注解

注解很厉害，它可以增强我们的java代码，同时利用反射技术可以扩充实现很多功能

它们被广泛用于三大框架底层 ，传统我们通过xml文本文件声明方式而现在

最主流的开发都是基于注解方式，代码量很少，框架可以根据注解去自动生成很多代码

从而减少代码量，程序更容易读，例如最火爆 的SpringBoot就完全基于了注解技术的实现

注解设计非常精巧，初学的时候觉得很另类甚至多余，甚至垃圾有了java代码还干嘛还要

有@注解？但熟练之后你就会感叹，它竟然可以超越java代码的功能，让java代码瞬间变得强大，大家慢慢体会吧

注解的分类

JKD自带的注解

@override

@Deprecated标记就表明这个方法已经过时了，但我就是要用，别提示我过期了

@SupperssWarnings(‘deprecation’)忽略警告

@SafeVaragers jdk1.7出现堆污染，不常用

@FunctionallInterface jdk1.8出现 配合函数式变成拉姆达表达式 ，不常用

元注解

描述注解的注解就5个

@Target注解用在哪里，类上，方法上，属性上

@Retention 注解的生命周期：源文件中 、class文件中、运行中

@Inherited允许子注解继承

@Documented生成javadoc的时候会包含注解，不常用

@Repeatable注解可重复类型注解 可以在同一个地方使用多次不常用



图形用户界面, 文本, 应用程序

描述已自动生成

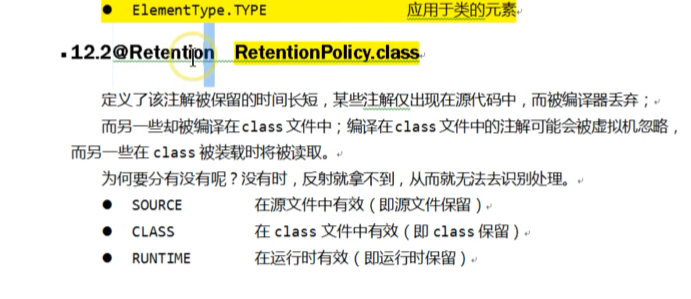
Type:可以出现在类上接口上

FIELD: 成员变量上

Method:方法上

Parameter:参数上

Constructor出现在构造方法上



图形用户界面, 文本, 应用程序

描述已自动生成

自定义注解

定义注解

注解的语法写法和常规的java语法不同