**ACC Software Bootcamp - Week 24 Review**

**Part 1 - Concepts Review (60% weightage)**

Feb 18, 2017

**Your name: \_\_\_\_Matthew (answers are in red)**

**INSTRUCTIONS (read carefully):**

• Please make a copy of this document and write your answers within this document itself. After you are done, please share a link to your doc back to the instructors.

• This test is open book. You may use all resources available to you, including discussions among your peers. Please keep the discussion on Slack, so others do not get distracted. Only use **finalreview** channel. Share tips, links, advice, but not answers and code. All discussions on Final Review channel will not be considered when evaluating your submission.

• There is no time limit for this portion of the review. We suggest to try to complete this in the morning.

• If a code snippet on the review has a bug, please feel free to fix it and then answer.

• To make sure you have enough time for all questions, return to the difficult ones later. Do not get stuck on them and spend all your time.

• When providing code, we prefer codepen or plunker. Copy paste of code is fine as well.

• If you have any questions for the instructors, please ask on the finalreview channel as well.

**EVALUATION CRITERIA:**

• Evidence of understanding of the underlying concepts related to the question (50% weightage). Can you answer the "why?" in each question?

• How complete is your answer and/or your code (25% weightage). Code is complete if it can handle all possible inputs, e.g. user error.

• Does your code work, where applicable? (25% weightage).

**QUESTIONS:**

• In JavaScript, how do you append a value to an array? Does it have to be the same type as the previous elements?

The push method appends a value to an array. Example: myArray.push(“add this”);

A new value in an array does not need to be the same type as previous elements. Example: var myArray = ["Hello”, 5, {name : Matthew}];

I

• Consider the following JS:

**var** myvar = {**this**:1,that:2,nothing:['one','two','three']};

What would be the value of the following:

myvar.that outputs 2

myvar['this'] outputs 1

myvar.nothing[1] outputs’two’

myvar['nothing'] outputs [‘one’, ’two’, ’three’]

• What do the following CSS selectors select? For bonus points, please provide an example via a Plunker, jsbin or CodePen (come back to this after you have attempted other questions).

• div + p

selects all <p> elements that are placed immediately after <div> elements

• div p

selects all <p> elements inside <div> elements

• div, p

selects all <div> elements and all <p> elements

• div > p

selects all <p> elements that are immediate children of a <div> element

• Let's say you are scraping a website. How would you select every anchor tag that points to a page on nytimes.com? For bonus points, please explain how this selector works.

a[href\*=“nytimes”]

Selects every <a> element whose href attribute value contains the substring “nytimes”

• What does the following code snippet do?  
  
 div.black {

color: #fff;

}

selects div elements with class of black and changes color to white

• Please explain the Cascading in CSS. Describe how,

"Cascading" means that styles can fall (or cascade) from one style sheet to another, enabling multiple style sheets to be used on one HTML document. Inline styles take priority over internal style sheets, which take priority over external style sheets.

• who owns the stylesheet (browser, reader or author),

Author supplies the stylesheet.

• where is the styles are loaded from (linked, embedded, inline) and

linked styles are loaded from an external style sheet, example:

<link rel="stylesheet" href="/stylesheets/style.css" type="text/css">

embedded style sheets are loaded from the HTML head in between <style> tags

inline style sheets apply styles to a specific element, using the style attribute

• specificity

Specificity is the means by which browsers decide which CSS property values are the most relevant to an element and, therefore, will be applied. Specificity is based on the matching rules which are composed of different sorts of CSS selectors.

• Please explain what the following snippet does

div {

line-height: 125%;

}

selects div elements and increases the line height to 125%

• Please write, test and copy here code that would create a navigation bar and pin it to the top of the viewport (it should not move as you scroll down the page). If you use a plunker or similar test environment, please paste its link here.

https://plnkr.co/edit/nm3slqgZsn7V4A0ukmb5?p=preview

• What would be the output of the following code snippet (this has been altered from what was shared earlier)

Output is 0.

Changed ? to : to fix ternary operator.

• Please describe what would be the output of the following code snippet

This is bootstrap styling. On an xs sized device each div in the row would take up a full row of 12 columns. On a md sized device, each div in the row would take up 3 column, all 4 divs would be a full 12 column row.

• What steps can you take to optimize your CSS?

reduce redundant / duplicate code

reduce whitespace (minify / compress

don’t use frameworks like Bootstrap which slow down

• Write a JS function that returns prime numbers upto a specified max (pass the **max** as an argument to this function) (60% credit). For additional 30% credit, do not return 17 and 53 in the results. Now wrap it in an immediately invoked function expression (IIFE) for extra 10% credit.

function getPrimes(max) {

var num = [], i, j, primes = [];

for (i = 2; i <= max; ++i) {

if (!num[i]) {

primes.push(i);

for (j = i << 1; j <= max; j += i) {

num[j] = true;

}

}

}

return primes.toString();

}

getPrimes(100)

///////

function getPrimes(max) {

var num = [], i, j, primes = [];

for (i = 2; i <= max; ++i) {

if (!num[i]) {

primes.push(i);

for (j = i << 1; j <= max; j += i) {

num[j] = true;

}

}

}

primes.splice(6,1);

primes.splice(14,1);

return primes.toString();

}

getPrimes(100)

//////

(function (max) {

var num = [], i, j, primes = [];

for (i = 2; i <= max; ++i) {

if (!num[i]) {

primes.push(i);

for (j = i << 1; j <= max; j += i) {

num[j] = true;

}

}

}

primes.splice(6,1);

primes.splice(14,1);

return primes.toString();

})

(100)

• What steps can you take to optimize your CSS?

this is a duplicate question, ignoring

• What will the following code output? Why?

person = 'Nolan Ryan';

**var** sport = {

person: 'Jose DeLeon',

thrower: **function**(){

**return** person + ' threw an awesome ball.';

}

}

**var** announcement = sport.thrower()

console.log(announcement);

output: Nolan Ryan threw an awesome ball.

The function only works with the global variable. The variable inside the object is not defined.

• What will the following code output? Why?

console.log(5 != '5' ? 'Match' : 'Unmatched');

output: Unmatched

the condition (5 != '5') is false

if it was 5 == ‘5’ it would be Match

• Write a query to return all transactions for a given date. You may use any flavor of SQL.

select \* from transaction where trans\_date = '11-sep-01'

// works with the database I have with no timestamp

select \* from transaction where trans\_date <= TO\_DATE(sysdate)

TRANSACTION TABLE

• Write a Query that returns data from the LEADS and REMARKS tables below such thats

• All leads are returned, even if there are no remarks for that lead. (Hint: you would need an outer join).

• List LEAD\_ID.

• List FULLNAME (First and Last Name separated by a space).

• If the LEAD\_ACTIVE = ‘Y’, return the text “Lead is Active”. (Hint: you may use CASE). Else, return the text “Lead is Unknown”.

• REMARKS from REM\_DESC column (display only the first 100 characters, hint: use SubString function)

LEAD TABLE:

REMARKS TABLE:

To test your query, you may connect to the same Oracle database we were using in class using SQL Developer. Both tables are available on this database with the data loaded. In case you do not have it, here are the connection details:

SELECT leads.first\_name1, leads.last\_name, leads.lead\_id, leads.lead\_active,

substr(remarks.rem\_desc,0, 100)

FROM leads

left OUTER JOIN remarks

ON leads.lead\_id=remarks.lead\_id;

• What color would you expect the item Sweet Halwa to show as

red

• What response do you expect from the following

true

• What do you expect the following snippet to output?

Number("1") - 1 == 0;

output is true

• What do you expect the following snippet to output? Please explain. This explores the understanding of coercion.

(true + false) > 2 + true;

Output is false. (1+0) < 2+1

• What is the output of the following? Please explain (credit given for the explanation only and not the answer). This tests your understanding of the 2 pass execution model discussed in Advanced JS lecture. I suggest tracing through it to explain why you get the answer that you get.

function bar() {

return foo;

foo = 10;

function foo() {}

var foo = '11';

}  
 console.log(typeof bar());

The output is function.

typeof function() {} === ‘function’;

it doesn’t see var foo = ’11’

• Write a JS function that takes a sentence, splits it into an array of words and capitalizes the first letter of each word. Every other word needs to be lower case. Provide working code (a Plunk links works as well).

function capitalizeEachWord(str) {

return str.replace(/\w\S\*/g, function(txt) {

return txt.charAt(0).toUpperCase() + txt.substr(1).toLowerCase();

});

}

var x = "this is a string";

var y = capitalizeEachWord(x);

var tochop = (y);

var chopped = tochop.split(" ");

console.log(chopped)

• What is the output of the following? Why?

var arr = [];  
arr[0] = 'a';  
arr[1] = 'b';  
arr.foo = 'c';  
console.log(arr.length);

The output is 2. arr.foo is not assigning a new entry to the array, and outputs 0.

• Does adding margin-top and padding-top have any effect on inline elements? If you do not know, try a simple test in plunker. Should not take much time.

Yes, the top styling extends the length of the inline element.

• What would be color of the text "Sausage"

Blue

• What would be color of the text "Sausage". Why?

HTML

<ul class="shopping-list" id="awesome">

<li>

<span>Milk</span>

</li>

<li class="favorite" id="must-buy">

<span class="highlight">Sausage</span>

</li>

</ul>

CSS

ul#awesome #must-buy {  
 color: red;  
 }  
 .favorite span {  
 color: blue!important;  
 }

Sausage is blue because !important will be applied.

• While using ExpressJS, what is the difference between the following headers

content-type

Checks the content type.

accept

Checks if the specified content types are acceptable.

• In the following code, what would you get? Why?

2 is alerted because of hoisting.

• What does the following snippet output? Why? Credit granted only for explanation.

Returns 1.

pass 1 into s, outer invokes inner, inner invokes console.log

you pass 2 into r, but it does not do anything with the r

• What would the following code output? Why?

**var** x = 0.1;

**var** y = 0.2;

console.log(x + y == 0.3);

output is false, because variables are not equal to whole numbers.

(x + y) == 0.30000000000004

• What would the following code output to the browser? Why? (Hints: Consider how the event loop works. What happens to events? What order are they executed?)

<!DOCTYPE **html**>

<**html**>

<**head**>

<**meta charset="utf-8"**>

<**title**></**title**>

</**head**>

<**body**>

<**p id="demo"**></**p**>

<**script type="text/javascript"**>

**var message** = **"1 "**

setTimeout(**function**(){

**message** += **'2 '**

**document**.getElementsById(**'demo'**).**innerHTML** = **message**;

}, 0);

**message** += **'3 '**

</**script**>

</**body**>

</**html**>

the order is 1,3,2

• What does the following code do? How can you reverse the order?

**var points** = [40, 100, 1, 5, 25, 10];

**points**.sort(**function**(roo,raa){**return** roo-raa});

Sorts items from smallest to largest.

Reverse order with {return raa-roo} instead of {return roo-raa}

• You have committed and pushed a bad change to a git repo. How can you fix it, without destroying history, since others might have pulled your bad change and based their changes on yours?

git commit -m "Something terribly misguided"

git reset HEAD~

<< edit files as necessary >>

git add ...

• Can you think of 2-3 methods you refer to a particular commit? Briefly describe each method.

Use git checkout <sha1> to check out specific commit

Identify the hash of the commit, using git log, then use git revert <commit> to create a new commit that removes these changes

git commit --amend, edit commit message and save the commit (change commit message)

• How do you delete a branch after completing work on it.

git branch -d <<branch\_name>>

• What is the difference between "cloning" and "forking" a repository? Why would you use one or the other?

When you are Forking a repository you are creating a copy of repository under the GitHub Id. Any changes made to the original original repository will be reflected back to your forked repository. However, if you make any changes to your forked repository you will have to explicitly create a pull request to the original repository. When your pull request is approved by the administrator of the original repository, then your changes will be merged with the existing original code-base. Until then, your changes will be reflected only in the copy you forked.

A Clone is where you have proper duplication, and separation between, two versions of a repository. When one repository is amended, the new content must be actively copied to the other repository using a push command. And changes in the other repository are fetched.

The forked repo allows you to punish work for code review purposes, but not for active development. It is not on your computer, it is on the github server. The cloned rep is active, you do work in it but other people probably don’t have access to it on your computer.

• Review this code. What would happen to the p#example element?

HTML:

<p id="example">Hello</p>

CSS:

#example {  
 margin-bottom: -5px;  
}

• It would move #example up by 5 pixels

• It would move #example down by 5 pixels

• It would move all later elements up by 5 pixels

• None, but this is what would happen ...

It would move all later elements up by 5 px.

• Does the HTML below trigger a HTTP request when the page first loads?

<img src="mypic.jpg" style="visibility: hidden" alt="My Pic">

Yes, it still occupies space in the DOM. Unlike display: none

• What do the following two expressions evaluate to, in JS:

"1" + 2 + "3" + 4

4 + 3 + 2 + "1"

1234

91

• Can time of creation be determined from system generated ObjectID in MongoDB?

Yes, getTimestamp()

• Consider the following snippet of code. Please explain what everything does or means

>db.mycol.find({},{"title":1,\_id:0}).limit(2)

{"title":"MongoDB Overview"}  
{"title":"NoSQL Overview"}

queries the mycol collection

displays the title of the document

limit to 2 returns

CONGRATULATIONS !

END OF PART 1