CSE 154, Autumn 2012
Final Exam, Thursday, December 13, 2012

Name: ______________________________________

Quiz Section: __________________ **TA:** __________________

Student ID #: __________________

Rules:

- You have **110 minutes** to complete this exam.
  You may receive a deduction if you keep working after the instructor calls for papers.
- This test is open-book/notes. You may use any paper resources other than practice exams.
- You may *not* use any computing devices, including calculators, cell phones, or music players.
- Unless otherwise indicated, your code will be graded on proper behavior/output, not on style.
- Please do not abbreviate code, such as writing ditto marks ("") or dot-dot-dot marks (...).
  You may write `$` for `document.getElementById` and `$$` for `document.querySelectorAll`.
- You may not use JavaScript frameworks such as jQuery or Prototype when solving problems.
- If you enter the room, you must turn in an exam and will not be permitted to leave without doing so.
- You must show your **Student ID** to a TA or instructor for your submitted exam to be accepted.

Good luck! You can do it!

Problem	Description	Earned	Max
1	HTML / CSS Tracing		20
2	CSS		20
3	PHP		20
4	JS / Ajax / JSON		20
5	SQL		20
X	Extra Credit		1
TOTAL	**Total Points**		**100**

1. HTML / CSS Tracing

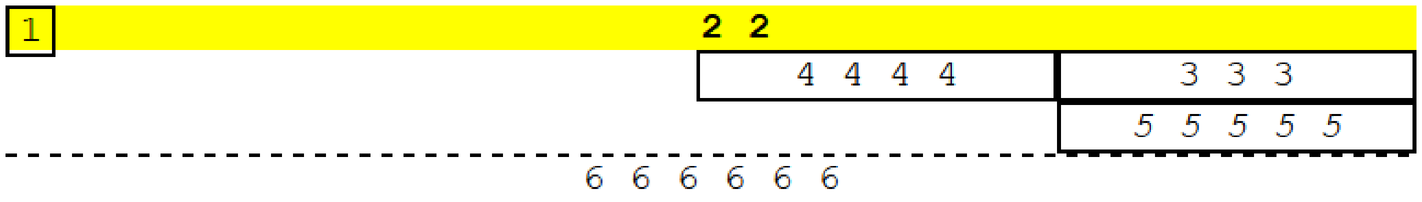
Draw a picture of how the following HTML/CSS code will look when the browser renders it on-screen. Assume that the HTML is wrapped in a valid full page with a `head` and `body`. Indicate a non-white background by shading lightly or by drawing diagonal lines like this. It is possible that some CSS rules shown will not apply to any elements.

```
<div>                                                                 HTML
  <span>1</span>
  <div id="div">2 2</div>
</div>
<span class="div">3 3 3</span>
<div>
  <div class="div">4 4 4 4</div>
  <div id="span">5 5 5 5 5</div>
  <div class="span">6 6 6 6 6 6</div>
</div>
```

```
div                 { border: 2px solid black;  padding: 1em; }         CSS
body > div          { margin: auto; width: 50%; }
div #div, p         { background-color: yellow; text-decoration: underline; }
span div, span.div  { border: 2px dashed black; }
div > div.div       { float: left; }
#div, .span         { clear: left; }
span#span           { background-color: yellow; }
```

2. CSS

Write the **CSS code** necessary to recreate the following appearance on-screen, exactly as shown.
The page uses the same HTML code as in the previous problem. You are **not allowed to modify the HTML**.



```
<div>
  <span>1</span>
  <div id="div">2 2</div>
</div>
<span class="div">3 3 3</span>
<div>
  <div class="div">4 4 4 4</div>
  <div id="span">5 5 5 5 5</div>
  <div class="span">6 6 6 6 6 6</div>
</div>
```

All text now uses a monospace font at the default size.

All borders shown are 2px thick and black in color.

The element with "2 2" now has a yellow background.

The elements with "3 3 3", "4 4 4 4", and "5 5 5 5 5" are now each exactly one fourth (1/4) of the page width.

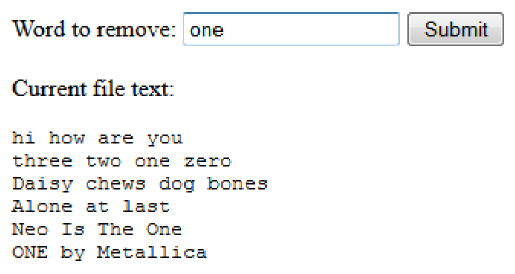
The element "2 2" now has **bold** text, and the element "5 5 5 5 5" now has *italic* text.

3. PHP

Write the PHP code for a web page `filter.php` that filters lines of text from a file. The page should contain a short form with a text box where the user can type a word. The page also displays the current contents of the file `text.txt` as a pre-formatted block. The form submits back to the same page, `filter.php`, as a POST request. When the word is submitted, your code should examine the contents of `text.txt` and remove any lines from the file that contain the given word, case-insensitively. Write the changes to the file so that any future viewings of the page will see the changes. You can write just the code dealing with the page's `body`; you don't need to output a head section or a complete page.

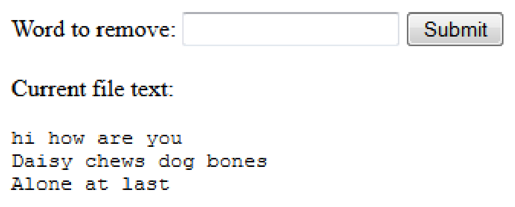
Match the exact word, not other words that contain it as a substring. For example, if the user submits the word "me" you would filter out lines containing the word "me", but not lines that just contain a word such as "men" or "game".

The following screenshots show the page as the user types the word "one" and after clicking Submit:



Current file text:

```
hi how are you
three two one zero
Daisy chews dog bones
Alone at last
Neo Is The One
ONE by Metallica
```



Current file text:

```
hi how are you
Daisy chews dog bones
Alone at last
```

If the user makes a POST but somehow does not submit the query parameter for the word, or if the word they submit does not consist entirely of upper/lowercase letters, issue an HTTP 400 error and do not display the rest of the page. Use the browser's default styling; you do not need to write any CSS for this problem.

3. PHP (additional writing space)

4. JavaScript / Ajax / JSON

Write the JavaScript code for a basic vocabulary quiz built using Ajax and JSON that allows the user to try to guess the definitions to randomly chosen words from the server. The quiz data comes from a web service named `word.php`, located on your web server in the same directory as your code. Contact this service with a GET parameter of `part` for a part of speech such as `noun` or `adjective`. It outputs JSON data about a random dictionary word and several possible definitions for the word (at least 2 definitions, of which exactly 1 is correct) in the following format. For example, a request to `word.php?part=noun` might return:

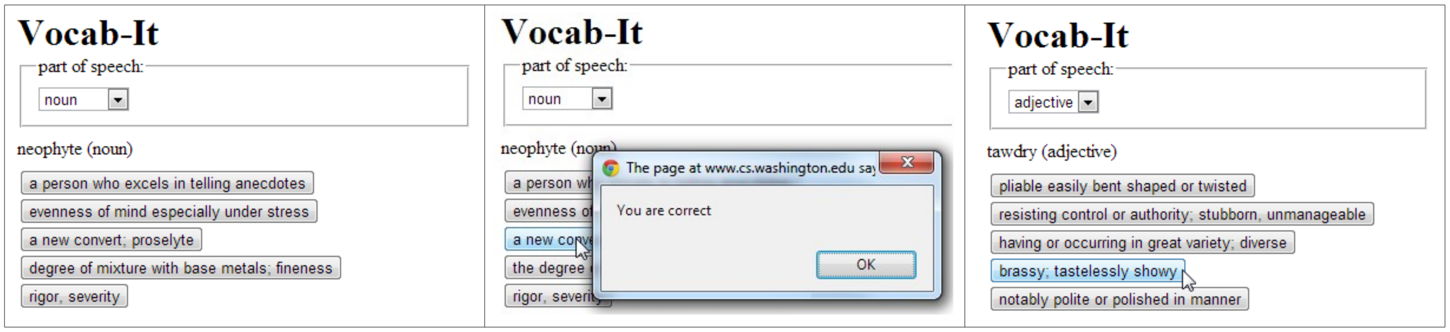
```
{"word": "neophyte",
 "part": "noun",
 "choices": [
   {"definition": "a person who excels in telling anecdotes", "correct": false},
   {"definition": "evenness of mind especially under stress", "correct": false},
   {"definition": "a new convert; proselyte", "correct": true},
   {"definition": "degree of mixture with base metals; fineness", "correct": false},
   {"definition": "rigor, severity", "correct": false}
 ]
}
```

When the page loads, contact the web service with Ajax. Display the random word and its part of speech in the `"word"` area. Display all of the possible definitions as buttons in the `"choices"` area. When the user clicks a button to guess the definition, display an alert message of either "You are correct" or "You are incorrect" appropriately, and then once the alert box is closed, start a new quiz by fetching a new word and displaying it and its definitions to the user. At any time the user can change the part of speech from the select box, which should affect any future words.

The relevant existing HTML in the page is the following:

```
<h1>Vocab-It</h1>
<fieldset>
  <legend>part of speech:</legend>
  <select id="part">
    <option>noun</option> <option>verb</option> <option>adjective</option>
  </select>
</fieldset>
<div id="word"></div>
<div id="choices"></div>
<div id="result"></div>
```

For the example JSON shown above, the page would look as follows. The three screenshots show the page's initial state, the state after a button is clicked, and then the state after the alert box is closed and a new word is fetched.


Vocab-It
part of speech:
noun
neophyte (noun)
a person who excels in telling anecdotes
evenness of mind especially under stress
a new convert; proselyte
degree of mixture with base metals; fineness
rigor, severity
Vocab-It
part of speech:
noun
neophyte (noun)
The page at www.cs.washington.edu says
OK
Vocab-It
part of speech:
adjective
tawdry (adjective)
pliable easily bent shaped or twisted
resisting control or authority; stubborn, unmanageable
having or occurring in great variety; diverse
brassy; tastelessly showy
notably polite or polished in manner

You are correct

You may assume that the JSON data is valid in the format described previously, and that the .php service is reachable. You do not need to handle any Ajax errors. Do not use any JavaScript libraries such as jQuery or Prototype.

Write your answer on the next page.

4. JavaScript / Ajax / JSON (writing space)

5. SQL

Write an SQL query to search the `imdb` database for all actors who appeared in a romantic comedy film with actor Woody Allen that was made in 1999 or later. A romantic comedy is a film that is classified as being in both the "Romance" and "Comedy" genres. Show the actors in alphabetical order by movie name ascending, breaking ties by actor last name ascending and then by first name ascending. Each actor/film combination should be listed only once. Woody Allen's first and last name are "Woody" and "Allen" respectively, and you may assume that he is the only actor in the database with that exact first/last name pairing. Recall the `imdb` tables:

actors

id	first_name	last_name	gender
433259	William	Shatner	M
797926	Britney	Spears	F
831289	Sigourney	Weaver	F
...			

movies

id	name	year	rank
112290	Fight Club	1999	8.5
209658	Meet the Parents	2000	7
210511	Memento	2000	8.7
...			

roles

actor_id	movie_id	role
433259	313398	Capt. James T. Kirk
433259	407323	Sgt. T.J. Hooker
797926	342189	Herself
...		

directors

id	first_name	last_name
24758	David	Fincher
66965	Jay	Roach
72723	William	Shatner
...		

movies_directors

director_id	movie_id
24758	112290
66965	209658
72723	313398
...	

movies_genres

movie_id	genre
209658	Comedy
313398	Action
313398	Sci-Fi
...	

```
+-----------------+-------------+-------------------+
| first_name      | last_name   | name              |
+-----------------+-------------+-------------------+
| Anthony         | Arkin       | Anything Else     |
| James           | Babbin      | Anything Else     |
| ...             | ...         | ...               |
| Eric            | Tonken      | Anything Else     |
| Ricardo         | Bertoni     | Small Time Crooks |
| ...             | ...         | ...               |
| Jesse (I)       | Levy        | Small Time Crooks |
| Lawrence Howard | Levy        | Small Time Crooks |
| ...             | ...         | ...               |
| Karla           | Wolfangle   | Small Time Crooks |
| Frank (III)     | Wood        | Small Time Crooks |
+-----------------+-------------+-------------------+
83 rows in set
```

When run on the `imdb` database, your query would produce the results at left *(a subset of the total rows are shown)*.

If you join too many tables together that are not needed for the query, you will not receive full credit.

You should solve this problem using only the SQL syntax shown in class and the textbook.

X. Extra Credit

What is a fun web site that you think the TAs should look at while we are grading your exam? And why?
(This is just for fun; any URL you write on this page will receive credit.)