Completing the App

In this chapter, I will complete the basic version NoteFlash example app that I started in Chapter 3. I continue to use approaches and techniques that have a lot in common with regular web app development, but I also start to mix in more Windows-specific functionality that is available through the WinJS API. I provide a brief overview of each Windows app feature as I use it and explain where in the book you can get more details.

Revisiting the Example App

In this chapter, I will build directly on the NoteFlash project from Chapter 3. As you will recall, I put the basic structure of the app in place, defined a navigation function, and defined the styles that will be applied throughout the app. I also used the Page Control item template to generate a set of related HTML, CSS, and JavaScript files I used to create the content that allows the user to select the notes to be tested on. You can see how that turned out in Figure 4-1. In this chapter, I’ll create additional content that will perform the testing based on the user’s selection.

Figure 4-1. The NoteFlash selector page
 DEFINING THE NOTES DATA

The next step in building out the example app is to define the notes the user will be tested on. To do this, I have added a new JavaScript file to the js folder called notes.js, the content of which you can see in Listing 4-1. (Right-click on the js folder in the Solution Explorer, select Add ➤ New Item, and use the JavaScript File item template.)

Tip Add a JavaScript file by right-clicking on the js folder in the Solution Explorer window and selecting Add ➤ New Item from the pop-up menu. Select JavaScript File from the list of file types, set the name of the file to be notes.js, and click the Add button.

Listing 4-1. Defining the note data

(function () {
    "use strict";

    var Note = WinJS.Class.define(function (note, character, hand) {
        this.note = note;
        this.character = character;
        this.hand = hand;
    });

    WinJS.Namespace.define("Notes", {
        leftHand: [
            new Note('C', 80, "left"), new Note('D', 81, "left"),
            new Note('E', 82, "left"), new Note('F', 83, "left"),
            new Note('G', 84, "left"), new Note('A', 85, "left"),
            new Note('B', 86, "left"), new Note('C', 87, "left"),
            new Note('D', 88, "left"), new Note('E', 89, "left"),
            new Note('F', 90, "left"), new Note('G', 91, "left"),
            new Note('A', 92, "left"), new Note('B', 93, "left"),
            new Note('C', 94, "left")
        ],
        rightHand: [
            new Note('C', 82, "right"), new Note('D', 83, "right"),
            new Note('E', 84, "right"), new Note('F', 85, "right"),
            new Note('G', 86, "right"), new Note('A', 87, "right"),
            new Note('B', 88, "right"), new Note('C', 89, "right"),
            new Note('D', 90, "right"), new Note('E', 91, "right"),
            new Note('F', 92, "right"), new Note('G', 93, "right"),
            new Note('A', 94, "right")
        ],
    });

    }());

This file requires some explanation. I have employed two useful features from the WinJS API: classes and namespaces. I explain each of them in the following sections.