

Java05

Task1

我创建了两个类

- 主类Main

```
package haha;

public class Main{

    public static void main(String[] args) {
        //我这里选择用构造函数
        Person zhangsan = new Person("张三",18,1);
        //题目给的person类成员函数(Java中好像叫方法 这里我个人习惯把它称为函数了)都是private
        //这边为了方便下面调用展示我就把函数的private去掉了
        zhangsan.eat();
        zhangsan.sleep();
        zhangsan.dadoudou();
    }

}
```

- Person类

```

package haha;

public class Person{
    private String name;
    private int age;
    private int sex;

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

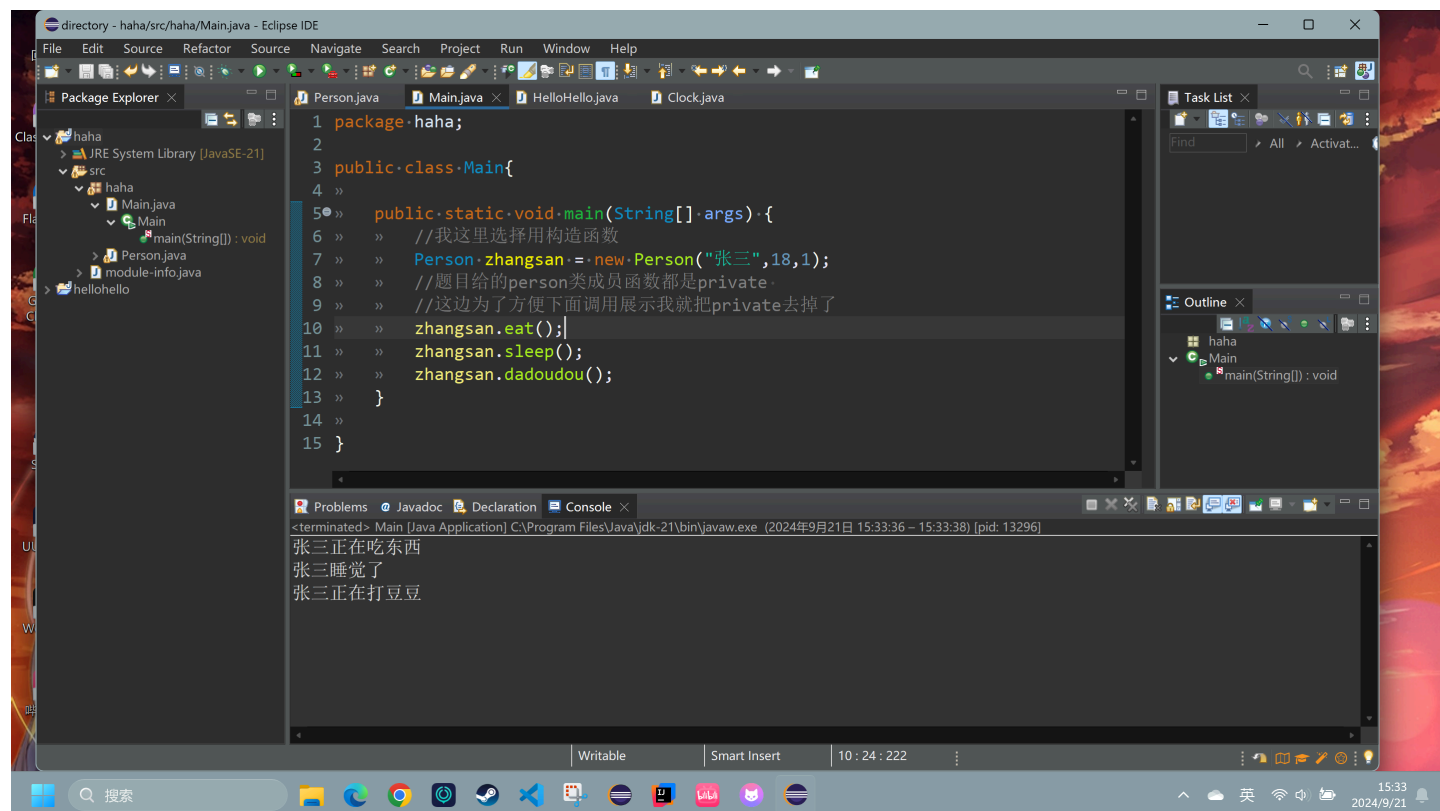
    void sleep() {
        System.out.println(name+"睡觉了");
    }

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

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

```

如图 成功运行



1. 在创建对象时构造函数会被自动调用 其中的this指的就是该对象 `this.name = name`就是把()内的字符串赋值给该对象的name
2. 类是一类事物 它规定了这类事物应该具有哪些属性哪些变量 而对象就是这类事物中的一个 对象具有类规定的各种变量和属性 不同对象变量和属性的具体值不一样

举个简单例子就是如果类是人 那么对象就是一个个的个体 人都有名字 那么对于两个个体 虽然他们都有名字 但一个是张三 一个是李四

3.

- public:在任何地方都可以访问
- private:只能在当前这个类的{}里面能访问
- friendly:即不加限定词 在同一个package里的类都可以访问 不包括子类
- protected:和friendly一样 但是包括子类
-
- 对1的补充 题干中貌似要求实现复制对象吗??这里简单查阅了一下 如下

只需要再额外创建一个构造函数就可以了 试了一下一个类可以有多个构造函数 只要后面参数不一样就可以

```
package haha;

public class Person{
    private String name;
    int age;
    private int sex;

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

    void sleep() {
        System.out.println(name+"睡觉了");
    }

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

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

    Person(Person origin){
        this.name = origin.name;
        this.age = origin.age;
        this.sex = origin.sex;
    }
}
```

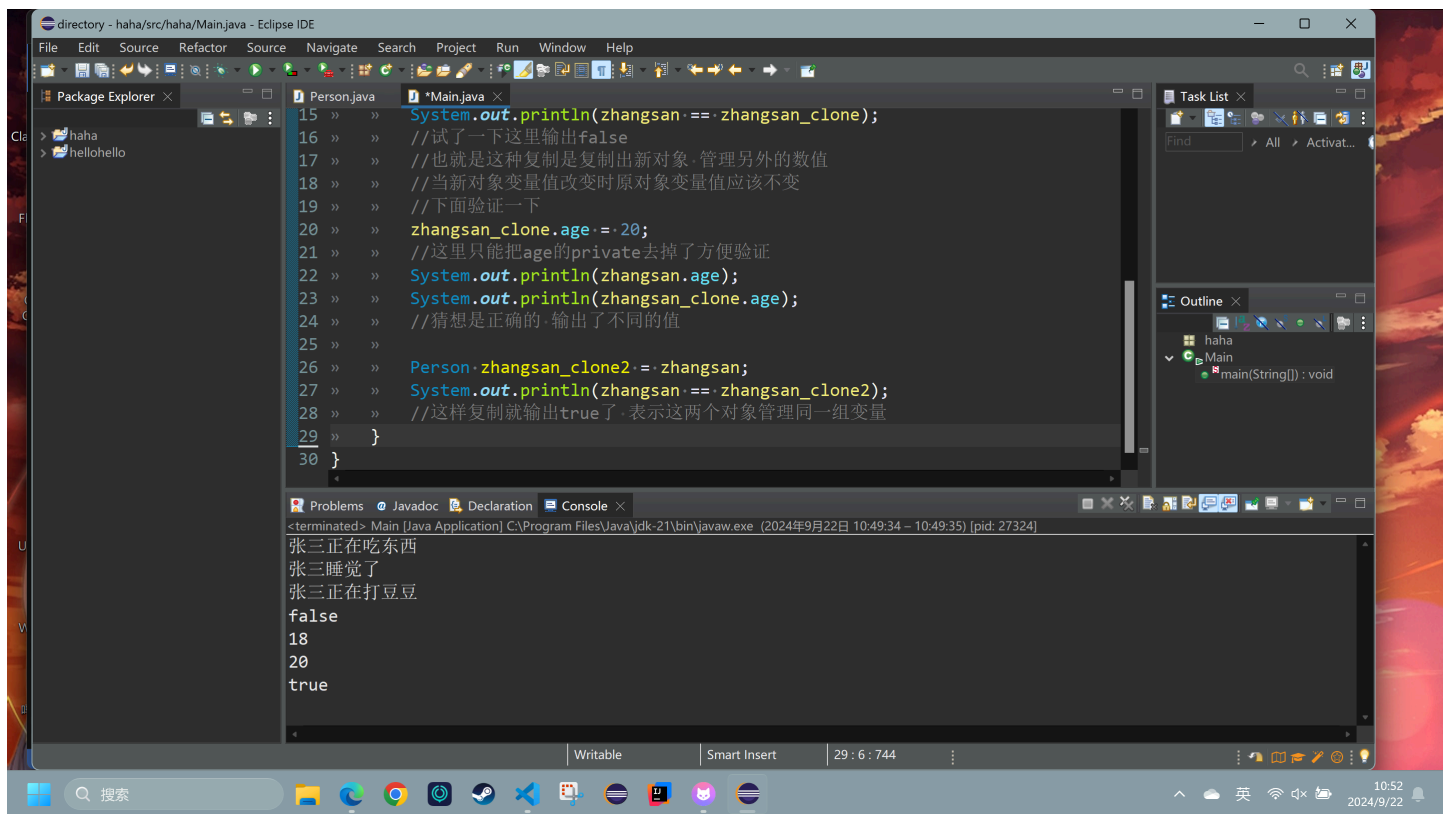
```
package haha;

public class Main{

    public static void main(String[] args) {
        //我这里选择用构造函数
        Person zhangsan = new Person("张三",18,1);
        //题目给的person类成员函数都是private
        //这边为了方便下面调用展示我就把private去掉了
        zhangsan.eat();
        zhangsan.sleep();
        zhangsan.dadoudou();

        Person zhangsan_clone = new Person(zhangsan);
        System.out.println(zhangsan == zhangsan_clone);
        //试了一下这里输出false
        //也就是这种复制是复制出新对象 管理另外的数值
        //当新对象变量值改变时原对象变量值应该不变
        //下面验证一下
        zhangsan_clone.age = 20;
        //理论上成员变量应该都是private会比较安全 这里为了方便验证把age的private去掉了
        System.out.println(zhangsan.age);
        System.out.println(zhangsan_clone.age);
        //猜想是正确的 输出了不同的值

        Person zhangsan_clone2 = zhangsan;
        System.out.println(zhangsan == zhangsan_clone2);
        //这样复制就输出true了 表示这两个对象管理同一组变量
    }
}
```



Task2

4. "count方法不应该统计到已经被销毁的Person对象"这个真难住我了 对象可以手动赋值为null 但这并不意味着对象被销毁 java有垃圾自动回收系统 你的对象为null也不会立刻被销毁 不过我这里就姑且认为null就是销毁吧
- 要实现这个功能 原想法是利用计数器cnt 当对象被创建或者复制时cnt++ 设置函数dispose 让对象为null并cnt--

```
void dispose(Person a) {
    a = null;
    cnt--;
}
```

最后非常傻的结果就是对象只能是值传递 改变a结果main里面的对象并没有变化

又试了一下this = null这种语法是错误的

后面试了对象数组 或者用System.gc() 但这个也不是强制消除的

最后也没找到什么合适的方案 或者直接在main里面把对象赋值为null后手动cnt--??感觉这不是我想要的效果 试了很多方式 最后了解到ArrayList 能让我把对象从列表中移除...虽然不是真正意义上的销毁 但我实在找不到替代方案了 求助

- Person类

```
package haha;

import java.util.ArrayList;

public class Person{
    private String name;
    int age;
    private int sex;

    static ArrayList<Person> people = new ArrayList<Person>();
    //在这里创建people

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

    void sleep() {
        System.out.println(name+"睡觉了");
    }

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

    Person(String name,int age,int sex){
        this.name = name;
        this.age = age;
        this.sex = sex;
        people.add(this);
        //这里我开始还担心能不能用this添加进去 测试后是可以的
    }

    Person(Person origin){
        this.name = origin.name;
        this.age = origin.age;
        this.sex = origin.sex;
        people.add(this);
    }

    void dispose() {
        people.remove(this);
    }

    static int count() {
        return people.size();
    }
}
```

```
}
```

- 主类

```
package haha;

public class Main{
    public static void main(String[] args) {
        //我这里选择用构造函数
        Person zhangsan = new Person("张三",18,1);
        //题目给的person类成员函数都是private
        //这边为了方便下面调用展示我就把private去掉了
        zhangsan.eat();
        zhangsan.sleep();
        zhangsan.dadoudou();

        Person zhangsan_clone = new Person(zhangsan);

        System.out.println("目前对象数为"+Person.count());
        //输出结果为2
        zhangsan.dispose();
        System.out.println("目前对象数为"+Person.count());
        //输出结果为1
    }
}
```