

Java对象和类疑问解答

一个类必须有构造器吗？

非必须

没有明确写构造器，Java会自动加一个无参构造器

构造器=构造方法

构造器 (Constructor) 一般用来进行对象的初始化工作

给成员变量赋初值 (比如传递参数进来设置name、age等)
进行对象所需的其他初始化逻辑 (如分配资源、执行检测等)
也可什么都不写 (空的构造器)

构造器里可以写复杂逻辑

最常见的写法 (赋初值)

```
public class Dog {
    String name;
    int age;

    // 构造器：用参数初始化成员变量
    public Dog(String name, int age) {
        this.name = name; // this表示当前对象
        this.age = age;
    }

    public Dog(String name, int age) {
        if (age < 0) {
            throw new IllegalArgumentException("年龄不能为负数");
        }
        this.name = name;
        this.age = age;
        // 可以加：如链接数据库、读配置文件等初始化动作
    }

    public class Dog {
        String name;
        int age;

        public Dog() {
            this.name = "无名";
            this.age = 0;
        }
    }
}
```

无参构造器 (给默认值)

构造器 (Constructor)：用来创建对象并初始化对象的成员变量 (属性)，只在new时执行一次。

构造器名字和类名必须完全一致，没有返回值 (连void都不能写)

对比普通方法

不能和类名完全相同且没有返回类型 (就是说普通方法必须有返回类型)

```
public void name() {
    // 可以实现自己的功能
}
```

普通方法用来实现对象的具体动作 (比如set/get、打印、计算等)，可以反复调用。

普通方法是对对象创建好后，调用来完成某些具体功能的

理解构造器和方法？

与set/get方法和普通方法 (Method) 对比

set和get方法 (不是构造器)，用来赋值和取值

```
private String name;

public void setName(String name) { // 方法
    this.name = name;
}

public String getName() { // 方法
    return this.name;
}
```

Java约定 (Bean规范)：set/get后面的第一个字母大写 (即使变量名是小写)；驼峰命名法 (camelCase)，是规范，不是强制

```
public void setName(String name) { ... }
public String getName() { ... }
```

用于安全地设置/获取属性值，有命名规范
主要用于封装，外部不能直接访问private属性，只能通过set/get间接访问或修改

普通方法可以有任何名字，功能不限于属性读写
用来实现实际功能，比如计算、打印、操作等名称不需要像set/get那样有固定格式

实际代码对比

```
public class Student {
    private String name; // 私有属性

    public void setName(String name) { // 方法：设置属性
        this.name = name;
    }

    public String getName() { // 方法：获取属性
        return this.name;
    }
}

public void study() {
    System.out.println(name + "在学习Java! ");
}

public class Book {
    private String title;
    private int price;

    // set/get方法
    public void setTitle(String title) { this.title = title; }
    public String getTitle() { return this.title; }
    public void setPrice(int price) { this.price = price; }
    public int getPrice() { return this.price; }

    // 普通方法
    public void showInfo() {
        System.out.println("书名: " + title + ", 价格: " + price);
    }
}
```

```
public class Person {
    // 这是构造器
    public Person() {
        // 构造的时候执行
    }

    // 这是普通方法
    public void sayHello() {
        System.out.println("Hello!");
    }
}
```