

```

else
{
    select_File_Process();
}

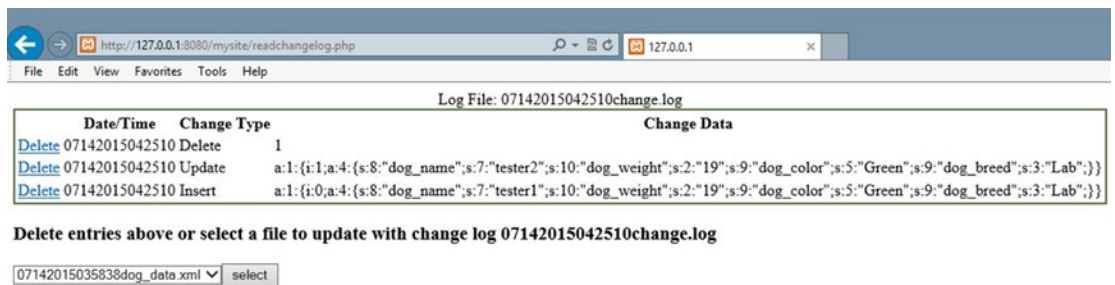
```

Since the code has grown in length, it will be much easier to follow the logical flow (and to modify the code when necessary) if much of the work is done in methods. As with many applications, the main flow of the program becomes one embedded if else statement.



**Figure 6-1.** The `readchangelog.php` file requesting selection of a change log file

Upon opening the application, a list box will allow the users to select which valid change log file to use (and possibly update). The else portion of the statement will direct the program to `select_File_Process`, which will handle this request.



**Figure 6-2.** The `readchangelog.php` file displaying the selected log and requesting the valid data file

Once the user has selected a change file, the contents of the file will be displayed in the same manner as shown in Chapter 5. The `display_Process` method will provide this information. The user can decide to remove some entries in the change file selected. If deletions are needed, the `delete_Process` method will complete the process, using the same techniques shown in Chapter 5. In addition, the same method will allow the user to select the most recent valid data file to apply changes.

Once the data file has been selected, `update_XML_File_Process` will apply the changes to the file using the `dogdata` program (in Example 6-3). The process will display a changes completed message to the user.

The `select_File_Process` method uses similar logic as found in the `getbreeds.php` program (in Example 4-5).