

Applets

Overview

In this lab, you'll review the functionality of an existing Swing desktop application, and then create an equivalent applet that can be opened in a browser.

Source folders

Student project: **StudentApplets**
Solution project: **SolutionApplets**

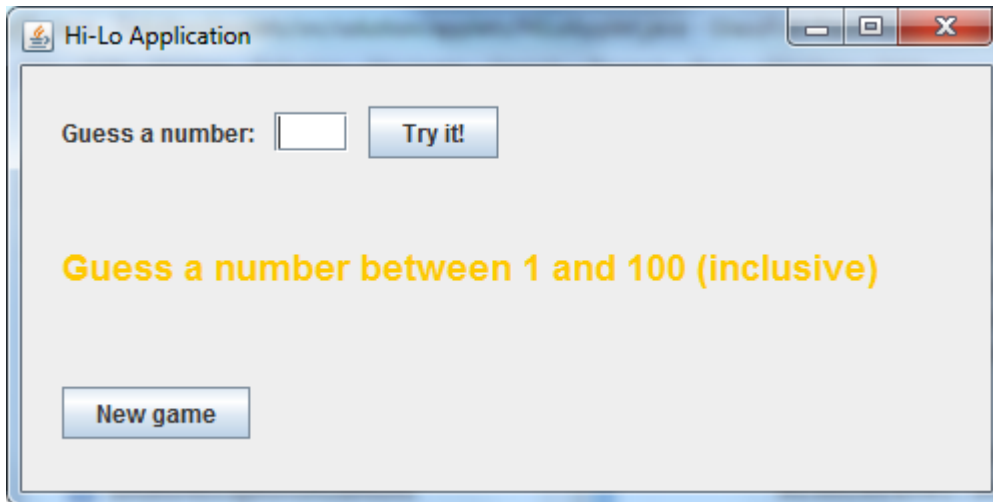
Roadmap

There are 4 exercises in this lab, of which the last exercise is "if time permits". Here is a brief summary of the tasks you will perform in each exercise; more detailed instructions follow later:

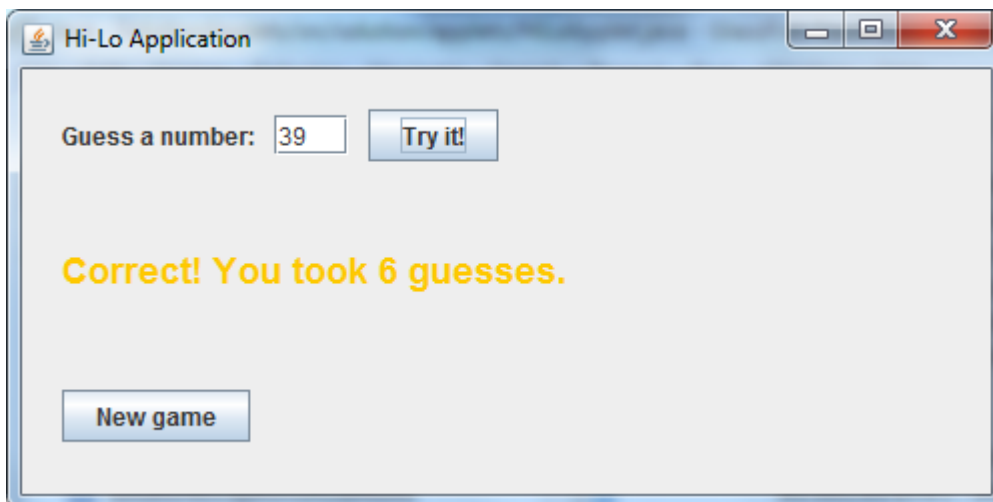
1. Familiarization with the Swing desktop application
2. Creating an equivalent Swing applet
3. Hosting the applet in an HTML page
4. Additional suggestions

Exercise 1: Familiarization with the Swing desktop application

Open the *student* project. The project contains a prewritten Swing desktop application class named `HiLoWindow`. Run this class, to display a Swing application window as follows:



The application is a simple "high-low" game. You have to guess a number between 1 and 100. For example, enter 50 in the text box and click the *Try it!* button. The window will display a message indicating whether the target number is higher or lower. When you eventually guess correctly, the application will tell you how many guesses you took:



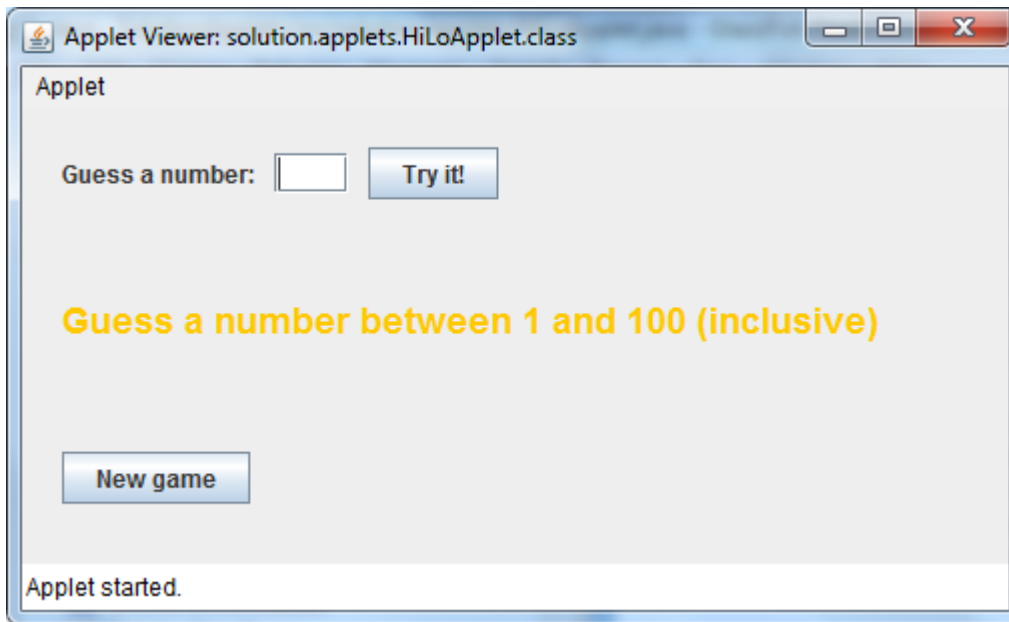
You can then click *New game* to generate a new random number and to reset your "guess count" to zero, ready for a new game.

When you're happy with how everything works, close the application. Review the code in the *student* Swing application, and make sure you understand how it all works.

Exercise 2: Creating an equivalent Swing applet

Create a new class named `HiLoApplet`, and implement a Swing applet equivalent to the desktop application you just played with. You'll be able to port most of the code directly from the `HiLoWindow` class, but you'll need to do a certain amount of refactoring to fit into the applet lifecycle.

You can test the applet in AppletViewer for now (i.e. within Eclipse, right-click on the applet class, and select **Run As | Java Applet**). The applet should appear as follows, and should be functionally complete:



Exercise 3: Hosting the applet in an HTML page

Add an HTML page to your project, to host your Java applet. Here are some hints:

- Package your applet into a JAR file.
- In the HTML page, refer to the applet in the JAR file.
- To test everything works, make sure the HTML page and the JAR file are located in the same folder on your machine, and then open the HTML page in a browser window.

Exercise 4 (If time permits): Additional suggestions

- Add an applet parameter in the HTML page, to specify the upper limit for the random-number generator (e.g. set the value to 300). Modify the applet code so that it attempts to get this parameter value, and reverts to a default value of 100 if no such parameter is defined).
- Take a look at the following URL, which provides information about various advanced applet techniques:

<http://download.oracle.com/javase/tutorial/deployment/applet/index.html>