实验二 类和对象

1. 实验目的
2. 了解类的成员设计
3. 掌握类与对象的关系。
4. 理解静态方法与实例方法的使用差异
5. 理解成员方法的参数类型
6. 掌握对象数组
7. 实验预备知识
8. Java中关于数组的核心类
9. System类

Java中在System类中提供了特殊的方法拷贝数组：arraycopy()——如果数组是对象型，则拷贝的是引用，而不是对象，即对象本身不变

1. java.util. Arrays类

a)拷贝数组

public static double[] copyOf(double[] original, int newlength)

public static double[] copyOfRange(double[]original, int from,int to)

b)排序 sort

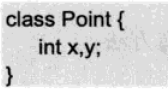
c)二分查找 binarySearch （先排序再二分查找）

1. 类和对象知识

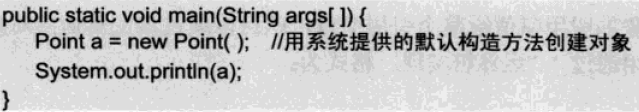
类是构造对象的模板或蓝图，由类构造对象的过程称为创建类的实例

1. 实验内容
2. 验证性实验
3. 基础类（1-12小题）

1）构建Point类

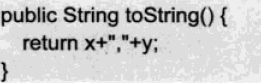


2）创建对象



观察输出结果是对象的引用地址，记录实验结果。

3）为Point类编写一个toString（）方法，**注意必需要public权限**



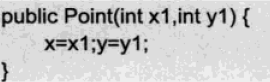
观察输出结果，思考原因。

4）在main方法中添加如下代码，定义另一对象引用变量b，让b和a引用同一对象。



观察输出结果，思考原因。

1. 定义构造方法。



6）用新定义构造方法创建对象，改变引用变量b，使其指向新建对象。



输出a、b观察有何变化，在实验报告上画图表示，体会对象与对象引用的关系。

7）加入如下代码：

b=null;

再输出b，观察结果变化。

8）定义对象数组



输出整个数组：



观察结果，分析总结对象数组元素的赋值特点

9）添加如下代码：



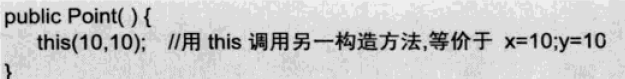
重新观察输出结果。

1. 添加如下代码：



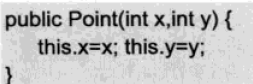
观察编译错误，体会什么时候系统会自动提供默认构造方法。

11）编写无参构造方法



重新观察输出结果，注意c[0]和c[7]

12）将有参构造方法的参数名定为x,y，则程序做如下修改：



理解this的作用

（b）理解static代码块

static块可以用来优化程序性能，是因为它的特性:只会在类加载的时候执行一次。下面看个例子:

class Person{

private Date birthDate;

public Person(Date birthDate) {

this.birthDate = birthDate;

}

boolean isBornBoomer() {

Date startDate = Date.valueOf("1946");

Date endDate = Date.valueOf("1964");

return birthDate.compareTo(startDate)>=0 && birthDate.compareTo(endDate) < 0;

}

}

isBornBoomer是用来这个人是否是1946-1964年出生的，而每次isBornBoomer被调用的时候，都会生成startDate和birthDate两个对象，造成了空间浪费，如果改成这样效率会更好：

class Person{

private Date birthDate;

private static Date startDate,endDate;

**static{**

**startDate = Date.valueOf("1946");**

**endDate = Date.valueOf("1964");**

**}**

public Person(Date birthDate) {

this.birthDate = birthDate;

}

boolean isBornBoomer() {

return birthDate.compareTo(startDate)>=0 && birthDate.compareTo(endDate) < 0;

}

}

因此，很多时候会将一些只需要进行一次的初始化操作都放在static代码块中进行。

1. 思考性实验

（1）不等长二维数组的使用。

建立类YanghuiClass，试用不等长二维数组打印如下所示的出杨辉三角形（注意缩进“\t”），试用Scanner类接受控制台输入的一个整型来控制需要打印的杨辉三角形的行数。

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

（2）对象数组

编写一个学生类Student，包含的属性有学号、姓名、年龄，创建n个学生实例，将所有学生存在一个数组中，自己定义数据，并实现如下操作：

* 将所有学生年龄增加1
* 按照数组中顺序显示所有学生的信息
* 查找显示所有年龄大于20岁的学生名单。
* （可选）按照学生年龄进行排序，并输出（联合java.util.Arrays和Comparable/Comparator接口）

（3）使用static 变量进行对象间信息共享

建立一个Book类，包括title，salenumber和gensalenumber三个成员变量，分别表示书名，一本书的出售量，以及所有书的出售量。再设计一个sale方法，使得每卖出一本书，相应的出售量有所增加。

建立一个BookTestStaticVarial类，在main函数中模拟有3种书，假设一共卖出书籍n次(n接受用户输入)，统计每种书的销售量和总销售量。

（4）创建链表

创建如下图所示的链表，每个结点定义如下：

class Node{

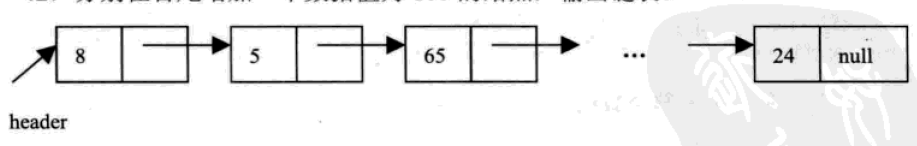
int data;

Node next;

}

创建一个初始为10个结点的单向链表，结点数据data可以用随机函数产生Math.Random()，要求

* 输出该链表
* 分别在首尾增加一个数据值为100的结点，输出链表



四、实验结果和分析

请按照“实验内容”部分，依次在此位置贴上：运行结果、分析说明、核心代码

五、实验小结