

The all alphabetic method uses a *regular expression* to check for alphabetic characters. Most program languages (including PHP) enable the programmer to create a regular expression that will check a string for a required format.

```
var letters = /^[a-zA-Z ]+$/;
```

This line sets up the expression to determine if the string only has lowercase and/or uppercase alphabetic characters (a through z, A through Z). You will work more in detail with regular expressions within PHP in a later chapter.

For more information on JavaScript regular expressions, visit

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Regular_Expressions

For more information on the JavaScript match method, visit

http://www.w3schools.com/jsref/jsref_match.asp

One of the methods that exists in all JavaScript strings is the match method. This method compares the contents of the string (the `_string`) to the regular expression (alphabetic characters in this example). If the string meets the requirements of the expression (only contains alphabetic characters), then the method will return true. Otherwise, it will return false. In this example, if the string is alphabetic, true is returned to the if statement in the `validate_dog_name` (or `validate_dog_color` or `validate_dog_breed`) method. If any part of the string contains something other than alphabetic characters, a false is returned to the if statement (which in turn will make the results of the if statement false even if the string size is correct).

Going back to the bottom of the JavaScript code, you determine what to return to the HTML program (`lab.html`).

```
if (!validate_dog_weight(form.dog_weight.value))
    {
        error_message += "Invalid dog weight. ";
    }
if (error_message.length > 0)
    {
        alert(error_message);
        return false;
    }
else
    {
        return true;
    }
```

Each if statement at the bottom of the `validate_input` method will build an error message (such as "Invalid dog weight.") and place it in the `error_message` property if the validation fails (if a false is returned from the if statement). The `+=` symbol concatenates what is currently in the property with what is being placed into the property (so you don't overwrite any messages that are already there).

The last if statement (`if (error_message.length > 0)`) checks the length of the `error_message` property. If it is greater than zero (error messages have been passed into the property), the program will display the messages in an alert box and return false to the HTML program (`lab.html`). If the length is zero, there were no error messages; true is passed back to the HTML program.

```
<form method="post" action="lab.php" onSubmit="return validate_input(this)">
```