



链滴

静态代码块执行顺序

作者: [lyn](#)

原文链接: <https://ld246.com/article/1497791441933>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

执行顺序：静态代码块 > 构造代码块 > 构造方法

理由：静态代码块（`static{}`）在类加载的时候执行一次。

构造代码块（`{}`内的部分）在每一次创建对象时执行，始终在构造方法前执行。

构造方法在新建对象时调用（就是`new`的时候）。

注意：a.静态代码块在类加载的时候就执行，所以它的优先级高于入口`main()`方法。 b.当三种形式不一次出现，同优先级是按照先后顺序执行。

带继承的例子

```
public class HelloB extends HelloA {  
  
    public HelloB(){  
        System.out.println("B的构造方法");  
    }  
    {  
        System.out.println("B的构造代码块");  
    }  
    static{  
        System.out.println("B的静态代码块");  
    }  
    //public static HelloB hB = new HelloB();  
    public static void main(String[] args){  
        new HelloB();//调用B的构造方法  
    }  
}  
  
class HelloA{  
    public HelloA(){  
        System.out.println("A的构造方法");  
    }  
    {  
        System.out.println("A的构造代码块");  
    }  
    static{  
        System.out.println("A的静态代码块");  
    }  
}
```

输出结果为：

A的静态代码块

B的静态代码块

A的构造代码块

A的构造方法

B的构造代码块

B的构造方法

结论

- a.父类始终先调用（继承先调用父类），并且这三者之间的相对顺序始终保持不变
- b.因为静态代码块在类加载时执行，所以先输出的是父类和子类的静态代码块
- c.调用B的构造方法创建对象时，构造块和构造方法会一起按顺序执行，还是父类的先调用