

CS 2334: Debugger

Using the Eclipse Debugger

Sometimes we want to be able to see what our program is “thinking” at various times....

Recall Lab 2 ...

The screenshot shows the Eclipse IDE interface. The Package Explorer on the left shows a project structure with a 'src' folder containing 'Driver.java', 'Gadget.java', 'GadgetTest.java', 'UtilityBelt.java', and 'UtilityBeltTest.java'. The main editor window displays the code for 'Driver.java' with the following content:

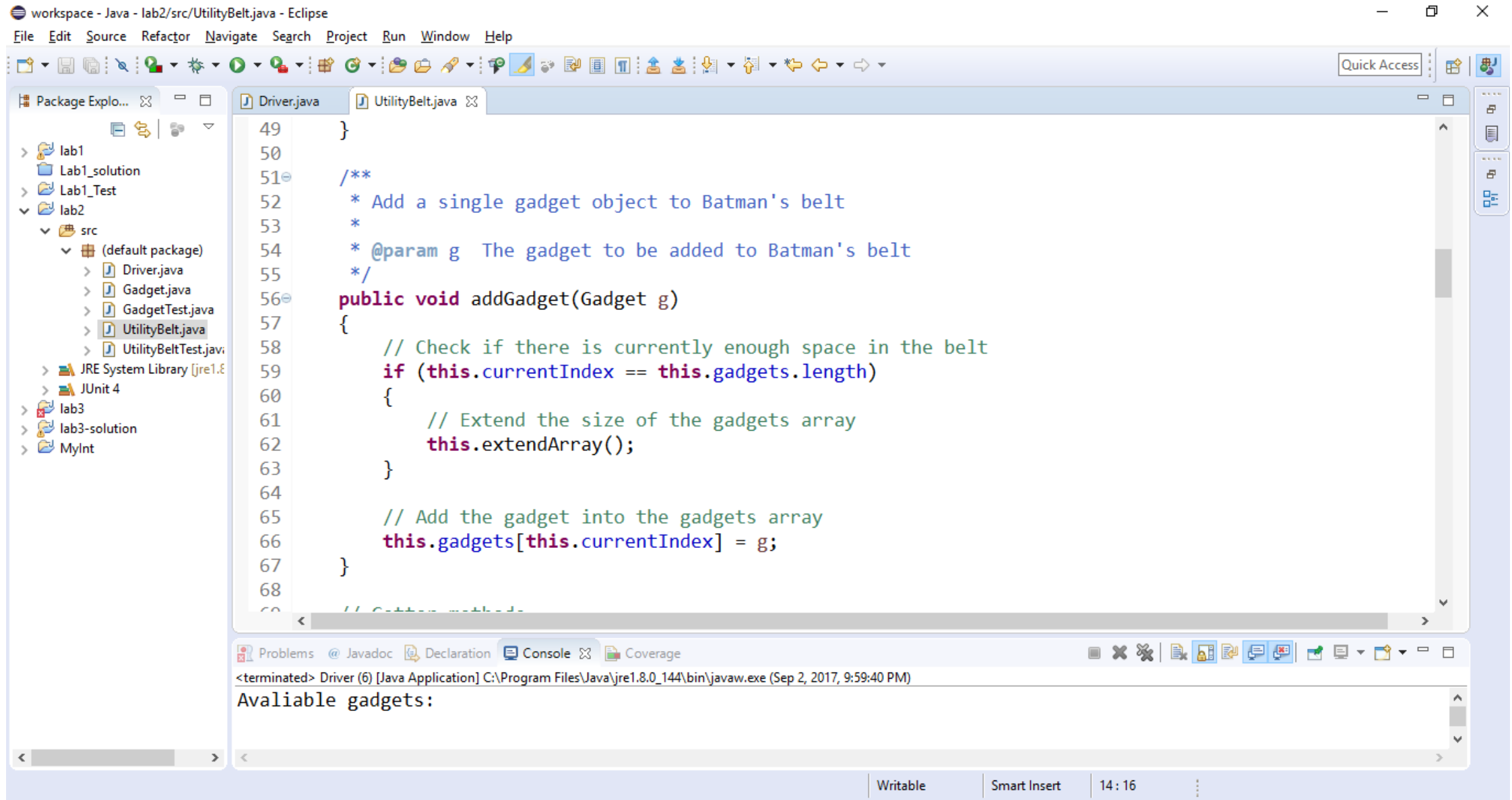
```
17  */
18  public static void main(String[] args)
19  {
20      /*
21       * You don't have to change any of this particular code however you may
22       * find it useful to play around with how the code works here.
23       */
24      //Create a batarang gadget to add to Batman's utility belt
25      Gadget batarang = new Gadget("Batarang", 47.24, 40.20);
26      //Create a shark repellent item
27      Gadget sharkRepellent = new Gadget("Shark Repellent", 30.0, 15);
28
29      // Create a belt and add the gadgets to it
30      UtilityBelt belt = new UtilityBelt();
31      //Add the batarang
32      belt.addGadget(batarang);
33      //Add the shark repellent
34      belt.addGadget(sharkRepellent);
35
36      // Tell Batman what he has
37      System.out.println(belt);
38  }
```

The Console window at the bottom shows the output of the program:

```
<terminated> Driver (6) [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Sep 2, 2017, 9:59:40 PM)
Available gadgets:
```

A red arrow points from the text "The program prints nothing!" to the empty console output area.

Maybe addGadget() is the problem...



The screenshot shows the Eclipse IDE interface. The Package Explorer on the left shows a project structure with folders 'lab1', 'lab2', and 'lab3'. Under 'lab2', there is a 'src' folder containing 'Driver.java', 'Gadget.java', 'GadgetTest.java', 'UtilityBelt.java', and 'UtilityBeltTest.java'. The main editor window displays the code for 'UtilityBelt.java'. The code includes a comment for the 'addGadget' method and its implementation. The console window at the bottom shows the output of a Java application, indicating it terminated and displaying the text 'Available gadgets:'.

```
workspace - Java - lab2/src/UtilityBelt.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explo...
lab1
  Lab1_solution
  Lab1_Test
lab2
  src
    (default package)
      Driver.java
      Gadget.java
      GadgetTest.java
      UtilityBelt.java
      UtilityBeltTest.java
  JRE System Library [jre1.8
  JUnit 4
lab3
lab3-solution
MyInt

49 }
50
51 /**
52  * Add a single gadget object to Batman's belt
53  *
54  * @param g The gadget to be added to Batman's belt
55  */
56 public void addGadget(Gadget g)
57 {
58     // Check if there is currently enough space in the belt
59     if (this.currentIndex == this.gadgets.length)
60     {
61         // Extend the size of the gadgets array
62         this.extendArray();
63     }
64
65     // Add the gadget into the gadgets array
66     this.gadgets[this.currentIndex] = g;
67 }
68
69 // Getter methods
```

<terminated> Driver (6) [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Sep 2, 2017, 9:59:40 PM)
Available gadgets:

Writable Smart Insert 14:16

Setting a Breakpoint

The screenshot shows the Eclipse IDE interface. The Package Explorer on the left shows a project structure with a package named 'src' containing several Java files, including 'UtilityBelt.java'. The main editor window displays the source code of 'UtilityBelt.java'. A context menu is open over line 56, with the 'Toggle Breakpoint' option selected. The code in the editor is as follows:

```
49     }
50
51     /**
52      * Add a single gadget object to Batman's belt
53      *
54      * @param g The gadget to be added to Batman's belt
55      */
56     public void addGadget(Gadget g)
57     {
58         // Check if there is currently enough space in the belt
59         if (currentIndex == this.gadgets.length)
60             return;
61         // Add the gadget to the gadgets array
62         gadgets[currentIndex] = g;
63     }
64 }
```

The context menu options are:

- Toggle Breakpoint (Ctrl+Shift+B)
- Disable Breakpoint (Shift+Double Click)
- Go to Annotation (Ctrl+1)
- Add Bookmark...
- Add Task...
- Show Quick Diff (Ctrl+Shift+Q)
- Show Line Numbers
- Folding
- Preferences...
- Breakpoint Properties... (Ctrl+Double Click)

The Console window at the bottom shows the output of the application, which is currently empty except for the header text: "Availiable gadgets:".

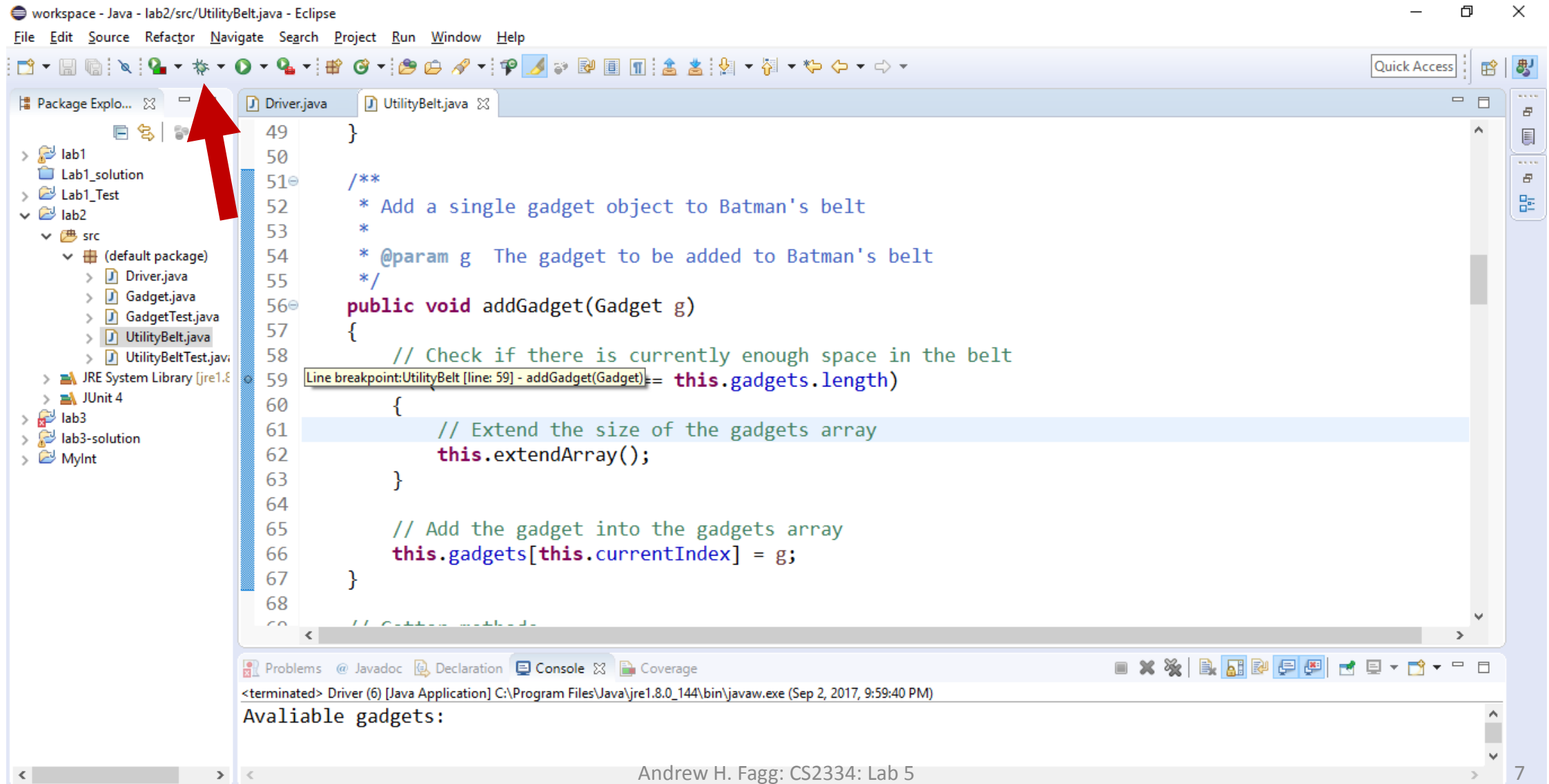
Breakpoint is now set

The screenshot shows the Eclipse IDE interface. The Package Explorer on the left shows a project structure with a package named 'src' containing several Java files, including 'UtilityBelt.java'. The main editor window displays the source code of 'UtilityBelt.java'. A red oval highlights a breakpoint set on line 59, which reads: `Line breakpoint:UtilityBelt [line: 59] - addGadget(Gadget) == this.gadgets.length`. The code in the editor includes a Javadoc comment for the `addGadget` method and its implementation, which checks for space in the `gadgets` array and extends it if necessary before adding the gadget.

```
49     }
50
51     /**
52      * Add a single gadget object to Batman's belt
53      *
54      * @param g The gadget to be added to Batman's belt
55      */
56     public void addGadget(Gadget g)
57     {
58         // Check if there is currently enough space in the belt
59         Line breakpoint:UtilityBelt [line: 59] - addGadget(Gadget) == this.gadgets.length
60     }
61     // Extend the size of the gadgets array
62     this.extendArray();
63 }
64
65 // Add the gadget into the gadgets array
66 this.gadgets[this.currentIndex] = g;
67 }
68
69 // Getter methods
```

At the bottom of the IDE, the Console window shows the output of a terminated Java application: `<terminated> Driver (6) [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Sep 2, 2017, 9:59:40 PM)`. Below this, the text `Availiable gadgets:` is visible, indicating the start of a list of available gadgets.

Now enter debugging context and execute program



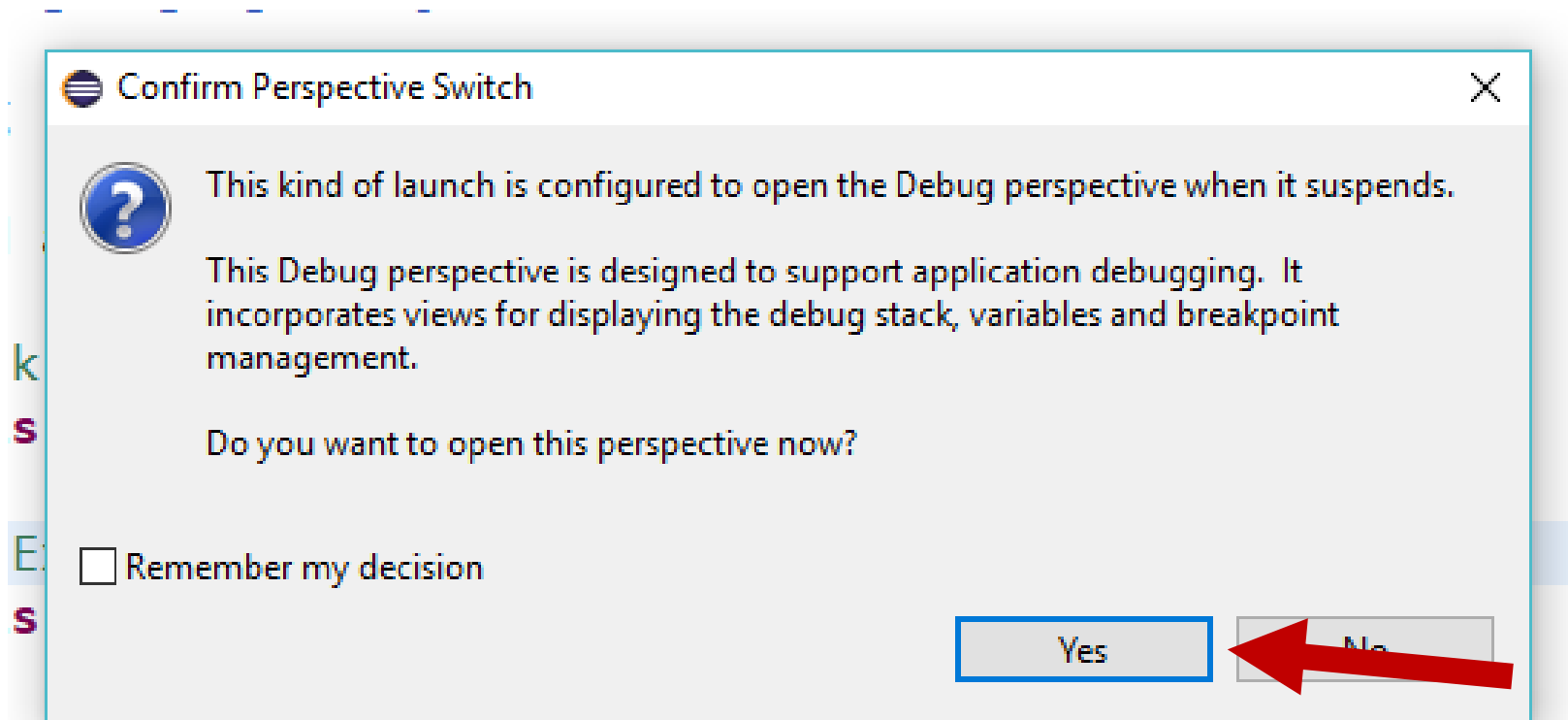
The screenshot shows the Eclipse IDE interface. The Package Explorer on the left shows a project structure with a package named 'src' containing several Java files, including 'UtilityBelt.java'. A red arrow points to the 'Run and Debug' icon in the toolbar. The main editor displays the source code of 'UtilityBelt.java' with a breakpoint set at line 59: `this.gadgets.length`. The console at the bottom shows the output: `<terminated> Driver (6) [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Sep 2, 2017, 9:59:40 PM) Available gadgets:`. The status bar at the bottom indicates the current file is 'UtilityBelt.java' and the line number is 61.

```
workspace - Java - lab2/src/UtilityBelt.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explo...
lab1
  Lab1_solution
  Lab1_Test
lab2
  src
    (default package)
      Driver.java
      Gadget.java
      GadgetTest.java
      UtilityBelt.java
      UtilityBeltTest.java
  JRE System Library [jre1.8.0_144]
  JUnit 4
lab3
lab3-solution
MyInt

49 }
50 }
51 /**
52  * Add a single gadget object to Batman's belt
53  *
54  * @param g The gadget to be added to Batman's belt
55  */
56 public void addGadget(Gadget g)
57 {
58     // Check if there is currently enough space in the belt
59     Line breakpoint:UtilityBelt [line: 59] - addGadget(Gadget) == this.gadgets.length
60     {
61         // Extend the size of the gadgets array
62         this.extendArray();
63     }
64
65     // Add the gadget into the gadgets array
66     this.gadgets[this.currentIndex] = g;
67 }
68
69 // Getter methods
```

<terminated> Driver (6) [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Sep 2, 2017, 9:59:40 PM)
Available gadgets:

Andrew H. Fagg: CS2334: Lab 5



k
S
E
S

Debugging View: Current Point in Program

The screenshot displays the Eclipse IDE in a debugging state. The main editor shows the source code of `UtilityBelt.java` with a breakpoint set at line 59, which is highlighted in green. A red arrow points to this line. The code snippet is as follows:

```
57 {
58     // Check if there is currently enough space in the belt
59     if (this.currentIndex == this.gadgets.length)
60     {
61         // Extend the size of the gadgets array
62         this.extendArray();
63     }
64
65     // Add the gadget into the gadgets array
66     this.gadgets[this.currentIndex] = g;
67 }
```

The Debug console on the left shows the current thread: `Thread [main] (Suspended (breakpoint at line 59 in UtilityBelt))`. The Variables view on the right shows the current state of variables:

Name	Value
> this	UtilityBelt (id=17)
> g	Gadget (id=18)

The Outline view on the right shows the class structure of `UtilityBelt`:

- gadgets : Gadget[]
- currentIndex : int
- UtilityBelt()
- extendArray() : void
- addGadget(Gadget) : void
- getTotalPrice() : double
- getTotalWeight() : double
- getSize() : int
- getGadgets() : Gadget[]
- weighGadgetByName(String) : double
- priceGadgetByName(String) : int
- toString() : String

The Console view at the bottom shows the output of the program: `Driver (6) [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Sep 2, 2017, 10:14:12 PM)`.

Debugging View: Current Variable Values

The screenshot displays the Eclipse IDE in a debugging state. The top-left pane shows the Debug Console with the following structure:

- Driver (6) [Java Application]
- Driver at localhost:63009
 - Thread [main] (Suspended (breakpoint at line 59 in UtilityBelt))
 - UtilityBelt.addGadget(Gadget) line: 59
 - Driver.main(String[]) line: 32

The top-right pane shows the Variables view with the following table:

Name	Value
> ● this	UtilityBelt (id=17)
> ● g	Gadget (id=18)

A red arrow points to the 'Gadget (id=18)' value. The bottom-left pane shows the code editor with the following code:

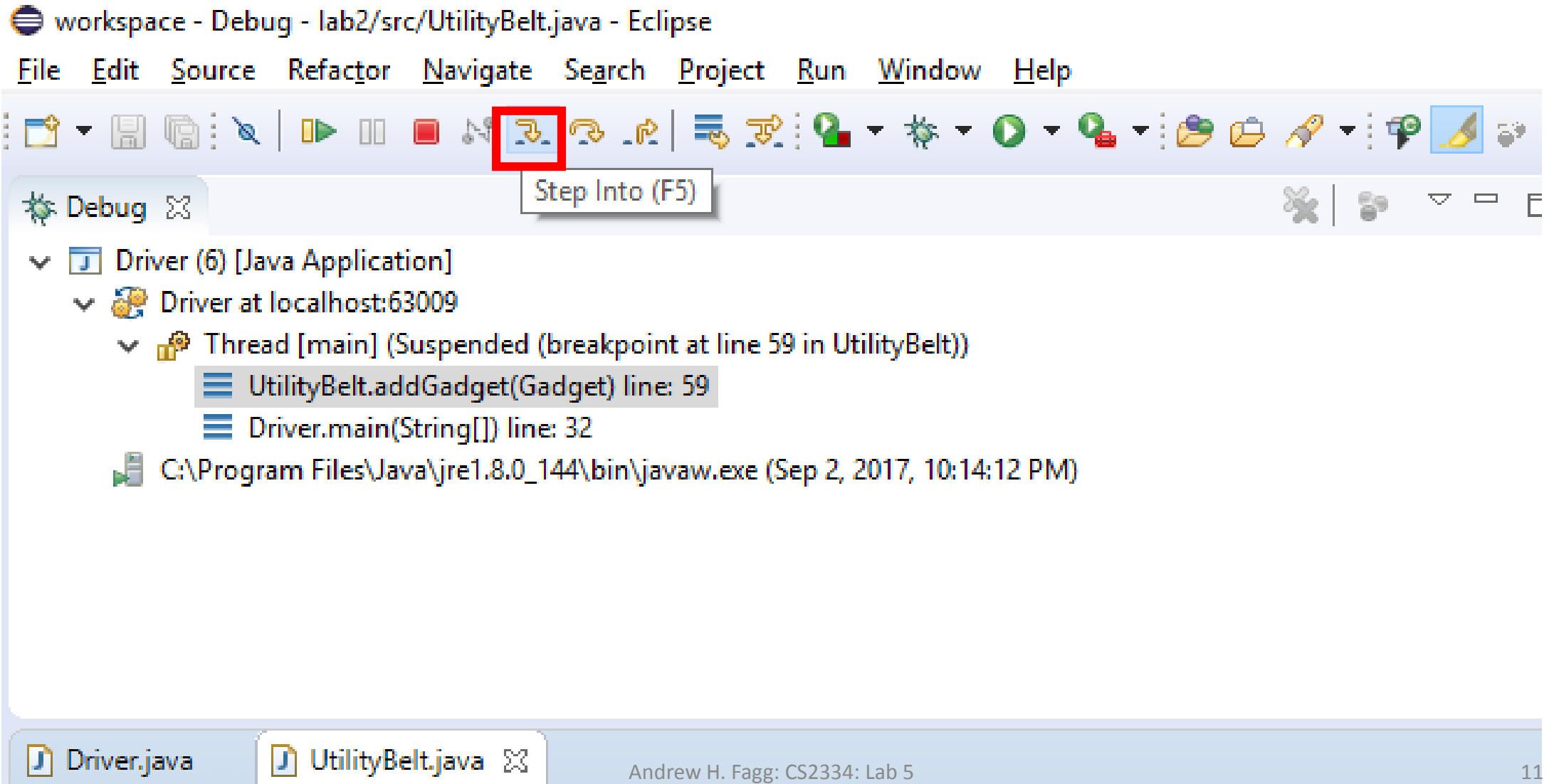
```
57 {
58     // Check if there is currently enough space in the belt
59     if (this.currentIndex == this.gadgets.length)
60     {
61         // Extend the size of the gadgets array
62         this.extendArray();
63     }
64
65     // Add the gadget into the gadgets array
66     this.gadgets[this.currentIndex] = g;
67 }
```

The bottom-right pane shows the Outline view with the following structure:

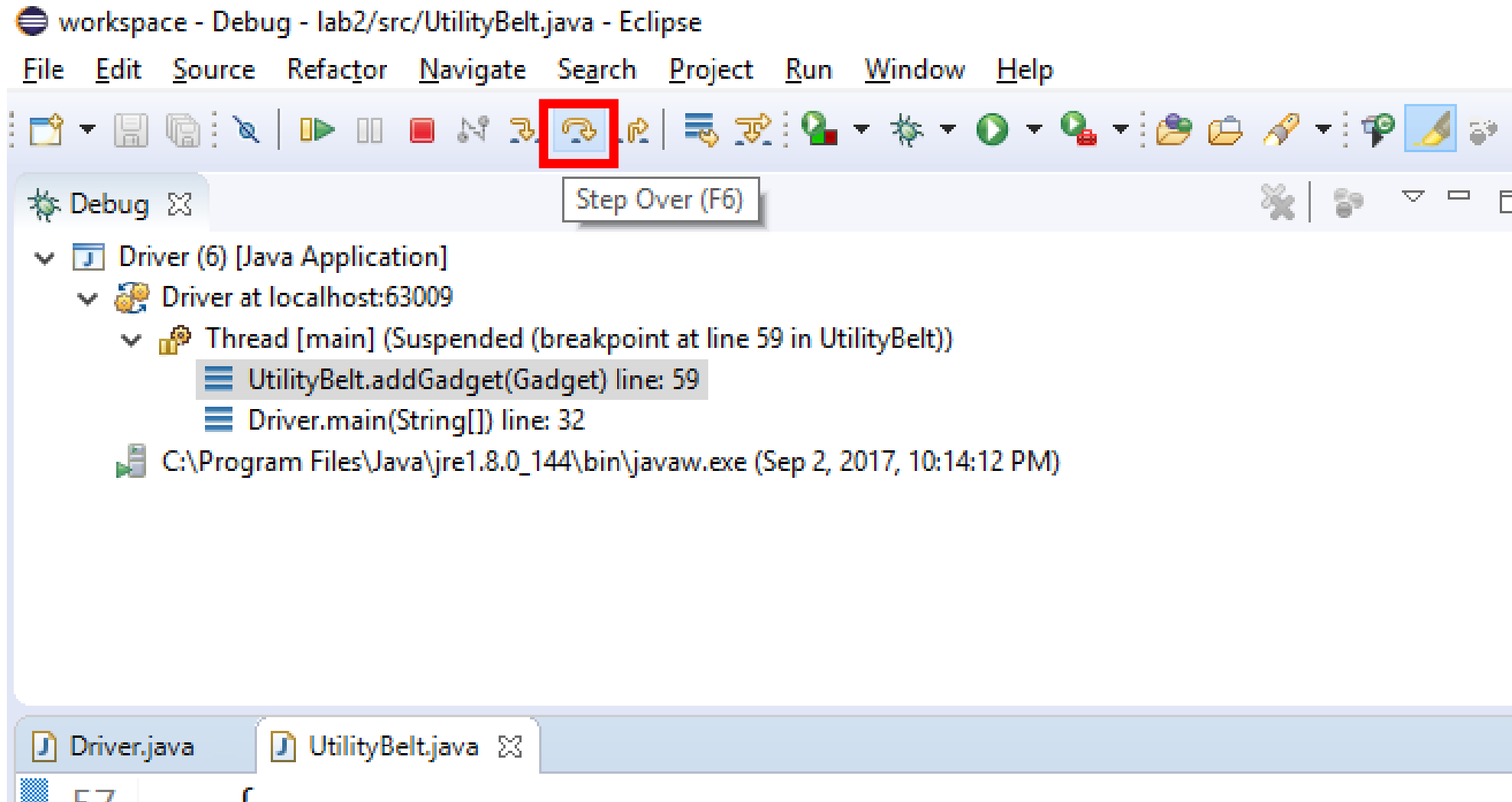
- UtilityBelt
 - gadgets: Gadget[]
 - currentIndex: int
 - UtilityBelt()
 - extendArray(): void
 - addGadget(Gadget): void
 - getTotalPrice(): double
 - getTotalWeight(): double
 - getSize(): int
 - getGadgets(): Gadget[]
 - weighGadgetByName(String): double
 - priceGadgetByName(String): int
 - toString(): String

The bottom status bar shows: Andrew H. Fagg: CS2334: Lab 5

Step Into: Execute one line of code and stop (even if that is within a method)



Step Over: Execute one line of code and stop (stay within the current method)



Step Over: now at next line

workspace - Debug - lab2/src/UtilityBelt.java - Eclipse

File Edit Source Refactor Navigate Search Project Run Window Help

Debug

- Driver (6) [Java Application]
 - Driver at localhost:63009
 - Thread [main] (Suspended)
 - UtilityBelt.addGadget(Gadget) line: 66
 - Driver.main(String[]) line: 32

C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Sep 2, 2017, 10:14:12 PM)

Variables

Name	Value
> this	UtilityBelt (id=17)
> g	Gadget (id=18)

Outline

- UtilityBelt
 - gadgets : Gadget[]
 - currentIndex : int
 - UtilityBelt()
 - extendArray() : void
 - addGadget(Gadget) : void
 - getTotalPrice() : double
 - getTotalWeight() : double
 - getSize() : int
 - getGadgets() : Gadget[]
 - weighGadgetByName(String) : double
 - priceGadgetByName(String) : int
 - toString() : String

```
65 // Add the gadget into the gadgets array
66 this.gadgets[this.currentIndex] = g;
67 }
```

Console

Driver (6) [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Sep 2, 2017, 10:14:12 PM)

Writable Smart Insert 66:1

Step Over: now at the next-next line

workspace - Debug - lab2/src/UtilityBelt.java - Eclipse

File Edit Source Refactor Navigate Search Project Run Window Help

Debug

- Driver (6) [Java Application]
 - Driver at localhost:63009
 - Thread [main] (Suspended)
 - UtilityBelt.addGadget(Gadget) line: 67
 - Driver.main(String[]) line: 32

C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Sep 2, 2017, 10:14:12 PM)

(x)= Variables Breakpoints

Name	Value
> this	UtilityBelt (id=17)
> g	Gadget (id=18)

Driver.java UtilityBelt.java

```
58 // Check if there is currently enough space in the belt
59 if (this.currentIndex == this.gadgets.length)
60 {
61     // Extend the size of the gadgets array
62     this.extendArray();
63 }
64
65 // Add the gadget into the gadgets array
66 this.gadgets[this.currentIndex] = g;
67 }
68
```

Outline

- UtilityBelt
 - gadgets : Gadget[]
 - currentIndex : int
 - UtilityBelt()
 - extendArray() : void
 - addGadget(Gadget) : void
 - getTotalPrice() : double
 - getTotalWeight() : double
 - getSize() : int
 - getGadgets() : Gadget[]
 - weighGadgetByName(String) : double
 - priceGadgetByName(String) : int
 - toString() : String

Console Tasks

Driver (6) [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Sep 2, 2017, 10:14:12 PM)

Writable Smart Insert 67 : 1

The Batarang has been added to the UtilityBelt but `currentIndex` has not been updated!

The screenshot shows the Visual Studio Code interface. The top part is the debugger window, which is split into two panes: 'Variables' and 'Breakpoints'. The 'Variables' pane shows the state of the program at a breakpoint. The 'this' object is expanded to show the following variables and their values:

Name	Value
UtilityBelt	UtilityBelt (id=17)
currentIndex	0
gadgets	Gadget[] (id=20)
[0]	Gadget (id=18)
name	"Batarang" (id=445)
price	40.2
weight	47.24
g	Gadget (id=18)

Two red arrows point to the `currentIndex` value (0) and the `name` value ("Batarang"), highlighting that the new gadget has been added but the index has not moved.

Below the debugger is the code editor, showing the following code snippet:

```
currently enough space in the belt  
== this.gadgets.length)  
  
e of the gadgets array  
);
```

The 'Outline' pane on the right shows the class structure for `UtilityBelt`:

- gadgets : Gadget[]
- currentIndex : int
- UtilityBelt()
- extendArray() : void
- addGadget(Gadget) : void
- getTotalPrice() : double
- getTotalWeight() : double

Live demonstration....