

Java面试陷阱第三波 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/145/2021_2022_Java_E9_9D_A2_E8_AF_95_c104_145300.htm

6. class Something { int i. public void doSomething() { System.out.println("i = " i). } } 有什么错呢? 看不出来啊。 答案: 正确。输出的是"i = 0"。 int i属于instant variable (实例变量，或叫成员变量)。 instant variable有default value。 int的default value是0。

7. class Something { final int i. public void doSomething() { System.out.println("i = " i). } } 和上面一题只有一个地方不同，就是多了一个final。这难道就错了吗? 答案: 错。 final int i是个final的instant variable (实例变量，或叫成员变量)。 final的instant variable没有default value，必须在constructor (构造器)结束之前被赋予一个明确的值。可以修改为"final int i = 0."。

8. public class Something { public static void main(String[] args) { Something s = new Something(). System.out.println("s.doSomething() returns " doSomething()). } public String doSomething() { return "Do something ...". } } 看上去很完美。 答案: 错。看上去在main里call doSomething没有什么问题，毕竟两个methods都在同一个class里。但仔细看，main是static的。 static method不能直接call non-static methods。可改成"System.out.println("s.doSomething() returns " s.doSomething())."。同理，static method不能访问non-static instant variable。

9.此处，Something类的文件名叫OtherThing.java class Something { private static void main(String[] something_to_do) { System.out.println("Do something ..."). } } 这个好像很明显。 答案: 正确。从来没有人

说过Java的Class名字必须和其文件名相同。但public class的名字必须和文件名相同。 10.最难的一题: interface Playable { void play(). } interface Bounceable { void play(). } interface Rollable extends Playable, Bounceable { Ball ball = new Ball("PingPang"). } class Ball implements Rollable { private String name. public String getName() { return name. } public Ball(String name) { this.name = name. } public void play() { ball = new Ball("Football"). System.out.println(ball.getName()). } } 这个错误不容易发现。 答案: 错。 "interface Rollable extends Playable, Bounceable"没有问题。 interface可继承多个interfaces, 所以这里没错。 问题出在interface Rollable里的"Ball ball = new Ball("PingPang")."。 任何在interface里声明的interface variable (接口变量, 也可称成员变量), 默认为public static final。 也就是说"Ball ball = new Ball("PingPang")."实际上是"public static final Ball ball = new Ball("PingPang")."。 在Ball类的Play()方法中, "ball = new Ball("Football")."改变了ball的reference, 而这里的ball来自Rollable interface, Rollable interface里的ball是public static final的, final的object是不能被改变reference的。 因此编译器将在"ball = new Ball("Football")."这里显示有错。 100Test 下载频道开通, 各类考试题目直接下载。 详细请访问 www.100test.com