





(续)		
类 型	标 志	描 述
CONSTANT_InvokeDynamic_info	18	表示一个动态方法调用点
CONSTANT_Module_info	19	表示一个模块
CONSTANT_Package_info	20	表示一个模块中开放或者导出的包

ACC_FINAL	0x0010	是否被声明为 final。只有类可设置
ACC_SUPER	0x0020	是否允许使用 invokespecial 字节码指令的新语法的语在 JDK 1.0.2 发生过大变，为了区别这条指令 1.0.2 之后编译出来的类的这个标志都必须为真
ACC_INTERFACE	0x0200	标识这是一个接口
ACC_ABSTRACT	0x0400	是否为 abstract 类型，对于接口或者抽象类类由类图例为和

ss
r Class
aces

表6-9 字段访问标志

标识字符	含 义	标识字符	含 义
B	基本类型 byte	J	基本类型 long
C	基本类型 char	S	基本类型 short
D	基本类型 double	T	基本类型 time_t

- 类名 `org.springframework.test.class` 是这个类的全限定名，仅仅是把类全名中的“`org`”替换成了“`了`”而已，为了连接类的多个全限定名之间不产生混淆，在调用时加一个“`。`”，表示全限定名结束。
- 包名 `inc` 指没有类名和参数修饰符的方法或者类名修饰，这个类中的`inc`方法和`m`字段的简单名称分别是“`inc`”和“`m`”。

标志名称	标志值	含 义
ACC_PUBLIC	0x0001	方法是否为 public
ACC_PRIVATE	0x0002	方法是否为 private

- 按照先参数列表、后返回值的顺序书写
- 参数列表在返回参数的严格顺序放在一组小括号“()”之内。
 - 方法 `indexOf` 的返回值为 `int`
 - 方法 `java.lang.String.toString` 的返回值为 `java.lang.String` ;
 - 方法 `int indexOf(char[] source, int sourceOffset, int sourceCount, char[] target, int targetOffset, int targetCount, int[] matchStart)`

`code_length`代表字节码长度。

Java语言里面的惯例是：在任何实例方法里面，都可以通过“this”关键字访问到此方法所属的对象。这个访问机制对Java程序的编写很有意义。

因此的实现非常简单，仅仅是通过在Java编译器编译的时候把对this关键字的访问转换为对一个普通方法参数的访问，然后在虚拟机调用时由传参人传入参数而已。

因此在此实现方式的局部变量表中至少会存在一个当前对象实例的局部变量，而局部变量中也会预留出第一个空槽位来存放对实例对象的实际方法参数值从开始计算。

因此编译生成后的Class文件中应该有这样的代码：
`byte[] local_0 = new byte[1];`

- 用户需要指定一个执行任务的主类（包含main方法的那个类），虚拟机首先初始化这个主类
 - `java.lang.invoke.MethodHandle`实现最后的解析结果力
 - `REF_getStatic`、`REF_putStatic`、`REF_invokeStatic`、`REF_newInvokeSpecial`这四种类型的访问方法
 - 并且这个方法调用拉取的类没有进行过初始化，则需要先触发其初始化。

元数据验证 —— 对字节码描述的信息进行语义分析，以保证其描述的信息符合《Java语言规范》的要求 ——

验证 (Verification) ——

· 保证代码的完整性
· 你未加载类

解析 (Resolution)

- 符号引用 → 符号引用是一种符号或标识符所引用的符号, 符号可以以任意的形式存在, 只需要用打散无意义的符号来代替即可
- 直接引用 → 直接引用是可以直接引用符号的标识符, 相对标识符来说是一个能标识出它标识的符号

标识符的变换是构造类库或 `clint` 方法的过程

`<clint>` 方法是由从源程序类或类集中的所有类名和函数名动作链接符号表 (stack) 中的符号与包含的。
编译程序将源程序中的符号与在符号表中找到的符号进行比较, 若在符号表中找不到, 则符号表中添加。
编译符号表与符号引用表是在符号表中添加的, 添加在它们之间的位置, 在源程序的编译过程中可以知道, 但是不能确定。

(Extension Class Loader)

- 加载Java扩展类库的类
- sun.misc.Launcher\$ExtClassLoader

负责加载由JRE的C:\jre\lib\ext\目录下加载

