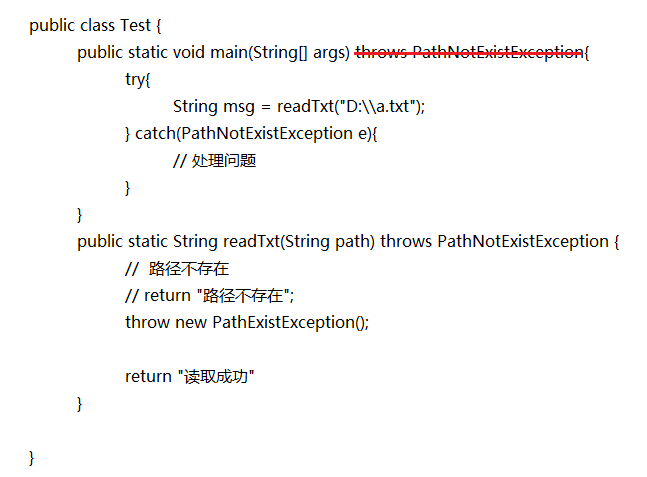
BigInteger---能用存储任意大小的整数，底层实际上是一个字节数组形式存储这个整数的补码

## 日期类

Date---重点掌握日期格式转换---SimpleDateFormat

Calendar---了解即可

## 异常



在Java中，异常实际上用于进行问题的反馈和处理的

### Throwable

Error---不需要处理

Exception---可以处理

编译时异常(checked exception)---编译的时候就已经出现，要求必须处理。

CloneNotSupportedException

ParseException

运行时异常(unchecked exception)---编译期间不报错，而是运行的时候出现。可以处理，也可以不处理 --- RuntimeException

ArithmeticException

IndexOutOfBoundsException

NullPointerException

ClassCastException

NumberFormatException

写一个异常类然后继承对应的异常---往往会提供两个构造方法

### 异常的捕获方式

1. 每一个异常对应一个catch，适合于每一个异常的处理方式都不一样

2. 捕获一个父类异常，适合于异常的统一处理

3. 将同一组的异常之间用|隔开，适合于分组处理 --- 从JDK1.7开始的

如果代码没有出现异常，那么此时有无try对代码的影响都不大；如果出现了异常，这个时候会影响效率

总结：重载和重写的异同点

重载是指在同一类中存在了方法名一致儿而参数列表不同的方法。重载只和方法签名有关，和返回值类型、修饰符以及异常无关。重载本身是一种编译时多态。

重写是指在父子类中产生了方法签名完全一致的非静态方法。在方法重写的时候，需要保证子类方法的权限范围要大于等于父类方法的权限范围。同时，子类方法的返回值类型要根据父类方法的返回值类型来确定。如果父类方法的返回值类型是基本类型或者是void，那么子类方法的返回值类型要相同。如果父类方法的返回值类型是引用类型，那么子类方法的返回值类型要和父类方法的返回值类型之间构成继承关系。最后，需要保证子类方法抛出的编译时异常与父类方法抛出的编译时异常有继承关系。方法的重写本身是一种运行时多态。

而无论是方法的重载还是方法的重写，都是方法的不同的体现形式，都是一种行为多态。

### finally代码块

无论try和catch谁执行，它都会在最后执行一次

如果项目开发阶段，出现异常，需要打印栈轨迹，进行修改

如果项目上线阶段，记录错误日志，跳转错误页面

