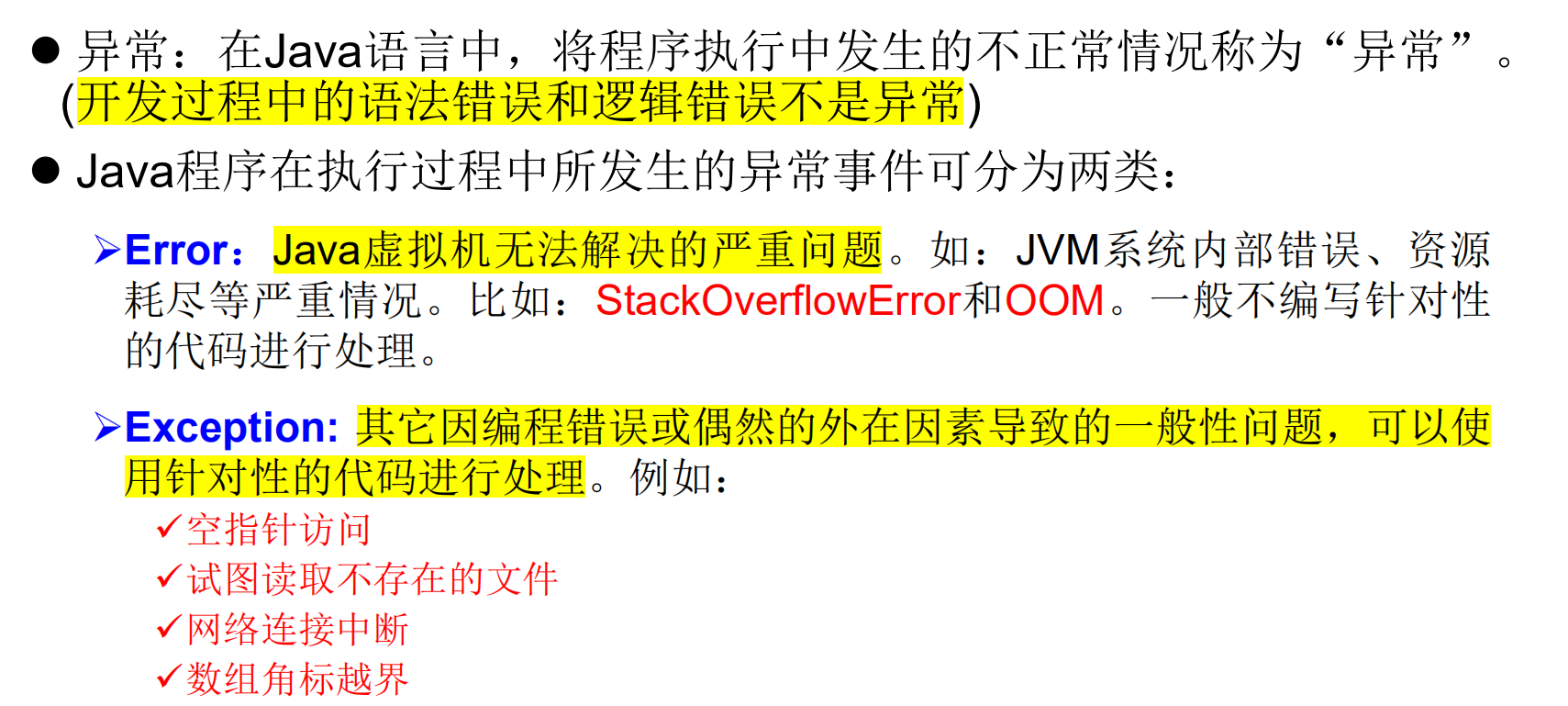
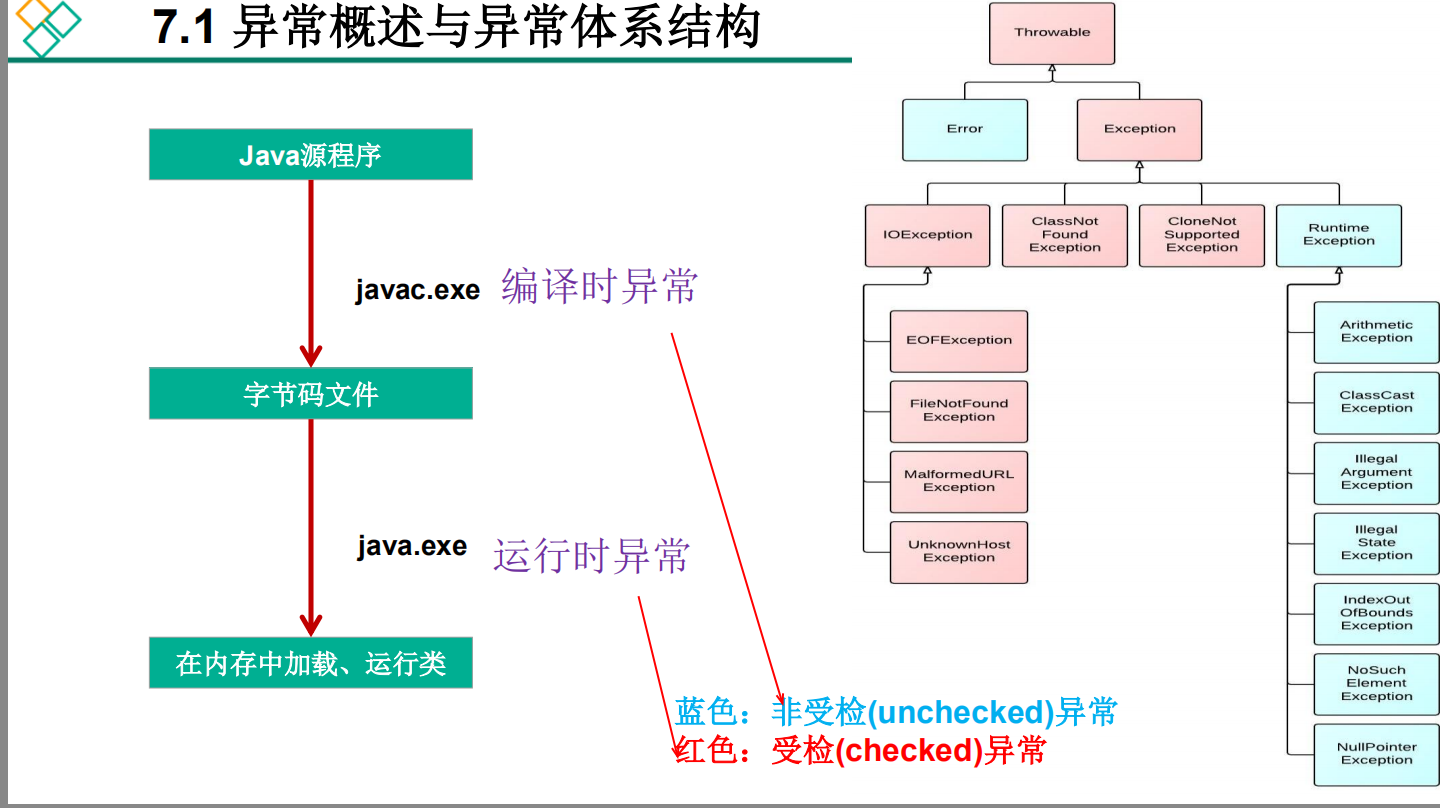
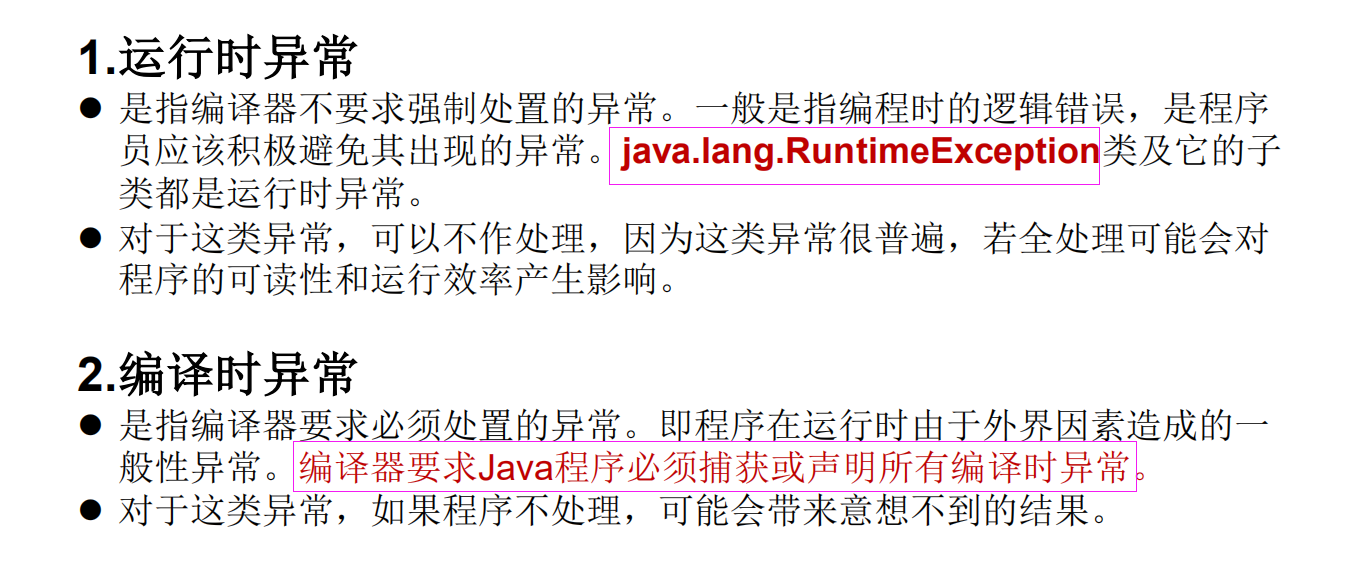
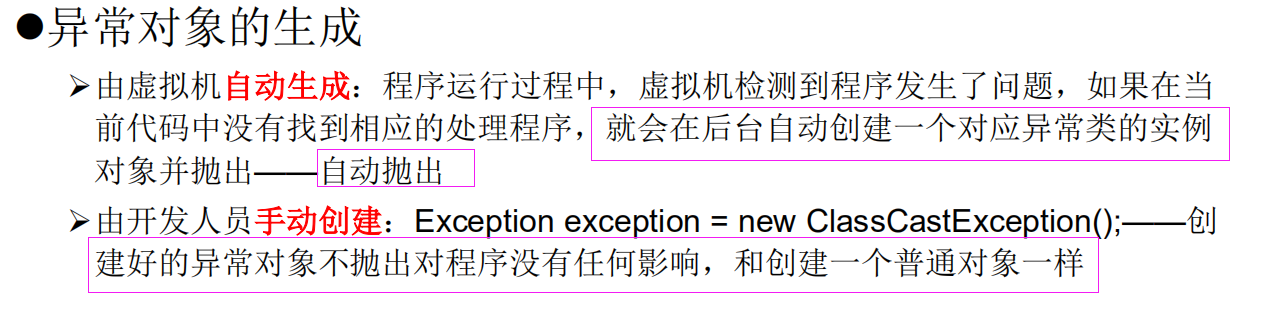


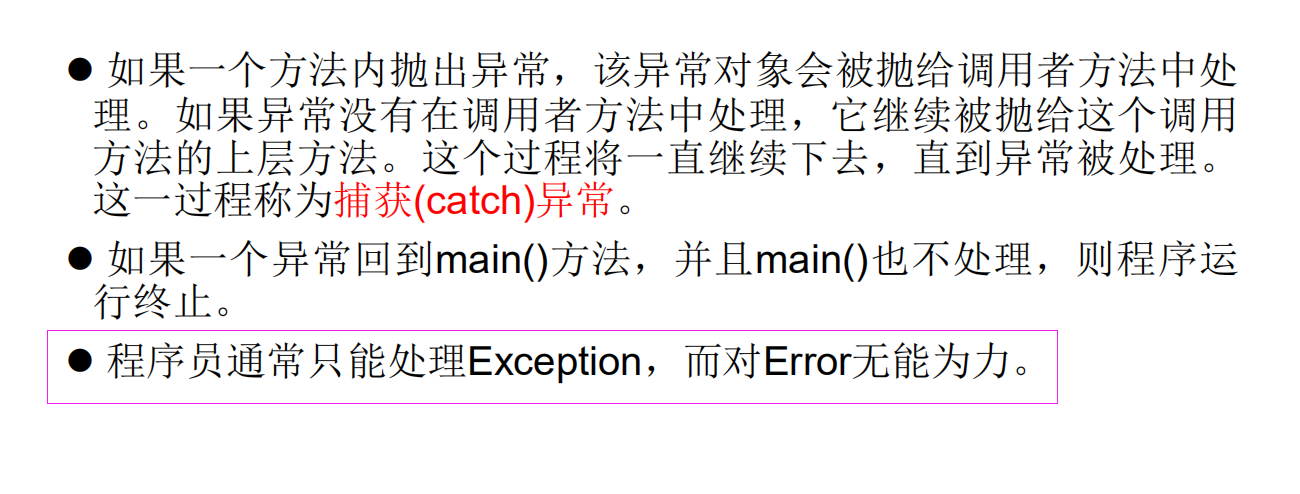
# 抓

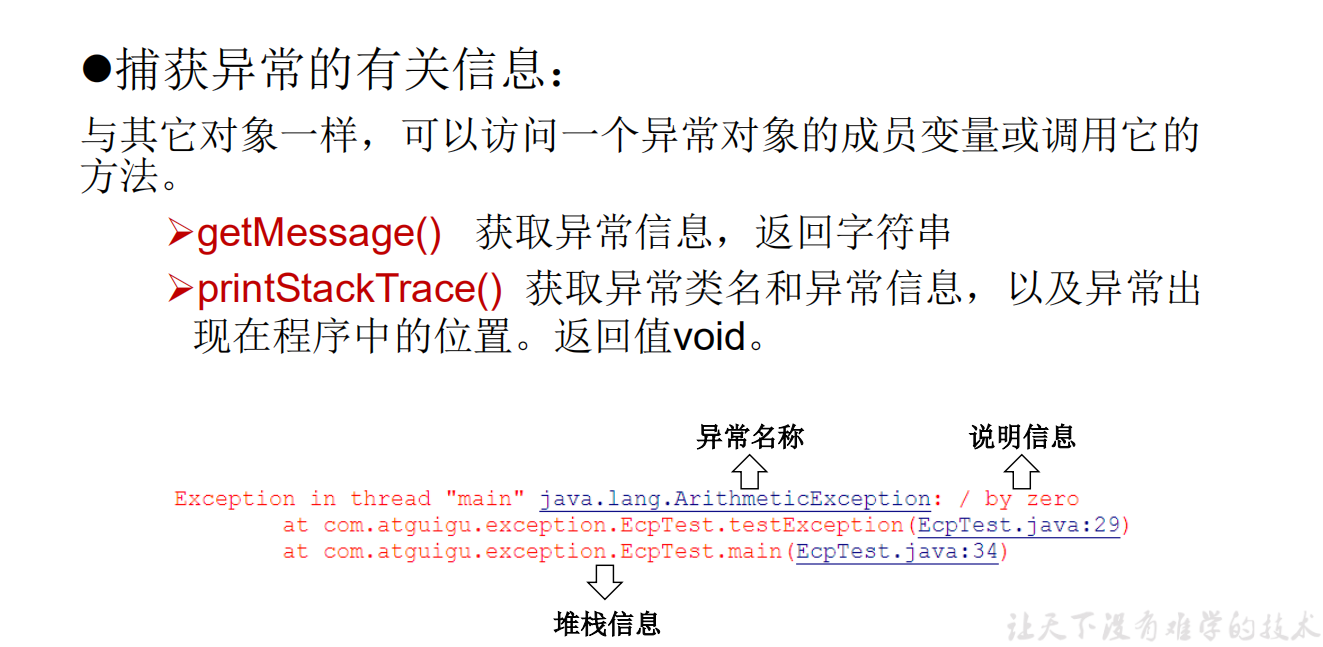


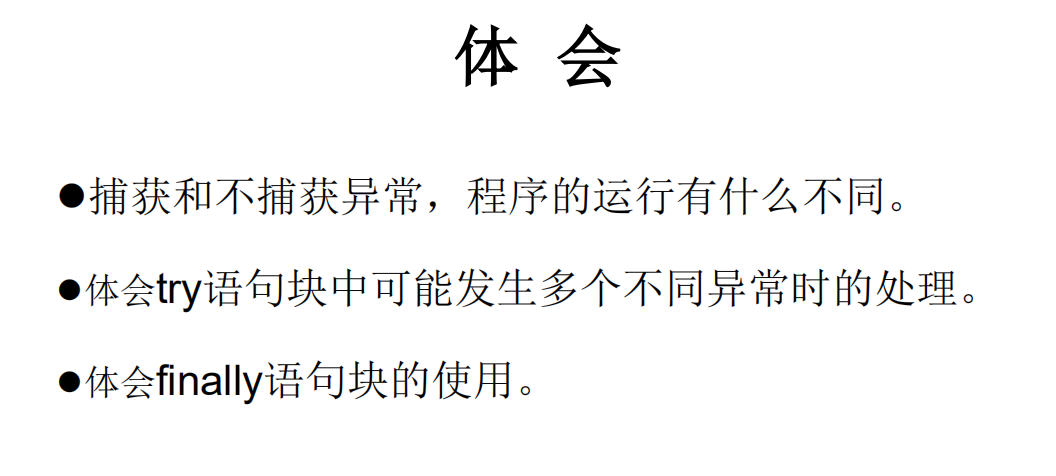


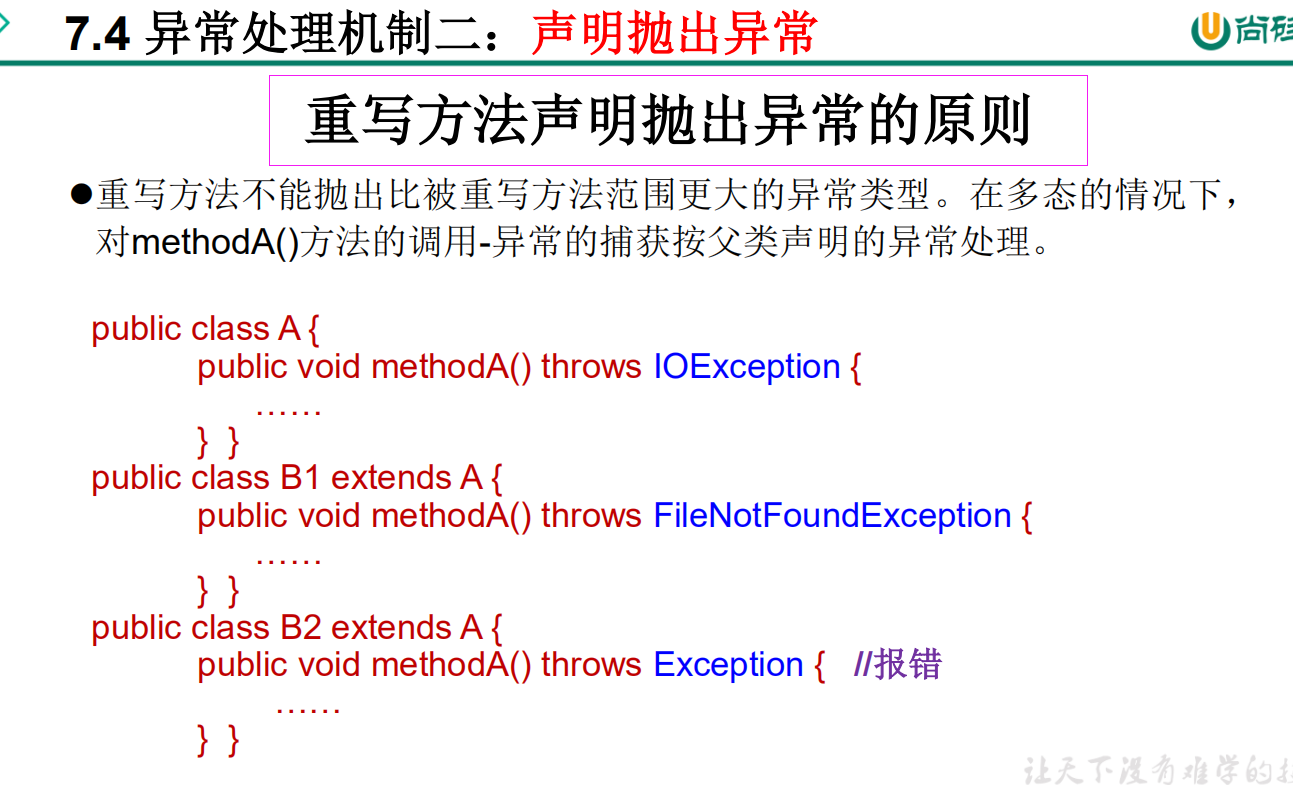








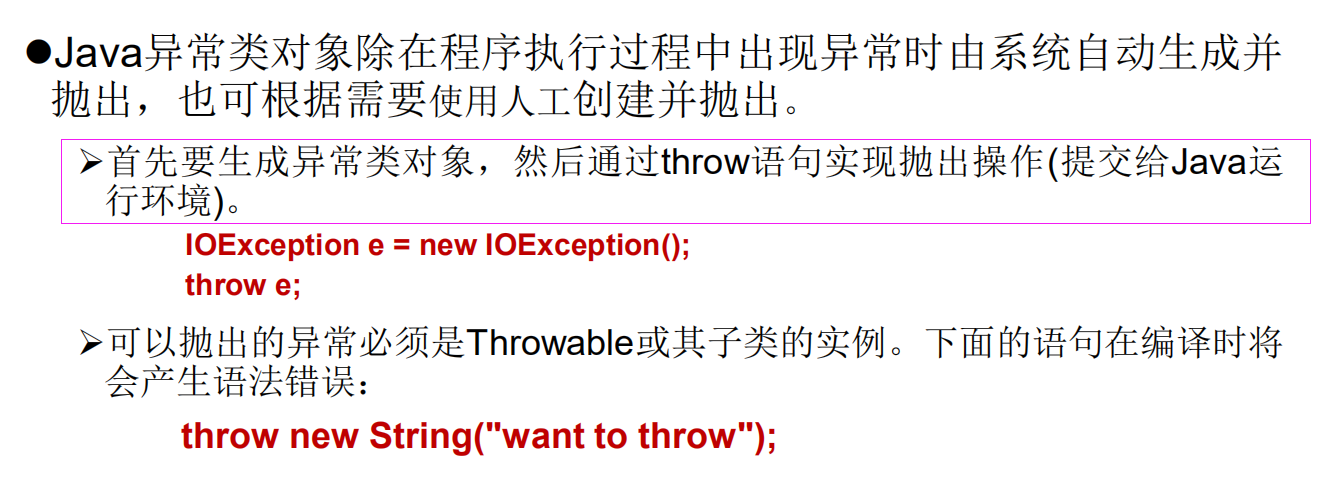




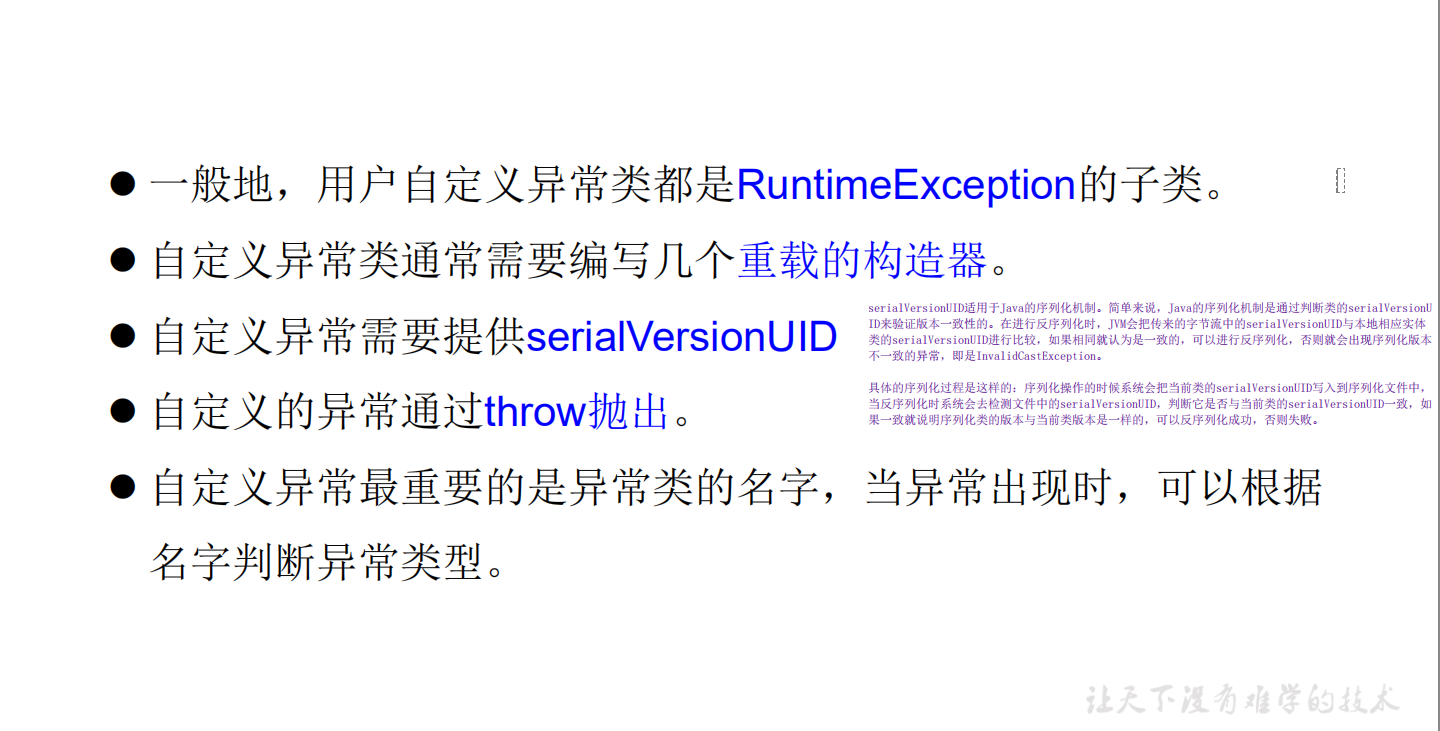


# 抛

抛异常的时候，抛的是Exception的子类，如果抛的是不受检查的运行时异常就不用处理，如果抛的是受减产的异常，还必须处理。



# 用户自定义异常



serialVersionUID适用于Java的序列化机制。简单来说，Java的序列化机制是通过判断类的serialVersionUID来验证版本一致性的。在进行反序列化时，JVM会把传来的字节流中的serialVersionUID与本地相应实体类的serialVersionUID进行比较，如果相同就认为是一致的，可以进行反序列化，否则就会出现序列化版本不一致的异常，即是InvalidCastException。

**具体的序列化过程是这样的**：序列化操作的时候系统会把当前类的serialVersionUID写入到序列化文件中，当反序列化时系统会去检测文件中的serialVersionUID，判断它是否与当前类的serialVersionUID一致，如果一致就说明序列化类的版本与当前类版本是一样的，可以反序列化成功，否则失败。

如果我们不希望通过编译来强制划分软件版本，即实现序列化接口的实体能够兼容先前版本，就需要显式地定义一个名为serialVersionUID，类型为long的变量，不修改这个变量值的序列化实体都可以相互进行串行化和反串行化。

# 总结

