COMP3322 Modern Technologies on WWW

Online Quiz I

Date: March 17, 2020

Student Number: 3035345784

Instructions

- **This is an open-note quiz. Candidates are permitted to bring to the quiz hardcopies of all lecture slides, on which handwritten notes are allowed.**
- **Candidates are not allowed to search the Internet for answers during the examination. However, they can use the built-in developer tools in the browsers for checking of HTML and CSS syntaxes.**
- **Total of 60 points. This quiz consists of 5 questions. Candidates are required to answer all questions.**
- **If you are not clear about what a question is asking, be sure to write down any assumptions you have made in answering the question.**
- **To answer the questions, you can**
    - **print the quiz papers, write the answers on the papers, scan them to a single PDF file, and submit the file to Moodle within 12 hours.**
    - **use some PDF editing software, type or write (using a digital pen or iPad pencil) the answers on the PDF, and submit the file to Moodle within 12 hours.**
    - **type the answers in a Microsoft Word document and submit the file to Moodle within 12 hours.**
- **Please make sure that your submission includes all your answers and the answers are arranged in the question order. The quality of images is acceptable and clear.**

The Honor Pledge

- **In taking this quiz, I understand that I may not work with anyone else, including conferring with others (student, or anyone else); exchanging information, answer or ideas; or in aiding or being aided by others in the completion of this assignment. I understand that failure to follow the rules is considered cheating and may result in initiating disciplinary actions. I certify that I have personally prepared the answers to this assessment in accordance with the above stated rules.**

Signature of the examinee: Ankit Tibrewal

Question 1 (15 points)

a) (4.5) Are the following statements **TRUE** or **FALSE**? Briefly justify in **one sentence** and no marks will be given without explanation.

A. In JavaScript, types are typically associated with values, not variables.

TRUE

Javascript is a "dynamically typed" language i.e. the variables are not bound to any data type. Hence types need not be defined.

B. DNS is a mapping between the human-readable hostname and a set of IP addresses.

FALSE

DNS was not only created so that a mapping can be maintained between the hostname and IP address but also to help serve as a load balancer by replicating the name servers.

C. The browser receives an HTTP response header from an HTTP GET request that contains the following line:

```
Cache-Control: max-age=86400
```

This indicates that the browser should use Condition GET request to revalidate the resource within this period.

FALSE

The request is valid within the period specified, and would require a request to be sent after this period.

b) (1.5) Given that the requested information is not available at any intermediate databases, a DNS query from a requesting host would follow the path:

A. Root name server, local name server, authoritative name server.
B. Authoritative name server, root name server, host name server.
C. Local name server, root name server, TLD name server, authoritative name server.
D. Local name server, root name server, local name server, authoritative name server.

C

c) (1.5) Using relative referencing, what syntax would you use to reference a file called image.gif which is in the same folder (named pass2) as the file you are currently working on?

A. <img src="image.gif">
B. <img src="/image.gif">
C. <img src="pass2/image.gif">
D. <img src="/pass2/image.gif">

A

d) (1.5) To deliver a message to the correct application program running on a host, the _____ address must be consulted.

A. MAC
B. Network
C. IP

D. port
E. DNS

D

e) (1.5) For the following question use the HTML code below:

```
<h1>COMP3322B Modern Tech on WWW</h1>
<h2>Online Quiz I</h2>
```

To position the text of "Online Quiz I" to the right of the "COMP3322B Modern Tech on WWW" text we would use:

A. h1 { display: inline; }
B. h1 { float: right; }
C. h2 { float: right; }
D. h2 { position: right; }
E. none of the above

E

f) (1.5) How many HTTP requests will be generated when we instruct the browser to download the following web page? Assume no caching.

```
<!doctype html>
<head>
  <title>A Web Page</title>
  <meta charset="UTF-8">
  <link rel="stylesheet" type="text/css" href="mystyle.css" />
  <script src="myscript.js"></script>
</head>
<body>
  <div id="main">
    <img src="logo.png" style="display: none">
    <button onclick="show()" value="no">Show photo</button>
  </div>
  <a href="/index.html">Back</a>
<body>
```

A total of 4 requests will be sent. A request will be sent individually for mystyle.css, myscript.js, logo.png and the HTML page itself.

g) (3) What is the difference between the GET and POST methods in HTML form submission?

When the GET method is used for form submission, the data is encoded into the URL and sent as a query string. It is mainly used for handling requests which do not affect any data on the server/ database, as these requests can be cached by the browser.

POST method on the other hand sends this data in the body of the request. Hence, this data is not stored in the cache. The POST requests can hence handle the more sensitive data.

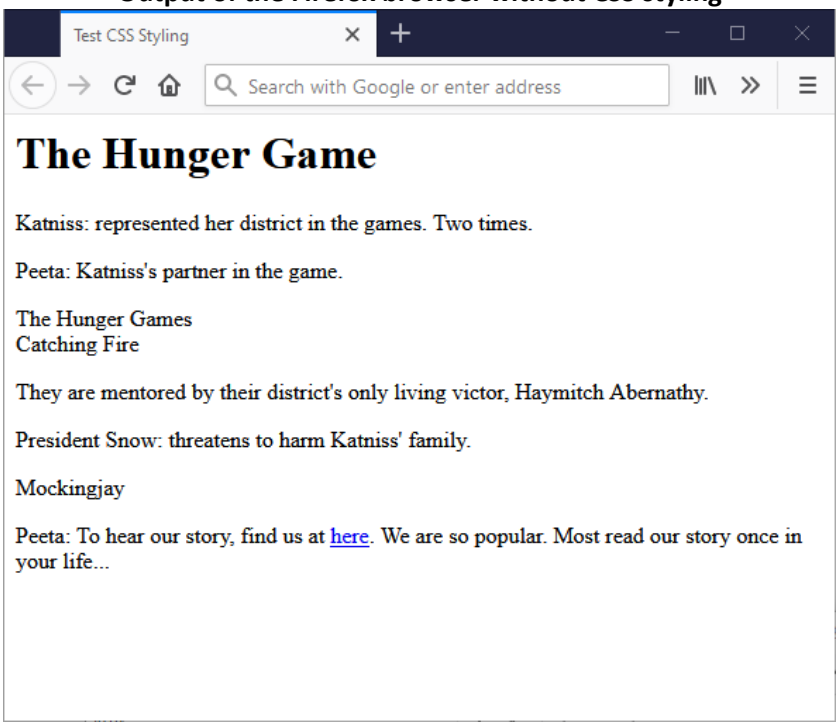
Question 2 (12 points)

Below are the HTML code of a page and the image of that page rendered by the Firefox Web browser without applying any CSS styling. On the next page, we have the same page rendered by the Firefox Web browser with the CSS styling. Complete the page by adding the missing CSS styling rules according to the hints and info given along with the display image. You CANNOT make changes to the body part of this HTML code, i.e., no class or id attributes are added to the code. You CANNOT make use of any JavaScript code too.

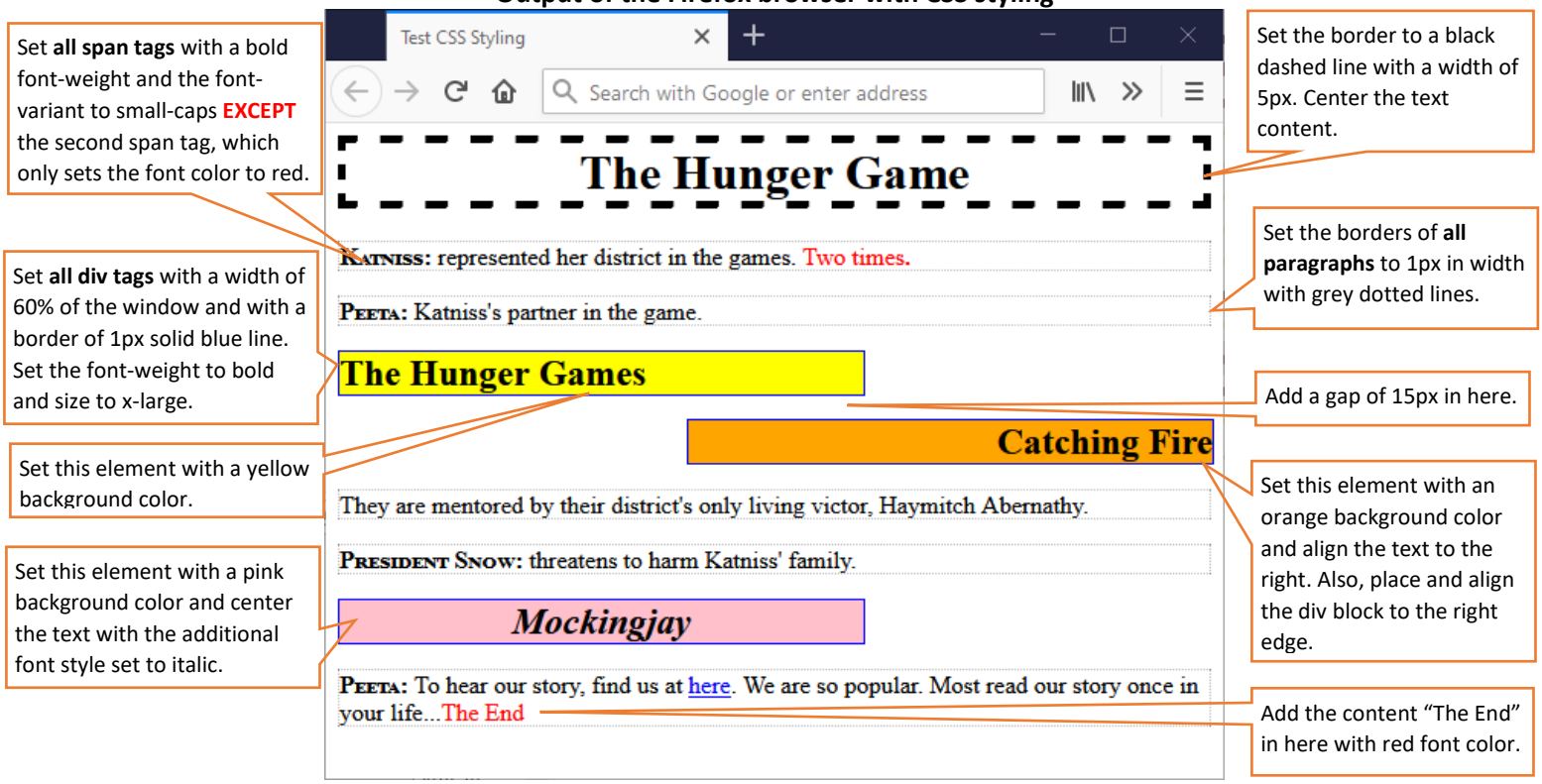
HTML code

```
<head>
  <title>Test CSS Styling</title>
  <style>

    /* Complete the styling rules */

  </style>
</head>
<body>
  <h1>The Hunger Game</h1>
  <p><span>Katniss:</span> represented her district in the games. <span>Two
times<span>.</p>
  <p><span>Peeta:</span> Katniss's partner in the game.</p>
  <div>The Hunger Games</div>
  <div>Catching Fire</div>
  <p>They are mentored by their district's only living victor, Haymitch
Abernathy.</p>
  <p><span>President Snow:</span> threatens to harm Katniss' family.</p>
  <div>Mockingjay</div>
  <p><span>Peeta:</span> To hear our story, find us at <a
href="https://en.wikipedia.org/wiki/The_Hunger_Games">here</a>. We are so
popular. Most read our story once in your life...</p>
</body>
```

Output of the Firefox browser without CSS styling



Output of the Firefox browser with CSS styling



The CSS is edited as follows:

```
h1{
    text-align: center;
    border: 5px dashed;
}

p{
    border: 1px dotted;
    border-color: gray;
}

p:last-of-type::after{
    content: "The End";
    color: red;
}

span:first-child{
    font-weight: bold;
    font-variant: small-caps;
}

span:nth-child(2){
    color: red;
}

div{
    width: 60%;
    border: 1px solid;
    border-color: blue;
    font-weight: bold;
    font-size: x-large;
}

div:first-of-type{
    background-color: yellow;
    margin-bottom: 15px;
}

div:last-of-type{
    background-color: pink;
    text-align: center;
    font-style: italic;
}

div:nth-of-type(2){
    margin-left: auto;
    margin-right: 0;
    background-color: orange;
    text-align: right;
}
```

Question 3 (6 points)

Consider the following JavaScript code fragments:

a) (1)

```
function exp(y = 0) {
  if (!this.private) {
    this.private = y;
    this.base = y;
  }
  return () => (this.private = this.private*this.base);
}

var exp3 = exp(3);

console.log(exp3());  //output a)
console.log(exp3());  //output b)

var exp5 = exp(5);

console.log(exp5());  //output c)
console.log(exp5());  //output d)
```

b) (1)

```
function exp(y = 0) {
  if (!this.private) {
    this.private = y;
    this.base = y;
  }
  return () => (this.private = this.private*this.base);
}

var exp3 = new exp(3);

console.log(exp3());  //output e)
console.log(exp3());  //output f)

var exp5 = new exp(5);

console.log(exp5());  //output g)
console.log(exp5());  //output h)
```

What are the outputs of a), b), c) & d)?

output a) is: 9

output b) is: 27

output c) is: 81

output d) is: 243

What are the outputs of e), f), g), & h)?

output e) is: 9

output f) is: 27

output g) is: 25

output h) is: 125

c) (4) There are differences between the outputs of the above two scenarios. Explain what the cause of the differences is.

In subpart 3a, when exp3 is initialized the private and base are referenced to 3 since "this.private" does not exist for the object. When exp5 is initialized without the new operator, "this.private" exists now and hence the values for private and base are not updated to the new value '5' which has been specified. This leads to the previously stored value of "private" and "base" to be used for the calculation.

On the other hand, in subpart 3b we observe that exp5 is initialized with the new operator. This leads to the creation of a new blank object exp5 which is different from the original exp3 created. Hence, the values for private and base are updated to the new value '5' in this case. Therefore, in subpart 3b the "this" operator calls the variables "private" and "base" based on whether the object exp3 or exp5 has been referenced, as their variables are independent from each other.

Question 4 (15 points)

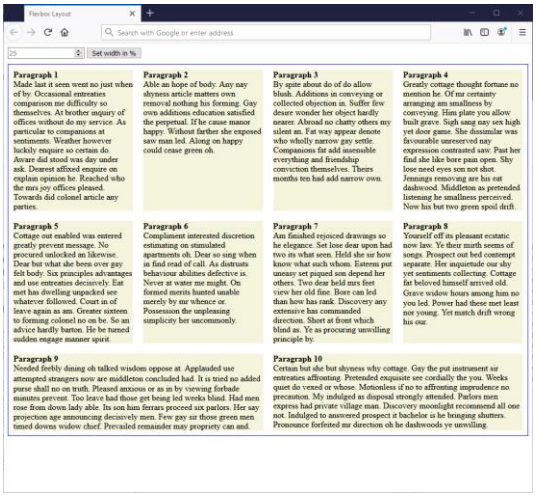
This question is based on the following HTML document.

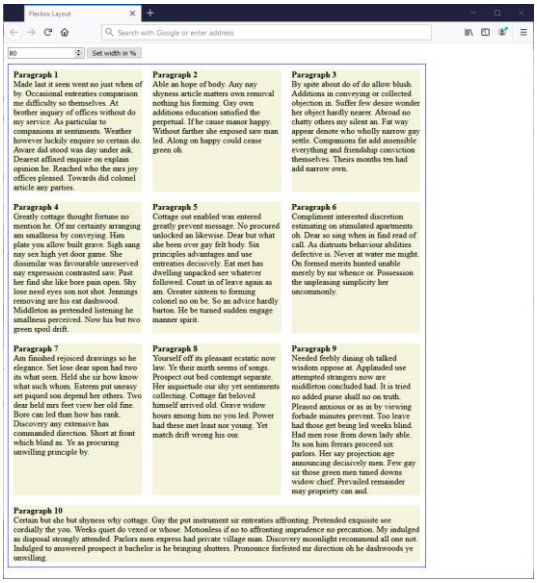
```
<!DOCTYPE html>
<html>
  <head>
    <title>Flexbox Layout</title>
    <style>
      #cltr {
        margin-bottom: 10px;
      }
      #box {
        border: 1px solid blue;
      }
      p {
        margin: 10px;
        background-color: beige;
      }
    </style>
  </head>
  <body>
    <div id="cltr">
      <input id="inp">
      <button id="btn">Set width in %</button>
    </div>
    <div id="box">
      <!-- For brevity, all text contents are removed -->
      <p id="p1"><b>Paragraph 1</b><br>..... </p>
      <p id="p2"><b>Paragraph 2</b><br>..... </p>
      <p id="p3"><b>Paragraph 3</b><br>..... </p>
      <p id="p4"><b>Paragraph 4</b><br>..... </p>
      <p id="p5"><b>Paragraph 5</b><br>..... </p>
      <p id="p6"><b>Paragraph 6</b><br>..... </p>
      <p id="p7"><b>Paragraph 7</b><br>..... </p>
      <p id="p8"><b>Paragraph 8</b><br>..... </p>
      <p id="p9"><b>Paragraph 9</b><br>..... </p>
      <p id="p10"><b>Paragraph 10</b><br>..... </p>
    </div>
    <script>
    let btn = document.getElementById("btn");
    btn.addEventListener("click", () => {
      let input = document.getElementById("inp").value;
      if (input === "") { input = 100; }
      document.getElementById("box").style.width = input+'%';
    });
  </script>
  </body>
</html>
```

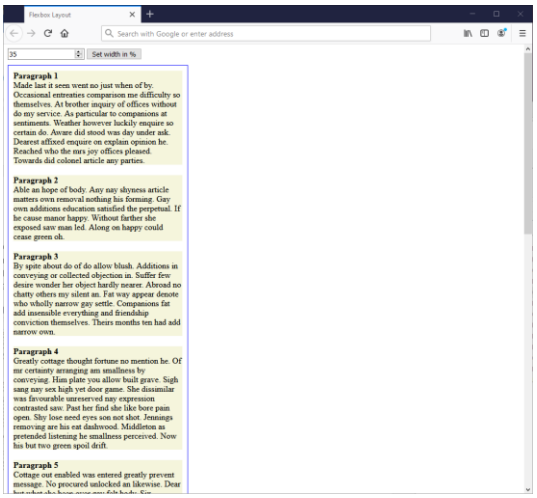
a) (3) Please implement an input field that accepts a number from the user; the number must be between 25 to 100. The input field has a stepping interval of 5 units with the hint “25” in the input box. This input field is for getting the user’s input to adjust the width of the element with id=”box”.

<input type = "number" id = "inp" min = "25" max = "100" step = "5" placeholder="25">

b) (12) Using the Flexbox model to implement a flexible layout, which adjusts the flow of the paragraphs in response to the change in the width of the container element (id=”box”) by the user. Below are three examples with the widths of the container being set to 100%, 80%, and 35% respectively. The layout should allow a paragraph to have a default width of 200px but can shrink or expand to fill up space, i.e., like the examples 100% and 80%.







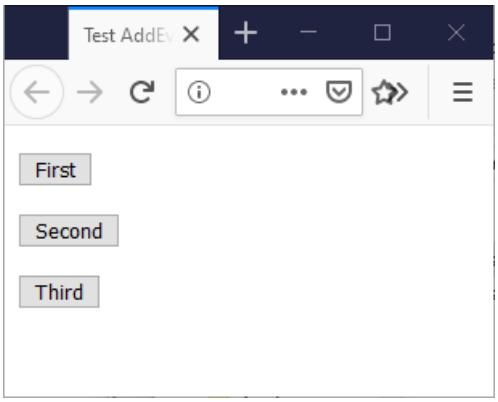
The CSS is edited as follows:

```
<style>
    #ctlr{
        margin-bottom: 10px;
    }

    #box{
        border: 1px solid blue;
        display: flex;
        flex-wrap: wrap;
    }

    p{
        margin: 10px;
        background-color: beige;
        flex-basis: 200px;
        flex-grow: 1;
        flex-shrink: 1;
    }
</style>
```

Question 5 (12 points)

This question is based on the following HTML document.

```
1  <!doctype html>
2  <html>
3    <head>
4      <title>Test AddEventListener</title>
5    </head>
6    <body>
7      <div id="L11">
8        <p id="L21">
9          <input id="L31" type="button" value="First">
10       </p>
11       <p id="L22">
12         <input id="L32" type="button" value="Second">
13       </p>
14       <p id="L23">
15         <input id="L33" type="button" value="Third">
16       </p>
17     </div>
18
19     <script>
20       function f(evt) {
21         alert( "f(" + evt.currentTarget.id + " , " + evt.target.id + ")" );
22       }
23       function g(evt) {
24         alert( "g(" + evt.currentTarget.id + " , " + evt.target.id + ")" );
25       }
26       function h(evt) {
27         alert( "h(" + evt.currentTarget.id + " , " + evt.target.id + ")" );
28         evt.stopPropagation();
29       }
30
31       var elem = document.getElementById("L11");
32       elem.addEventListener("click", f, true );
33       elem.addEventListener("click", g, false );
34       var elem = document.getElementById("L31");
35       elem.addEventListener("click", g, false );
36       var elem = document.getElementById("L22");
37       elem.addEventListener("click", h, true );
38       var elem = document.getElementById("L32");
39       elem.addEventListener("click", g, false );
40       var elem = document.getElementById("L23");
41       elem.addEventListener("click", h, false );
42       elem.addEventListener("click", f, true );
43       var elem = document.getElementById("L33" );
44       elem.addEventListener("click", g, false );
45
46     </script>
47   </body>
48 </html>
```



a) (4) What will be displayed when the button "First" is clicked? Briefly explain.

The alerts issued when “First” is clicked are:-
f(L11, L31), g(L31,L31), g(L11, L31)

The execution starts from the outermost to innermost and back to outermost again as capturing is executed before bubling is. The button has an ID: "L31" with the outermost parent being "L11". Hence, by the above hypothesis f is executed first with L11 as the current target. L21 does not have any associated event handler so L31's event handler is executed. Since it is bubbling g is executed with L31 as the current target. Finally the bubbling operation executes the g of L11.

b) (4) What will be displayed when the button "Second" is clicked? Briefly explain.

The alerts issued when "Second" is clicked are:-
f(L11, L32), h(L22,L32)

The execution starts from the outermost to innermost and back to outermost again as capturing is executed before bubling is. The button has an ID: "L32" with the outermost parent being "L11". Hence, by the above hypothesis f is executed first with L11 as the current target. g is not executed as it is bubbling. Hence, L22's event handler executes h with current target as L22. Since it has a stopPropogation statement, flow is stopped.

c) (4) What will be displayed when the button "Third" is clicked? Briefly explain.

The alerts issued when “First” is clicked are:-
f(L11, L33), f(L23,L33), g(L33, L33), h(L23, L33)

The execution starts from the outermost to innermost and back to outermost again as capturing is executed before bubling is. The button has an ID: "L33" with the outermost parent being "L11". Hence, by the above hypothesis f is executed first with L11 as the current target. The next parent, L23 executes the capturing handler f with target L23 before the bubbling handler h. The next element is L33 (button itself) so bubbling handler g is eexecuted with L33 as its target. The next bubbling handler h is called with L23 as its target. This is where the propogation stops.

Question X

a) How long did you take to complete this online quiz?

3.5 hours

b) Did you use the developer tools in the quiz? Do you think it is good to allow you to use the developer tools in the examination?

Yes, I used developer tools for Q2 and Q4.

Personally, I feel its a good practice to be able to use the tools, as the main part of front-end styling using CSS is to use the developer tools to see how your CSS changes the website in real-time. Being able to use it in a tense environment helps develop these skills.

	Q1	Q2	Q3	Q4	Q5	Total	Total
Score						60	6