Methods in Java

Introduction

What is a Program?

A program is a collection of useful actions that a computer performs to solve a problem (or several problems).

It is up to the programmer to choose the actions, put them into the correct order, and implement them using the appropriate language.

As problems get more complicated, programs contain more actions, becoming longer and more difficult to manage.

What is a Method?

A method is a program within a program, which is also called a sub-program. You may also hear terms such as functions and procedures, which are the same as methods in Java.

We have already made use of the most important method of all, the main method. The main method starts all of our programs.

There are also utility methods, such as those for input, output, and specialized math operations.

Typical Java Program Structure

```
class JavaProgram
{
   public static void main(String[] args)
   {
      // useful instructions here!
   }
}
```

Java Structure with Methods
(methods can be declared before main)

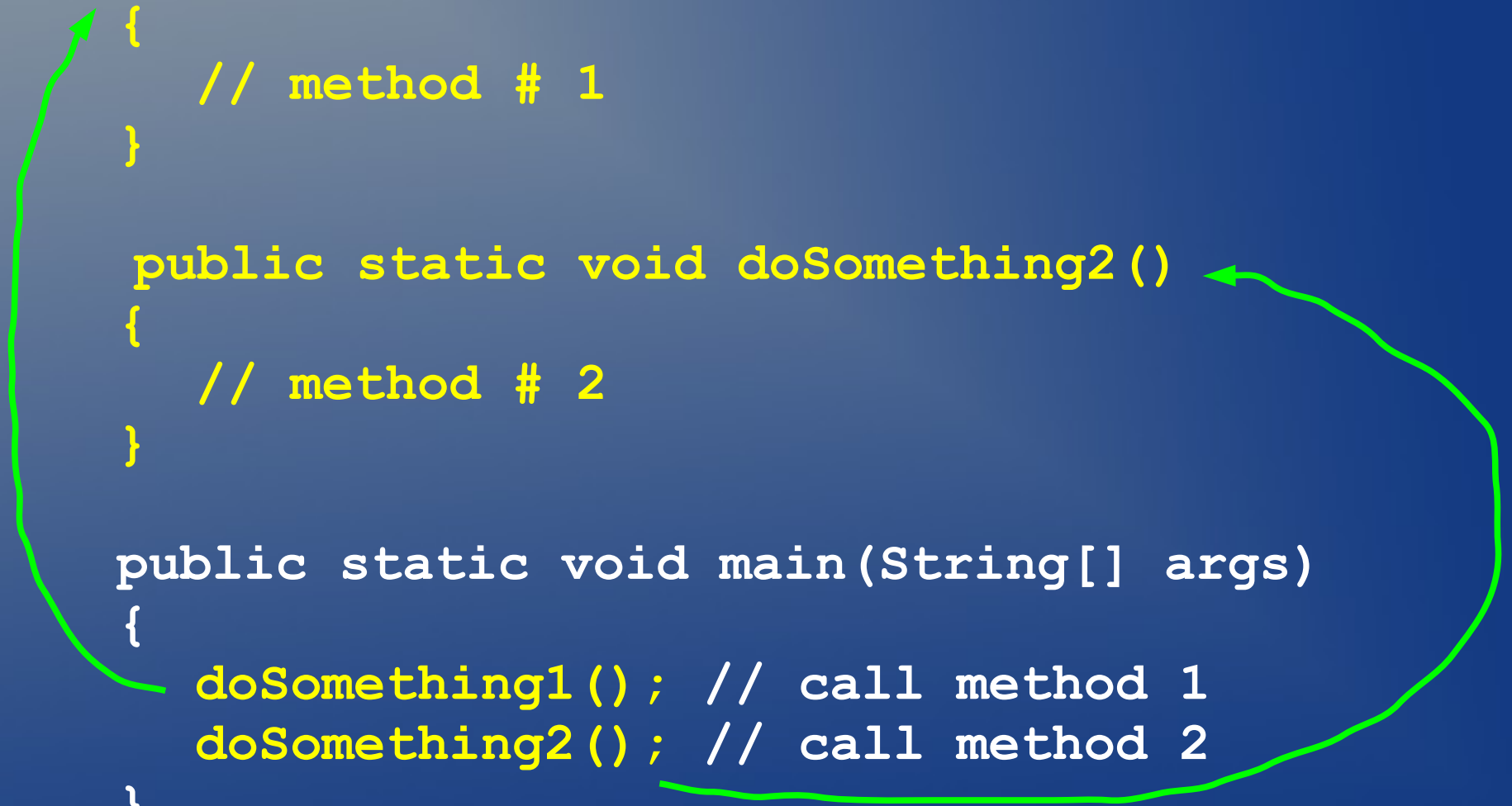
```
class JavaProgram
{
   public static void doSomething()
   {
      // useful instructions here!
   }

   public static void main(String[] args)
   {
      doSomething(); // call method
   }
}
```

Java Structure with Methods
(methods can be declared after main)

```
class JavaProgram
{
   public static void main(String[] args)
   {
      doSomething(); // call method
   }

   public static void doSomething()
   {
      // useful instructions here!
   }
}
```

Multiple Methods in Java

```
class JavaProgram
{
   public static void doSomething1()
   {
      // method # 1
   }

   public static void doSomething2()
   {
      // method # 2
   }

   public static void main(String[] args)
   {
      doSomething1(); // call method 1
      doSomething2(); // call method 2
   }
}
```



Methods Can Call Other Methods

```
class JavaProgram
{
    public static void doSomething1()
    {
        doSomething2(); // call method 2
    }

    public static void doSomething2()
    {
        // method # 2
    }

    public static void main(String[] args)
    {
        doSomething1(); // call method 1
    }
}
```

Remember to always start with main method!

Method Demo 1: Copy & Paste

```
class MethodDemo1
{
   public static void doSomething1()
   {
      System.out.println("method 1 called by main");
   }

   public static void doSomething2()
   {
      System.out.println("method 2 called by main");
   }

   public static void main(String[] args)
   {
      System.out.println("calling method 1");
      doSomething1(); // call method 1
      System.out.println("calling method 2");
      doSomething2(); // call method 2
   }
}
```

Method Demo 2: Copy & Paste

```
class MethodDemo2
{
   public static void doSomething1()
   {
      System.out.println("method 1 called by main");
      System.out.println("method 1 calling method 2");
      doSomething2(); // call method 2
   }

   public static void doSomething2()
   {
      System.out.println("method 2 called by method 1");
   }

   public static void main(String[] args)
   {
      System.out.println("calling method 1");
      doSomething1(); // call method 1
   }
}
```