```
/*
 * June 24th : Introduction to Java Programming
 * III Semester - CSE
 */
```

Programming Text Editor

Do NOT use Notepad (Windows) or TextEdit (Mac) for programming. Install a *programming* text editor, which does syntax color highlighting. For example,

- For Windows: Sublime Text, Atom, NotePad++, TextPad.
- For Mac: Sublime Text, Atom, jEdit, gEdit.
- For Ubuntu: gEdit.

Step 1: Write the Source Code:

```
/*
 * First Java program, which says hello.
 */
public class Hello {   // Save as "Hello.java"
   public static void main(String[] args) {  // Program entry point
      System.out.println("Hello, world!");   // Print text message
   }
}
```

Save the source file as "`Hello.java`". A Java source file should be saved with a file extension of "`.java`". The filename shall be the same as the classname - in this case "`Hello`". Filename and classname are *case-sensitive*.

Step 2: Compile the Source Code:

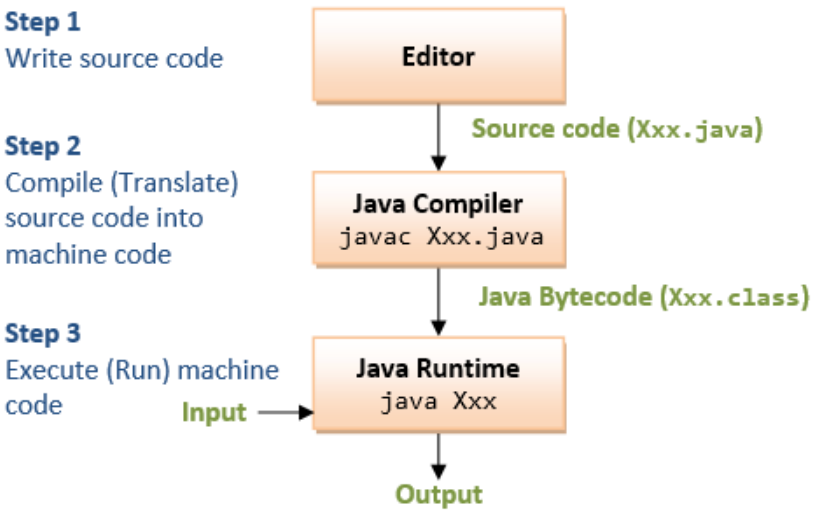
Compile the source code "`Hello.java`" into Java bytecode (or machine code) "`Hello.class`" using JDK's Java Compiler "`javac`".

```
// Change directory (cd) to the directory (folder) containing the source file
"Hello.java"
javac Hello.java
```

Step 3: Run the Program:

Run the machine code using JDK's Java Runtime "`java`", by issuing this command:

```
java Hello
Hello, world!
```



Java Program Template

```
/*
 * Comment to state the purpose of the program.
 */
public class Classname {   // Choose a meaningful Classname. Save as "Classname.java"
   public static void main(String[] args) {  // Entry point of the program
      // Your programming statements here!!!
   }
}
```

Important Note:

Take note of the source-code **indentation**!!! Whenever you open a block with `'{'`, indent all the statements inside the block by 3 (or 4 spaces). When the block ends, un-indent the closing `'}'` to align with the opening statement.

Go through the PrintTest.java – analyze Print() and Println() statements.

Exercises:

1. Write 4 programs, called PrintCheckerPattern, PrintSquarePattern, PrintTriangularPattern and PrintStarPattern to print each of the following patterns.

```
* * * * *      * * * * *    * * * * *        *
 * * * * *     *       *     *     *     * *   * *
* * * * *      *       *      *   *         * *
 * * * * *     *       *       * *          *   *
* * * * *      * * * * *        *          *     *

   (a)            (b)          (c)           (d)
```

`Try: Use nested loops to print the above patterns.`

2. Write a program called **`CozaLozaWoza`** which prints the numbers `1` to `110`, `11` numbers per line. The program shall print "Coza" in place of the numbers which are multiples of 3, "Loza" for multiples of 5, "Woza" for multiples of 7, "CozaLoza" for multiples of 3 and 5, and so on. The output shall look like:

```
1 2 Coza 4 Loza Coza Woza 8 Coza Loza 11
Coza 13 Woza CozaLoza 16 17 Coza 19 Loza CozaWoza 22
23 Coza Loza 26 Coza Woza 29 CozaLoza 31 32 Coza
......
```