Section A: Multiple Choice Questions (30%)

1. What is the correct HTML for making a text input field? (2%)

A. <input type="textfield">
B. <textfield>
C. <input type="text">
D. <textinput type="text">

2. How do you display hyperlinks without an underline? (2%)

A. a {decoration:no-underline;}
B. a {text-decoration:none;}
C. a {underline:none;}
D. a {text-decoration:no-underline;}

3. How do you display a border like this: (2%)

The top border = 10 pixels
The bottom border = 5 pixels
The left border = 20 pixels
The right border = 1pixel?

A. border-width:10px 1px 5px 20px;
B. border-width:5px 20px 10px 1px;
C. border-width:10px 20px 5px 1px;
D. border-width:10px 5px 20px 1px;

4. What is the correct JavaScript syntax to change the content of the HTML element below? (2%)

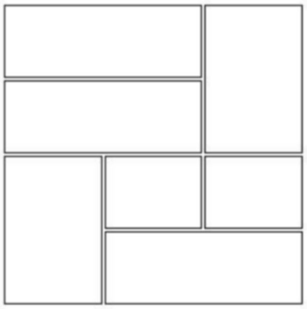
```
<p id="demo">This is a demonstration.</p>
```

A. document.getElementById("demo").innerHTML = "Hello World!";
B. #demo.innerHTML = "Hello World!";
C. document.getElement("p").innerHTML = "Hello World!";
D. document.getElementByName("p").innerHTML = "Hello World!";

5. If you wanted to round the corners of a block element, which style property would you apply? (2%)

A. border:
B. border-arc:
C. border-deg:
D. border-radius:

6. Consider the table shown below. Which line of code would create the last row of the table? (2%)



   A. <tr><td></td><td rowspan=”2”></td></tr>
   B. <tr><td colspan=”2”></td><td rowspan=”2”></td></tr>
   C. <tr><td rowspan=”2”></td></tr>
   D. <tr><td colspan=”2”></td></tr>

7. How many alert boxes will the following loop produce? (2%)

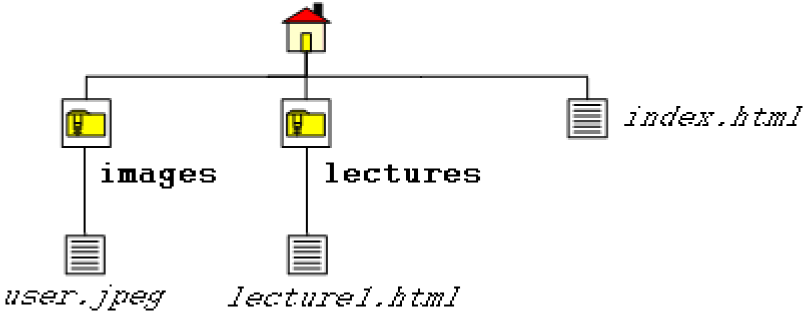
```
for ( var Index = 0; Index <= 2; ++Index )
alert( Index );
```

   A. 0
   B. 1
   C. 2
   D. 3

8. Which of the following is the code for accessing the contents of an input box named userData on a form?

   A. document.forms[0].userData
   B. document.forms[0].userData.value
   C. document.forms[0].userData.contents
   D. document.forms[0].userData.data

9. Given a file structure in the figure below, if you are working on the source code of index.html, which of the following is the correct relative URL for lecture1.html? (2%)



   A. lecture1.html
   B. lectures/lecture1.html
   C. ../lectures/lecture1.html
   D. ./lecture1.html

10. The syntax for adding a textfield of 3 rows, each of width 40 is: (2%)

A. <textfield rowspan=”3” colspan=”40”>blah blah</textfield>

B. <textfield rows=”3” width=”40”>blah blah</text>

C. <text row=”3” width=”40”>blah blah</text>

D. <textarea rows=”3” cols=”40”>blah blah</textarea>

11. Indicate whether each of the following statements is True or False (1.5% each - 6%)

A. This is a valid JSON. True

```
{
   "id": 1,
   "name": "A green door",
   "price": 12.50,
   "tags": ["home", "green"]
}
```

B. In an HTML form, the method attribute is used to specify the script file that processes the form data (False)

C. Domain Name Servers map symbolic computer names to their host names (False, to IP Address)

D. XML elements cannot be empty (False)

12. What is the output when you run the following programs: . . .(0.5% each – 4%)

```
var i = 25;
var j = “25”;
var k = 2 + “5”;
var l = “2” + “5”;
```

(a) document.write (i == j); true

(b) document.write (i === j); false

(c) document.write (j == k); true

(d) document.write (j === k); true

(e) document.write (i == k); true

(f) document.write (i === k); false

(g) document.write (k == l); true

(h) document.write (k === l); true

Section B: Longer Answer Questions (30%)

1. Fill in the following gaps: (1.5% each gap - 9%) Other responses might also be correct, as long as a response is correct will get the points.

A. The purpose of __________(Transport Layer/TCP) is to ensure the integrity of network communication.

B. __________ (Dynamic) routing is a technique to select the best route according to the network conditions at the time of transmission.

C. What is the required attribute of the <video> element in HTML5, when the video is in a single format? __________(src)

D. Consider the following JavaScript statement:

var myLinks =getElementsByTagName (“a”);

What is the data type of the result? __________(an array).

E. If you want to control which element sits on the top (when elements overlap) you use the __________ (z-index) property.

F. The __________ (content) part of the box (box model), is where text and images appear.

2. Select all valid XML tag names (1%)

[X] <_Element1>
[ ] <1Tag>
[X] <My.Element>
[ ] <#Elem&ent>
[X] <My-Element_Name]
[ ] <XmL>

3. Fill in the empty cells of the table with the correct word (0.3% each field – 3.6%)

Image Type	**File Extension**	**Compression** (lossy or lossless)	**Transparency** (yes or no)	**Animation** (yes or no)	**Colors** (hundreds or millions)
GIF	.gif	Lossless	yes	yes	hundreds
JPEG	.jpg or jpeg	Lossy	no	no	millions
PNG	.png	Lossless	yes	no	millions

PNG can also be lossy, so that’s correct too.

4. Write the appropriate position property value for each of the following description (1% each - 4%).

Property Value	**Description**
static	This is the default setting – no special positioning.
absolute	The box is taken out of normal flow and no longer affects the position of the other elements.
relative	This property value moves an element in relation to where it would have been in normal flow.
fixed	Configures the location of an element within the browser viewpoint – i.e. the element doesn’t change position if scrolling content.

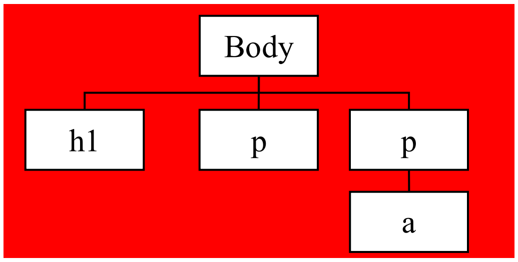
5. Explain the key differences between Canvas and SVG, no need to write more than 4-5 bullet points (4%).

- In Canvas elements are drawn programmatically while in SVG are parts of the page’s DOM
- In Canvas drawing is done with pixels while in SVG is done with vectors.
- In Canvas animations are not built-in while in SVG animations are built-in
- Canvas has high performance for pixel-based drawing operations while SVG is based on standard XML syntax, which provides better accessibility

6. Consider the following HTML file:

```
<!DOCTYPE html>
<html>
<head>
<style>
/* unvisited link */
□{
   background-color: lightblue;
}

/* visited link */
□{
   background-color: lightblue;
}

/* mouse over link */
□{
   background-color: yellow;
}

/* selected link */
□{
   background-color: yellow;
}
</style>
</head>
<body>

<h1>This is a Heading</h1>
<p>This is a paragraph.</p>
<p><a href="http://www.ntnu.no">NTNU</a></p>
</body>
</html>
```

A. Build (draw) the Document / HTML tree of the code above (0.4%)



B. Write the CSS Pseudo-Classes needed (4 boxes) to set the background color for visited and unvisited links to "lightblue", and the background color for the mouse over and active links states to "yellow". (8% - 2% each)

```
    /* unvisited link */
    a:link {
       background-color: lightblue;
    }
    /* visited link */
    a:visited {
       background-color: lightblue;
    }
    /* mouse over link */
    a:hover {
       background-color: yellow;
    }
    /* selected link */
    a:active {
       background-color: yellow;
}
```

Section C: Coding Related Questions (40%)

Note: your code need not match my code exactly. I would be shocked if it did! The solutions are just few of many ways to solve the problems. Capturing the ideas is what's important.

1. Write code for a JavaScript function that: (6% - 1.5% each)
    a. reads in two numbers from an HTML form
    b. determines which number is the largest
    c. outputs to screen which number is the largest (in the format "44 is larger than 22")

   and

    d. write an appropriate HTML form which will call this function

```
function largest(){
var numA = parseInt(document.myForm.numA.value);
var numB = parseInt(document.myForm.numB.value);
if (numA > numB){
      document.write(numA + " is larger than "+numB);
}
else if (numA < numB){
      document.write(numB + " is larger than "+numA);
}
else{
      document.write("Both numbers are equal");
}
}
```

```
<form name="myForm" action="">
Enter first number <input type="text" name="numA"><br>
Enter second number <input type="text" name="numB"><br>
<input type="button" value="click" onclick=largest();>
</form>
```

2. Write a CSS rule that configures a class called “company” in bold, serif font, and 1.25em in size. (2%)

   ```
   .company { font-weight: bold;
                   font-family: serif;
                   sonf-size: 1.25em; }
   ```

3. Consider the following HTML:

   ```
   <!DOCTYPE html>
   <html>
   <head>
   <style>
   …
   </style>
   </head>
   <body>


   <p>In my younger and more vulnerable years my father gave me some advice that I've
   been turning over in my mind ever since. 'Whenever you feel like criticizing anyone,' he
   told me, 'just remember that all the people in this world haven't had the advantages that
   you've had.'</p>


   </body>
   </html>
   ```

   Write the CSS Pseudo-Classes needed to (6% - 2% each):

   a. Set text color to red, for the first line of the <p> element.

   ```
   p::first-line {

      color: red;

   }
   ```

   b. Set text color to "blue", and the text size to "xx-large", for the first letter of the <p> element.

   ```
   p::first-letter {

      color: red;

      font-size: xx-large;

   }
   ```

c. Insert the image "smiley.gif" before, and after <p> elements.

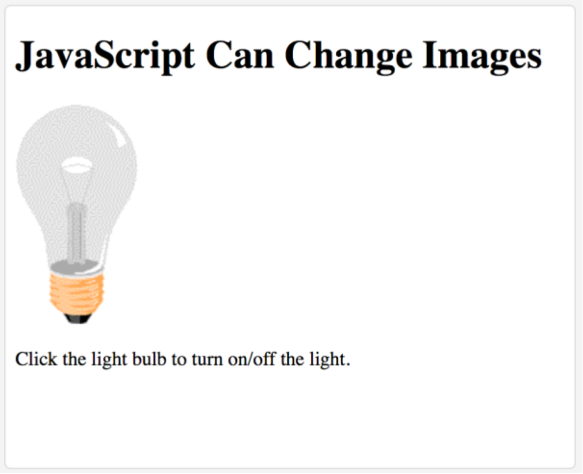
Hint: it will look something like this, we need you to write the two missing CSS Pseudo-Classes.

```
□{
            content: url(smiley.gif);
    }
□{
            content: url(smiley.gif);
    }
```

```
p::before{
        content: url(smiley.gif);
}
p::after{
        content: url(smiley.gif);
}
```

4. Below is an outline for a JavaScript program that turns on and off the image of a bulb, when clicked, as shown in the before and after pictures below.

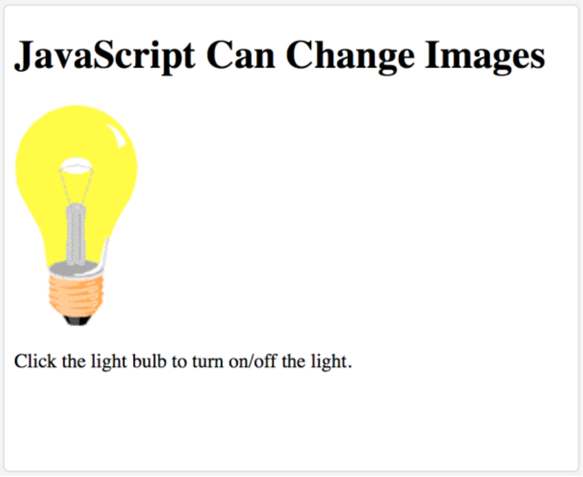
Before



JavaScript Can Change Images

Click the light bulb to turn on/off the light.

After



JavaScript Can Change Images

Click the light bulb to turn on/off the light.

Your task is to fill in the missing JavaScript code so that the program works correctly. Notice that there are two images: pic_bulboff.gif and pic_bulbon.gif. (10% - 2% each box)

```
<!DOCTYPE html>
<html>
<body>

<h1>JavaScript Can Change Images</h1>
<img id="myImage" onclick="changeImage()" src="pic_bulboff.gif" width="100"
height="180">

<p>Click the light bulb to turn on/off the light.</p>

<script>
function      □() {
    □
    if (image.src.match("□")) {
        □
    } else {
        □
    }
}
</script>
</body>
</html>
```

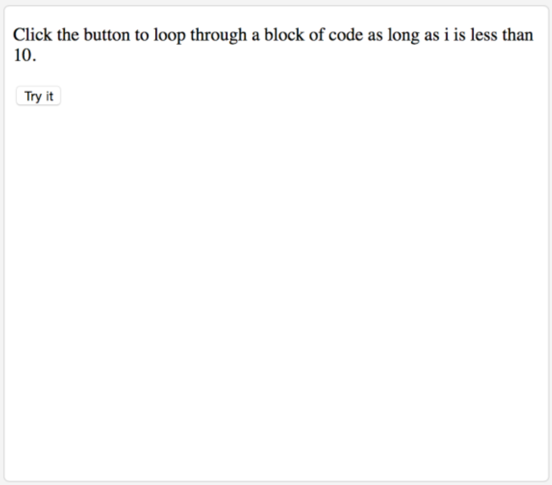
Here is an indicative solution, you don’t have to use all the boxes or have exactly the same code. There are many correct solutions!

```
<!DOCTYPE html>
<html>
<body>
<h1>JavaScript Can Change Images</h1>
<img id="myImage" onclick="changeImage()" src="pic_bulboff.gif" width="100"
height="180">

<p>Click the light bulb to turn on/off the light.</p>

<script>
function changeImage() {
    var image = document.getElementById('myImage');
    if (image.src.match("bulbon")) {
        image.src = "pic_bulboff.gif";
    } else {
        image.src = "pic_bulbon.gif";
    }
}
</script>
```

```
</body>
</html>
```

5. Below is an outline for a JavaScript program, with a “Try it” button that, when clicked, produces the output as shown in the before and after pictures below.

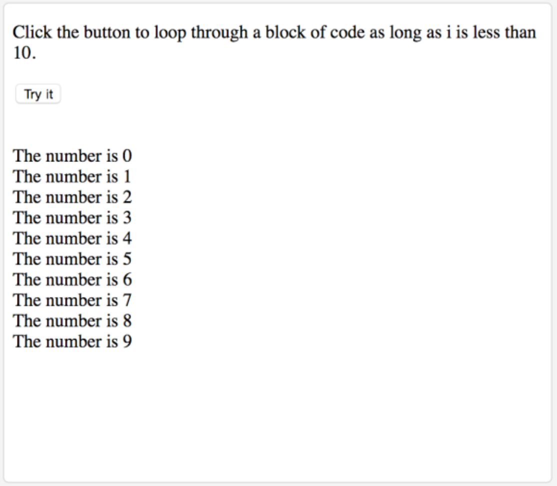
Before



Click the button to loop through a block of code as long as i is less than 10.

Try it

After



Click the button to loop through a block of code as long as i is less than 10.

Try it

```
The number is 0
The number is 1
The number is 2
The number is 3
The number is 4
The number is 5
The number is 6
The number is 7
The number is 8
The number is 9
```

Your task is to fill in the missing JavaScript code so that the program works correctly. You must use a “while” loop statement, as indicated. (10% - 2% each box)

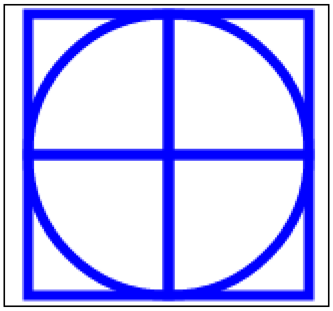
```
<!DOCTYPE html>
<html>
<body>

<p>Click the button to loop through a block of code as long as i is less than 10.</p>

<button onclick="myFunction()">Try it</button>

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

<script>
function myFunction() {
   □
   □
   while (i < 10) {
      □
      □
   }
   □
}
</script>
```

```
</body>
</html>
```

Here is an indicative solution, you don’t have to use all the boxes or have exactly the same code. There are many correct solutions!

```
<!DOCTYPE html>
<html>
<body>
<p>Click the button to loop through a block of code as long as i is less than 10.</p>

<button onclick="myFunction()">Try it</button>

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

<script>
function myFunction() {
    var text = "";
    var i = 0;
    while (i < 10) {
        text += "<br>The number is " + i;
        i++;
    }
    document.getElementById("demo").innerHTML = text;
}
</script>

</body>
</html>
```

6. Write JavaScript code to draw the following figure inside a <canvas> element:
   Don’t forget to access the canvas element from your script and use the 2D drawing API. We expect you to approximately draw the following figure, eg. line width, not fully centered to the canvas, color of the lines etc. are not problems. (6%)



```
<!DOCTYPE html>
<html> <body>
<canvas id="Canvas1" width="150" height="150" style="border:1px solid
#d3d3d3;">Your browser does not support canvas.</canvas>

<script>
…

</script>
</body> </html>
```

Here is an indicative solution, you don’t have to use this type of drawings (e.g., rectangle can simply be four different lines etc.). Again there are many correct solutions!

```
<script> window.onload = function() {
 var theCanvas = document.getElementById('Canvas1');
 if (theCanvas && theCanvas.getContext) {
 var ctx = theCanvas.getContext("2d");
 if (ctx) {

  // draw the rectangle
  ctx.beginPath();
  ctx.strokeStyle = "blue";
  ctx.lineWidth = 5;
  ctx.strokeRect(5,5,140,140);

  // draw the cicle
  ctx.beginPath();
  ctx.lineWidth = 5;
  ctx.arc(75,75,70,0,Math.PI*2,true);
  ctx.stroke();

  // draw the crossed lines
  ctx.beginPath();
  ctx.lineWidth= 5;
  ctx.moveTo(5, 75);
  ctx.lineTo(145, 75);
  ctx.moveTo(75, 5);
  ctx.lineTo(75, 145);
  ctx.stroke();
  ctx.stroke();

 }}}</script>
```