

First Program in Java

Like any other programming language, the java programming language is used to create applications. So, we start from building a classical “Hello World” application, which is generally used as the first program for learning any new language.

HelloWorldApp

1. Open notepad editor from *Start → ProgramFiles → Accessories → Notepad*.
2. Write the following code into it.

Note: Don’t copy paste the given below code. Probably it gives errors and you can’t able to remove them at the beginning stage.

```
1.  /* The HelloWorldApp class implements an application that
2.     simply displays "Hello World!" to the standard output.
3.  */
4.  public class HelloWorldApp {
5.      public static void main(String[] args) {
6.
7.          //Display the string. No global main
8.          System.out.println("Hello World");
9.      }
10. }
```

3. To save your program, move to File menu and choose save as option.
4. Save your program as “HelloWorldApp.java” in some directory. Make sure to add double quotes around class name while saving your program. For this example create a folder known as “examples” in D: drive

Note: Name of file must match the name of the public class in the file (at line 4). Moreover, it is case sensitive. For example, if your class name is MyClasS, than file name must be MyClasS. Otherwise the Java compiler will refuse to compile the program.

For the rest of this handout, we assume that program is saved in D:\examples directory.

HelloWorldApp Described

Lines 1-3

Like in C++, You can add multiple line comments that are ignored by the compiler.

Lines 4

Line 4 declares the class name as `HelloWorldApp`. In java, every line of code must reside inside class. This is also the name of our program (`HelloWorldApp.java`). The compiler creates the `HelloWorldApp.class` if this program successfully gets compiled.

Lines 5

Line 5 is where the program execution starts. The java interpreter must find this defined exactly as given or it will refuse to run the program. (However you can change the name of parameter that is passed to main. i.e. you can write `String[] argv` or `String[] someParam` instead of `String[] args`)

Other programming languages, notably C++ also use the `main()` declaration as the starting point for execution. However the main function in C++ is global and reside outside of all classes where as in Java the main function must reside inside a class. In java there are no global variables or functions. The various parts of this main function declaration will be covered at the end of this handout.

Lines 6

Again like C++, you can also add single line comment

Lines 7

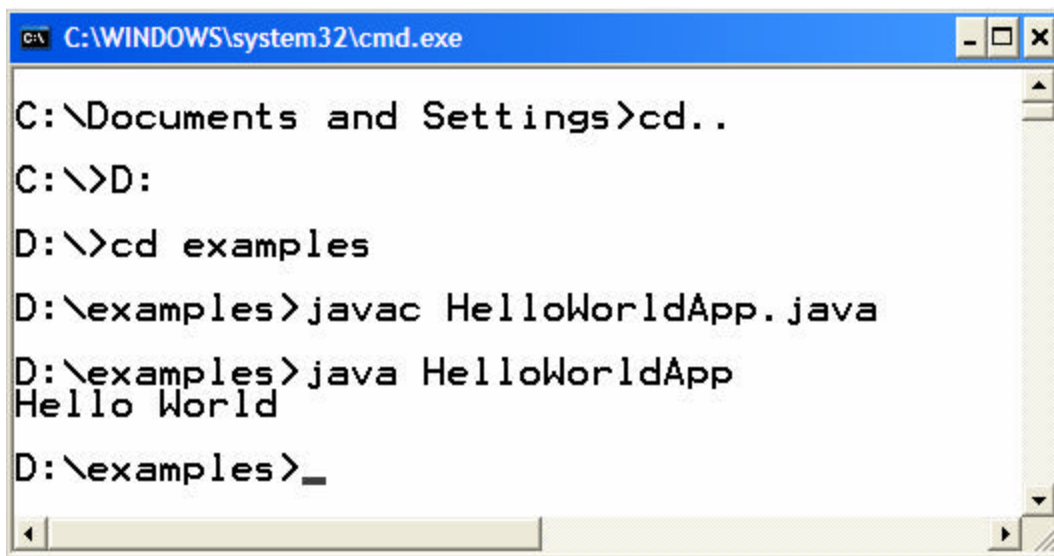
Line 7 illustrates the method call. The `println()` method is used to print something on the console. In this example `println()` method takes a string argument and writes it to the standard output i.e. console.

Lines 8-9

Line 8-9 of the program, the two braces, close the method `main()` and the class `HelloWorldApp` respectively.

Compiling and Running HelloWorldApp

1. Open the command prompt from *Start → Program Files → Accessories*. OR alternatively you can write *cmd* in the run command window.
2. Write *cd..* to came out from any folder, and *cd [folder name]* to move inside the specified directory. To move from one drive to another, use [Drive Letter]: See figure given below
3. After reaching to the folder or directory that contains your source code, in our case HelloWorldApp.java.
4. Use “javac” on the command line to compile the source file (“.java” file).
 - D:\examples> javac HelloWorld.java
5. If program gets successfully compiled, it will create a new file in the same directory named HelloWorldApp.class that contains the byte-code.
6. Use “java” on the command line to run the compiled .class file. Note “.class” would be added with the file name.
 - D:\examples> java HelloWorld
7. You can see the *Hello World* would be printed on the console. Hurrah! You are successful in writing, compiling and executing your first program in java 😊



```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings>cd..
C:\>D:
D:\>cd examples
D:\examples>javac HelloWorldApp.java
D:\examples>java HelloWorldApp
Hello World
D:\examples>_
```

Points to Remember

- Recompile the class after making any changes
- Save your program before compilation
- Only run that class using java command that contains the main method, because program executions always starts form main

An Idiom Explained

- You will see the following line of code often:
 - `public static void main(String args[]) { ... }`
- About main()
 - “main” is the function from which your program starts
 - Why public?
 - Since main method is called by the JVM that is why it is kept public so that it is accessible from outside. Remember private methods are only accessible from within the class
 - Why static?
 - Every Java program starts when the JRE (Java Run Time Environment) calls the main method of that program. If main is not static then the JRE have to create an object of the class in which main method is present and call the main method on that object (In OOP based languages method are called using the name of object if they are not static). It is made static so that the JRE can call it without creating an object.
 - Also to ensure that there is only one copy of the main method per class
 - Why void?
 - Indicates that main () does not return anything.
 - What is String args[] ?
 - Way of specifying input (often called command-line arguments) at startup of application. More on it latter
