

第二章 标示符、关键字、变量-答案

1.程序阅读改错

题目要求：

1. 请阅读并分析以下四个源文件中的内容，找出其中会导致程序无法通过编译或运行的错误
2. 修改代码使它们能够编译和运行

Test1.java

```
1 package com.briup.md01;
2 public class Test1 {
3     public static void main(String[] args) {
4         System.out.println("What's wrong with this program?");
5     }
6 }
7 class TestAnother1 {
8     public static void main(String[] args) {
9         System.out.println("What's wrong with this program?");
10    }
11 }
```

Test2.java

```
1 public class Test2 {
2     public static void main(String[] args) {
3         System.out.println("What's wrong with this program?");
4     }
5 }
```

Test3.java

```
1 public class Test3 {
2     public static void main(String [ ] args) {
3         System.out.println("What's wrong with this program?");
4     }
5 }
```

Test4.java

```
1 public class Test4 {
2     public static void main(String[] args) {
3         System.out.println("What's wrong with this program?");
4     }
5 }
```

2.变量声明改错

题目要求：

请观察下列代码中的变量声明语句，指出哪几句会产生编译报错或精度丢失，并解释原因。

```
1 {
2     byte a = 56;
3     byte b = 156;
4     int c = 2000000000 + 2000000000;
5     long d = 999999999;
6     long e = 9999999999;
7     long f = 9999999999L;
8 }
```

3.进制格式考查

题目要求：

定义3个int类型变量，分别使用二进制、八进制、十六进制三种字面常量语法为其赋值，使其值均等于十进制值85。

4.找零问题

题目要求：

小明有2元，买东西花了1.1元，问找零多少，请用Java代码描述上述过程！

注意：观察程序运算输出是否能得到0.9，如果不能，请解决该精度问题。

5.字面值常量考查

题目要求：

请定义2个变量，分别用每天对应的毫秒数以及微秒数（表达式描述）给其赋值。输出变量值，观察是否正确。如果输出有问题，请解决！

6.数据类型问答

题目要求：

基本类型有哪些，取值范围分别是什么？

7.数据类型问答

题目要求：

请回答引用数据类型有哪些，分别是什么？

8. 数据类型问答

题目要求：

请简述基本类型和引用类型变量的区别。

9. 编程题

题目要求：

请编码输出下图内容：

-----商场库存清单-----

品牌型号	尺寸	价格	库存数
MacBookAir	13.3	5699.0	5
ThinkPadT490	14.0	8499.0	10
MateBook 14	14.0	7199.0	18

总库存数： 33

商品库存总金额： 243067.0

案例分析：

输出内容包含3部分，表头、表格、表尾，具体要求如下：

- 表头部分：可以采用多条System.out.println语句输出，其是固定数据， 直接输出即可
- 表格中间：请定义变量描述商品型号、尺寸、价格、库存，然后按格式进行输出（注意变量数据类型）
- 表格尾巴：先对商品数据进行数学计算，得出结果后输出内容

10. 程序分析题

题目要求：

分析以下代码的运行结果。

```
1 public class Test10 {
2     public static void main(String[] args) {
3         short s = 1;
4         s = s + 2;
5         short x = 1;
6         x += 1;
7     }
8 }
```