

如实例中所示，当我们在class ThreadTestCommonVariable中声明一个静态方法，并在静态方法Main中使用时系统会发生报错，同样在类中声明的done变量也无法在静态方法Main中使用。

这就引出了静态类和静态方法的一些概念了

静态类：

在类（class）上加入static修饰，表示该类无法被实例化，并且该类中，无法实例化变量和函数。

**那么在此静态类中添加函数、变量、属性，都必须是静态的**。

当我们需要将一个类下面的所有成员都被共享的话，就可以把这个类声明为静态类。

静态成员：静态成员属于类所有，非静态成员属于**类的实例**所有。

每创建一个类的实例，都会在内存中为非静态成员新分配一块存储。

静态方法：**静态方法属于类所有**，类实例化前即可使用。

非静态方法可以访问类中的任何成员，静态方法只能访问类中的静态成员。

因为静态方法在类实例化前就可以使用，而类中的非静态变量必须在实例化之后才能分配内存。所有静态方法无法使用非静态变量。

由上述理论可以知道，tt.Go()调用的是实例tt中的实例方法，而我们在class中声明的Go()是一个静态方法，所有就会发生报错；同样Main方法属于静态方法，所以在该方法中无法直接调用非静态变量done。