

# 36从零开始学Java36之方法的重载是咋回事

前言

配套开源项目资料

一. 方法重载

1. 概念

2. 要求

二. 代码案例

1. 普通方法重载

2. 构造方法重载

三. 结语

四. 配套视频

五. 今日作业

1. 第一题

作者：孙玉昌，昵称【**一一哥**】，另外【**壹壹哥**】也是我哦

千锋教育高级教研员、CSDN博客专家、万粉博主、阿里云专家博主、掘金优质作者

## 前言

壹哥在之前给大家讲解构造方法的时候说过，在一个类中，可以定义多个构造方法，这叫做方法的重载！但是关于方法重载，具有有哪些要求和细节？很多小白还并不清楚，所以在今天的这篇文章中，壹哥给大家详细说说方法重载相关的内容。

-----前戏已做完，精彩即开始-----

全文大约【1800】字，不说废话，只讲可以让你学到技术、明白原理的纯干货！本文带有丰富的案例及配图视频，让你更好地理解 and 运用文中的技术概念，并可以给你带来具有足够启迪的思考.....

# 配套开源项目资料

Github:

## GitHub – SunLtd/LearnJava

Contribute to SunLtd/LearnJava development by creating an account on GitHub.

GitHub

Gitee:



## 一一哥/从零开始学Java

从零开始学Java系列 稀土掘金专栏地址: <https://juejin.cn/column/7175082165548351546> CSDN专...

Gitee

## 一. 方法重载

### 1. 概念

所谓方法的重(chong)载(重复出现), 就是指在**一个类中可以有多个名称相同但参数不同的方法**, 不过方法重载与返回值的类型无关。在Java中, **无论是普通方法, 还是构造方法, 都可以进行重载**。我们在调用重载的方法时, 会根据传递的参数不同, 来执行不同的方法。

通过方法的重载, 我们就可以把一系列功能相似, 但参数不同的方法, 用同一个方法名进行定义。这样就避免了出现繁多的方法名, 如果我们每次新建一个方法, 就取个新的方法名, 就会降低程序的可读性。使用同一个方法名称, 我们就很容易记住这些方法, 使用起来就很方便灵活。

### 2. 要求

根据方法重载的概念可知, 方法重载时有如下要求:

1. **同名不同参(参数类型、个数、顺序);**
2. **与返回值、访问修饰符无关。**

也就是说, 方法重载时, **要求方法的名称相同, 但参数不同, 只要参数的类型、个数、顺序有一个地方不同即可**。但方法的重载与返回值和使用的访问修饰符没有关系!

我们可以自己进行重载方法的定义，当然java源码中也有大量的重载方法，比如我们常见的String类中就有，如下图所示：

```
String str="hello";  
str.indexOf
```

- **indexOf**(int ch) : int - String
- **indexOf**(String str) : int - String
- **indexOf**(int ch, int fromIndex) : int - String
- **indexOf**(String str, int fromIndex) : int - String
- **lastIndexOf**(int ch) : int - String
- **lastIndexOf**(String str) : int - String
- **lastIndexOf**(int ch, int fromIndex) : int - String
- **lastIndexOf**(String str, int fromIndex) : int - String

比如String字符串中就定义了多个indexOf()和lastIndexOf()重载方法，当然也有其他的重载方法。这些方法名称相同，但参数肯定不同，和返回值没有关系！

说了这么多之后，接下来还是让我们通过一些案例看看方法的重载怎么实现吧。

## 二. 代码案例

### 1. 普通方法重载

接下来我们设计一个案例，用于比较两个数字的大小。我们在比较两个数值时，数值的类型是不固定的，可能是两个int类型的数值，也可能是两个double类型的数值，也可能是两个double、一个int类型的数值.....在这种情况下，我们就可以使用方法的重载来实现数值之间的比较。具体实现代码参考如下：

```
1  /**
2   * @author 一一哥Sun
3   * QQ: 2312119590
4   * CSDN、掘金、知乎找我哦
5   */
6  public class Demo01 {
7
8      public static void main(String[] args) {
9
10         Demo01 demo=new Demo01();
11         int max01 = demo.max(10, 20);
12         System.out.println("较大的整数是: "+max01);
13
14         float max02 = demo.max(20.0f, 39.8f);
15         System.out.println("较大的单精度数是: "+max02);
16
17         double max03 = demo.max(22.88, 10.88);
18         System.out.println("较大的双精度数是: "+max03);
19
20         double max04 = demo.max(44.88, 5);
21         System.out.println("较大的双精度数是: "+max04);
22     }
23
24     // 输出较大的那个整数
25     public int max(int a, int b) {
26         // 采用三目运算符进行简化判断
27         return a >= b ? a : b;
28     }
29
30     // 输出较大的那个单精度数
31     public float max(float a, float b) {
32         // 采用三目运算符进行简化判断
33         return a >= b ? a : b;
34     }
35
36     // 输出较大的那个单精度数
37     public double max(double a, double b) {
38         // 采用三目运算符进行简化判断
39         return a >= b ? a : b;
40     }
41
42     //定义一个私有的方法，该方法的两个参数类型不同，这同样也属于方法重载。
43     private double max(double a, int b) {
44         // 采用三目运算符进行简化判断
45     }
```

```
46         return a >= b ? a : b;  
47     }  
48  
49 }
```

以上代码，编译、运行都是完全没问题的。虽然3个max()方法名称相同，但它们的形参列表不同，所以系统可以正常区分出这3个方法。在运行时，Java虚拟机会根据传递过来的不同参数来调用不同的方法。

## 2. 构造方法重载

除了普通方法可以重载之外，构造方法也可以重载。但是我们要注意，构造方法没有返回值。

```
1  /**
2   * @author 一一哥Sun
3   * QQ: 2312119590
4   * CSDN、掘金、知乎找我哦
5   */
6  public class Car {
7
8      // 汽车品牌
9      public String logo;
10
11     // 汽车颜色
12     private String color;
13
14     // 汽车价格
15     private double price;
16
17     // 无参构造方法
18     public Car() {
19         System.out.println("汽车类...");
20     }
21
22     // 有参构造方法
23     public Car(String logo) {
24         System.out.println("汽车类...logo=" + logo);
25     }
26
27     // 有参构造方法
28     public Car(String logo, String color) {
29         System.out.println("汽车类...logo=" + logo + ",color=" + color);
30     }
31
32     // 有参构造方法
33     public Car(String logo, String color, double price) {
34         System.out.println("汽车类...logo=" + logo + ",color=" + color + ",
35         price=" + price);
36     }
37 }
```

我们在构造Car汽车类对象时，可以如下所示：

```
1 public class Demo02 {  
2  
3     public static void main(String[] args) {  
4  
5         Car car1 = new Car();  
6  
7         Car car2 = new Car("奥迪A6");  
8  
9         Car car3 = new Car("奥迪A6", "黑色");  
10  
11        Car car4 = new Car("奥迪A6", "黑色", 50.0);  
12    }  
13  
14 }
```

这样通过传入不同的参数，就可以构造出不同的汽车对象。

-----正片已结束，来根事后烟-----

## 三. 结语

至此，壹哥就把方法重载相关的内容讲解完毕了，现在你知道哪些情况属于方法重载了吗？方法重载要点如下：

- 方法重载是指同一类中有多个方法的名称相同，但参数不同；
- 重载的方法功能都类似，比如String的indexOf()方法；
- 重载方法的返回值类型尽量要相同。

另外如果你独自学习觉得有很多困难，可以加入壹哥的学习互助群，大家一起交流学习。

## 四. 配套视频

如果你不习惯阅读技术文章，或是对文中的技术概念不能很好地理解，可以来看看壹哥帮你筛选出的视频教程。与本文配套的Java学习视频，链接如下：

<https://player.bilibili.com/player.html?bvid=BV1Ja411x7XB&p=53&page=53>

## 五. 今日作业

### 1. 第一题

两个方法的返回值不同，方法名和参数等其他都相同，这算不算方法重载？