

类的基本定义和声明: 类和属性的讲解使用代码注释。

```
public class Person {  
    String name; // 属性变量可以声明, 不用初始化, 具有默认值。  
                // String 的默认值是 null.  
    int age; // 年龄, int 的默认值是 0  
    public void showName() {  
        System.out.println("姓名:" + name);  
    }  
    public int getAge() {  
        return age;  
    }  
}
```

类的实例化:

```
public class Test4 {  
    public static void main(String[] args) {  
        Person person = new Person();  
        person.name = "张三";  
        person.showName();  
        System.out.println(Person.getAge());  
    }  
}
```

属性的详细讲解:

修饰符

类型

属性名 = 初值;

public

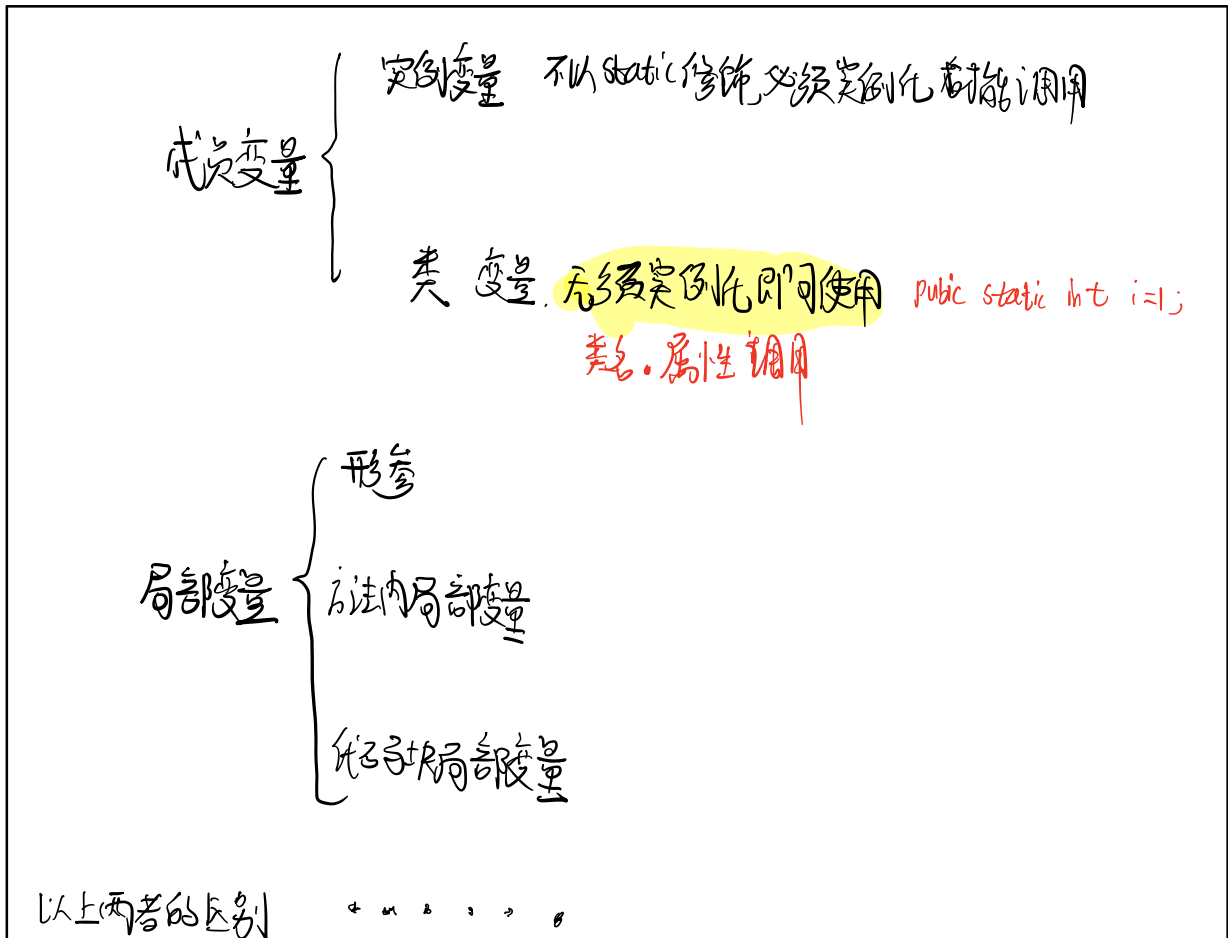
类

可以不初始化,
有默认值

private 只能为键的
方法访问

或
基本类型

补充：变量的类型



方法的详细讲解:

语法:

修饰符 返回值类型 方法名(参数) {
 方法体语句;
}

→ public
protected
private

!!! 定义变量时通过类, 变量改变;

没有是也通过类, 可以

类变量是每个对象共有! 还是各有副本。

每个对象共有, 没有其他约束的情况下可通过 对象. 变量改变

✗ 方法中可以调用其他方法, 但不能定义其他方法

对象的产生:

对象被创建时, 它的所有成员变量都自动初始化

数值型	0
char	\u0000
boolean	false
引用类型	null

匿名对象:

new Student().show();

不用将创建的对象赋予变量

使用场景: ①

??? 是否不能通过类直接调用方法

可以调用静态方法

② 传递给方法作为参数

* 类中的 static 方法不能访问非 static 变量, 也不能访问非 static 方法。

方法的重载 (Overload)

在一个类中, 允许出现一个或几个同名方法, 只要它们的参数个数或类型或顺序
仅凭返回值是无法重载的。

可变参数的形参:

① 借助数组

② Java 特有的 ... 方式

... and printInfo (String ... args)

```
public void printInfo() {
    // args[]; 传数组一样传进去
}
```

两种方法 {
 printInfo("1", "2", "3");
 printInfo(new String[]{"1", "2", "3"});

→ 从0到该参数

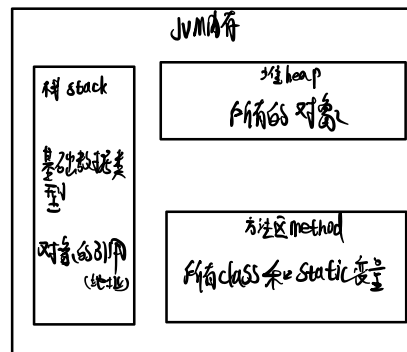
· public void printInfo(int i, String... args);

... 必须写在最后

· Java 这法不支持可变类型的参数，但可以通过重载和数组参数代替。

方法的参数传递：

JVM内存模型：



Java中的参数传递方法 {
 基本数据类型 值传递
 引用数据类型 引用传递

包package和引用：

包的层级结构：包的路径

```
animal | person | white men
        | cat      |
```

↳ dog

animal, person, whiteman

声明自己所在包和引入包中的类

package 包名;

import animal.person.whiteman // import animal.person.*

同一个包下, 无须 import

JDK中自带的包:

java.lang java核心类

.net 网络相关

.io 包含各种输入输出流的类

.text java 操作系统相关

.sql 数据库

.awt GUI

.applet applet运行所需类