



05 - 如何编写ASM代码



舍是境界 300 关注

0.229 2022.01.13 07:12:24 字数 447 阅读 16

在刚开始学习ASM的时候，编写ASM代码是不太容易的。或者，有些人原来对ASM很熟悉，但由于长时间不使用ASM，编写ASM代码也会有一些困难。在本文当中，我们介绍一个ASMPrint类，它能帮助我们将.class文件转换为ASM代码，这个功能非常实用。

ASMPrint类

下面是ASMPrint类的代码，它是利用org.objectweb.asm.util.TraceClassVisitor类来实现的。在使用的时候，我们注意修改一下className、parsingOptions和asmCode参数就可以了。

```
1 import jdk.internal.org.objectweb.asm.ClassReader;
2 import jdk.internal.org.objectweb.asm.util.ASMifier;
3 import jdk.internal.org.objectweb.asm.util.Printer;
4 import jdk.internal.org.objectweb.asm.util.Textifier;
5 import jdk.internal.org.objectweb.asm.util.TraceClassVisitor;
6
7 import java.io.IOException;
8 import java.io.PrintWriter;
9
10 public class ASMPrint {
11     public static void main(String[] args) throws IOException {
12         // (1) 设置参数
13         String className = "sample.HelloWorld";
14         int parsingOptions = ClassReader.SKIP_FRAMES | ClassReader.SKIP_DEBUG;
15         boolean asmCode = true;
16
17         // (2) 打印结果
18         Printer printer = asmCode ? new ASMifier() : new Textifier();
19         PrintWriter printWriter = new PrintWriter(System.out, true);
20         TraceClassVisitor traceClassVisitor = new TraceClassVisitor(null, printer, printer);
21         new ClassReader(className).accept(traceClassVisitor, parsingOptions);
22     }
23 }
```

在现在阶段，我们主要是使用这个类，来帮助我们生成ASM代码；

ASMPrint类使用示例

假如，有如下一个HelloWorld类：

```
1 public class HelloWorld {
2     public void test() {
3         System.out.println("Test Method");
4     }
5 }
```

对于ASMPrint类来说，其中

- className值设置为类的全限定名，可以是我们自己写的类，例如sample.HelloWorld，也可以是JDK自带的类，例如java.lang.Comparable。
- asmCode值设置为true或false。如果是true，可以打印出对应的ASM代码；如果是false，可以打印出方法对应的Instruction。



热门故事

妻子去世半年，我再娶一个小十岁的女人有错吗？

代替公主和亲后，我成了敌国后宫“升职”最快的妃子

直播间打赏五十万，女主播主动私信我要见面

生完二胎，老公给我雇了一个“90后”小保姆

推荐阅读

坚持再坚持

阅读 1,889

爱自己，才是终身浪漫的开始

阅读 887

如果你有一儿一女，你会把你的财产平分吗？还是给儿子留多点？

阅读 2,493

腾讯三面：你换一个角度跟我讲讲MySQL的B+树

阅读 1,874

netty（六）NIO、BIO与AIO

阅读 1,436

- parsingOptions值设置为ClassReader.SKIP_CODE、ClassReader.SKIP_DEBUG、ClassReader.SKIP_FRAMES、ClassReader.EXPAND_FRAMES的组合值，也可以设置为0，可以打印出详细程度不同的信息。

运行结果示例：

```
1 package asm.sample;
2 import java.util.*;
3 import jdk.internal.org.objectweb.asm.*;
4 public class HelloWorldDump implements Opcodes {
5
6     public static byte[] dump () throws Exception {
7
8         ClassWriter cw = new ClassWriter(0);
9         FieldVisitor fv;
10        MethodVisitor mv;
11        AnnotationVisitor av0;
12
13        cw.visit(52, ACC_PUBLIC + ACC_SUPER, "sample/HelloWorld", null, "java/lang/Object", ne
14
15        {
16            fv = cw.visitField(ACC_PRIVATE + ACC_FINAL + ACC_STATIC, "intValue", "I", null, new In
17            fv.visitEnd();
18        }
19        {
20            mv = cw.visitMethod(ACC_PUBLIC, "<init>", "()V", null, null);
21            mv.visitCode();
22            mv.visitVarInsn(ALOAD, 0);
23            mv.visitMethodInsn(INVOKEVIRTUAL, "java/lang/Object", "<init>", "()V", false);
24            mv.visitInsn(RETURN);
25            mv.visitMaxs(1, 1);
26            mv.visitEnd();
27        }
28        {
29            mv = cw.visitMethod(ACC_PUBLIC, "test", "()V", null, null);
30            mv.visitCode();
31            mv.visitInsn(ICONST_1);
32            mv.visitVarInsn(ISTORE, 1);
33            mv.visitInsn(ICONST_2);
34            mv.visitVarInsn(ISTORE, 2);
35            mv.visitVarInsn(ILOAD, 1);
36            mv.visitVarInsn(ILOAD, 2);
37            mv.visitInsn(IADD);
38            mv.visitVarInsn(ISTORE, 3);
39            mv.visitInsn(RETURN);
40            mv.visitMaxs(2, 4);
41            mv.visitEnd();
42        }
43        {
44            mv = cw.visitMethod(ACC_PUBLIC + ACC_STATIC, "main", "(Ljava/lang/String;)V", null, n
45            mv.visitParameter("args", 0);
46            mv.visitCode();
47            mv.visitFieldInsn(GETSTATIC, "java/lang/System", "out", "Ljava/io/PrintStream;");
48            mv.visitLdcInsn("HelloWorld");
49            mv.visitMethodInsn(INVOKEVIRTUAL, "java/io/PrintStream", "println", "(Ljava/lang/Strin
50            mv.visitInsn(RETURN);
51            mv.visitMaxs(2, 1);
52            mv.visitEnd();
53        }
54        cw.visitEnd();
55
56        return cw.toByteArray();
57    }
58 }
```



热门故事

妻子去世半年，我再娶一个小十岁的女人有错吗？

代替公主和亲后，我成了敌国后宫“升职”最快的妃子

直播间打赏五十万，女主播主动私信我要见面

生完二胎，老公给我雇了一个“90后”小保姆

推荐阅读

坚持再坚持

阅读 1,889

爱自己，才是终身浪漫的开始

阅读 887

如果你有一儿一女，你会把你的财产平分吗？还是给儿子留多点？

阅读 2,493

腾讯三面：你换一个角度跟我讲讲MySQL的B+树

阅读 1,874

netty（六）NIO、 BIO与AIO

阅读 1,436

小结

本文主要介绍了ASMPrint类和它的使用示例，内容总结如下：

- ASMPrint类，是通过org.objectweb.asm.util.TraceClassVisitor实现的。
- ASMPrint类的作用，是帮助我们生成ASM代码。当我们想实现某一个功能时，不知道如何下手，可以使用ASMPrint类生成的ASM代码，作为思考的起点。

