

# Java 程序设计 Lab02

```
1  /**
2   * 学号:
3   * 姓名:
4   * JDK版本:
5   * 代码文件编码方式:
6   * IDE:
7   */
```

## 实验目的

- 理解并掌握类的概念
- 理解并掌握对象的概念
- 理解类与对象的关系
- 理解面向对象中抽象过程
- 理解面向对象中的消息
- 理解 Java 程序的基本结构并能灵活使用
- 理解并掌握 Java 类的定义（成员变量、成员方法和方法重载）
- 理解并掌握 Java 类的构造函数（默认构造函数、带参数构造函数），理解重载的构造函数并灵活使用
- 理解 Java 垃圾内存自动回收机制
- 理解并掌握 Java 类变量和类方法

## 实验题目

### 1. 简答题

#### Problem

编译下面这个类：

```
1  public class Sample {
2      int x; // 1
3      long y = x; // 2
4      public void f(int n) {
5          int m; // 3
6          int t = n + m; // 4
7      }
8      public static void main(String[] args) {
9          Sample t=new Sample();
10         t.f(5);
11         System.out.println(t.x);
12     }
13 }
```

注释标记的哪些行会导致编译错误，为什么？

## Solution

### 2. overload

#### Problem

阅读下面代码：

```
1 public class Overload{
2     Overload(int m){}
3     Overload(double m){}
4     int Overload(int m){return 23;}
5     void Overload(double m){}
6 }
```

对于 `Overload` 类，下面哪些叙述是错误的，为什么错？

- a. `Overload(int m)` 与 `Overload(double m)` 互为重载的构造方法。
- b. `int Overload(int m)` 与 `void Overload(double m)` 互为重载的非构造方法。
- c. `Overload` 类有 2 个构造方法，尝试调用默认构造方法 `Overload()` 会无法通过编译。
- d. `Overload` 类有 3 个构造方法。

## Solution

### 3. overload2

#### Problem

阅读下面这段代码：

```
1 public class Overload2 {
2     int a,b;
3     int c=-1;
4     public Overload2(){a=-2;}
5     public Overload2(int p1,int p2){a=p1; b=p2;}
6     public Overload2(int p1,int p2,int p3){
7         this(p1,p2);
8         c=p3;
9     }
10
11     @Override
12     public String toString(){
13         return String.format("(a,b,c)=(%d,%d,%d)",a,b,c);
14     }
15     public static void main(String args[]){
16         Overload2 s = new Overload2();
17         System.out.println(s);
18         System.out.println(new Overload2(5,6));
19         System.out.println((new Overload2(2,3,4)).toString());
20     }
21 }
22
```

运行 `java Overload2`，程序的输出是什么？

题外话：

根据 `void java.io.PrintStream.print()` 系列函数的实现，当传入的 `String` 或 `Object` 是 `null` 时，最后都会导致输出字符串“null”，而不是什么都不输出。

进行字符串连接时也一样：“”+null 的结果是“null”。

因此当涉及相关操作时需要稍微注意一下。

具体原因参见 `java.io.PrintStream`、`java.lang.String` 的源代码。

## Solution

输出结果：

```
1 |
```

## 4. 简答题 finalize

### Problem

阅读下面这段代码：

```
1  public class User {
2      public static int userCount = 0;
3      public User() {
4          ++userCount;
5      }
6
7      @Override
8      protected void finalize() throws Throwable {
9          super.finalize();
10         System.out.println("bye");
11         --userCount;
12     }
13     public static void main(String[] args) {
14         System.out.println(User.userCount); // 1
15         User u = new User();
16         System.out.println(User.userCount); // 2
17         u = null;
18         System.out.println(User.userCount); // 3
19         System.gc();
20         System.out.println(User.userCount); // 4
21     }
22 }
```

注意代码中用注释标出的四行 `System.out.println(User.userCount)`，某些行的输出结果有多种可能。

运行 `java User`，程序的输出可能是什么？

注释标记行	可能的输出内容
1	
2	
3	
4	

有什么办法可以让 `finalize()` 在 `main()` 结束前一定被调用吗？

题外话：

和 C++ 不同，在 Java 中显式重写 `finalize()` 通常是不必要的。

`java.lang.System.gc()` 只会在需要的时候执行垃圾收集，你的调用只是一个“建议”。

## Solution

## 5. 编程题

写一个音乐类，并编写测试类显示音乐信息（音乐名称、音乐类型）。编写一个手机类，并编写测试类显示手机信息（手机品牌、手机型号）

- 1 目录名为：5
- 2 文件名有：

## 6. 编程题

编写程序,在其中定义两个类：

- Person 类：
  - 属性有 name、age 和 sex
  - 提供你认为必要的构造方法
  - 方法 `setAge()` 设置人的合法年龄 (0~130)
  - 方法 `getAge()` 返回人的年龄
  - 方法 `work()` 输出字符串“working”
  - 方法 `showAge()` 输出 age 值
- TestPerson 类：
  - 创建 Person 类的对象，设置该对象的 name、age 和 sex 属性
  - 调用 `setAge()` 和 `getAge()` 方法，体会 Java 的封装性。
  - 创建第二个对象，执行上述操作，体会同一个类的不同对象之间的关系。

- 1 目录名为：6
- 2 文件名有：

## 7. 编程题

编写一个 Java 命令程序，只从标准输入读取一行用户输入，判断这行输入是否是一个没有前导 0 的无符号整数；如果是，则还要判断该数字是否是一个回文数。输出你的判断和执行结果。

对于“没有前导 0 的无符号整数”的定义：

- 是一个字符串 s

- $s$  的长度至少是 1，没有上限要求
- $s$  的字符集  $\Sigma=\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ ，其他所有字符都不应该出现在  $s$  中
- 当  $s$  的长度大于 1 时，若从其首部开始有若干个连续字符 0，那么这些字符 0 都叫做  $s$  的“前导 0”
  - 比如数字串 00010020，有三个前导 0
  - 比如数字串 01，有 1 个前导 0
  - 比如数字串 102030，没有前导 0
  - 比如数字串 0，没有前导 0

本题对于回文数的定义：

- 是一个字符串  $s$
- $s$  的长度至少是 1，没有上限要求
- $s$  的字符集  $\Sigma=\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ ，其他所有字符都不应该出现在  $s$  中
- 将  $s$  中的字符逆序排列并去除前导 0 得到的数字串  $r$ ，有  $s$  和  $r$  完全相同
  - $s=123$  时， $r=321$ ，不相同， $s$  和  $r$  都不是回文数
  - $s=12321$  时， $r=12321$ ，相同， $s$  是回文数
  - $s=12100$  时， $r=121$ ，不相同， $s$  不是回文数，但  $r$  是回文数
  - $s=1$  时， $r=1$ ，相同， $s$  是回文数
  - $s=0$  时， $r=0$ ，相同， $s$  是回文数

本题将输入的一行字符视为一个完整的字符串，如果输入的是诸如“121 121”这样包含空格的串，虽然 121 是回文数，但是整个串不应该被认为是回文数。如果将字符集扩充为包含空格的其他字符集，那么“121 121”就是一个该字符集下的回文串，不过本题的字符集限定为由 0~9 这十个数字组成的字符集。

- 1 目录名为：7
- 2 文件名有：