

## 自动装箱

笔记本: java基础

创建时间: 2019/5/4 10:02

更新时间: 2019/5/4 10:08

作者: ANNER

URL: <https://blog.csdn.net/chenliguan/article/details/53888018>

---

# Java两种数据类型

- 基本数据类型，分为boolean、byte、int、char、long、short、double、float;
- 引用数据类型，分为数组、类、接口。

为了编程的方便还是引入了基本数据类型，但是为了能够将这些基本数据类型当成对象操作，**Java为每一个基本数据类型都引入了对应的包装类型（wrapper class）**，int的包装类就是Integer，从Java 5开始引入了自动装箱/拆箱机制，使得二者可以相互转换。

## 转换机制

1.自动装箱：将基本数据类型重新转化为对象

```
public class Base2Object {  
    public static void main(String[] args) {  
        Integer num=1;  
        System.out.println(num.toString());  
    }  
}
```

1是属于基本数据类型的，原则上它是不能直接赋值给一个对象Integer的，但jdk1.5后你就可以进行这样的声明。自动将基本数据类型转化为对应的封装类型，成为一个对象以后就可以调用对象所声明的所有的方法。

2. 自动拆箱：将对象重新转化为基本数据类型

因为对象时不能直接进行运算的，而是要转化为基本数据类型后才能进行加减乘除。对

比:

```
public class Base20Object {  
    public static void main(String[] args) {  
        Integer num=1;  
        System.out.println(num.toString());  
        Integer num2=new Integer( value: 10);  
        System.out.println(++num2);  
    }  
}
```

java对于Integer与int的自动装箱与拆箱的设计，是一种模式：叫享元模式（flyweight）。加大对简单数字的重利用，**Java定义在自动装箱时对于值从-128到127之间的值，它们被装箱为Integer对象后，会存在内存中被重用，始终只存在一个对象。**而如果超过了从-128到127之间的值，被装箱后的Integer对象并不会被重用，即相当于每次装箱时都新建一个 Integer对象。