Annie Cheng (ac962)
2/23/16
CS 2300

Project 2 Design Rationale

Form Checking and Data Validation

In my **Add Form**, I have four input text fields (name, image url, favorite toy, special talent) and 3 select boxes (breed, weight, and personality). For the select boxes, I used HTML's *required attribute* to ensure that the form will not submit unless the user selects an option from each of those fields. I also used the *required attribute* for all the text fields and the *maxlength attribute* to set the max number of characters for the name (30), favorite toy (50), and special talent (50) fields. Aside from minor input validation in the HTML, most of my validation occurs in the *valid.js file* when *validForm()* is called upon a click of the submit button:

- For the name, I only allowed users to enter letters (uppercase and lowercase) and spaces as well as dashes and underscores for rare name cases (but often socially-accepted).
- For the image url, I checked to see if the url returned a displayable image by ensuring that the image received from the url has a height greater than 0.
- For the favorite toy and special talent fields, I allowed all characters with the exception that any text field containing only spaces was considered an invalid entry. I purposely chose to allow any characters for these fields to encourage more creative entries (i.e. $$ to denote money).
- For the name, favorite toy, and special talent fields, I did not allow entries containing only spaces.
- When validating, if any field had an incorrect entry, I gave a red border to that field and displayed a helpful error message.

In my **Search Form**, I have four input text fields (search, name, favorite toy, special talent) and 3 select boxes (breed, weight, and personality). None of the fields are required but I checked that the user filled in at least one field. In my name field, I used the *"autofocus pattern" attribute* and a *regex* to only allow submission with legal name characters (as defined before). In the search field, I allowed any character to comply with searches of the favorite toy and special talent fields, which have no restricted characters (as defined before).

Note: For both forms, I used *strip_tags()* to prevent HTML code injection as user input. Also, I allow duplicate entries because the point of the website is simply to showcase pups for fun. Pups can share the same name, weight, breed, favorite toy, special talent, and even image url (more of the same cute pup, the better!).

Sample Queries: Search Over Multiple Categories
- **Example:** An existing entry that would result from a certain query:
    - **Search for** "Wiggles" in name field and "rubber bone" in favorite toy field
    - **Results in** Wiggles: name is "Wiggles" and favorite toy is "rubber bone"

- **Excludes** all other entries including <u>Nibbles</u>: favorite toy ("rubber bone") is matched but name ("Nibbles" which doesn't contain "Wiggles") isn't matched
- **Example:** An existing entry that would *not* result from a certain query:
    - **Search for** "Wiggles" in name field and "rubber bone" in favorite toy field
    - **Results in** <u>Wiggles</u> and <u>Nibbles</u>: favorite toy for both ("rubber bone") is matched, so "Nibble" matched regardless of name mismatch
    - **Excludes** all other entries

<u>Searching Functionality</u>

For the search form, I implemented a **case-insensitive "and" search** to allow for users to quickly find a specific pup by narrowing down multiple categories. Users can choose to search a single field or multiple fields but at least one field must be filled. All text fields and select boxes search in their respective categories except for the search field, which searches across all the fields for a match in any specific field.

Search Field Examples:
- **"Fluff"** in search field results in <u>1 match (Fluff, pomeranian)</u> because it matches the name field.
- **"Fluff 4"** in search field results in <u>no matches</u> because not one field contains the combination of both "Fluff" and "4".

<u>WOW Factor and Other Notes</u>

On the main page, I allow for pup sorting by date added and alphabetically by name, breed, weight, and personality. In addition, I put in a lot of thought and effort into making a simple, clean-looking website with a common theme, good design, and playful interactivity. Aside from the puppy images, I designed and created all of the other assets.