Web Site Design and Development

Lecture 24

CS 0134
Fall 2018
Tues and Thurs
1:00 – 2:15PM

Useful CSS for forms

- **Given the nature of form controls, they come with a lot of default styling but there are still some useful styling options we can use**
- **Aligning**
  - If you have a form made up of a series of labels and their fields, it is often nice to have all of the fields line up. You can do this by styling the label elements with this code from the book

    -
      ```
      label {
              float: left; /* puts all the labels to the left of their fields */
              width: 4em; /*makes all of the labels the same width */
              text-align: right; /*lines the label text all up to the right */
      }
      ```

Useful CSS for forms continued

  - You can also style the input elements to give a desired amount of space between the label and input with margin-left and a desired amount of space between inputs with margin-bottom
- **Style focused elements**
  - You can style focused elements but using the :focus pseudo-class
- **Style all elements**
  - You can set, among other CSS properties, the border, background and color properties on most form controls to make your form fit more closely with your overall design

Data validation

- **HTML5 provides some attributes for data validation**
  - autocomplete: sets if a field should be allowed to auto-complete. This can be set to “on” or “off”. “on” is the default in all modern browsers
  - required: boolean attribute that sets if a field is required to have a value
  - novalidate: boolean attribute that tells the browser not to validate a field or form

Data validation continued

- **CSS pseudo-classes for formatting validation states**
  - :required will match all required fields, can also use input[required]
  - :valid will match all valid fields
  - :invalid will match all invalid fields
- **HTML5 also introduced Regular expressions**

Regular Expressions

- A Regular Expression is a pattern that can be used to match text that has that pattern
- This is useful with forms because we can write patterns that allow us to test if the user has submitted data in a form we like
- You can set the pattern to be tested by using the pattern attribute

Regular Expressions continued

- You should also set the title attribute to something that describes the pattern you are looking for as this will be displayed to the user when they hover over the field or in the message the browser shows when the users input does not match the desired pattern
- For a full reference on regular expressions, please refer to https://www.w3schools.com/jsref/jsref_obj_regexp.asp
- Example

```
<input type=”tel” pattern=”\d{3}[\-]\d{3}[\-]\d{4}” title=”Phone
numbers must be in the format 555-555-5555”>
```

Lets try!

Datalist

- **The datalist element can be used to present suggested options to a user for a text element and will work as an auto-complete when the user types in the field.**
- **The children of the datalist element are option elements**
- **When used within a datalist element, an option element has the following attributes**
  - value: the value submitted with the form
  - label: the description of the option
- **Depending on the browser, one or both of the above attributes are displayed to the user in a drop down below the text box**
- **To associate a datalist with an element, you assign the datalist's id to the elements list attribute**

Datalist example

```
<input type=”email” name=”contact”
list=”emails”>
<datalist id=”emails”>
    <option value=”support@example.com”
label=”Support”>
    <option value=”sales@example.com”
label=”Sales”>
    <option value=”orders@example.com”
label=”Orders”>
</datalist>
```

Contact form Group Exercise