Java中的方法（Method）概念介绍

1. 方法（Method）又称函数，就是定义在类中的具有特定功能的一段独立小程序。
2. 定义方法的格式：

修饰符 返回值类型 方法名（参数类型 形参1，参数类型 形参2，…..）{

执行语句；

return \*\*\*；

}

1. 方法名定义规范：要有意义，且第一个单词首字母小写，后面的单词首字符大写。
2. 方法中都具有return语句，特殊情况是 返回类型为void时，只不过return语句省略了，但是实际上有return; 语句。
3. 方法没有具体的返回值，那么返回类型利用关键字 void 来指明，return语句可以省略，也可以加上return语句，但是后面不能加任何东西，直接return;。
4. 任何一个方法（函数）都是以return语句结尾的，所以方法体中必须有return语句，只不过返回类型为void的return； 语句可以省略不写，编译器自动补上。
5. 定义方法的好处：（封装和复用）
6. 可以将功能代码进行封装，
7. 便于对功能进行复用，
8. 方法只有被调用时，才会被执行，
9. 方法的出现提高了代码的复用性；
10. 注意： 方法中只能调用方法，不能在方法内部定义另外一个方法。
11. 方法中一旦执行return语句，就说明方法结束了。
12. 定义方法时，方法的结果一定要返回给调用者，交由调用者处理。这个很重要，不要对结果进行处理，交给调用者对结果进行处理。
13. 输出语句中什么都不写是合法的，功能是输出一个空行，如 System.out.println();任何执行System.out.println(\*\*\*);的语句，相当于执行两句：print（\*\*\*）；再执行 println（）；

System.out.print(LINE\_SEPARATOR);的作用等同于System.out.println();

如果System.out.println(LINE\_SEPARATOR);相当于先执行System.out.print(LINE\_SEPARATOR);再执行System.out.println();

但是小括号里不允许是一个返回类型为void的方法调用，此时会报错。如myPrint()是一个返回类型为void的方法，如果System.out.println(myPrint()); //此句代码是非法的

1. 定义一个方法前：找到两个明确目标：
2. 这个方法的结果是什么，从而确定返回值的类型；
3. 这个方法是否需要参数参加运算，从而确定形参的个数和参数的类型。
4. 参数值类型和返回值类型没有直接关系。
5. 栈：先进后出，后进先出。程序调用方法就是在栈中进行的。