

Chapter 09 内部类、Object、包装类、String

Key Point :

- Object 类
- finalize 方法
- toString 方法
- equals 方法
- getClass 方法
- 包装类
- 三种类型(原始类型、包装类、字符串)与6种转换
- 内部类的基本概念
- 串池
- StringBuilder类

问题：

1. (Object 类) 在 Object 类中，定义的 finalize 方法在_____时调用，toString()方法返回值表示_____，equals 方法的作用为_____，getClass 方法作用为_____。

2. (toString 方法) 写出下列代码的运行结果

```
class Student{  
  
    private int age;  
  
    private String name;  
  
    public Student(){}  
  
    public Student(String name, int age){
```

```

        this.name = name;

        this.age = age;
    }

    public String toString(){

        return name + " " + age;

    }

}

public class TestStudent{

    public static void main(String args[]){

        Student stu1 = new Student();

        Student stu2 = new Student( "Tom" , 18);

        System.out.println(stu1);

        System.out.println(stu2);

    }

}

```

3. (equals) 写出下面程序运行的结果

```

public class TestEquals{

    public static void main(String args[]){

        String str1 = new String( "Hello" );

        String str2 = new String( "Hello" );

        System.out.println(str1.equals(str2));

        System.out.println(str1 == str2);

    }

}

```

```
}
```

4. (getClass) 写出下面程序运行的结果

```
class Animal{}

    class Dog extends Animal{}

    public class TestGetClass{

        public static void main(String args[]){

            Animal a1 = new Dog();

            Animal a2 = new Animal();

            System.out.println(a1 instanceof Animal);

            System.out.println(a1.getClass() == a2.getClass());

        }

    }
```

5. (内部类) Java 中的内部类包括

- A. 成员内部类
- B. 静态内部类
- C. 局部内部类
- D. 匿名内部类
- E. 公开内部类
- F. 抽象内部类

6. 为 Worker 类增加 equals 和 toString 方法。

7. 从命令行上读入一个字符串，用两种不同的方法，把该字符串转换为一个 int 类型

方法一：把 String 直接转换为 int

方法二：把 String 转换为 Integer，再把 Integer 转换为 int 类型

8. （ toString，字符串加法）有下面代码

```
class Student{

    private int age;

    private String name;

    public Student(){

    }

    public Student(String name, int age){

        this.name = name;

        this.age = age;

    }

    public String toString(){

        return name + " " + age;

    }

}

public class TestStudent{

    public static void main(String args[]){

        Student stu1 = new Student( "tom" , 18);

        System.out.println(/*1*/);

    }

}
```

问：在/*1*/位置，填入什么代码能编译通过？

A. stu1 + " " + 100

B. 100 + " " + stu1

C. " " + 100 + stu1

D. stu1 + 100 + " "

9. （局部内部类）*有下面代码

```
class OuterClass{  
    private int value1 = 100;  
  
    private static int value2 = 200;  
  
    public void method(int value3){  
        final int value4 = 400;  
  
        class InnerClass{  
            public void print(){  
                //1  
            }  
        }  
    }  
}
```

问：下面哪些代码放在//1 处能够编译通过？

A. System.out.println(value1);

B. System.out.println(value2);

C. System.out.println(value3);

D. System.out.println(value4);

10. 已知接口 Light 定义如下：

```
interface Light{

    void shine();

}

//定义 Lamp 类 :
```

```
class Lamp{

    public void on(Light light){

        light.shine();

    }

}
```

```
//写一个类 TestLamp , 部分代码如下 :
```

```
public class TestLamp{

    public static void main(String args[]){

        Lamp lamp = new Lamp();

        //1

        //2

    }

}
```

把 TestLamp 类补充完整，要求：

- I. 在//1 处使用局部内部类技术，调用 lamp 的 on 方法要求输出“ shine in red”
- II. 在//2 处使用匿名内部类技术，调用 lamp 的 on 方法要求输出“ shine in yellow”

11. 获取邮箱 “zhengcg@zparkhr.com” 中的用户名“ zhengcg” 。

12. 验证邮箱 “zhengcg@zparkhr.com” 是否是一个合法的邮箱格式。

提示：

- I. 邮箱必须包含 "@" 和 "."
- II. 最后一个 "." 的位置必须大于 "@" 的位置。

13. 将随机获取的 UUID (含义是通用唯一识别码 Universally Unique Identifier) 中的 "-" 去掉。

提示：`java.util.UUID.randomUUID().toString()` //可以获取随机 UUID

格式：`e6c57443-1667-4d75-98f6-a8863d95e58f`

14. 在给定的字符串 "ABCDEFGHijklmn1234567" 中获取随机的 4 个字符 , 并使用 `StringBuilder` 拼接成字符串。(随机获取到的 4 个字符中可以出现重复字符)

提示：创建随机数对象 `java.util.Random`。

```
java.util.Random random = new java.util.Random();
```

```
random.nextInt(100); //可以获取到 0~99 中任意一个随机数
```

15. 给定一个由数字组成的字符串

如:"1239586838923173478943890234092" ; 统计出每个数字出现的次数。