

In Example 4-1, the format of the form tag will automatically cause a response if a true or false is returned from the `validate_input` method. If true is returned (there were no validation problems), the HTML program will send the form properties and values to the `lab.php` program on the web server. If false is returned (there was at least one validation problem), the HTML program will not pass the information and will stay visible in the browser. The alert box will display for the user to see what errors have occurred. When the user closes the alert box, they will still be on the HTML page (with the information they had previously entered). They can correct any incorrect information and click the Submit button (again) to revalidate the information they have entered.

The screenshot shows a web browser window titled "Dog Object" with the address bar displaying "127.0.0.1:8080/mysite/lab.html". Below the browser window, the page content is as follows:

Dog Object Creator

Please complete ALL fields. Please note the required format of information.

Your Dog's Name (max 20 characters, alphabetic)

Your Dog's Breed (max 14 characters, alphabetic)

Your Dog's Color (max 14 characters, alphabetic)

Your Dog's Weight (numeric only)

Figure 4-2. The `lab.html` file with incorrect entries

The screenshot shows the same web browser window as Figure 4-2, but with an alert box displayed. The form fields and the "Click to create your dog" button are still visible in the background. The alert box contains the following text:

Invalid dog name. Invalid dog breed. Invalid dog color. Invalid dog weight.

Figure 4-3. The `lab.html` file after failing JavaScript verification