



18 - MethodVisitor创建对象



舍是境界

200

关注

0.31 2022.01.28 07:47:23 字数 143 阅读 13

目标

假如有一个GoodChild类，内容如下：

```
1 public class GoodChild {
2     public String name;
3     public int age;
4
5     public GoodChild(String name, int age) {
6         this.name = name;
7         this.age = age;
8     }
9 }
```

预期目标是生成一个HelloWorld类：

```
1 public class HelloWorld {
2     public void test() {
3         GoodChild child = new GoodChild("Lucy", 8);
4     }
5 }
```

编码实现

```
1 import lsieun.utils.FileUtils;
2 import org.objectweb.asm.ClassWriter;
3 import org.objectweb.asm.MethodVisitor;
4
5 import static org.objectweb.asm.Opcodes.*;
6
7 public class HelloWorldGenerateCore {
8     public static void main(String[] args) throws Exception {
9         String relative_path = "sample/HelloWorld.class";
10        String filepath = FileUtils.getFilePath(relative_path);
11
12        // (1) 生成byte[]内容
13        byte[] bytes = dump();
14
15        // (2) 保存byte[]到文件
16        FileUtils.writeBytes(filepath, bytes);
17    }
18
19    public static byte[] dump() throws Exception {
20        // (1) 创建ClassWriter对象
21        ClassWriter cw = new ClassWriter(ClassWriter.COMPUTE_FRAMES);
22
23        // (2) 调用visitXxx()方法
24        cw.visit(V1_8, ACC_PUBLIC + ACC_SUPER, "sample/HelloWorld", null, "java/lang/Object", null);
25
26        {
27            MethodVisitor mv1 = cw.visitMethod(ACC_PUBLIC, "<init>", "()V", null, null);
28            mv1.visitCode();
29            mv1.visitVarInsn(ALOAD, 0);
30            mv1.visitMethodInsn(INVOKE_SPECIAL, "java/lang/Object", "<init>", "()V", false);
31            mv1.visitInsn(RETURN);
32            mv1.visitMaxs(1, 1);
33            mv1.visitEnd();
34        }
```



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

回魂夜

阅读 2,156

故乡的年味

阅读 1,831

晨

阅读 753

一波值得收藏的宝藏句子

阅读 717

读金庸II 仰望高山，才觉渺小

阅读 5,226



```
35
36     {
37         MethodVisitor mv2 = cw.visitMethod(ACC_PUBLIC, "test", "()V", null, null);
38         mv2.visitCode();
39         mv2.visitTypeInsn(NEW, "sample/GoodChild");
40         mv2.visitInsn(DUP);
41         mv2.visitLdcInsn("Lucy");
42         mv2.visitIntInsn(BIPUSH, 8);
43         mv2.visitMethodInsn(INVOKESPECIAL, "sample/GoodChild", "<init>", "(Ljava/L
44         mv2.visitVarInsn(ASTORE, 1);
45         mv2.visitInsn(RETURN);
46         mv2.visitMaxs(4, 2);
47         mv2.visitEnd();
48     }
49
50     cw.visitEnd();
51
52     // (3) 调用toByteArray()方法
53     return cw.toByteArray();
54 }
55 }
```

验证结果

```
1 import java.lang.reflect.Method;
2
3 public class HelloWorldRun {
4     public static void main(String[] args) throws Exception {
5         Class<?> clazz = Class.forName("sample.HelloWorld");
6         Object obj = clazz.newInstance();
7
8         Method m = clazz.getDeclaredMethod("test");
9         m.invoke(obj);
10    }
11 }
```

Frame的变化

对于HelloWorld类中test()方法对应的Instruction内容如下：

```
1 public void test();
2 Code:
3     0: new           #11           // class sample/GoodChild
4     3: dup
5     4: ldc           #13           // String Lucy
6     6: bipush        8
7     8: invokespecial #16           // Method sample/GoodChild."<init>":(Ljava/L
8    11: astore_1
9    12: return
```

该方法对应的Frame变化情况如下：

```
1 test()V
2 [sample/HelloWorld] []
3 [sample/HelloWorld] [uninitialized_sample/GoodChild]
4 [sample/HelloWorld] [uninitialized_sample/GoodChild, uninitialized_sample/GoodChild]
5 [sample/HelloWorld] [uninitialized_sample/GoodChild, uninitialized_sample/GoodChild, j
6 [sample/HelloWorld] [uninitialized_sample/GoodChild, uninitialized_sample/GoodChild, j
7 [sample/HelloWorld] [sample/GoodChild]
8 [sample/HelloWorld, sample/GoodChild] []
9 [] []
```

小结

• 从Instruction的角度来讲，创建对象的指令集合：

- new
- dup
- invokespecial

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

回魂夜

阅读 2,156

故乡的年味

阅读 1,831

晨

阅读 753

一波值得收藏的宝藏句子

阅读 717

读金庸II 仰望高山，才觉渺小

阅读 5,226

- 从Frame的角度来讲，在创建新对象的时候，执行new指令之后，它是uninitialized状态，执行invokespecial指令之后，它是一个“合格”的对象。

1人点赞 >

Java系列

...

更多精彩内容，就在简书APP



"小礼物走一走，来简书关注我"

赞赏支持

还没有人赞赏，支持一下



舍是境界 200 专注高性能，高可用，高扩展架构领域
总资产147 共写了77.2W字 获得408个赞 共16个粉丝

关注

签名文件一键自动生成



推荐阅读

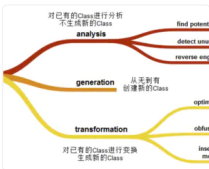
16 - MethodVisitor生成构造方法

先回顾下asm能做什么 在当前阶段，我们关注如何生成代码 <init>()方法在.class文件中，构造方法的名...



舍是境界 阅读 13 评论 0 赞 1

更多精彩内容 >

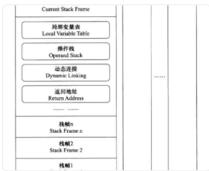


深入理解java虚拟机（七）-虚拟机字节码执行引擎

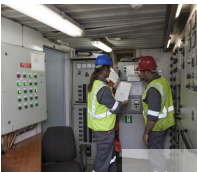
本文基于周志明的《深入理解java虚拟机 JVM高级特性与最佳实践》所写。特此推荐。运行时栈帧结构 栈帧是用于支...



阳光的技术小栈 阅读 580 评论 0 赞 5



安全检测找我们！你想要的我们都有！



深入理解 JVM 内存分配模型

程序运行时，内存到底是如何进行分配的？虚拟机在执行 java 程序的过程中会...