# 反射机制性能问题

## 反射机制会导致程序的性能比较慢

### setAccessible

* 启用和禁用访问安全检查的开关，值为true则指示反射的对象在使用时应该取消Java语言的访问检查。值为false则指示反射的对象应该实施Java语言访问检查。并不是为true就能访问为false就不能访问
* 禁止安全检查，可以提高反射的运行速度

## 反射操作泛型

* Java采用泛型擦除的机制来引入泛型。Java中的泛型仅仅是给编译器javac使用的，确保数据的安全性和免去强制类型转换的麻烦。但是，一旦编译完成，凡有的和泛型有关的类型全部擦除。
* 为了通过反射操作这些类型以迎合时机开发的需要，Java就新增了ParameterizedType，GenericArrayType，TypeVariable和WildcardType几种类型来代表不能被归一到Class类中的类型但是又和原始类型齐名的类型
* ParameterizedType：表示一种参数化的类型，比如Collection<String>
* GenericArrayType: 表示一种元素类型是参数化类型或者类型变量的数组类型
* TypeVariable：是各种类型变量的公共接口
* WildcardType：代表一种通配符类型的表达式，不如？，？extends Number，？super Integer [wildcard是一个单词 就是“通配符“]

## 反射操作注解

* 可以通过反射APi：getAnnotations,getAnnotation获得相关的注解信息