

四、class类文件结构

魔数magic number

- 前四个字节为魔数，表示coffee baby。用于表示该文件是否为一个能够被JVM接受的class文件；

版本号

- 共四个字节，前两个字节为此版本号，后两个字节为主版本号。《JAVA虚拟机规范》明确规定JVM拒绝执行超过其版本号的class文件；

常量池

- 用于存放编译期间产生的字面量和符号引用；

字面量：

- 接近java语言层面的概念，像文本字符串、使用final修饰的常量值都属于字面量；

符号引用：

- 接近编译原理的概念，java中的符号引用包括：

- 1.类或接口的全限定名；
- 2.字段的简单名称和描述符；
- 3.方法的简单名称和描述符；
- 4.方法句柄和方法类型；
- 5.动态常量和调用点限定符；
- 6.JDK9中涉及的被模块导出或开放的包；

常量池的入口是一个U2类型的常量容量计数器，用于描述常量池中的常量个数。常量池中的每一项常量都是一个表，截至目前为止，总共有17种类型的常量，在JDK8中，有14种类型的常量，分别是：

- UTF-8编码的字符串；
- 整型字面量；
- 长整型字面量；
- 浮点型字面量；
- 双精度浮点型字面量；
- 字符串类型字面量；
- 类或接口的符号引用；
- 字段的符号引用；
- 类方法的符号引用；
- 接口方法的符号引用；
- 字段或部分符号引用；
- 方法句柄；
- 方法类型；
- 动态常量；
- 动态调用点限定符；
- 用于表示这是一个模块；
- 用于表示这是一个被模块导出或开放的包；

访问标志access\_flag

- 用于描述该类或该接口的访问信息，共2字节，16bit，每个bit对应一个标志位，共使用其中九个；
- 1.是否是一个接口；
- 2.是否是一个枚举；
- 3.是否是一个注解；
- 4.是否是一个模块；
- 5.是否使用final修饰；
- 6.是否使用static修饰；
- 7.是否使用public修饰；
- 8.是否并非由用户代码产生；
- 9.是否支持invokeSpecil字节码指令的新语义；

类索引、父类索引、接口索引集合

- JVM通过这三项确定该类或该接口的继承关系；
- 类索引表示该类或该接口的全限定名；
- 父类索引表示该类所继承父类的全限定名；
- 接口索引集合是一组U2类型的数据，数组第一个元素用于表示实现接口的数量，之后的每一个元素与implements或extends关键字之后的接口顺序保持一致；

字段表集合

- 用于描述类或接口中声明的变量，包括实例变量、静态变量，不包括局部变量；
- 入口是一个U2类型的字段表容量计数器；

字段表结构：

固定部分。访问表示access\_flag：

- 是否使用public修饰；
- 是否使用private修饰；
- 是否使用protected修饰；
- 是否使用final修饰；
- 是否使用static修饰；
- 是否使用volatile修饰；
- 是否使用transient修饰；
- 是否为enum字段；
- 是否由编译器自动产生；

- 固定部分。用于描述字段的简单名称；

- 固定部分。用于描述字段的描述符；

- 可选部分。每个字段表可携带一些属性信息，比如使用final修饰的静态变量在前端编译器编译期间就会将值写入到class文件的字段表集合对应字段表的ConstantValue属性表中；

方法表集合

- 用于描述类或接口中声明的方法；
- 入口是一个U2类型的方发表容量计数器；

方法表结构：

固定部分。访问标志access\_flag：

- 是否使用public修饰；
- 是否使用protected修饰；
- 是否使用private修饰；
- 是否使用final修饰；
- 是否使用static修饰；
- 是否使用abstract修饰；
- 是否使用synchronized修饰；
- 是否使用native修饰；
- 是否使用strictfp修饰；
- 是否由编译器产生的桥接方法；
- 是否由编译器自动产生；
- 是否接口不定参数；

属性表集合

- 用于描述整个class文件层面的属性信息；