初识函数

什么是函数

- 为什么要有函数?
- 函数的概念

  函数就是能够实现特定功能的程序模块。
- 函数的作用

  1．让代码易于复用，避免编写重复的代码，提高编程的效率;

  2．让程序更加模块化，从而提高程序的可读性，便于后期的维护。
- 函数的使用

  1.提供函数定义　　2.提供函数原型　　3.调用函数

函数的定义

- 语法

```
返回值类型 方法名(形式参数列表)
{
    函数语句;
    return 返回值;
}
```

函数头（函数定义首部）

函数体

- 示例

```
int getSum(int a, int b) {
    int sum = a + b;
    return sum;
}
```

1. 返回值类型：用于限定返回值的数据类型，如果没有返回值用void代替；
2. 形式参数列表：由各种类型变量组成的列表，多个参数之间用逗号隔开，用于接收调用方法时传入的参数，如果没有参数可以省略或者为void；
3. 函数语句：用于完成功能的代码；
4. return：用于结束方法，并返回数据给调用者，如果没有返回值，可以省略。

函数原型（函数声明）

- 什么是函数原型

  函数原型（也称为函数声明）告诉编译器函数的名称、函数返回数据的类型、函数预期接收的形参个数及形参的类型和顺序。

- 函数原型的语法：`int getSum(int, int);` → **函数签名**

- 为什么需要函数原型

  1．告诉编译器函数的调用规则；

  2．提高编译器编译的效率；

  3．解决调用其它文件中函数时被限制的问题。

```
int getSum(int, int);
int main() {
    int sum = getSum(1, 2);
    return 0;
}

int getSum(int a, int b) {
    int sum = a + b;
    return sum;
}
```

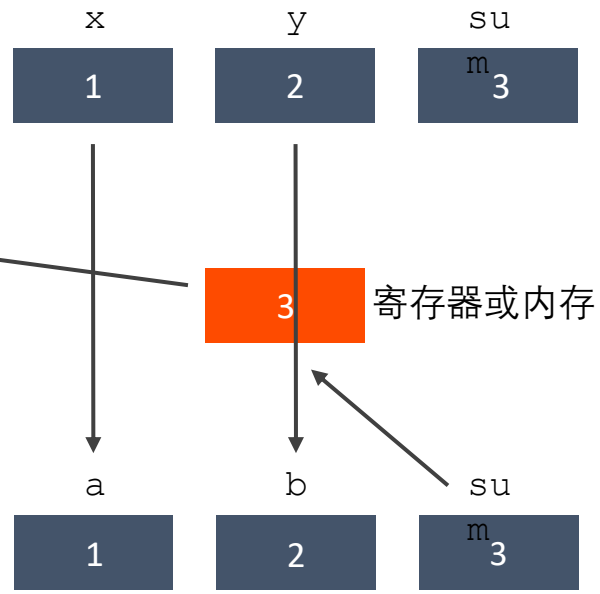
- 如果首次使用函数之前定义函数，则不需要函数原型，此时，函数头就是函数原型。

函数的调用

语法：方法名(实际参数值列表)；

```
int main() {
        int x = 1;
        int y = 2;
        int sum = getSum(x, y);
}
```

```
int getSum(int a, int b) {
        int sum = a + b;
        return sum;
}
```



Thanks