Week 10 Notes

* There are a couple reasons to keep do form validation:
* **We want to get the right data, in the right format.** Our applications won't work properly if our users' data is stored in the wrong format, is incorrect, or is omitted altogether.
* **We want to protect our users' data**. Forcing our users to enter secure passwords makes it easier to protect their account information.
* **We want to protect ourselves**. There are many ways that malicious users can misuse unprotected forms to damage the application (see [Website security](https://developer.mozilla.org/en-US/docs/Learn/Server-side/First_steps/Website_security)).
* The following are examples of how regex work
* a — Matches one character that is a (not b, not aa, and so on).
* abc — Matches a, followed by b, followed by c.
* ab?c—Matches a, optionally followed by a single b, followed by c. ( ac or abc)
* ab\*c—Matches a, optionally followed by any number of bs, followed by c. ( ac , abc, abbbbbc, and so on).
* a|b — Matches one character that is a or b.
* abc|xyz — Matches exactly abc or exactly xyz (but not abcxyz or a or y, and so on).

Most browsers support the [Constraint Validation API](https://developer.mozilla.org/en-US/docs/Web/API/Constraint_validation), which consists of a set of methods and properties available on the following form element DOM interfaces:

* [HTMLButtonElement](https://developer.mozilla.org/en-US/docs/Web/API/HTMLButtonElement) (represents a [<button>](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/button) element)
* [HTMLFieldSetElement](https://developer.mozilla.org/en-US/docs/Web/API/HTMLFieldSetElement) (represents a [<fieldset>](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/fieldset) element)
* [HTMLInputElement](https://developer.mozilla.org/en-US/docs/Web/API/HTMLInputElement) (represents an [<input>](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/input) element)
* [HTMLOutputElement](https://developer.mozilla.org/en-US/docs/Web/API/HTMLOutputElement) (represents an [<output>](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/output) element)
* [HTMLSelectElement](https://developer.mozilla.org/en-US/docs/Web/API/HTMLSelectElement) (represents a [<select>](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/select) element)
* [HTMLTextAreaElement](https://developer.mozilla.org/en-US/docs/Web/API/HTMLTextAreaElement) (represents a [<textarea>](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/textarea) element)
* We can add and remove css pseudo classes to display to the user the state of their input.
* Fetch must be used with an async function
* Reponse needs the json method to parse the reponse into json
* EXAMPLE:  
  Text

  Description automatically generated
* How to extract the body of a response statement:

The credentials attribute can be set to include all or some credential data of the caller.  
The [Request](https://developer.mozilla.org/en-US/docs/Web/API/Request) and [Response](https://developer.mozilla.org/en-US/docs/Web/API/Response) interfaces share the following methods to extract a body. These all return a promise that is eventually resolved with the actual content.

* [Request.arrayBuffer()](https://developer.mozilla.org/en-US/docs/Web/API/Request/arrayBuffer) / [Response.arrayBuffer()](https://developer.mozilla.org/en-US/docs/Web/API/Response/arrayBuffer)
* [Request.blob()](https://developer.mozilla.org/en-US/docs/Web/API/Request/blob) / [Response.blob()](https://developer.mozilla.org/en-US/docs/Web/API/Response/blob)
* [Request.formData()](https://developer.mozilla.org/en-US/docs/Web/API/Request/formData) / [Response.formData()](https://developer.mozilla.org/en-US/docs/Web/API/Response/formData)
* [Request.json()](https://developer.mozilla.org/en-US/docs/Web/API/Request/json) / [Response.json()](https://developer.mozilla.org/en-US/docs/Web/API/Response/json)
* [Request.text()](https://developer.mozilla.org/en-US/docs/Web/API/Request/text) / [Response.text()](https://developer.mozilla.org/en-US/docs/Web/API/Response/text)