

## Objectives

- CSS Review
- Discuss:
  - How Google Search Works
  - What Images You Can Use
- HTML Forms

## CSS Review

- Why CSS?
- What is the syntax of a CSS rule?
- What is the order of applying rules in the cascade?
- How can we apply a rule to a particular element?
- How did Lab 1 go?

## Identify the Errors

```
<style>
body {    background-color: white
h1, {    gray;    font-family: sans-serif; }
h2, p {    color: }
<em> {    font-style: italic; }
</style>
```

## Fixed CSS

```
<style>
body {    background-color: white; }
h1, {    color: gray;    font-family: sans-serif; }
h2, p {    color: blue; }
<em> {    font-style: italic; }
</style>
```

Could be other ways to fix the errors

## Reflection

- Why do we have two languages: HTML and CSS?
  - Why do we typically store them in separate files instead of having both in one file?
- Sometimes, people talk about “coding” and count HTML as coding
  - HTML is not what we typically refer to when we say “coding”. Why not? How is HTML different?

## A Little More on Bootstrap



April 25, 2019

Sprenkle - CSCI335

7

## Google Search Discussion

- How does Google's search work?
- What are some of its novel features?
- What are some recommendations for web masters?
  - Any surprises?
- Why is search engine optimization important?
  - It's so important that it has its own acronym: SEO

April 25, 2019

Sprenkle - CSCI335

8

## Google Search Discussion

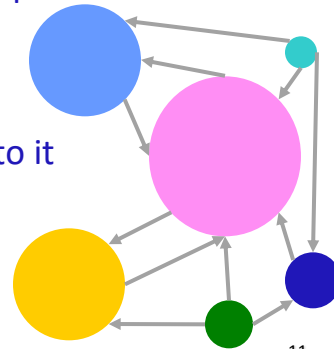
- What are Google search's limitations?
  - How can it be manipulated?
- What features would you like to add to Google's searching technique (or other search engine)?
- What is your preferred search engine? Why?

## Google Search Discussion

- Knowing more about Google search, will you change your queries any?
  - How can you make them more effective?
  - How can you save yourself time?
- Now that you're an author of HTML documents, will you change them to get a higher rank on Google?
  - How could you do that?

## Simplified Google PageRank

- Developed by Larry Page and Sergei Brin
  - Then graduate students at Stanford University
- How do we know what a page is about?
  - One approach: What text do people use to link to it?
- How should we rank pages?
  - By number of links to it
  - By popularity of pages that link to it



April 25, 2019

Sprenkle - CSCI335

11

## INTERNET COPYRIGHT

April 25, 2019

Sprenkle - CSCI335

12

## What Can You (Legally) Take From the Web?

April 25, 2019

Sprenkle - CSCI335

13

## What Can You (Legally) Take From the Web?

- **What** you take, **why** you take, **who** you are, what technology you use
  - Commercial or non-profit
  - Amount and substantiality of portion used
  - Nature of copyrighted work
- What is fair use? *Tough question!*
  - Rules change as technologies change

April 25, 2019

Sprenkle - CSCI335

14

User Interface


## FORMS

April 25, 2019

Sprenkle - CSCI335

15

## Web Interfaces

- Menus
  - Sidebars, links, navigation
- GUIs
- Forms  Today's focus

April 25, 2019

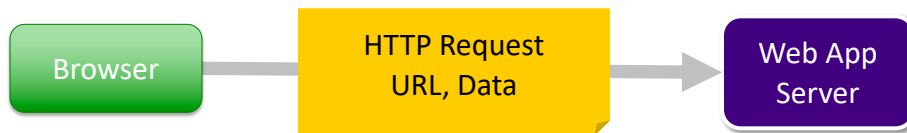
Sprenkle - CSCI335

16



## HTML form Tag

- Used to provide web application input from the user
- Contains various **controls** or **widgets** (sources of input) and labels for controls
- Must have a **submit** button that transmits all input data to server as a query string



April 25, 2019

Sprenkle - CSCI335

17

## Examples of Forms/Input

The image shows three examples of web forms and input fields:

- Google Search:** The Google logo is at the top. Below it is a search bar with a microphone icon on the right. Underneath the search bar are two buttons: 'Google Search' and 'I'm Feeling' with a smiley face icon.
- Member Login:** A blue header bar says 'Member Login'. Below it are two input fields for 'E-Mail:' and 'Password:'. There is a 'Remember Me' checkbox. At the bottom are two buttons: 'LOGIN' and 'SIGN UP!'. A link 'Forgot your password?' is at the bottom right.
- Friends are coming over for dinner. You wow them with...:** A yellow box contains a list of humorous suggestions for impressing friends, each followed by a small circular icon with a smiley face:
  - Your amazing cooking and homemaking skills. Julia Childs weeps with pride.
  - Your comedic skills. Pass out the tissues.
  - Your wisdom and masterful conversation.
  - Your ability to throw together a memorable event with seemingly little effort.
  - Your ability to be scathingly sarcastic while still loveable.
  - Your amazing music collection and kick ass stereo system.
  - Your beautiful smile.

April 25, 2019

Sprenkle - CSCI335

18

## HTML form Tag

- **form** attributes:
  - **action** (required): URL to send request to
    - Relative or absolute
  - **method**: `get` or `post`
    - Default: `Get`
    - More on differences later...

```
<form action="/search">  
  <!-- input fields, etc. -->  
</form>
```

## Input From Forms

- Types of input data
  - Text
  - Checkbox
  - Radio buttons
  - Select from list
  - Password
- Data is identified by a **name**, has a **value**
  - Specified by input fields' **name** attributes

## Simplified Form for Google Search

```
<form action="/search">
  <p>
    <input type="text" name="q" size="55"/>
    <input type="submit" value="Google Search"
      name="btnG"/>
  </p>
</form>
```

- Form is submitted to Google's `/search` page with data `q` set to whatever user entered in box, e.g.,
  - [http://www.google.com/search?q=user\\_query](http://www.google.com/search?q=user_query)

April 25, 2019

Sprenkle - CSCI335

[search.html](#)

21

## Another Way to Invoke A Web Application

- Example: Google
- Create a link to
  - <http://www.google.com/search?q=web+applications>
- Browser sends a **GET** request to the server's **search** page
  - 1 Parameter:
    - **name** is `q`, which has **value** `web+applications`
      - The + is the encoding for a space

April 25, 2019

Sprenkle - CSCI335

22

## Query Strings

- Do you always want your input data to show up in the URL?

## get vs post

- **get** passes parameters to server as a query string
  - Limited to *browser's* URL length
- **post** embeds the parameters in *HTTP request body*
  - **Not** in the URL

# post

## Advantages

- Information is more private (not shown in URL)
- Can't be bookmarked

## Disadvantages

- Can't be bookmarked
- Browser can't easily go back (POSTDATA error)

# input Tag

- Used to create many of the widgets
- **type** attribute specifies the type of widget
- Must be inside a block-level element
- Contains attributes
- Examples:
  - text, checkbox, radio
- Often requires **name** attribute
  - Names the data that will be sent to the Web application

## Text input

Enter Text:

- A horizontal box that the user can input text into

```
<input type="text" name="name" size="25"/>
```

### Attributes:

<b>size</b>	Width of text box in characters; scrolls if user types more
<b>maxlength</b>	Maximum number of characters browser accepts in a box
<b>value</b>	Provide a default value

Examples of when to use this?

April 25, 2019

Sprenkle - CSCI335

27

## Password input

Password:

- A horizontal box that the user can input text into **but** the text displays as \*s or •s

```
<input type="password" name="mypassword" size="10" maxlength="10"/>
```

### Attributes:

<b>size</b>	Width of text box in characters; scrolls if user types more
<b>maxlength</b>	Maximum number of characters browser accepts in a box

April 25, 2019

Sprenkle - CSCI335

28

## Labeling input fields: **label**

- Label a control with the **label** element
- Better than labeling with other text because
  - Can get separated during maintenance
  - **label** has special presentation
  - Improved usability

```
<p>
  <label>Password:
  <input type="password"
        name="mypassword" size="10"
        maxlength="10"/>
  </label>
</p>
```

textbox.html

April 25, 2019

Sprenkle - CSCI335

29

## Multiple Choice Input: **checkbox**

- Use when user has multiple choices for something and can choose  $\geq 1$

Milk  Bread  Eggs

- Only items that user checks are sent by the form to the action location

April 25, 2019

Sprenkle - CSCI335

checkbox.html

30

## Multiple Choice Input: checkbox

- All checkboxes in a group have the same **name**
- Checkbox requires a **value** attribute
  - **value** is submitted in the form data iff the checkbox is 'checked'
- To make a checkbox checked, add the **checked** attribute (which doesn't have a value)

```
<label>
  <input type="checkbox" name="groceries"
        value="milk" checked/>Milk
</label>
```

April 25, 2019

Sprenkle - CSCI335

31

## Multiple Choice Input: checkbox

```
<label>
  <input type="checkbox" name="groceries"
        value="milk" checked/>Milk
</label>
<label>
  <input type="checkbox" name="groceries"
        value="bread"/>Bread
</label>
<label>
  <input type="checkbox" name="groceries"
        value="eggs"/>Eggs
</label>
```

Notice order of label/input,  
Label and value are different

Milk  Bread  Eggs

April 25, 2019

Sprenkle - CSCI335

32



## Multiple Choice Input: **checkbox**

- Discussion: When designing a form, when should a checkbox be **checked** by default?

```
<label>  
  <input type="checkbox" name="groceries"  
        value="milk" checked/>Milk  
</label>
```

April 25, 2019

Sprenkle - CSCI335

33

## Multiple Choice Input: **checkbox**

- Discussion: When designing a form, when should a checkbox be **checked** by default?
  - Common value that people will always want
- Have you ever seen a checkbox turned on when you think it shouldn't be?

```
<label>  
  <input type="checkbox" name="groceries"  
        value="milk" checked/>Milk  
</label>
```

April 25, 2019

Sprenkle - CSCI335

34

## Multiple Choice Input: **radio**

- Only one radio button in a group can be on or pressed
  - Groups of radio buttons are identified with the same **name**

Why is **radio** appropriate?

```
<label> <input type="radio" name="age"
           value="under20" checked="checked" />0-19
</label>
<label> <input type="radio" name="age"
           value="20-35" />20-35
</label> ...
<label> <input type="radio" name="age"
           value="over65" />&gt;65
</label>
```

Same name

0-19  20-35 ...  >65

April 25, 2019

Sprenkle - CSCI335

35

## Alternative for **Label**

- Use **for** attribute to specify which control you're labeling
  - **for**'s value is the control's **id**

```
<label for="age.under18">0-17</label>
<input id="age.under18" type="radio"
       name="age" value="under18" />
```

Important for usability

April 25, 2019

Sprenkle - CSCI335

36

## Menus with `<select>`

- Displays large number of options more compactly
- Can emulate radio buttons (only one selection, default) or checkboxes (multiple selections)

<b>name</b>	Name of the data
<b>size</b>	# of items to display
<b>multiple</b>	Allows multiple selections if value is <b>multiple</b>

```
<select name="age">
  ...
</select>
```

CSCI335

37

## option tag

- Value options are in **option** tags, nested inside of **select** tags
- Can preselect an option with **selected** attribute set to "selected"

```
<select name="age">
  <option value="under18"
    selected>0-17</option>
  <option value="18-25">18-25</option>
  ...
  <option value="over65">&gt; 65</option>
</select>
```

What is your age?  0-17  
 18-25  
 26-45  
 46-64  
 > 64

April 25, 2019

Sprenkle - CSCI335

38

## select Tag Example

- Emulating checkboxes

```
<select name="groceries" multiple="multiple">  
  <option value="milk">Milk</option>  
  <option value="bread">Bread</option>  
  <option value="eggs">Eggs</option>  
</select>
```

### Alternative Example:

Which types of films do you like to watch?

 Action  
 Comedy  
 Foreign

## Using select

- Any advantages or disadvantages to using **select** rather than radio buttons or checkboxes?

## Using `select`

- Any advantages or disadvantages to using `select` rather than radio buttons or checkboxes?
  - `select` takes up less space when only one selection BUT user must click to see all options

April 25, 2019

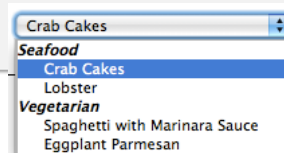
Sprenkle - CSCI335

41

## Option Groups: `optgroup`

- Tag used to group options with a label
  - Can also apply a style to label

```
<select name="entree">
  <optgroup label="Seafood">
    <option value="crabcakes">Crab Cakes</option>
    ...
  </optgroup>
  <optgroup label="Vegetarian">
    <option value="spaghetti">Spaghetti</option>
    ...
  </optgroup>
</select>
```



April 25, 2019

Sprenkle - CSCI335


42

## textarea Tag

- Allows users to input multiple lines of text

<b>name</b>	Name of the data
<b>rows</b>	# of rows for text area
<b>cols</b>	# of characters wide for text area

```
<p>Please provide your yearbook memories:</p>
<p><textarea name="memories" rows="3"
cols="40"> (Be brief and concise.)
</textarea></p>
```



Can't close the tag when opened.  
Needs content, even if empty.

Displayed by default in the  
text area.

April 25, 2019

Sprenkle - CSCI335

43

## Grouping Input: fieldset, legend

- **fieldset**: groups related **input** fields
- **legend**: supplies an optional caption

```
<fieldset>
  <legend>Credit cards:</legend>

  <input type="radio" name="creditcards" id="visa" />
  <label for="visa">Visa</label> <br />

  <input type="radio" name="creditcards"
    id="mastercard" />
  <label for="mastercard">MasterCard</label> <br />

  <input type="radio" name="creditcards" id="amex" />
  <label for="amex">American Express</label><br />
</fieldset>
```

April 25, 2019

Sprenkle - CSCI335

[See buttons.html](#)

44

## submit and reset Buttons

- type = **submit**
  - When clicked, browser sends parameters to the server
  - Browser shows server's response
- type = **reset**
  - when clicked, browser changes the controls back to their original state

```
<input type="submit" value="Submit Query"/>
<input type="reset" value="Reset"/>
```

Submit Query    Reset

April 25, 2019

Sprenkle - CSCI335

45

## Styling Forms with CSS Attribute Selectors

```
element[attribute=value] {
    properties; ...
}
```

```
input[type="text"] {
    color: blue;
    font-style: italic;
    margin-bottom: 2em;
}
```

*example text*

- CSS attribute selector affects an element only if it has the given attribute set to the given value
- Often used with forms because **input** element represents many different controls

April 25, 2019

Sprenkle - CSCI335

46

## Considerations for Choosing Controls

- How easy is it for the user to enter the input?
- What is the range of possible values?
- How many possible values are there?
- How much flexibility does the user have?
  - How much *should* they have?
- Should the data be hidden in the browser?

April 25, 2019

Sprenkle - CSCI335

47

## What **input** would you use?

Input Data	Input Type
Sensitive data	
Date (month, day, and year)	
Abstract for a paper	
Title for a song	
Household Income Bracket	

April 25, 2019

Sprenkle - CSCI335

48



## TODO

- Lab 2: HTML Forms
  - Due tonight at 11:59 p.m.
- Requirements, Design, Work Plan Documents on Wiki – Friday
  - Coordination/cooperation, division of responsibilities
  - Does this work?
  - Due Friday at midnight