

如何创建对象

1> 先声明再创建

Cat cat; // 声明对象 cat \leftrightarrow 在栈中有 cat 类 内容为 [null]

cat = new Cat(); // 创建 \rightarrow 在堆中开辟对应空间, 将地址填在栈中的内容.

2> 直接创建

Cat cat = new Cat();

如何访问属性

对象名.属性名; 例: cat.name = "小白";

类和对象的内存分配机制 ☆

(base) MacBook-Pro:0508 songfei\$ java Object04

小明

200

Exception in thread "main" java.lang.NullPointerException

at Object04.main(Object04.java:16)

(base) MacBook-Pro:0508 songfei\$

```
Object04.java
1 public class Object04{
2     // 编写一个main方法
3     public static void main(String[] args){
4         /* 查看以下输出内容
5         */
6
7         Person a = new Person();
8         a.age = 10;
9         a.name = "小明";
10        Person b;
11        b = a;
12        System.out.println(b.name);
13        b.age = 200;
14        b = null;
15        // b的地址为null, 即栈b中的地址与堆中的对象空间之间的联系断了, 所以访问出现问题;
16        // 但是这属于异常, 编译仍会通过
17        System.out.println(a.age);
18        System.out.println(b.name);
19    }
20 }
```

2021.5.9

P202 - P Java - 成员方法

成员方法, 简称方法, 人类除了有一些属性外, 还有一些行为 (说话, 跑步, 学习...)

```
Person p1 = new Person();
int returnRes = p1.getSum(10, 20);
System.out.println("getSum的值为" + returnRes);
```

```
public int cal01(int num1, int num2){
    return num1 + num2;
}
```

方法调用小信

1> 当程序执行到方法时, 就会开辟一个独立的空间 (栈空间)

2> 当方法执行完毕, return语句返回调用方法执行的地方

3> 返回后, 继续执行后面的代码, 此时方法的独立空间被销毁, 空间释放

JVM 内存 [方法的调用机制分析]

栈

getSum 栈 (独立空间)

num1 \rightarrow 10

num2 \rightarrow 20

return 10+20 (30)

main 栈

Person p1 = new...

int returnRes = p1.getSum(10, 20)

值 30 返回给 returnRes

堆

对象

方法区

成员方法的好处

- 1> 提高代码的复用性
- 2> 可以将实现的细节封装起来, 然后供其他用户来调用即可.

方法的定义

访问修饰符
(控制方法使用范围)

```
public 返回数据类型 方法名(形参列表) { // 方法体  
    语句;  
    return 返回值; // 不是必须的  
}
```

成员方法的使用细节.