

# Writing a first Java program

All Java programs start as text files that are later used to create “class” files, which are the actual runnable programs. This means that Java programs can be written in any plain text editor such as the Windows Notepad application.

Follow these steps to create a simple Java program that will output the traditional first program greeting:



Hello.java

## Beware



Java is a case-sensitive language where “Hello” and “hello” are distinctly different – traditionally Java program names should always begin with an uppercase letter.

## Don't forget



Java programs are always saved as their exact program name followed by the “.java” extension.

- 1 Open a plain text editor, like Notepad, and type this code exactly as it is listed – to create a class named “Hello”
 

```
class Hello
{
}
```
- 2 Between the curly brackets of the **Hello** class, insert this code – to create a “main” method for the **Hello** class
 

```
public static void main ( String[] args )
{
}
```
- 3 Between the curly brackets of the **main** method, insert this line of code – stating what the program will do
 

```
System.out.println( "Hello World!" );
```
- 4 Save the file at any convenient location, but be sure to name it precisely as **Hello.java** – the complete program should now look like this:

```

Hello.java - Notepad
File Edit Format View Help
class Hello
{
    public static void main ( String[] args )
    {
        System.out.println( "Hello World!" );
    }
}

```

## ...cont'd

The separate parts of the program code on the opposite page can be examined individually to understand each part more clearly:

### The Program Container

```
class Hello { }
```

The program name is declared following the **class** keyword and is itself followed by a pair of curly brackets. All of the program code that defines the **Hello** class will be contained within these curly brackets.

### The Main Method

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

This fearsome-looking line is the standard code that is used to define the starting point of nearly all Java programs. It will be used in most examples throughout this book exactly as it appears above – so it may be useful to memorize it.

The code declares a method named “main” that will contain the actual program instructions within its curly brackets.

Keywords **public static void** precede the method name to define how the method may be used and are explained in detail later.

The code ( **String[] args** ) is useful when passing values to the method and is also fully explained later in this book.

### The Statement

```
System.out.println( "Hello World!" );
```

Statements are actual instructions to perform program tasks and must always end with a semi-colon. A method may contain many statements inside its curly brackets to form a “statement block” defining a series of tasks to perform, but here a single statement instructs the program to output a line of text.

Turn to the next page to discover how to compile and run this program.

Don't forget



All stand-alone Java programs must have a **main** method. Java applets are different and their format is explained later.

Hot tip



Create a MyJava directory in which to save all your Java program files.