本文内容是对方法的梳理和总结，本文内容包括：

**方法-定义和调用**
**方法-形参与实参**
**方法-可变参**
**方法-返回值**
**方法-变量作用域和生命周期**

先看一幅图

看到这幅图，你想到了什么？

喝茶？

不，你想到了茶壶入口能装入茶叶，茶壶出口能倒出茶水

你又想到了什么？

- 茶壶 PK 方法
- 入口 PK 形参

- 出口 PK 返回值

方法就是具有特定功能的程序段，可传入数据，可传出数据。

就像茶壶具有泡茶的特定功能，可传入茶叶，可倒出茶水。

# 1 方法-定义和调用

```
// main是主调方法
public static void main(String[] args) {
    menu();
}

// menu是被调方法
public static void menu() {
    System.out.println("-------------- 欢迎使用订餐系统 --------
------");
    System.out.println("--------------    1：我要注册    --------
------");
    System.out.println("--------------    2：我要登录    --------
------");
    System.out.println("--------------    3、我要退出    --------
------");
    System.out.println("-----------------------------------
------");
}
```

小结

1. 方法设计原则：一个方法只做一件事，例如menu()方法只显示菜单，如果有多件事要做，就设计多个方法
2. 主调方法：发出调用的方法
3. 被调方法：被调用的方法
4. 强调：被调方法执行过程中，主调方法处于阻塞状态，当被调方法执行结束后，程序执行返回到主调方法，主调方法停止阻塞，继续执行

# 2 方法-形参与实参

需求：完成用户登录功能

- 设计login()方法，传入参数用户名和密码，显示登录结果（正确的用户名和密码：`admin`，`P@$$wOrd`）
- 在main()方法中，从键盘输入用户名和密码，调用login()方法验证用户名和密码是否正确，输出验证结果

```java
public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    System.out.println("请输入用户名：");
    String username = scanner.next();
    System.out.println("请输入密码：");
    String password = scanner.next();

    // username, password 是实际参数
    login(username, password);
}

// username，password是形式参数
public static void login(String username, String password) {
    if ("admin".equals(username) && "123".equals(password)) {
        System.out.println("登录成功");
    } else {
        System.out.println("登录失败");
    }
}
```

小结

1. 方法参数传递规则：按照类型、个数、顺序一致原则传递参数
2. 实参与形参：主调方法的实参传递给被调用方法的形参
3. 对参数的理解：被调方法运行前，需要从主调方法得到的数据，是方法运行的前提条件

# 3 方法-可变参

- 方法的可变参数是指：方法参数的个数可变

- 可变参数的标志是三个点
- 可变参数必须是方法的最后一个参数
- 可变参数实际上是一个数组
- 可以向可变参数传入数组和或者散列值

需求：某公司有多个部门，每个部门人数不同，每个人的工资不同，如下：

- 开发部：李逵:1987，鲁智深:2001，燕青:2005
- 测试部：扈三娘:2009，孙二娘:1888

现要求设计一个方法，传入部门名称和每个员工的工资，在方法内显示部门的平均工资。

```
public static void main(String[] args) {
    // 调用showSalary方法，给可变参数传入散列值
    showSalary("开发部",1987,2001,2005);

    // 调用showSalary方法，给可变参数传入数组
    showSalary("测试部", new double[]{2009,1888});
}

// salary定义为可变参数
public static void showSalary(String dname, double ...salary)
{
    double sum = 0;
    if(salary!=null && salary.length >0){
        for (int i = 0; i < salary.length; i++) {
            sum += salary[i];
        }
    }
    System.out.printf("%s部门的平均薪资是%.2f" , dname ,sum/
salary.length);
}
```

小结

1. 三个点是方法可变参数的标志
2. 可变参数的定义：
    1. 可变参数必须是方法的最后一个参数
    2. 一个方法只能有一个可变参数

    3. 可变参数实际上是一个数组
3. 可变参数的调用：
    1. 实参可用是数组
    2. 实参可用是散列值
    3. 实参可用是null

# 4 方法-返回值

需求：实现用户登录

1. 设计login()方法，传入参数用户名和密码，返回布尔值
    - 返回true，用户名密码正确
    - 返回false，用户名或密码错误
    - 正确的用户名和密码：`admin`，`P@$$w0rd`
2. 在main()方法中，从键盘输入用户名和密码，调用login()方法验证用户名和密码是否正确，main()方法得到验证结果并输出验证结果

```
public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    System.out.println("请输入用户名：");
    String username = scanner.next();
    System.out.println("请输入密码：");
    String password = scanner.next();

    // 调用login方法，result接收方法返回值
    boolean result = login(username, password);
    System.out.println(result ? "登录成功" : "登录失败");
}

// 方法返回布尔值
public static boolean login(String username, String password)
{
    return  "admin".equals(username) &&
"123".equals(password)? true:false;
}
```

小结

1. 被调方法返回值：

- 方法运行后输出的运行结果
- 一个方法只能返回一个值
- 使用return语句返回值
- return 返回值的类型必须与方法声明返回值的类型一致
- 如果方法声明了返回值，那么方法就必须返回值

2. 主调方法接收返回值
    - 主调方法使用 = 运算符接收方法返回值

# 5 方法-变量作用域和生命周期

需求：读程序，说出运行结果，分析原因

```
public static void main(String[] args) {
    int age = 18;
    System.out.println("main()方法调用前：age = " + age);
    increment(age);
    System.out.println("main()方法调用后：age = " + age);
}
public static void increment(int age) {
    System.out.println("increment()方法中自增前：age = " + age);
    age++;
    System.out.println("increment()方法中自增后：age = " + age);
}
```

小结

1. 方法的形参是方法内的局部变量
2. 作用域：方法内的局部变量只能在方法内部使用，方法外部超出了方法变量的作用域，不能使用
3. 生命周期：方法内的局部变量在方法调用时申请内存，在方法调用结束后回收方法变量的内存