## java.lang 类 Class<T>

[java.lang.Object](mk:@MSITStore:C:\\Users\\Administrator\\Desktop\\[Java参考文档].JDK_API_1_6_zh_CN.CHM::/java/lang/../../java/lang/Object.html" \o "java.lang 中的类)

IMG_256**java.lang.Class<T>**

**类型参数：**

T - 由此 Class 对象建模的类的类型。例如，String.class 的类型是 Class<String>。如果将被建模的类未知，则使用 Class<?>。

**所有已实现的接口：**

[Serializable](mk:@MSITStore:C:\\Users\\Administrator\\Desktop\\[Java参考文档].JDK_API_1_6_zh_CN.CHM::/java/lang/../../java/io/Serializable.html" \o "java.io 中的接口), [AnnotatedElement](mk:@MSITStore:C:\\Users\\Administrator\\Desktop\\[Java参考文档].JDK_API_1_6_zh_CN.CHM::/java/lang/../../java/lang/reflect/AnnotatedElement.html" \o "java.lang.reflect 中的接口), [GenericDeclaration](mk:@MSITStore:C:\\Users\\Administrator\\Desktop\\[Java参考文档].JDK_API_1_6_zh_CN.CHM::/java/lang/../../java/lang/reflect/GenericDeclaration.html" \o "java.lang.reflect 中的接口), [Type](mk:@MSITStore:C:\\Users\\Administrator\\Desktop\\[Java参考文档].JDK_API_1_6_zh_CN.CHM::/java/lang/../../java/lang/reflect/Type.html" \o "java.lang.reflect 中的接口)

public final class **Class<T>**

extends [Object](mk:@MSITStore:C:\\Users\\Administrator\\Desktop\\[Java参考文档].JDK_API_1_6_zh_CN.CHM::/java/lang/../../java/lang/Object.html" \o "java.lang 中的类)

implements [Serializable](mk:@MSITStore:C:\\Users\\Administrator\\Desktop\\[Java参考文档].JDK_API_1_6_zh_CN.CHM::/java/lang/../../java/io/Serializable.html" \o "java.io 中的接口), [GenericDeclaration](mk:@MSITStore:C:\\Users\\Administrator\\Desktop\\[Java参考文档].JDK_API_1_6_zh_CN.CHM::/java/lang/../../java/lang/reflect/GenericDeclaration.html" \o "java.lang.reflect 中的接口), [Type](mk:@MSITStore:C:\\Users\\Administrator\\Desktop\\[Java参考文档].JDK_API_1_6_zh_CN.CHM::/java/lang/../../java/lang/reflect/Type.html" \o "java.lang.reflect 中的接口), [AnnotatedElement](mk:@MSITStore:C:\\Users\\Administrator\\Desktop\\[Java参考文档].JDK_API_1_6_zh_CN.CHM::/java/lang/../../java/lang/reflect/AnnotatedElement.html" \o "java.lang.reflect 中的接口)

Class 类的实例表示正在运行的 Java 应用程序中的类和接口。枚举是一种类，注释是一种接口。每个数组属于被映射为 Class 对象的一个类，所有具有相同元素类型和维数的数组都共享该 Class 对象。基本的 Java 类型（boolean、byte、char、short、int、long、float 和 double）和关键字 void 也表示为 Class 对象。

Class 没有公共构造方法。Class 对象是在加载类时由 Java 虚拟机以及通过调用类加载器中的 defineClass 方法自动构造的。

以下示例使用 Class 对象来显示对象的类名：

void printClassName(Object obj) {

System.out.println("The class of " + obj +

" is " + obj.getClass().getName());

}

还可以使用一个类字面值（JLS Section [15.8.2](http://java.sun.com/docs/books/jls/second_edition/html/expressions.doc.html" \l "251530)）来获取指定类型（或 void）的 Class 对象。例如：

System.out.println("The name of class Foo is: "+Foo.class.getName());