main 函数是一个程序的入口，也是出口，main 函数其实就是类里的一个方法，方法只能在类里声明了，所以 maion 函数必须声明在一个类里，一个程序里只有一个 main 函数，因为类与类之间是相与调用的。一个程序从开始运行就会先找 main 函数，然后再一步一步的执行，如果有多个 main 那么程序就会报错，因为他不知道该从那个门口（main 函数）进，然而类为什么可以有多个呢，你还记的 JAVA 的三大基本特征么，即封装，继承，多态。而封装就是我们所说的类，他是把所有相同的属性，方法归纳到一起，即组成一个类。这样给你打个比方吧

main 函数所在的一个类就相当于一个城市，而 main 函数就是城市的一个入口，并且只有一个入口，也可是出口，而城市中又有许多小区，我们叫他为类吧，而小区里的房子又基本相同，这也就验证了我们把同一类物品归为一类，即封装在一个类里，而每个小区又不同，这又验证了每一个类的实现功能 是不同的，而类与类这间的调用则是用 NEW 关键字，这个你应该会吧，继承，多态我就不用说了

JAVA 中的主函数是我们再熟悉不过的了，相信每个学习过 JAVA 语言的人都能够熟练地写出这个程序的入口函数，但对于主函数为什么这么写，其中的每个关键字分别是什么意思，可能就不是所有人都能轻松地答出来的了。我也是在学习中碰到了这个问题，通过在网上搜索资料，并加上自己的实践终于有了一点心得，不敢保留，写出来与大家分享。

主函数的一般写法如下：

public static void main(String[] args){…}

下面分别解释这些关键字的作用：

（1）public 关键字，这个好理解，声明主函数为 public 就是告诉其他的类可以访问这个函数。

（2）static 关键字，告知编译器 main 函数是一个静态函数。也就是说 main 函数中的代码是存储在静态存储区的，即当定义了类以后这段代码就已经存在了。如果 main()方法没有使用 static 修饰符，那么编译不会出错，但是如果你试图执行该程序将会报错，提示 main()方法不存在。因为包含 main()的类并没有实例化（即没有这个类的对象），所以其 main()方法也不会存。而使用 static 修饰符则表示该方法是静态的，不需要实例化即可使用。

（3）void 关键字表明 main()的返回值是无类型。

**（4）参数 String[] args，这是本文的重点。

第一、程序使用者可以在命令行状态下向某个类传递参数。看下面的例子：

public class ArgsDemo {

```
public static void main(String[] args) {

String str = new String();

for (int i = 0; i < args.length; i++) {

System.out.println(args[i]);

str += args[i];

}

System.out.println(str);

}

}
```

使用 javac ArgsDemo.java 命令生成 ArgsDemo.class 文件;然后使用“java ArgsDemo 参数一 参数二 参数三 …”的格式向 ArgsDemo 类传递参数。该示例程序将首先输出参数，然后输出所有参数的和。比如 java ArgsDemo a b c,将得到这样的输出：

```
a

b

c

abc
```

需要注意的是，如果这里的循环条件不是 i <args.length，而是 i <5，则在命令行中输入的参数必须是 5 个，否则将会报错，错误类型为：

```
Exception in thread “main”
java.lang.ArrayIndexOutOfBoundException:3

at ArgsDemo.main(ArgsDemo.java:5)
```

第二、可以在另一个类中向包含 main()的类传递参数，如下例：

```
public class A {

public static void main(String[] args)

{
```

```
for(int i=0;i <args.length;i++)
System.out.println(args[i]);
}
}
public class B {
public static void main(String[] args)
{
c = new A();
String[] b = {"111","222","333"};
c.main(b);
}
}
```

首先定义一个 class A，在 A 中定义一个 main()函数，在该函数中输出参数 args。然后定义一个 classB，在 B 中初始化一个 A 的实例 c，然后向 c 传递参数，并且调用 c 的 main 方法打印出传入的参数值。输出结果如下：

```
111
222
333
```

由于 main()函数是静态函数，即不需要实例化也能使用，所以 B 使用下面的写法也能完成相同的功能：

```
public class B {
public static void main(String[] args)
{
//A c = new A();
String[] b = {"111","222","333"};
```

```
A.main(b);

}

}
```

总结：参数 args 的主要作用是为程序使用者在命令行状态下与程序交互提供了一种手段。此外在其他类中直接使用 main()函数，并传递参数也是可行的，虽然这种方法不太常用，但毕竟为我们提供了一种选择。

（以上观点均系个人理解，不准确的地方欢迎批评指正。实例程序均调试通过）

==========

对 static 的理解还有问题。main 方法的代码是存在方法区的。方法不管是否为 static 的，都存在方法区。