12.4 Welcome App—Displaying Text and an Image

In this section, *without writing any code*, you'll build a GUI that displays text in a `Label` and an image in an `ImageView` (Fig. 12.2). You'll use visual-programming techniques to *drag-and-drop* JavaFX components onto Scene Builder's content panel—the design area. Next, you'll use Scene Builder's **Inspector** to configure options, such as the `Label`s' text and font size, and the `ImageView`'s image. Finally, you'll view the completed GUI using Scene Builder's **Show Preview in Window** option. In Section 12.5's **Tip Calculator** app, we'll discuss the Java code necessary to load and display an FXML GUI. Then, in Exercise 12.3, you'll create the Java application that displays the **Welcome** GUI you build in this section.

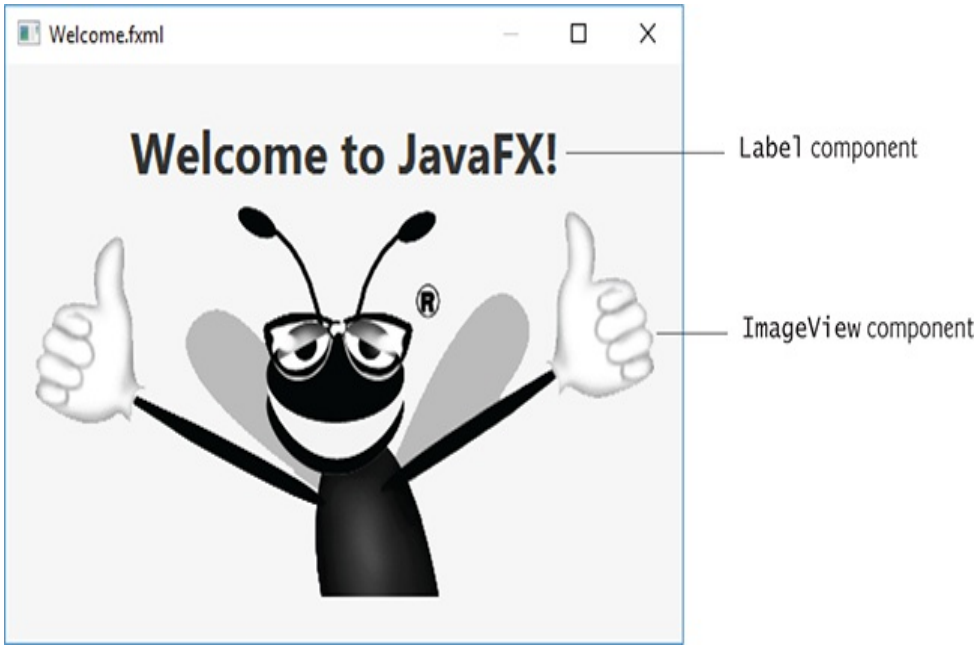

Fig. 12.2

Final **Welcome** GUI in a preview window on Microsoft Windows 10.

Description

12.4.1 Opening Scene Builder and Creating the File `Welcome.fxml`

Open Scene Builder so that you can create the FXML file that defines the GUI. The window initially appears as shown in Fig. 12.3. **Untitled** at the top of the window indicates that

Scene Builder has created a new FXML file that you have not yet saved.1 Select **File > Save** to display the **Save As** dialog, then select a location in which to store the file, name the file `Welcome.fxml` and click the **Save** button.

1. We show the Scene Builder screen captures on Microsoft Windows 10, but Scene Builder is nearly identical on Windows, macOS and Linux. The key difference is that the menu bar on macOS is at the top of the screen, whereas the menu bar is part of the window on Windows and Linux.

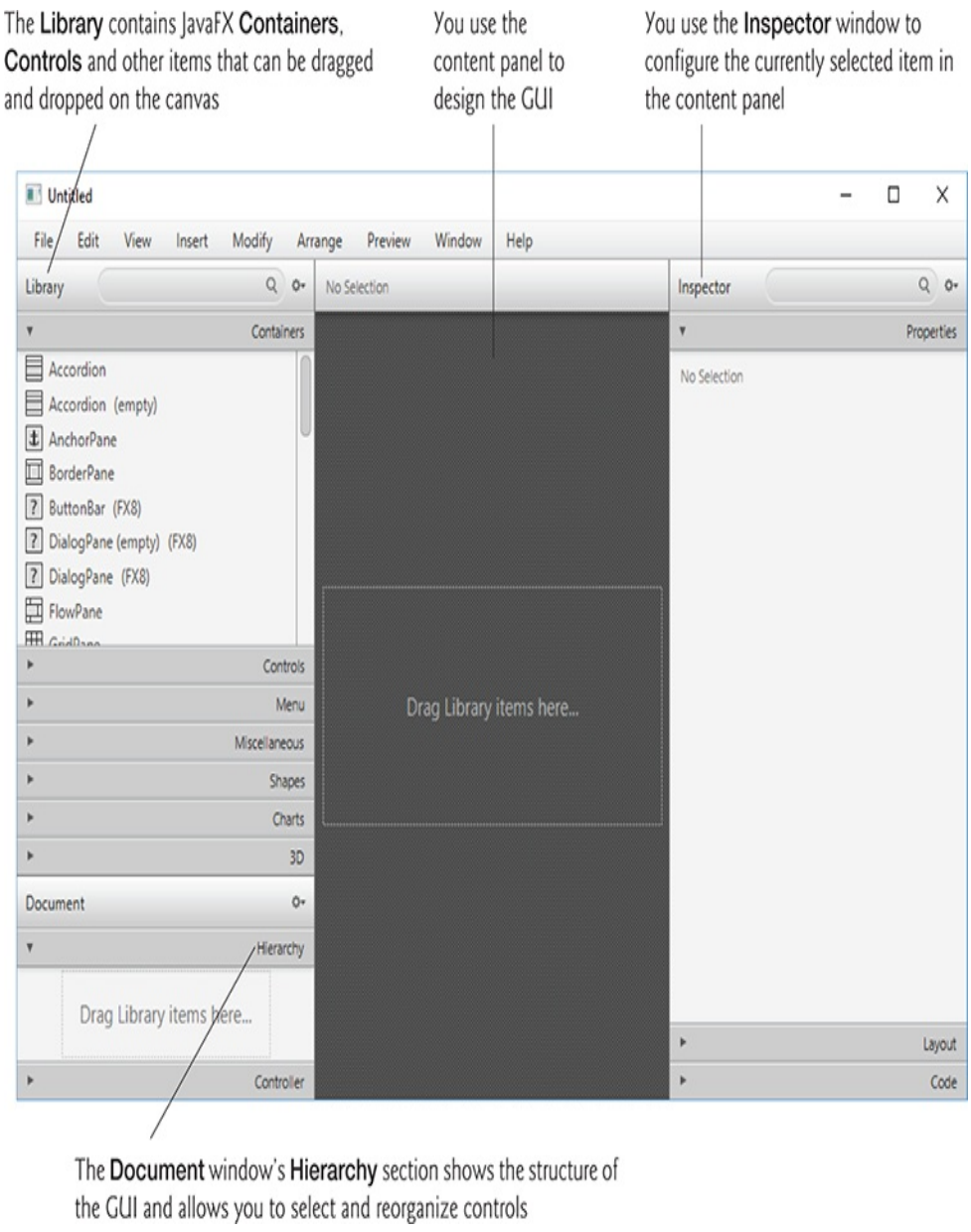

Fig. 12.3

JavaFX Scene Builder when you first open it.

Description

12.4.2 Adding an Image to the Folder Containing `Welcome.fxml`

The image you'll use for this app (`bug.png`) is located in the `images` subfolder of this chapter's examples folder. To make it easy to find the image when you're ready to add it to the app, locate the `images` folder on your file system, then copy `bug.png` into the folder where you saved `Welcome.fxml`.

12.4.3 Creating a `VBox` Layout Container

For this app, you'll place a `Label` and an `ImageView` in a `VBox` **layout container** (package `javafx.scene.layout`), which will be the scene graph's root node. Layout containers help you arrange and size GUI components. A `VBox` arranges its nodes *vertically* from top to bottom. We discuss the `GridPane` layout container in Section 12.5 and several others in Chapter 13. To add a `VBox`

to Scene Builder's content panel so you can begin designing the GUI, double-click **VBox** in the **Library** window's **Containers** section. (You also can drag-and-drop a `VBox` from the **Containers** section onto Scene Builder's content panel.)

12.4.4 Configuring the `VBox` Layout Container

You'll now specify the `VBox`'s alignment, initial size and padding.

Specifying the `VBox`'s Alignment

A `VBox`'s **alignment** determines the layout positioning of the `VBox`'s children. In this app, we'd like each child node (the `Label` and the `ImageView`) to be centered horizontally in the scene, and we'd like both children to be centered vertically, so that there is an equal amount of space above the `Label` and below the `ImageView`. To accomplish this:

1. Select the `VBox` in Scene Builder's content panel by clicking it. Scene Builder displays many `VBox` properties in the Scene Builder **Inspector**'s **Properties** section.
2. Click the **Alignment** property's drop-down list and notice the variety of potential alignment values you can use. Click `CENTER` to set the

Alignment.

Each property value you specify for a JavaFX object is used to set one of that object's instance variables when JavaFX creates the object at runtime.

Specifying the `VBox`'s Preferred Size

The **preferred size** (width and height) of the scene graph's root node is used by the scene to determine its window size when the app begins executing. To set the preferred size:

1. Select the `VBox`.
2. Expand the **Inspector**'s **Layout** section by clicking the right arrow () next to **Layout**. The section expands and the right arrow changes to a down arrow. Clicking the arrow again would collapse the section.
3. Click the **Pref Width** property's text field, type `450` and press *Enter* to change the preferred width.
4. Click the **Pref Height** property's text field, type `300` and press *Enter* to change the preferred height.

12.4.5 Adding and Configuring a `Label`

Next, you'll create the `Label` that displays `"Welcome to JavaFX!"`.

Adding a `Label` to the `VBox`

Expand the Scene Builder **Library** window's **Controls** section by clicking the right arrow (▶) next to **Controls**, then drag-and-drop a **Label** from the **Controls** section onto the `VBox` in Scene Builder's content panel. (You also can double-click **Label** in the **Containers** section to add the `Label`.) Scene Builder automatically centers the `Label` object horizontally and vertically in the `VBox`, based on the `VBox`'s **Alignment** property.

Changing the `Label`'s Text

You can set a `Label`'s text either by double clicking it and typing the new text, or by selecting the `Label` and setting its **Text** property in the **Inspector**'s **Properties** section. Set the `Label`'s text to `"Welcome to JavaFX!"`.

Changing the `Label`'s Font

For this app, we set the `Label` to display in a large bold font. To do so, select the `Label`, then in the **Inspector**'s **Properties** section, click the value to the right of the **Font** property. In the window that appears, set the **Style** property to `Bold` and the **Size** property to `30`. The design should now

appear as shown in Fig. 12.4.

Fig. 12.4

Welcome GUI’s design after adding and configuring a `Label`.

Description

12.4.6 Adding and Configuring an `ImageView`

Finally, you’ll add the `ImageView` that displays `bug.png`.

Adding an `ImageView` to the `VBox`

Drag and drop an **ImageView** from the **Library** window's **Controls** section to just below the `Label`, as shown in Fig. 12.5. You can also double-click **ImageView** in the **Library** window, in which case Scene Builder automatically places the new `ImageView` object below the `Label`. You can reorder a `VBox`'s controls by dragging them in the `VBox` or in the **Document** window's **Hierarchy** section (Fig. 12.3). Scene Builder automatically centers the `ImageView` horizontally in the `VBox`. Also notice that the `Label` and `ImageView` are centered vertically such that the same amount of space appears above the `Label` and below the `ImageView`.

Setting the `ImageView`'s Image

Next you'll set the image to display:

1. Select the `ImageView`, then in the **Inspector**'s **Properties** section click the ellipsis (**…**) button to the right of the **Image** property. By default, Scene Builder opens a dialog showing the folder in which the FXML file is saved. This is where you placed the image file `bug.png` in Section 12.4.2.
2. Select the image file, then click **Open**. Scene Builder displays the image and resizes the `ImageView` to match the image's aspect ratio—that is, the ratio of the image's width to its height.

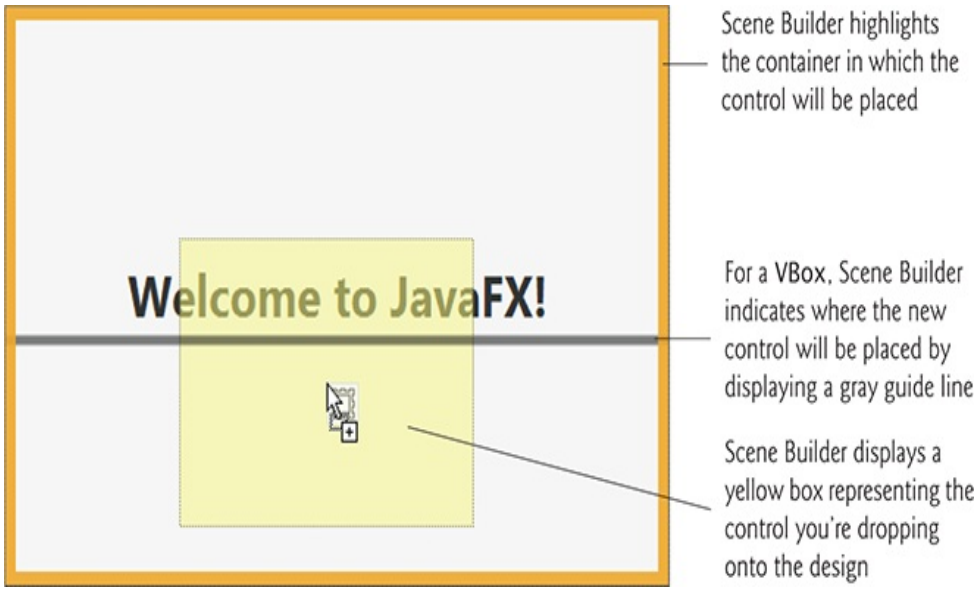

Fig. 12.5

Dragging and dropping the `ImageView` below the `Label`.

Description

Changing the `ImageView`'s Size

We'd like to display the image at its original size. If you reset the `ImageView`'s default **Fit Width** and **Fit Height** property values—which Scene Builder set when you added the `ImageView` to the design—Scene Builder will resize the `ImageView` to the image's exact dimensions. To reset these properties:

1. Expand the **Inspector**'s **Layout** section.
2. Hover the mouse over the **Fit Width** property's value. This displays the button  to the right property's value. Click the button and select **Reset to Default** to reset the value. This technique can be used with any property value to reset its default.
3. Repeat *Step 2* to reset the **Fit Height** property's value.

You've now completed the GUI. Scene Builder's content panel should now appear as shown in Fig. 12.6. Save the FXML file by selecting **File > Save**.

Fig. 12.6

Completed **Welcome** GUI in Scene Builder's content panel.

Description

12.4.7 Previewing the `Welcome` GUI

You can preview what the design will look like in a running application's window. To do so, select **Preview > Show Preview in Window**, which displays the window in Fig. 12.7.

Fig. 12.7

Previewing the **Welcome** GUI on Microsoft Windows 10—only the window borders will differ on Linux, macOS and earlier Windows versions.

Description