Tips about Coding Lectures and Debugging Instructions

#### **Welcome to the Coding Lectures**

Welcome to the Java Multithreading, Concurrency & Performance Optimization course. I am very excited to have you onboard and to give you a great learning experience.

For your convenience, from this point on, every lecture that involves a coding example also includes a downloadable resource with the source code for that particular lecture.  
To get the most out of the course it is **highly recommended** for you to download the source code and experiment with it in order to get comfortable with the concepts discussed in the lecture.

#### **Important Tips for following along the Coding Lectures**

Following and understanding the practical examples is crucial to your success.

It's important to acknowledge (and celebrate) that we are all different.  
We have different levels of experience with Java. We all think and read at a different pace, and for some of us, English is a second language.

The great news are: You are the pilot and you are in full control!  
As part of being an active student, you should use the video player buttonsto decide the right speed and pace for you.

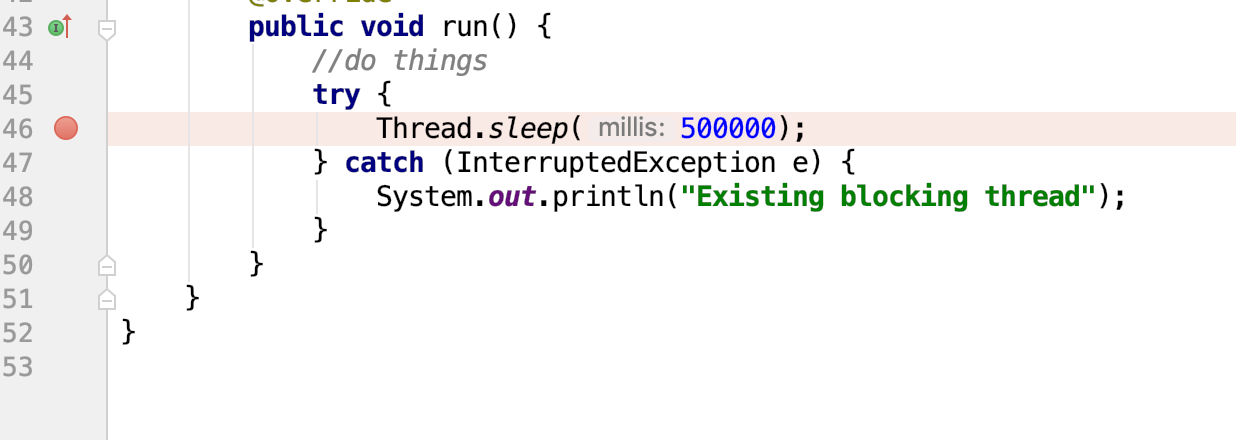
I encourage you to pause, replay and slow down at any point you feel necessary, as many times as necessary.

This is a normal process and is the best part about being a student on Udemy.

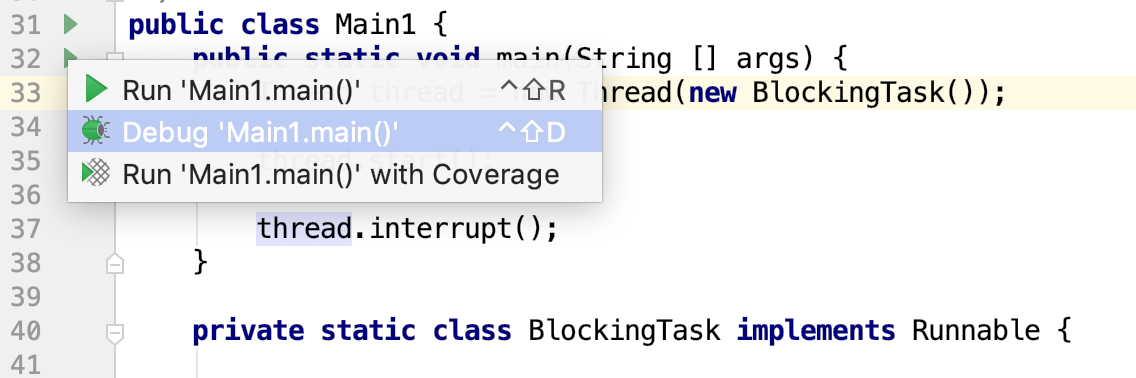
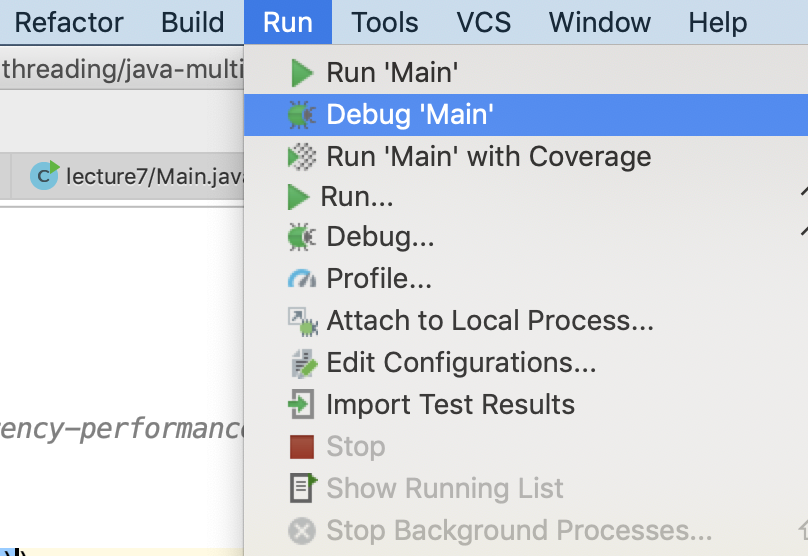
If in the end of the lecture you still have questions, I am here to help you in the Q&A section.  
  
And most importantly.  
Have fun!

#### **Debugging Java Threads in IntelliJ IDE**

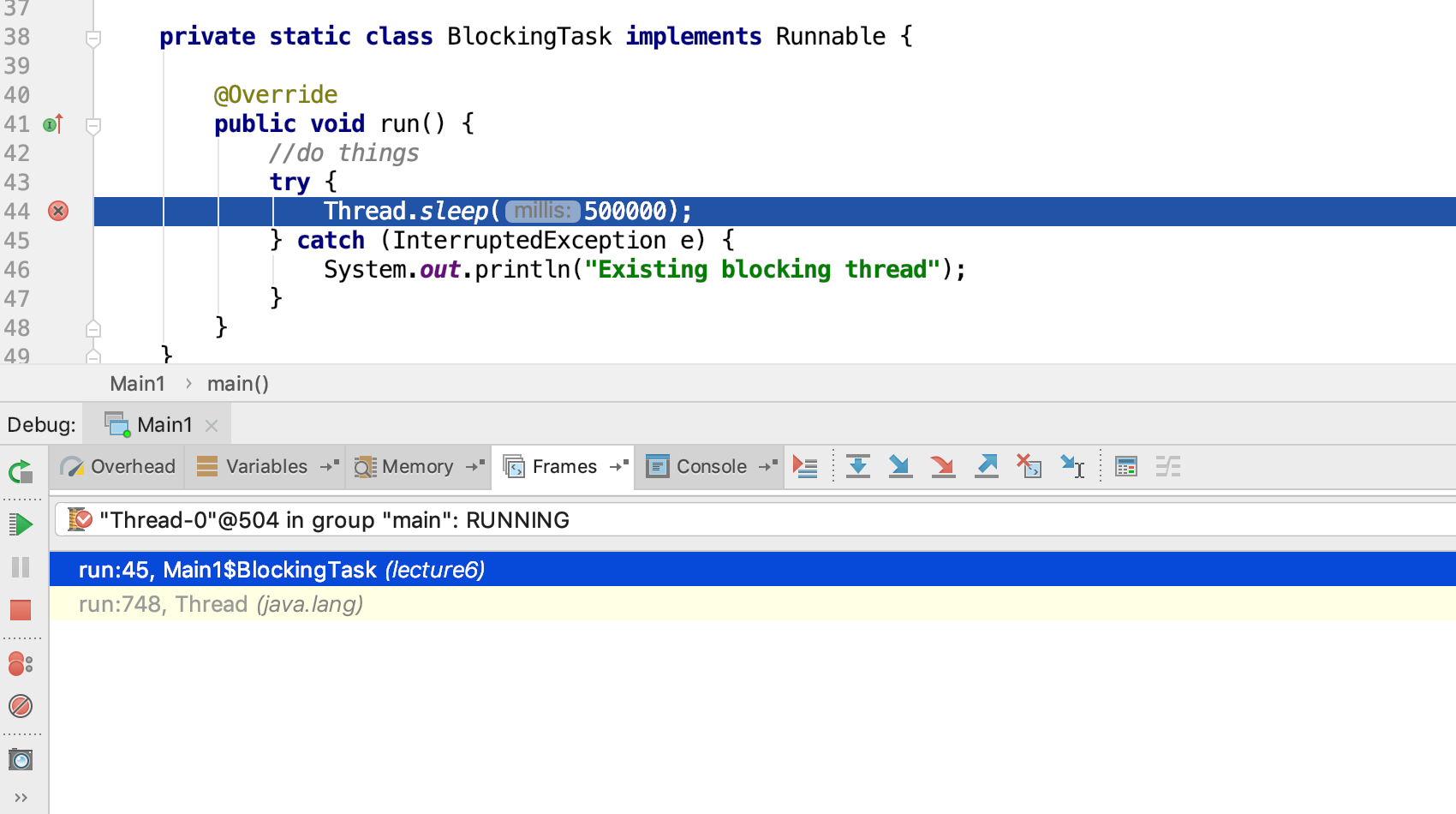
1. Set a breakpoint anywhere in your code. To set a breakpoint, right click on the line where you want the debugger to stop the execution of your application. The breakpoint appears as a red circle.



2. Go to Run -> Debug 'XXX' or right click on the green arrow next to the main() method and choose the Debug option



3. Once the flow of execution reaches the line of code where the breakpoint was set, the program execution will stop and the debug window should appear where you can inspect the state of your application.

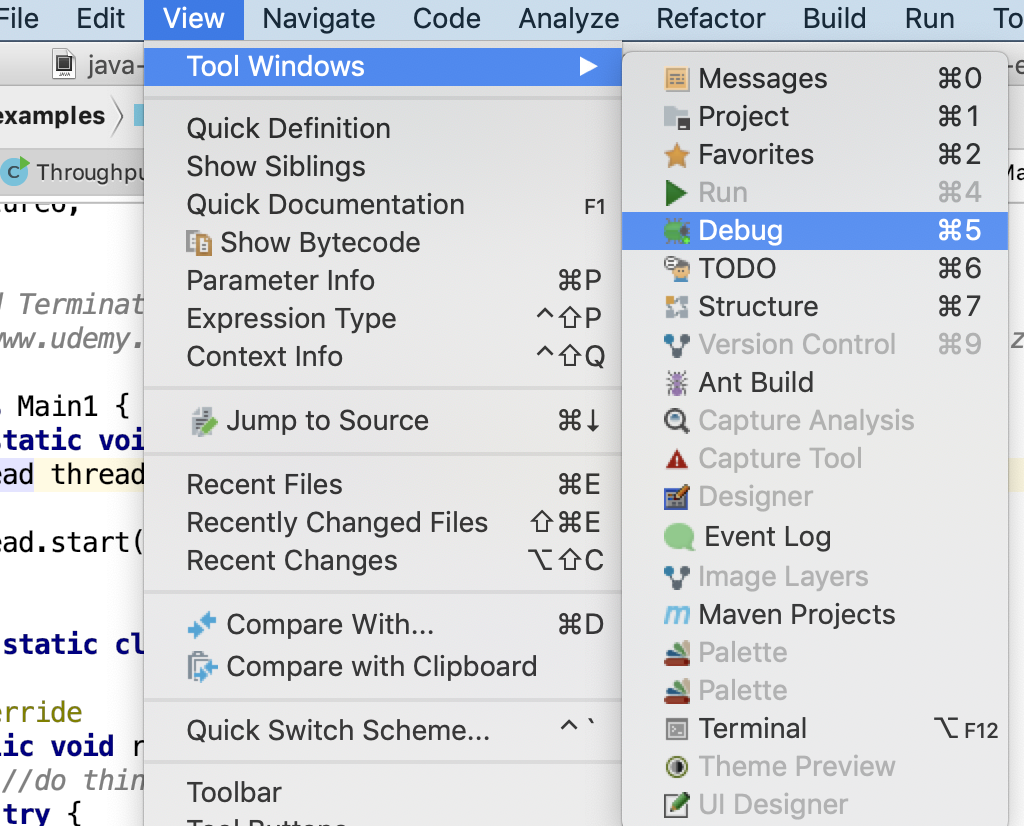


#### **FAQ**

**Debug Window is not showing up**

Sometimes this Debug view might be hidden.

To enable that view go to View -> Tool Windows -> Debug

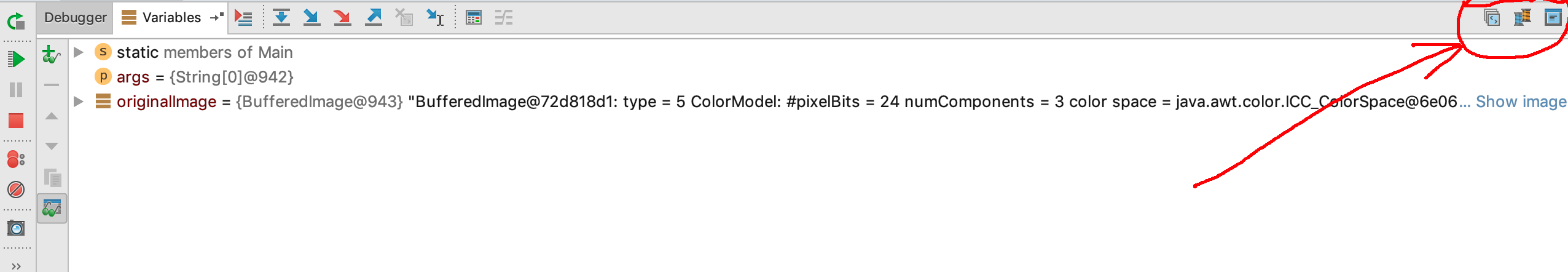
* 

**Not all Debug views are available**

The available windows for debugging a thread change from one version of Intellij to another.

The typical options are Variables, Frames and Console.

Sometimes all or some of those tabs are hidden.  
To make them all appear in the Debug window, click on their corresponding icon at the right most corner of the Debug window:



Resources for this lecture

* [all\_code\_examples.zip](javascript:void(0))