

面向过程：当需要实现某功能时，每一个具体步骤聚需亲力亲为，详细处理每个细节

面向对象：当需要实现某功能时，不关心具体的步骤聚，只需找已具有该功能的对象来实现

面向对象的语言中，有三大基本特征：封装、继承、多态

## 1.2 类和对象

环顾周围，你会发现很多对象，比如桌子，椅子，同学，老师等。桌椅属于办公用品，师生都是人类。那么什么是类呢？什么是对象呢？

### 什么是类

- **类**：是一组相关**属性**和**行为**的集合。可以看成是一类事物的模板，使用事物的属性特征和行为特征来描述该类事物。

现实中，描述一类事物：

- **属性**：就是该事物的状态信息。
- **行为**：就是该事物能够做什么。

举例：小猫。

属性：名字、体重、年龄、颜色。 行为：走、跑、叫。

### 什么是对象

- **对象**：是一类事物的具体体现。对象是类的一个**实例**（对象并不是找个女朋友），必然具备该类事物的属性和行为。

现实中，一类事物的一个实例：一只小猫。

举例：一只小猫。

属性：tom、5kg、2 years、yellow。 行为：溜墙根走、蹦跹的跑、喵喵叫。

### 类与对象的关系

- 类是对一类事物的描述，是**抽象的**。
- 对象是一类事物的实例，是**具体的**。
- **类是对象的模板，对象是类的实体。**

# 1.3 类的定义

## 事物与类的对比

现实世界的一类事物：

**属性：**事物的状态信息。 **行为：**事物能够做什么。

Java中用class描述事物也是如此：

**成员变量：**对应事物的**属性** **成员方法：**对应事物的**行为**

## 类的定义格式

```
public class ClassName {  
    //成员变量  
    //成员方法  
}
```

- **定义类：**就是定义类的成员，包括**成员变量**和**成员方法**。
- **成员变量：**和以前定义变量几乎是一样的。只不过位置发生了改变。**在类中，方法外。**
- **成员方法：**和以前定义方法几乎是一样的。只不过**把static去掉**，static的作用在面向对象后面课程中再详细讲解。  
*不一定要参数和返回值*

类的定义格式举例：

```
public class Student {  
    //成员变量  
    String name; //姓名  
    int age; //年龄  
    //成员方法  
    //学习的方法  
    public void study() {  
        System.out.println("好好学习，天天向上");  
    }  
  
    //吃饭的方法  
    public void eat() {  
        System.out.println("学习饿了要吃饭");  
    }  
}
```



# 1.4 对象的使用

## 对象的使用格式

创建对象：

```
类名 对象名 = new 类名();
```

使用对象访问类中的成员：

```
对象名.成员变量;  
对象名.成员方法();
```

对象的使用格式举例：

```
public class Test01_Student {  
    public static void main(String[] args) {  
        //创建对象格式：类名 对象名 = new 类名();  
        Student s = new Student();  
        System.out.println("s:"+s); //cn.itcast.Student@100363  
  
        //直接输出成员变量值  
        System.out.println("姓名: "+s.name); //null  
        System.out.println("年龄: "+s.age); //0  
        System.out.println("-----");  
  
        //给成员变量赋值  
        s.name = "赵丽颖";  
        s.age = 18;  
  
        //再次输出成员变量的值  
        System.out.println("姓名: "+s.name); //赵丽颖  
        System.out.println("年龄: "+s.age); //18  
        System.out.println("-----");  
  
        //调用成员方法  
        s.study(); // "好好学习，天天向上"  
        s.eat(); // "学习饿了要吃饭"  
    }  
}
```

## 成员变量的默认值

	数据类型	默认值
基本类型	整数 (byte, short, int, long)	0
	浮点数 (float, double)	0.0
	字符 (char)	'\u0000'
	布尔 (boolean)	false
引用类型	数组，类，接口	null