

CSE 190 M, Summer 2010
Final Exam, Part 2 (LECTURE)
Friday, August 20, 2010

Name: _____

Section: _____ **TA:** _____

Student ID #: _____

Rules:

- You have **60 minutes** to complete this part of the exam.
You may receive a deduction if you keep working after the instructor calls for papers.
- This test is open-book/notes.
- 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 (...).
- 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!

Problem	Description	Earned	Max
1	HTML/CSS Tracing		15
2	HTML/CSS Coding		15
3	JavaScript/DOM		15
TOTAL	Day's Total Points		45

Problem	Description	Earned	Max
4	Ajax/XML		15
5	PHP		20
6	SQL		20
TOTAL	Day's Total Points		55

TOTAL	Exam Total Points		100
--------------	--------------------------	--	------------

4. Ajax/XML

Suppose that there is a web service named `weather.php`, located on your web server in the same directory as your code. This service outputs XML data describing daily high temperatures Fahrenheit for various US cities. In this problem you will write Ajax JavaScript code to contact the web service (using a GET request), examine its XML data, and display a series of images representing the weather for each city. This code should run when the web page loads.

The XML data returned by the web service consists of an overall document element called `temperatures` that contains a series of one or more `city` elements inside it. Each city element has a `name` attribute containing the city's name and contains a series of one or more `high` elements, each of which represents one day's high temperature. The format matches the following example (though it might have more/fewer than 2 cities at different times):

```
<?xml version="1.0" encoding="UTF-8"?>
<temperatures>
  <city name="Seattle">
    <high>70</high>
    <high>67</high>
    <high>63</high>
  </city>


  <city name="San Diego">
    <high>79</high>
    <high>73</high>
    <high>68</high>
    <high>72</high>
  </city>
  ...
</temperatures>
```

The relevant existing HTML in the page is the following:

```
<h1>Weather Forecast for US Cities</h1>
<ul id="forecast"></ul>
```

When the page loads, read the XML data with Ajax. Turn each city's data into a bullet in the `forecast` list. In each bullet, write the name of the city followed by a colon, and then an image for each day's data in the XML. If the day has a temperature of 70°F or higher, use the image of `warm.gif`. For temperatures of 69°F or lower, use `cold.gif`. Separate neighboring images by 10 pixels. For the example XML shown above, the page would look as follows:

Weather Forecast for US Cities

- Seattle: 
- San Diego: 

You may assume that the XML data is valid in the format described previously, and that the `.php` service is reachable. You may also assume that **Prototype** is already linked to the page.

4. Ajax/XML (writing space)

5. PHP

Suppose the following HTML form `girlscout.html` exists for ordering Girl Scout Cookies online:

```
<form action="cookies.php" method="post">
  <div><input id="input1" type="text" name="samoas" size="2" /> Samoas x $3.50</div>
  <div><input id="input2" type="text" name="thinmints" size="2" /> Thin Mints x $4.00</div>
  <div>Shipping: <br />
    <label><input id="input4" type="radio" name="shipping" value="regular" /> Regular ($7)</label> <br />
    <label><input id="input5" type="radio" name="shipping" value="express" /> Express ($9)</label></div>
  <div><label><input id="input3" type="checkbox" name="donation" /> Donate $5 extra?</label></div>
  <div><input id="input6" type="submit" value="Buy" /></div>
</form>
```

Your task is to write the **PHP code** that would be saved in `cookies.php` to process the cookie order form. You should display an **h1 heading** saying "Your order:" followed by **pictures** of the cookie boxes the user has ordered. The user can order two kinds of cookies: Samoas and Thin Mints. You should display the image `samoas.jpg` once for each box of Samoas ordered, and display `thinmints.jpg` once for each box of Thin Mints ordered.

Your code should also compute the **total cost** of the order. Each box of Samoas costs \$3.50 and each box of Thin Mints costs \$4.00. There is a **shipping cost** added to the order, which costs either \$7.00 for regular shipping or \$9.00 for express shipping. The user can optionally choose to check a box to make a \$5.00 charitable **donation** with their order, which would be added to the order's total cost. If the donation box is checked, also print a "Thank you for your donation!" message at the bottom of your output.

The **screenshots** below show example values typed into the form, along with your page's resulting appearance:

<code>girlscout.html</code> appearance	Your <code>cookies.php</code> page's appearance
<p>Girl Scout Cookies</p> <p><input type="text" value="3"/> Samoas x \$3.50</p> <p><input type="text" value="2"/> Thin Mints x \$4.00</p> <p>Shipping:</p> <p><input type="radio"/> Regular (\$7)</p> <p><input checked="" type="radio"/> Express (\$9)</p> <p><input checked="" type="checkbox"/> Donate \$5 extra?</p> <p><input type="button" value="Buy"/></p>	<p>Your order:</p>  <p>Total order cost: \$32.5</p> <p>Thank you for your donation!</p>

You **don't have to round** the order cost to 2 decimal places. Your code doesn't need to output a complete HTML page; assume that your output will be put inside the page's `body`. **Assume valid form input**; the user will always submit numeric non-negative values for the number of Samoas and Thin Mints, and one of the two shipping methods will always be chosen. You will receive a small deduction if you use `print` or `echo` statements in your solution.

5. PHP (writing space)

6. SQL

Martin Scorsese has directed many movies, including Cape Fear, Casino, The Departed, Goodfellas, Raging Bull, and Taxi Driver. Scorsese is known for appearing as a character in his own films, in the background of a scene. Though he's known more as a director, he appears in our `imdb` database's table of actors due to appearances in his own films.

Write an SQL query to search the `imdb` database for all drama films (films with the "Drama" genre) that Martin Scorsese both directed *and* appeared in as an actor. For a movie to show in your results, Martin Scorsese must be that movie's director, and Martin Scorsese must also have played a character in that film. Show only the names of the movies, in **alphabetical order**. Each film should be listed only once. Recall the `imdb` database 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
...	

When run on the `imdb` database, your query would produce the following results:

name
Age of Innocence, The
Boxcar Bertha
Bringing Out the Dead
Color of Money, The
Gangs of New York
King of Comedy, The
Mean Streets
New York Stories
Raging Bull
Taxi Driver
Who's That Knocking at My Door?

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