

CSE134B Fall 2019  
Final Exam Question Bank

**Part 1: Core Ideas**

1. Client-Server Trade-Offs: Fill in the table showing the client and server-side programming trade-offs (aka Pros and Cons)

	Pros	Cons
Client Side	Faster Performance	Less secure, cannot be automated
Server Side	Secure	Centralised leads to SPOF

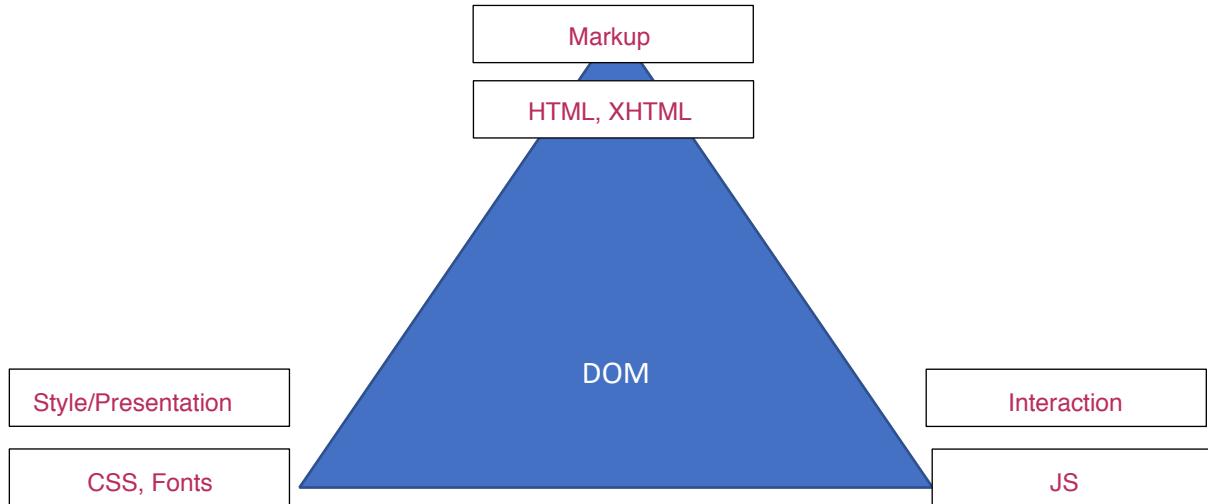
2. Explain which is better in your opinion, client-side or server-side and why. If neither is better than explain when each should be used and how to balance the two approaches. Be clear and provide examples for maximum points
3. Name the three parts of an HTTP request / response packet (aka Law of Three)

Request	Response
1. Request Line  GET /path HTTP/1.1	1. Response Line  HTTP 200 OK
2. Request Header  content-type: text/cms	2. Response Header
3. Body/ Payload	3. Body/ Payload

4. Toolbox 2.0: Fill-in the characteristics of the toolbox items

Client Side	Server Side	Characteristics
Browser Extensions  - Netscape plugins - ActiveX Controls - NaCL	Server API Programs  - Apache Modules - ISAPI Filters and Extensions - Nginx Dynamic Modules	Proprietary  Complex  Fast  Coupled
Java Applets	Java Servlets	Portable  Semi-coupled  Less complex
Client Side Scripting  1st Gen - Javascript - VBScript  2nd Gen - ASM - WebAssembly	Server Side Scripting  1st Gen - Classic Active Server Pages (ASP) - ColdFusion - PHP - Ruby  2nd Gen - JSP - ASP.NET	Fast  Portable  Semi-coupled  Less complex  Less fast
Helper Applications  - Scripted - Compiled	CGI Programs  - Scripted - Compiled	Portable/Not  Not coupled  complex/ Not  Slow

5. Fill in the diagram below labeling the idea and associated technology/technologies found in browser focused client-side web development.



Name the 4 pieces of RAIL, corresponding times / values and why important to meet

Letter	Stands For	Time / Value	Why important
R	Response.	~50ms	If user clicks and reaction isn't fluent
A	Animation.	60fps	Needs to render in time, user can detect break imm.
I	Idle.	0-50ms	Load expensive things in bg, main on UI thread
L	Load	1s	Need to get images on the screen ASAP

The professor says that Web sites in are made up of five aspects. He called these the 5 pillars.  
Name them.

- Content  
1. \_\_\_\_\_
- Structure  
2. \_\_\_\_\_
- Technology  
3. \_\_\_\_\_
- Delivery  
4. \_\_\_\_\_
- Design  
5. \_\_\_\_\_

## Part 2: HTML

1. Write an HTML comment with your name in it. <!--Paridhi Khaitan -->

2. Write the HTML5 document type statement <!DOCTYPE html>

3. Is <bogus foo="bar" baz>Bizz</bogus> markup well-formed ? Yes

Explain Because it is syntactically correct

Well formed Markup only needs to be  
syntactically correct  
Eg: Have closing/opening " etc.

4. Given the following markup snippet (4pts)

```
<ul id=myList>
    <li>First Point</li>
    <li>Second Point</li>
        <ol id="mySubList">
            <li>Sub point A
            <li>Sub point B
        </ol>
    <li>Third Point</li>
</ul>
```

Is the markup well-formed? No Explain <ol> cannot be a direct child of <ul>

Is the markup valid? No Explain because it is not well formed

5. Write a paragraph element containing the word ~~T~~ **U** **C** **S** **D**" in with **two** spaces between each letter using no CSS. (2pts)

<pre> U \_\_ C \_\_ S \_\_ D </pre> OR <p>. U&nbsp;&nbsp;C&nbsp;&nbsp;S&nbsp;&nbsp;D&nbsp;&nbsp; </p>

6. Write an HTML form tag to send data in the HTTP request line to the URL <https://somesite.com/deleterecord.php> (2pts)

<form action="https://somesite.com/deleterecord.php" method="POST"></form>

7. Write an HTML tag to insert the UCSD logo with fallback text of “UCSD”, a simple tooltip (aka advisory text) of “University of California, San Diego”. No JS needed!



8. Write a block element here <div></div>

9. Write an inline element here <span></span>

10. Write a logical element here \_\_\_\_\_ `<strong></strong> <em></em>`

11. Write a physical element here \_\_\_\_\_ `<b></b> <i></i> <big></big>`

12. The four core attributes for HTML tags are `id` , `title` , `style` & `class`

13. Write a `<div>` tag with `data-*` attributes that set your PID and your college. The contents of the `<div>` should be your name.

`<div data-PID="A15154494" data-college="Muir">Paridhi Khatian </div>`

14. Write an **XHTML** tag to insert a carriage return / line break \_\_\_\_\_ `<br><br>`

15. Given the markup below draw the DOM tree fragment that would be produced

```
<p>This is a <em>  
DOM tree  
</em>  
<br> test  
question.</p>
```

16. Images represent a significant amount of a web page and as such should be correctly used. Fill in the following table showing you understand image format characteristics and proper usage.

Image Format	Good For	Features / Characteristics	How Optimized
SVG			
GIF			
JPEG			
PNG			
WebP			

17. It appears that many tags like <section>, <aside>, <header>, <footer>, etc. all look the same why do we bother to use these? Be specific in what benefit such tags provide.

To make logical structure of the html. Also they could be used later to add structured JS

They're supported HTML markup, and help afford readability and maintainability

18. Apparently many HTML documents are invalid (in that they don't pass validation or conformance checking) explain the pros and cons of the looseness of HTML syntax.

### Part 3: CSS

1. Write CSS selectors with { ... } as the rules to : Example:

*someSelector {....}*

- A. select a <p> tag with the id 'firstParagraph'

*p#firstParagraph {...}*

- B. select **all** tags in class 'fancyTags'

*.fancyTags \*{...}*

- C. select **all** tags in the document

*\* {...}*

- D. select all tags that are direct descendants of the body

*body > \* {...}*

- E. select all h1 and h2 elements nested in section tags

*section h1, section h2 {...}*

- F. select all p tags in class 'fancy' upon hover

*.fancy p:hover {...}*

2. Write a paragraph element containing the word "UCSD" and make it blue text on a yellow background that is xx-large size. Use an inline style rule

*<p style="background: yellow; color: blue, font-size: xx-large">UCSD</p>*

3. Write a document wide style block (include the correct HTML tag) to set all <p> tags to be double spaced and justified (not ragged left / right text).

```
.p {  
    line-spacing: 2;  
    text-align: justify;  
}
```

4. Write a CSS rule to "normalize" or eliminate margins and padding values

*\*{margin: 0, padding: 0}*

5. Write an HTML tag to reference an external style sheet named "global.css"

*<link rel="stylesheet" type="text/css" href="https://example.com/css/global.css" >*

6. Write a CSS rule to set all `<a>` tags within a `<nav>` tag id = “mainNav” which is a direct descendent of a `<section>` tag to be bold and have an underline on hover.

```
section nav#mainNav a:hover{  
    font-weight: bold  
    text-decoration: underline  
}
```

7. Write a CSS rule that will try to **force** all `<p>` tags to be bold and in Arial font.

```
p{  
    font-weight: bold !important;  
    font-family: 'Arial' !important;  
}
```

8. Explain what `!important` does in CSS. Should you use this construct. Explain when it would be useful and when it would be harmful.

`!important` is used to override applied styles and prioritise the given style. It's useful when we're using an external stylesheet and applying our custom styles. It's harmful if done excessively because the precedence is lost

9. Write a `<div>` tag with an error message that says CSS Disabled in it. Now write an inline CSS rule to make sure it only shows up when CSS is off.

```
<div style="display: none"> CSS IS DISABLED </div>
```

10. CSS frameworks like Twitter Bootstrap exist to significantly ease our burdens in laying pages out with CSS. Explain at two downsides with adopting such a CSS framework and two upsides in doing so.

Downsides	Hard to integrate custom styling	slower performance because you have a lot of useless code
Upsides	Quick UI integration because someone has already written the code	Cross browser compatible and responsive

11. Write a CSS rule using a short-hand notation to set the margin of all div tags to have 5px on the left and right and 10px on the top and bottom.

```
div {  
    margin: 10px 5px;  
}
```

12. Write the color Red in at a minimum of four different ways that would be recognized by CSS.

```
rgba(255,0,0,1) #FF0000 #F00 red
```

#### Part 4: Forms and HTTP

1. Write a valid HTML form that collects your name, your PID, and your UCSD college and sends that data in the URL to “/gradefinal.php”. Your college should be found in a pull-down menu and your PID should not be echoed just like a login form would work. Use appropriate HTML elements to structure your form and attributes to make it usable, logical, and as secure as you can do easily with HTML for full points

```
<form action = “/gradefinal.php” method=“POST”>
    <label for=“name”>Name</label>
    <input type=“text” name=“name”>

    <label for=“PID”>PID </label>
    <input type=“password” name=“PID”>

    <select name=“colleges”>
        <option value=“ERC”> ERC </option>
        <option value=“Muir”> Muir </option>
    </select>

    <input type=“submit”>Submit </input>
</form>
```

2. Redo the question above using JavaScript to send the data using Ajax either using XHR or Fetch style APIs. Make sure your callbacks if used are called *handleResponse* and that data received is echoed back out to an *<output>* tag with an id of ‘result’. Hint: Careful with event handlers and don’t forget encoding.
3. Write a complete HTTP packet including the message body (aka ‘payload’) of your form from the previous examples if it were sent using your actual last name, pid and hoped for result.

```
POST /gradefinal.php HTTP/1.1
Host: /gradefinal.php
Content-Type: application
name=Paridhi&PID=A15154494&colleges=Muir
```

4. Redo the previous question questions with and without JavaScript but send the data using the other HTTP method that simple HTML forms support different than your previous answer. You can just write the changed lines for space.

GET /gradefinal.php?name=Paridhi&PID=A15154494&college=Muir HTTP/1.1  
(the whole thing becomes the query URL)

5. In homework #1 we saw many headers so of these headers seemed harmful to the site emitting them. Provide an example of one of these headers and explain why it is somewhat harmful.

X-Powered-By: ASP.net

Hackers can target specific viruses based on that version. Exposes the software version.

6. An HTTP response should contain a very important header that indicates what the content in the payload is. Write that header below and an example value for it.

content-type: text/html

Now explain what a browser does in the absence of that header.

in the absence of this header, the browser MIME sniffs the response and auto assigns the content type to the response

Now explain how what the browser does in the absence of that header can be harmful.

As this is purely based on guessing, the assigned header can be incorrect.  
MIME sniffing can expose the response to XSS (Cross Site Scripting)

#### Part 4: Other Things, Usability, ‘Engineerism’ and Related Effects

1. Describe progressive enhancement as a design philosophy in Web development as compared to graceful degradation. Which is better? If neither is better, describe which makes sense in what situation.

Progressive Enhancement: Enhances experiences for browsers that support it. Doesn't have any impact on those that don't .  
Bottom up approach where you develop the core first, then presentation

Graceful Degradation: Use modern features as default, and it won't break on old browsers but won't support it.  
Depends on the use case. If you're trying to update an existing website-> GD  
If you're trying to create a modern website-> PE

2. Explain the UX/DX tension the Professor continually stresses. Provide a concrete example of the tension it describes.

### 3. Usability Related Fill-ins

You are not the \_\_\_\_\_ user but also \_\_\_\_\_ users are not \_\_\_\_\_ designers

99% of times user's are at other sites and apps and they bring in

The 99% rule means that \_\_\_\_\_ how they understand those sites and apps into your site

The reason I tend to prefer multiple choice tests over fill-in questions is because per the

usability slides themselves \_\_\_\_\_ recognition is easier than

recall

From Nielsen 5 ideas that determine usability of a site would include Learnability,

Learnability

Rememberability

Reliability

Efficiency

User Satisfaction

and User Satisfaction.

7+- 2 (cognitive load)

When making choices apparently people can remember \_\_\_\_\_ items. This might be good when thinking about putting choices on a menu bar.

4. The Professor has also stressed that cultural, economic, and other effects shape the technology choices we often have to make as web developers. Provide one example to show that you understand this. Be clear and expansive and personal if appropriate to show that you see beyond syntax to other factors that may shape your decisions when building sites or apps.

5. The Professor loves the idea of trade-offs. If you have been to lecture more than a few times you have probably heard about many. Describe two (2) trade-offs that we may generally encounter as web developers and provide some details about what the tradeoff means.

6. If browsers “correct” things for us problems may ensue. Discuss unintended consequences or later challenges we might encounter because of the permissive nature of Web browsers handling HTML, HTTP, CSS, etc. Be concrete in your discussion.

Hackers can hide malicious code in other file formats that the browser thinks is code. If this code is run, our site will be hacked because we allowed the browser to MIME sniff

7. The professor has presented that historically Web development has been getting more and more client-side focused. Discuss why in his opinion it is inevitable that is the case.

Because the server side can be automated -> Cloud, Security etc. The client side is what can be controlled

8. State Postel’s Law and explain how that is a useful way to act as a web developer.

9. Explain how being a full-stack engineer is both nonsense and something maybe we should seriously aspire to be. Modify your description to be reasonable to show you understand the breadth and depth of the topic matter.

It is nonsense because the client side and the server side are too cast in itself and it very hard to be an expert in both. However, there are several positions for full stack developers and if we’re aware about both, we’ll be better developers and can integrate more easily.

10. Write the golden rule of performance. It should have four parts

Send less data. (1)	Less often (2)	From Nearby (3) _____ (4) Only when needed
------------------------	-------------------	---

For each part of the golden rule write at least one technique to accomplish the rule

- #1 Technique(s): Send Less Data: Eliminate and compress useless info,  
#2 Technique(s): Less Often: Caching  
#3 Technique(s): From Nearby: Use CDN (Content Distribution Networks)  
#4 Technique(s): When Needed: Lazy load, preload info

11. A) Explain the idea of responsive web design.

The website should be accessible by all browsers and screen sizes, and should look good on all of them. The UI should be fully integrated

B) The Professor is concerned that responsive web design is harmful to end users. Explain why he believes this despite the wide popularity of the approach.

It takes a hit on performance because more bytes= more code = slower performance  
Desktop version takes the greatest hit because under RWD (Responsive Web Design) is always mobile first dev

13.) What is the golden rule of security?

TRUST NO USER TRUST NO DATA

14.) The Golden Hammer Theme Question

a) What is a “golden hammer”?

A tool or idea that is assumed to be good and solves everything

b) Why are golden hammers dangerous to engineering thinking?

You end up using a familiar tool/framework to solve completely new projects with new constraints  
This is not good because you don't have enough arguments to determine whether it is needed

c) What is the current golden hammer in web dev?

Using Web Frameworks- React, Angular, Vue

d) What are the problems of the current golden hammer – technically speaking?

Usually results in worse performance. Some sites that do not require extensive scaling that React offers end up taking a hit on the performance because of it

e) Why might you want to wield that golden hammer anyway? The Professor may even encourage you to do so?

Because it's important to become aware of its benefits and limitations. There are tasks that golden hammers will solve very well- HOWEVER- know that it will not solve all tasks

f) Will you be more effective with that hammer now that you did what you did in this course? Explain why or why not.

Yes, because a lot of the Golden Hammer is merely popularity and now we're aware of the toll that it has on performance

g) Do you think the Professor will always have a running golden hammer example to use in this course? Explain why or why not?

## Part 5: JavaScript and the DOM

- Given the following script

```
var num = 3;
function whatIsIt () {
    var num = 6;
    num--;
    var showIt = function() { alert(num); };
    num--;
    return showIt;
}
var test = whatIsIt();
test(8);
```

What is alerted? \_\_\_\_\_ 4

Now explain why you answered the way you did?

num is decremented twice before the function “showIt” is called. When it is called, the current local value of num is 4 and therefore that is what is printed.

- JavaScript (ES5) has five primitive types what are they

string, boolean, number, null, undefined and \_\_\_\_\_

ES => EcmaScript (Standard Scripting Language For JavaScript)  
3. JavaScript (ES6) has another primitive type what is it? \_\_\_\_\_ symbol

- Given the following variables

```
let number1 = 8, number2 = 3;
```

Template literals use ` ` tags instead of single or double quotes. Use \${xyz} to embed placeholder text

Add the string “The result of  $8 * 3 = 24$ ” within a **b** tag using a template string into the first **div** tag in class “output”. Use an appropriate DOM selector and property to do this easily. *Tip: Do not hard code the individual numeric values into the string as they should be evaluated dynamically.* (2pts)

```
document.querySelector("div.output").appendChild(<b> The result of ${number1} * ${number2} = ${number1*number2}</b>)
```

- Create an **object literal** named “myDog” that has properties of name of “Zelda”, type of “Mini Schnauzer”, and a color of ‘grey’. The object also has a method called bark () that alerts out the word “Woof woof!”

Similar to a C++ struct  
An object of comma separated key value pairs  
Syntax Conventions:

- Colon separated key-value pairs
- Comma after each pair
- Last pair doesn't have comma

```
let myDog = {
    name: "Zelda",
    type: "Mini Schnauzer",
    color: "grey",
    bark: function(){
        alert("Woof woof!")
    }
}
```

object literal  
use: myDog.bark()  
(Can also be rewritten as arrow

2. On the homework you use something like `setTimeout ('someCode', 0)` to fix things. Explain what this does and why you had to use it.

JS Code only runs on one thread-> setTimeout allows a certain code to run after a specified period of time  
setTimeout is used to ensure that requests expire after a certain time period so that you don't continue  
DOM modification is done at the end of the code-> By putting setTimeout 0 you can ensure that the  
DOM modification is completed before making changes to the code

3. Evaluate the following code to values including errors if any. Be clear what type you are trying to answer with as we will not give points if we can't explicitly tell what it is. If something "depends" explain your answer.

a) "2" +true + true

2true (string)

b) "0" \* 4

0 (because it auto-convert string to num)

c) "0" \* 4

0 (because it auto-convert string to num)

d) 0 / 8

0

parses a string to return a floating pt number  
e) parseFloat("42.013x2");

42.013 (determines if the first element is a number, and then  
parses till it reaches the end of the number)

f) false || "true"

true (converts true to boolean value)

g) Number.Infinity - 1

Undefined

Number.POSITIVE\_INFINITY => Returns Infinity  
Number.NEGATIVE\_INFINITY => Returns -Infinity

h) 5 && 10

10

AND either returns the first FALSEY value or the last value  
OR either returns the first TRUTHY value or the last value

i) {} + {}

[object Object] [object Object]

j) 5 === "5"

false

## Part 6: All the Rest

1. If you did the previous question correctly your tag name was written a certain way. Explain what this and why we do that?

2. What is a PWA and how do they work? Draw a diagram(s) to explain the idea visually.

PWA- Progressive Web App. - It's an offline first approach; you just can't trust the network.  
Progressive: Can work on any browser, device, and take advantage of any features  
Discoverable: As it's a website, it should be discoverable in Web Engines  
Linkage: Use URI to indicate the current state of the application  
Responsive: All screen sizes  
Connectivity Independent: Work even in low connectivity-> Offline

Reliable  
Fast  
Engaging

3. Name the 4 APIs/technologies that are used to make native web components and describe the purpose of each. Note: One of the four has two answers, for an extra point name both in your answer and explain why it was replaced.      used to create versatile code with encapsulated functionality that can be reused whenever you want.

Tech/API 1: Custom Elements: Define your own elements and behaviour that can then be used as desired in your document

Tech/API 2: Shadow DOM: Protected sub-trees with their own CSS and DOM. This way custom components can be scripted and styled without the fear of collision against main DOM

Tech/API 3: HTML Templates: Use <template> and <slot> tags that are not displayed on the rendered page and can be reused several times. You can also use template libraries (like Handlebars)

Tech/API 4: JS Modules (alive) + HTML imports (dead): Module like system for HTML like import() require(), allow you to bundle JS, HTML, CSS and offers security of content  
All this is going to be replaced by JS modules.

4. Write only the markup for an example web component for a star rating that rates from 1-5 and has some question you are supposed to react to. Do not write JavaScript code to implement the component.

<star-rating min="1" max="5"> How chocolaty is Melody </star-rating>

5. MVC!

- a) **Model View Controller**  
Explain what MVC is and draw a picture to explain the idea. Do it solely from a server-side perspective.

Model-> The database (Stores and manipulates information)  
View -> Presentation layer of the model, Templates and constantly changes  
Controller -> URL Accepts input from user and gives response (Communication Layer)

- b) Redo the previous question to do MVC from a client-side perspective.
  - c) How does client-side MVC change our situation? Be specific with some problems you might run into with client-side MVC.
6. Describe progressive enhancement as a design philosophy in Web development as compared to graceful degradation. Which is better? If neither is better, describe which makes sense in what situation.
7. The professor made a number of odd comments like "Is your code in my markup or is your markup in my code?" to describe some separation of concerns thoughts. Try to describe these and even include markup or code snippets demonstrating you understand what he is getting at since it is a common challenge in building and maintaining apps built with the MVC pattern.

### Representation State Transfer & Create Read Update Delete

8. REST & CRUD Question - There is an endpoint located at the relative URL path /api/pets that references the Pets information show in the table below

<b>Id</b>	<b>Name</b>	<b>Type</b>	<b>Colors</b>	<b>Alive</b>
10	Rufus	Fish	White, Grey, Yellow	False
11	Cisco	Dog	Black	False
12	Pineapple	Hamster	Brown, White	True

We perform CRUD actions upon this table. If we are asked to add a Pet we use Name as *Tucker*, Type as *Dog*, Color as *Grey* and Alive as *True* and with the Id value being sequential.

Given the information above fill in the table below

CRUD Letter	Meaning	Associated HTTP Method(s)	Example HTTP Request Line	Example HTTP Body (if needed) Request Body?
C	Create  (Sends user generated data to the Web Server, helpful when you don't know the target URI )	POST	POST /api/pets HTTP/1.1	{ Name: "Tucker", Type: "Dog", Color: "Grey", Alive: "True" }
R (all)	Read	GET	GET /api/pets HTTP/1.1	None
R (one)	Read	GET	GET /api/pets /12 HTTP/1.1	None
U	Update  PUT (completely replaces what's present at the target URL with sth else) (NEED TO KNOW THE TARGET URI)		POST /api/pets/12 HTTP/1.1	{ Name: "Tucker", Type: "Dog", Color: ["Black", "Gold"], Alive: "True", }

D	Delete	DELETE	DELETE /api/pets/12 HTTP/1.1	None
---	--------	--------	------------------------------	------