Exercise 1:

* Q7: “Step Return” redirects me to line 30 of the “run” method

Screen Capture One: Graphical user interface, application

Description automatically generated

|  |
| --- |
| Question: What is the Eclipse keyboard shortcut for toggling a breakpoint? |
| CTRL + SHIFT + B |
| Question: What is the difference between “Step-Over”, and “Step-Into”, and “Step-Return”? |
| Assuming “Step-Over” and “Step-Into” are being used to debug functions, “Step-Over” executes the functions as one step and goes to the next line, while “Step-Into” goes into the function itself, where from there, you can debug the function’s content.  “Step Return” basically executes the function until it reaches its end, then goes to where it is called until the end. |
| Task: Practice tracing through the DebugStar sample program.  It is ok if you don’t understand all of the java code; but you should be able to trace the order in which statements are executed.  Based on your best understanding of the program, provide a list of methods that are called when the program executes (from start to end, in order of being called). You can skip library methods (like println, for example).  HINT: Use a combination of “Step-Into” “Step-Over” and “Step-Return”. Use the “Stack Trace” window |

|  |
| --- |
| List of Methods (in order of call) below. Please use the fully qualified name, eg. “DebugStar.run(String, int, int). Use the stack view to help you.  DebugStar.main(String[])  DebugStar.run(String, int, int)  DebugStar.getOperation(String)  Add.perform(int, int)  DebugStar.run(String, int, int)  DebugStar.getOperation(String)  Subtract.perform(int, int)  DebugStar.run(int)  Factorial.perform(int) |

Exercise 2:

Screen Capture One: Text

Description automatically generated

|  |  |  |  |
| --- | --- | --- | --- |
| Line Number | Type of error (compile-time, run-time, or logical) | Description | Correction |
| 26 | Compile-time |  | Add “;” after n |
| 27 | Compile-time |  | Initialize n to a value |
| 49 | Run-Time |  | Change “n > 2” to “n <= 2” |
| 49 | Run-time |  | Change line 49 to “System.out.print(fiboList.get(i - 1)); |
| 49 | Logical |  | Change line 49 to “System.out.print(fiboList.get(i – 1) + “, ”); |
| 40 | Logical |  | Change line 40 to “f.add(f.get(i – 2) + f.get(i – 1)); |
| 47 | Logical |  | Change line 47 to “int i = 1;” |

Screen Capture Two:

Text

Description automatically generatedText

Description automatically generated