

Java-04

一、Task1.对象和类

1.

```
public class Person {

    private String name;
    private int age;
    private int sex;

    public Person(String name, int age, int sex) {
        this.name = name;
        this.age = age;
        this.sex = sex;
    }

    public Person(Person a){
        this.name = a.name;
        this.age = a.age;
        this.sex = a.sex;
    }

    private void eat() {
        System.out.println(name+"正在吃东西");
    }

    private void sleep() {
        System.out.println(name+"正在睡觉");
    }

    private void dadoudou() {
        System.out.println(name+"正在打豆豆");
    }

    public void putout() {
        System.out.println(name);
        System.out.println(age);
        System.out.println(sex);
    }

}
```

代码如下

this的作用: 区分 局部变量 和 实例变量,

指代实例变量(即此处要设置的那个变量)

2. 创建对象(顺带演示一问中的复制)

```
public static void main(String[] args) {  
    Person a = new Person("xiaoming",18,1);  
    Person b = new Person(a);  
    a.putout();  
    b.putout();  
}
```

对象和类的关系: 对象是类的具体实例(一个类可以有多个对象),

类是对象的属性, 定义了对象拥有的属性

3. 访问修饰符:

- **默认** 只能被**相同包**下的类访问
- **private** 只能在**当前类内部**被访问(且不能修饰类和接口)
- **public** 可以被所有类访问
- **protected** 可以被**相同包下的类**或**不同包中子类**访问
(A extends B 表示A为B的子类)

二、Task2.static

- **静态变量** 也称为 **类变量**, 属于类而非类的实际变量
被类的所有实例共享, 可以被 **public private**,或 **protected** 等修饰符修饰
- **静态方法** 属于类而非类的实际方法, 无需创建实例就可调用
直接使用 **classname.methodname()** 调用方法
值得注意的是 静态方法 中不能使用 **this** 关键字