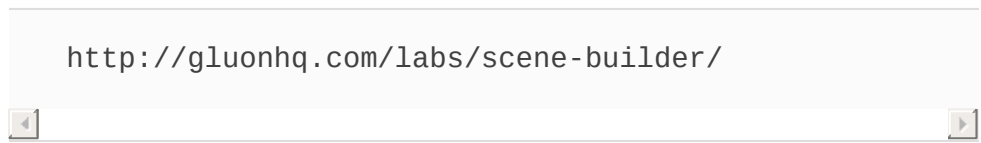# 12.2 JavaFX Scene Builder

Most Java textbooks that introduce GUI programming provide hand-coded GUIs—that is, the authors build the GUIs from scratch in Java code, rather than using a visual GUI design tool. This is due to the fractured Java IDE market—there are many Java IDEs, so authors can't depend on any one IDE being used, and each generates different code.

JavaFX is organized differently. The **Scene Builder** tool is a standalone JavaFX GUI visual layout tool that can also be used with various IDEs, including the most popular ones—Eclipse, IntelliJ IDEA and NetBeans. You can download Scene Builder at:

```
http://gluonhq.com/labs/scene-builder/
```

JavaFX Scene Builder enables you to create GUIs by dragging and dropping GUI components from Scene Builder's library onto a design area, then modifying and styling the GUI—all without writing any code. JavaFX Scene Builder's live editing and preview features allow you to view your GUI as you create and modify it, without compiling and running the app. You can use **Cascading Style Sheets (CSS)** to change the entire look-and-feel of your GUI—a concept sometimes called **skinning**. In Chapter 22, we'll introduce styling with CSS.

FXML (FX Markup Language)

As you create and modify a GUI, JavaFX Scene Builder generates **FXML (FX Markup Language)**—an XML vocabulary for defining and arranging JavaFX GUI controls without writing any Java code. XML (eXtensible Markup Language) is a widely used language for describing things—it's readable both by computers and by humans. In JavaFX, FXML concisely describes GUI, graphics and multimedia elements. *You do not need to know FXML or XML to study this chapter.* As you'll see in Section 12.4, JavaFX Scene Builder hides the FXML details from you, so you can focus on defining *what* the GUI should contain without specifying *how* to generate it—this is an example of *declarative programming*.

Software Engineering Observation 12.1

The FXML code is separate from the program logic that's defined in Java source code—this separation of the interface (the GUI) from the implementation (the Java code) makes it easier to debug, modify and maintain JavaFX GUI apps.