## Day06

### 6-1 方法的定义以及分类

* 方法：也叫做函数，**实现某个功能**
* 方法分类：
  + 系统提供的方法
    - 常用的系统提供的方法
      * 如:nextInt() next() nextDouble() print() println()……..
  + 用户自定义的方法

### 6-2 方法的基本语法格式

**[修饰符]** **返回值类型** **方法名称**(**[**参数列表**]**){

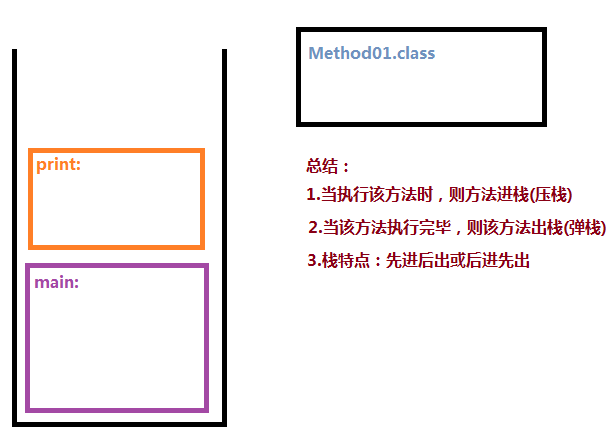
**方法体**

}

* []中的内容是可有可无的
* 暂时将方法的修饰符编写为**public static**
* 返回值类型有两种情况：
  + 第一种：无返回值类型，也就是编写为**void**
  + 第二种：有返回值类型，**目前可以编写9种数据类型之一**
* 方法名称，也就是标识符
  + 建议**编写单词**,也就是见名知意
  + 建议采用驼峰命名法，如：age getAge print
* 方法体实现具体的功能，根据题意编写

### 6-3 方法的调用

* 语法格式：
  + 类名.方法名称([参数列表]);
  + 方法名称([参数列表]); //注意：只能调用本类中的方法



### 6-4 无参无返回类型方法

### 6-5 无参带返回类型方法

* 该题只有一个结果时，此时就可以编写为带返回类型的方法完成
* 如果方法有返回类型则在方法体中必须编写return,通过return返回结果
* 方法的返回类型必须与return后面值的类型一致
* 通过return将结果返回到方法的调用处，因此在调用处进行处理
  + 使用变量接收，然后再输出变量中的值，如：

int a = Method03.sum(); //先调用方法，然后将方法返回的结果，如：3，再赋给左侧的变量名称a

System.out.println(a);

* 可以在一个方法体中编写任意多个return,但最终只执行一个相匹配的将值返回

### 6-6 break、continue、return之间区别

* break：表示中断，可以在switch case中或循环中使用，当遇到break则结束当前整个switch case或循环
* continue：表示继续，只能在循环中使用，当遇到continue时则结束本次(当次)循环，继续执行下一次循环
* return：表示返回，只能在方法中使用，当遇到return则返回到方法的调用处
  + 特殊形式：当方法是无返回类型时，则可以在方法体中编写return,但是必须编写为**return;**

### 6-7 带参无返回类型方法

* 当该题中有未知的数，因此可以编写为带参数的方法
* 带参方法的语法格式：

public static void 方法名称(**参数列表**){

方法体

}

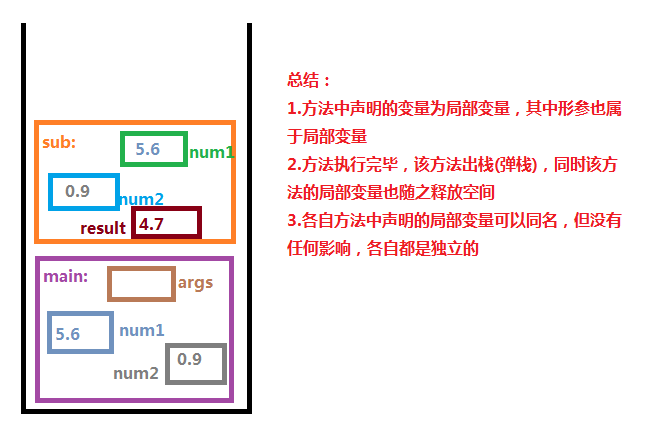
* 每个参数语法格式：数据类型 参数名称

多个参数之间使用英文逗号隔开

* 方法中未知的编写为参数，称为形式参数简称形参，**其中形参也属于局部变量**

调用方法时参数为已知的称为实际参数，简称实参

也就是实参的值赋给形参变量,必须保证实参与形参个数一致、类型一致、顺序一致



### 6-8 带参带返回类型方法

### 6-9 方法重载

在同一个类中，方法的名字相同，参数列表必须不同(类型不同、个数不同、顺序不同),与返回类型无关，构成方法的重载