

认真阅读此文，有助于你产生使用类和对象解决问题的思路

此文以周末电影院的作为基础

# 1 通过周末作业梳理【类和对象】

第一：你知道需要存储电影名称，导演，价格等数据

- 请问：怎么存储这些数据
- 回答：这些数据是每个电影私有的，需要在**电影对象**中存储，所以要创建电影对象

第二：顺着第一个问题继续问

- 请问：**电影对象怎么创建**
- 回答：对象是**类**实例化出来的，所以需要先**创建电影类**

第三：顺着第二个问题继续问

- 请问：电影类是怎么创建的
- 回答：
  - 将电影**对象私有的数据**设计成类的**实例属性**
  - 将所有电影对象**共性的数据**设置成**静态属性**
  - 方法的设计同理
- 所以你就想到了，要设计Movie类，表示电影类

```
1 public class Movie {
2     private String name; //属性私有化
3     private double price; //属性私有化
4
5     // 构造函数初始化实例属性值
6     public Movie(String name, double price){
7         this.name = name;
8         this.price = price;
9     }
10
11     // get方法，读取电影名称
12     public String getName() {
13         return name;
```

```

14     }
15
16     // get方法, 读取电影价格
17     public double getPrice() {
18         return price;
19     }
20
21     // set方法, 修改电影价格
22     public void setPrice(double price) {
23         this.price = price;
24     }
25
26     // set方法, 修改电影名称
27     public void setName(String name) {
28         this.name = name;
29     }
30 }

```

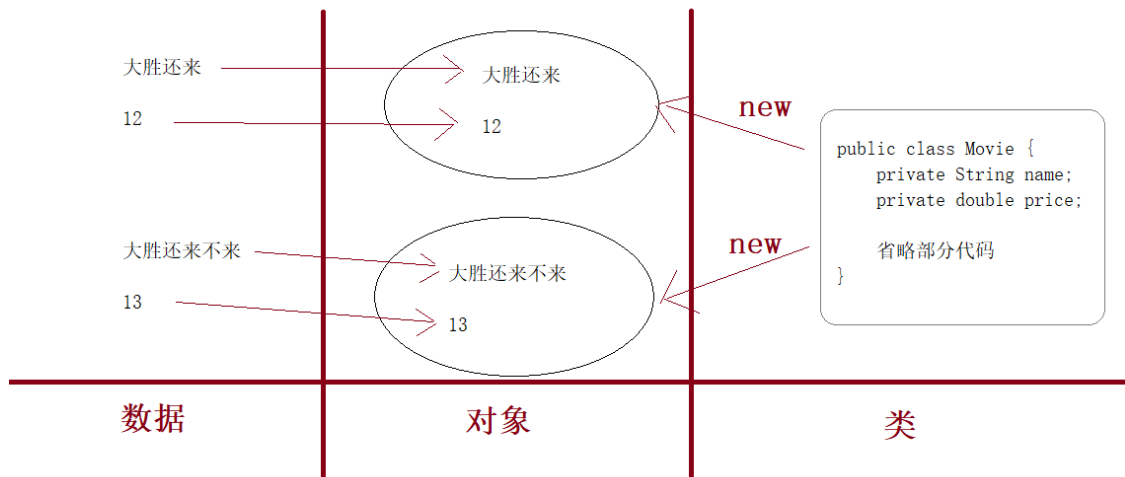
- 然后实例化Movie类, 就产生了电影对象

```

1 Movie movie1 = new Movie("大胜归来",12);
2 Movie movie2 = new Movie("大胜还来不来",13);

```

- 在给你画一个图示, 理解从数据 > 对象 > 类的思考过程



#### 第四: 小结

- 要存储数据, 先创建对象, 因为数据是存储在对象中的
  - 对象的私有数据存储在对象实例属性中
  - 对象的共性数据存储在类变量中 (静态属性)

- 要创建对象，先定义类，因为对象是类实例化出来的

## 2 通过周末作业梳理【对象数组】

第一：多个对象怎么管理

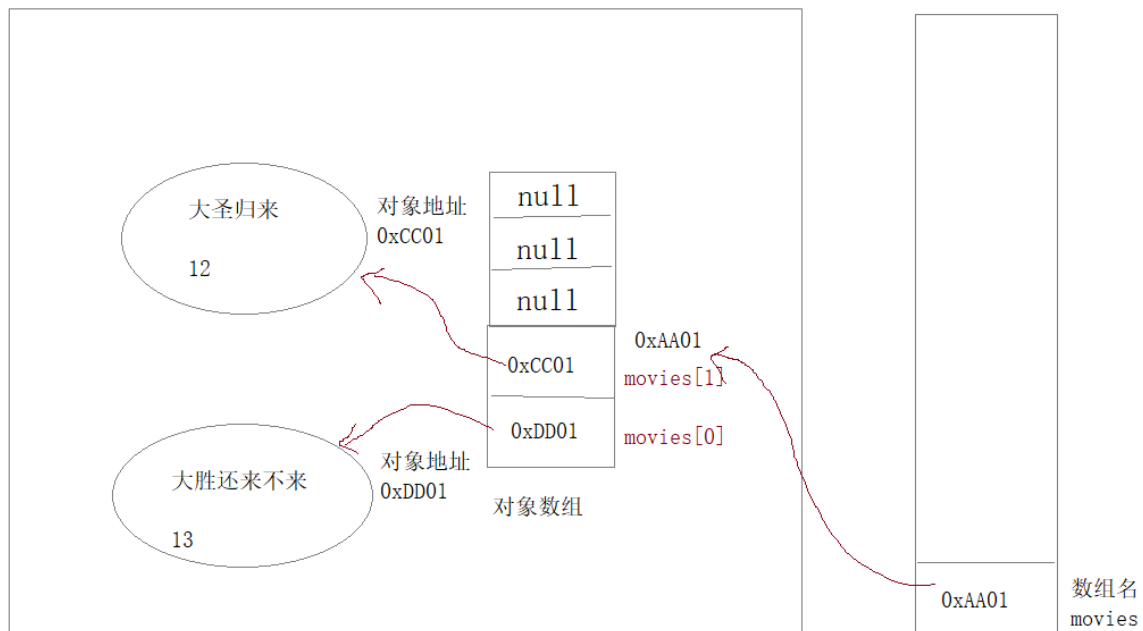
- 请问：现在有了多个电影对象，这些对象彼此独立，如何存储能更方便的操作这些对象？
- 回答：将电影对象放到电影数组中，通过数组方便操作电影对象

第二：接着第一个问题继续问

- 请问：电影数组定义在哪个类中？
- 回答：电影数组中存储了多部电影，多部电影属于电影院中的数据，因此应该定义电影院类，在电影院类中定义电影数组。电影院类应该作为程序运行的入口，也就是测试类。

```
1 public class Cinema {
2     private Movie movies[] = null;
3
4     public Cinema(int capacity){
5         movies = new Movie[capacity];
6     }
7
8     public static void main(String[] args) {
9         Cinema wanda = new Cinema(5);
10        wanda.movies[0] = new Movie("大胜归来",12);
11        wanda.movies[1] = new Movie("大胜还来不来",13);
12    }
13 }
```

第三：在给你画一个图示，理解对象数组，注意是对象数组，意思是数组的元素的一个对象



### 3 扩展梳理

以上梳理了数据存储的过程。根据以上的梳理，你自己思考一下，方法应该怎么设计。

小贴士：

- 数据存储在堆内存中
- 程序运行在栈内存中（栈帧，入栈，出栈）