

TOPIC 5 - JAVA DOCUMENTATION

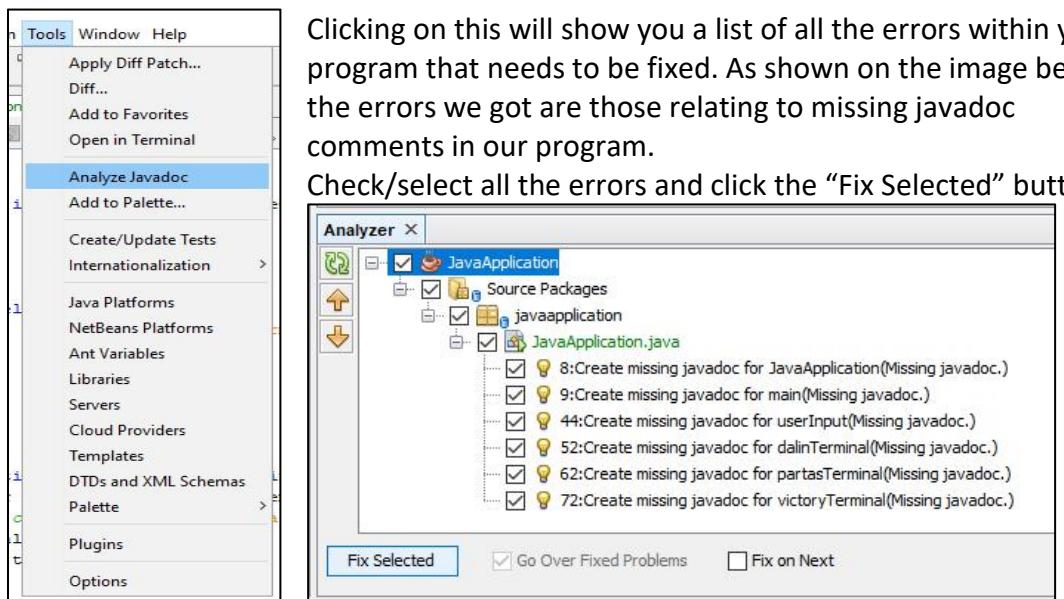
Documentation is where we put every detailed functional information for the reader to learn and guidelines for the user. In programming, we document our program for it to be easier to understand and for future purposes.

JAVA Documentation Using NetBeans IDE

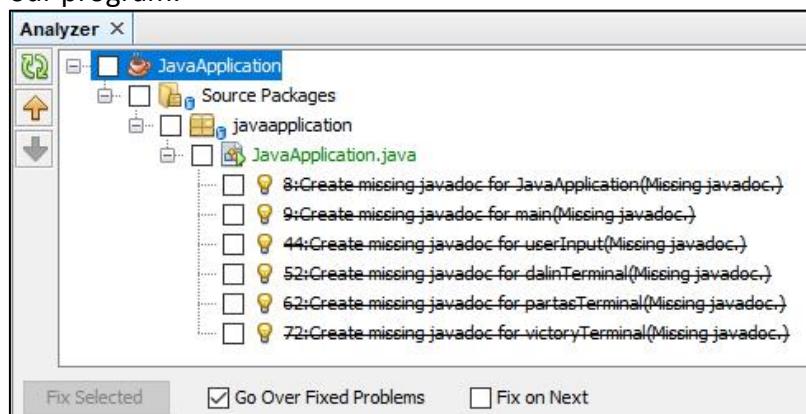
Javadoc is a tool that NetBeans uses for documenting Java. Javadoc documentations are being generated to HTML form. Under Javadoc we have a “doc comment”, the doc comment is where we place the description, function, explanation for the block or lines of codes. Documentation generator made by Sun Microsystems for Java (now claimed by Oracle Corporation) for creating API documentation in HTML arrange from Java source code.

Auto-generate javadoc comments in the file

Go to the Tools menu and choose “Analyze Javadoc”.



Once the “Fix Selected” process is done, we can see that each error on the list have been “lined” through the middle, indicating that they have been successfully fixed. Javadoc comments have been automatically added on the necessary locations within our program.



While NetBeans has automatically generated javadoc comments for us, notice that there are still missing contents within these comments which we have to provide manually, such as the description of the method and description of the method's parameter(s).

```

55 /**
56 *
57 * @param terminal
58 * @return
59 */
60 public static String userInput(String terminal) {
61     Scanner kbd = new Scanner(System.in);
62     System.out.print("Enter terminal name: ");
63     terminal = kbd.next();
64     return terminal;
65 }
66
67 /**
68 *
69 * @param line
70 * @throws IOException
71 */
72 public static void dalInTerminal(String line) throws IOException {
73     try (BufferedReader br = new BufferedReader(new FileReader("m..."))

```

Another way of generating javadoc comments in NetBeans is by simply doing this on the line before a method: type `/**` and press the Enter key. It will have the same results with the one above.

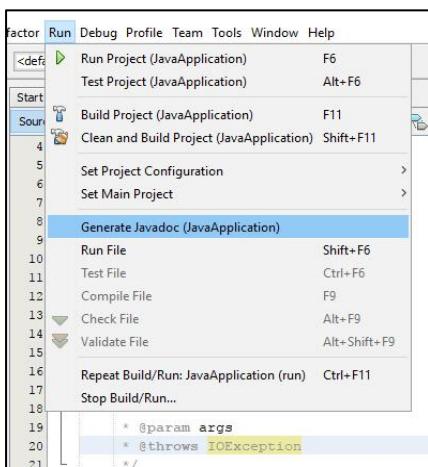
```

54 /**
55  *
56  * public static String userInput(String terminal) {
57  *     Scanner kbd = new Scanner(System.in);
58  *     System.out.print("Enter terminal name: ");
59  *     terminal = kbd.next();
60  *     return terminal;
61  * }
62
63 /**
64  *
65  * @param terminal
66  * @return
67 */
68 public static String userInput(String terminal) {
69     Scanner kbd = new Scanner(System.in);
70     System.out.print("Enter terminal name: ");
71     terminal = kbd.next();
72     return terminal;
73 }

```

Generate Java Documentation

Once our program already has the necessary javadoc comments, we can now generate the documentation.



```

8 /**
9  *
10 * A program that shows bus schedules.
11 *
12 * @author Group1
13 * @version 1.0
14 */
15 public class JavaApplication {
16
17 /**

```

Under the NetBeans "Run" menu, choose/click "Generate Javadoc". This will now create the documentation—in html format—in the "dist" folder of our project.

```

==[IDE]== 10 29, 18 12:53:37 AM Ignoring...
ignoring C:\Users\Acer\Desktop\Javadoc\TOPIC5\JavaApplication\dist
Javadoc\TOPIC5\JavaApplication\dist
==[IDE]== 10 29, 18 12:53:39 AM Ignoring... finished.

```

