

# 反射的基本的概念

使用反射机制可以动态获取当前 class 的信息 比如方法的信息、注解信息、方法的参数、属性等；

## 反射技术相关 API

```
class<?> aClass = Class.forName("com.mayikt.entity.UserEntity");
```

## 反射技术应用场景

1. JDBC 加载驱动连接 class.forName
2. Spring 容器框架 IOC 实例化对象
3. 自定义注解生效（反射+Aop）
4. 第三方核心的框架

## 自定义注解如何生效

Aop+反射技术

通过 aop 拦截目标方法，根据反射技术判断方法上是否有加上注解，如果有加注解，则执行封装业务逻辑操作。

## Java8 有哪些新特性

- 1.Lambda 表达式 - Lambda 允许把函数作为一个方法的参数（函数作为参数传递到方法中）。
- 2.方法引用 - 方法引用提供了非常有用的语法，可以直接引用已有 Java 类或对象（实例）的方法或构造器。与 lambda 联合使用，方法引用可以使语言的构造更紧凑简洁，减少冗余代码。
- 3.默认方法 - 默认方法就是一个在接口里面有了一个实现的方法。
- 新工具 - 新的编译工具，如：Nashorn 引擎 jjs、类依赖分析器 jdeps。
- 4.Stream API - 新添加的 Stream API（java.util.stream）把真正的函数式编程风格引入到 Java 中。
- 5.Date Time API - 加强对日期与时间的处理。

6.Optional 类 - Optional 类已经成为 Java 8 类库的一部分，用来解决空指针异常。

7.Nashorn, JavaScript 引擎 - Java 8 提供了一个新的 Nashorn javascript 引擎，它允许我们在 JVM 上运行特定的 javascript 应用。