Client-Side Form Validation

You should never rely upon any validation of user input on the client-side. The main purpose of client-side validation of user input is to improve the user experience of using your form. Immediate feedback on an invalid input is generally speaking much better than having to go back to correct something after you have tried to submit your form.

HTML 5 Form Validation

HTML5 comes with a lot of form validation functionality that you can make use of by adding attributes to your form elements.

http://www.w3.org/community/webed/wiki/HTML5_form_additions#Validation provides a very brief overview of the new validation features that are available.

https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/Forms/Data_form_validation provides a good description of HTML5 validation.

Example Code

The following HTML shows three input fields that are automatically validated.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>Form Example</title>
</head>
<body>
<header><h1>Example</h1></header>
<section>
  <form method="get" action="Process.php">
    <fieldset>
      <legend>HTML5 form input types with built in validation</legend>
      <p><label>URL: <input type="url" name="url"></label></p>
      <fieldset id="userDetails">
        <legend>Validating immediately with JavaScript</legend>
        <p><label>Number: <input type="number" name="number" id="number"
min="2" max="5"><span id="numMsg"></span></label></p>
        <p><label>Email: <input type="email" id="email" name="email">
<span id="emailMsg"></span></label></p>
      </fieldset>
    </fieldset>
    <p><input type="submit" id="submit"></p>
    <p><input type="reset"></p>
  </form>
</section>
<script src="Content/emailValidator.js"></script>
</body>
</html>
```

Save the above code and view it in a browser. The <input type="number"> element also has *min* and *max* attributes set. Put in different values and press the submit button. If you enter invalid values (a letter in the number field or a string that is not an email in the email field) you will get a message.

The following code sets that message to appear in the *<span>* element next to the bad input as soon as you move away from that field.

```
window.onload=initialise;

function initialise()
{
  registerValidation(
    document.getElementById('email'),
    document.getElementById('emailMsg')
  );

  registerValidation(
    document.getElementById('number'),
    document.getElementById('numMsg')
  );
}

function registerValidation(element, span)
{
  element.onblur = function() {
    span.textContent = element.validationMessage;
  }
}
```

Look at the JavaScript validation code to see how it is working. Can you add to it so that a message appears immediately for the *<input type="url">* element also?