09/23

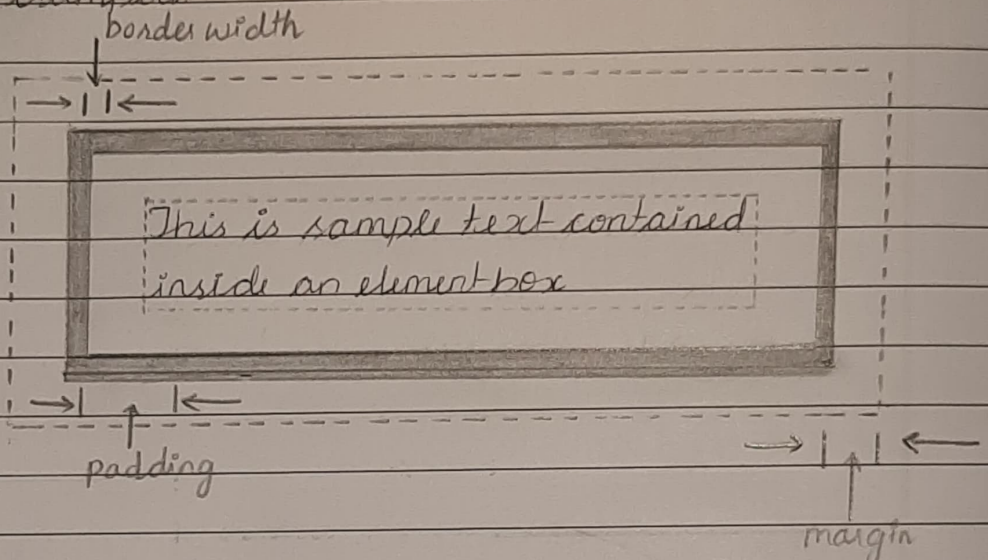
D D M M Y Y Y Y

Page No.

Assignment-1

1. Illustrate the box model with padding and margin properties in CSS

Ans
- borders have no gaps inside or outside of them
- to introduce gaps to make the elements look comfortable, not cramped



Padding and margin properties

- padding specifies the width of the area on the interior of an element's borders
- margin property specifies the width of the area on the exterior of an elements borders
- label {
  border : solid ;
  padding : 20px ;
  margin : 20px ;
  }
- border-width property specify elements different padding widths for the four different sides

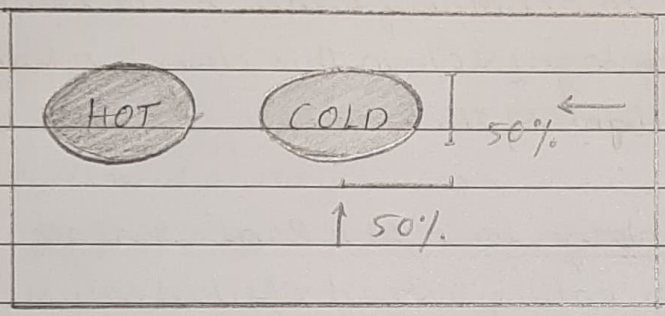
D D M M Y Y Y Y

Page No.

- use multiple values with one padding property
- use separate padding side properties - padding-top, padding-right, padding-bottom, and padding-left,
- use separate margin side properties - margin-top, margin-right, margin-bottom, and margin-left,
- margin and padding values - allows negative values

Example that uses padding and margin properties.

```
.label { padding: 20px;
         margin: 20px;
         display: inline-block; }
```




```
<meta charset = "UTF-8" >
<meta name = "author" content = "John Dean">
<title> hot and cold Labels </title>
<style>
  .hot { background-color: red; }
  .cold { background-color: blue; }
  .label {
        color: white;
        font: bold xx-large Lucida, monospace;
        border: solid black;
        border-radius: 50%;
        padding: 20px;
```

This implements curved corners

```
        margin: 20px;
        display: inline-block;
    }
</style>
</head>
```

← This is necessary for the vertical margins to work

- display: inline-block - element more "blockish", and the browser accomodates by adding space above the ovals.
- borders-radius property - allows to specify how much curvature you want at each of the four corners of an element.
- to specify the focal point for each of the corner's curves
- specified percentage (50% in this case) is relative to the width and height of the element

Using a percentage for a Web Page's Margin

- use percentage values instead of pixel values
- by, browsers leave a small blank area on the four sides of the window - default margin value of around 8 pixels
- body { margin: 10%; }
- margins can be reduced or increased

When to use the different length units

- CSS units that are used for specifying the size, distance, or length of something
- those units are called length units - px, cm, in, etc
- Use em for font-related properties (like font-size)
- Use % for properties that should grow and shrink with the size of the containing element (like margins

D D M M Y Y Y Y

Page No.

for the body element)

- Use absolute units sparingly - only when a fixed size is essential and the target device has a high resolution

2. explain how responsive webpage layout can display differently on different platform

Ans Responsive web design (RWD)

- Users view web pages on different types of platforms - desktops, laptops, tablets and smartphones
- If you want your code to be "responsive" (i.e, you want it be dynamically generate different platforms) you'll need to learn responsive web design
- Responsive images are images that display differently in different environments

Implementation strategies for two of those situations

* Resolution switching technique
* Art direction technique

- Resolution switching

Resolution switching is when you provide a list of images for different versions of the same.

```
<img
srcset = "pigeons - 320w. jpg 320w,
          pigeons - 480w. jpg 480w,
          pigeons - 800w. jpg 800w"
sizes = "(max-width : 340px) 90 vw,
         (max-width : 500px) 90vw,
         800px"
src = "pigeons - 800w. jpg" alt = "feeding pigeons in St. Mark's
                                   Square">
```

The srcset attribute holds the image filenames & their width

The sizes attribute & holds viewport width conditions and associated image-slot widths

VEMANA IT

picture where the images are identical in terms of aspect ratio, where aspect ratio is the ratio of an image's width to height.

- The img element is for a web page that employs resolution switching to display one of three diffuent pictures on a web page.
- The srcset attribute provides a comma-separated list of image filenames with an image width next to each filename.
- The image width values are the pixel values you see when you hover your mouse over the files in file explorer.
- When specifying an image width value, insert a space before the value and append a w after the value.
- the sizes attribute value helps the browser choose the most appropriate file from among the srcset attribute's list of image files. The sizes attribute provides a comma-separated list of values.
- each value has two-parts:-
  * a condition that checks the width of the browsers window's viewport and,
  * the width of the slot in which the image displays
- The viewport is the area below the adress bar where the page content displays.
- imagine, mobile devices have the smallest viewports, whereas laptops and desktop monitors have the largest viewports.
  * (max-width: 340px) 90vw
- The max width: 340px condition means that if the devices viewport is less than or equal to 340 pixels, then the webpage uses 90vw for its image slot width

D D M M Y Y Y Y

Page No.

- The vw value means the image slots width spans — of the viewport's width.
- The sizes attribute's third value has just one part - the width of the slot in which the image displays.

→ Art Direction

- Provide a list of images for different versions of the same picture
- With ~~so~~ resolution switching, the different versions are different sizes, but they have the same aspect ratio.

1. The srcset attribute holds the image filenames and their widths
2. The sizes attributes holds viewport width conditions and associated image-slot widths.

- On the other hand, with art-direction, the different versions can have different aspect ratios because of cropping the original picture in different ~~images~~ ways.
- A desktop computer layout, the browser's viewport would be wider, so having a wide landscape-oriented picture should be OK
- A mobile device layout, the browser's viewport would be narrower, so using that same landscape-version image would probably make the picture's main subject too small.
- To implement responsive images with the art direction, technique.

* wrap the images in a picture container
* The picture container holds a group of source elements
* each source element value has 2 parts
    - a condition that checks the width of the browser window's viewport.

- an image filename.
- The source element's media attribute provides the condition that checks the viewport's width, the source element's srcset attribute provides the image, filename

example:-

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<meta name="author" content="John Dear">
<title>Dominician Republic</title>
</head>

<body>
<h1>Joe Hartman's school in The Dominician Republic</h1>

<picture>
  <source media="(max-width:799px)" srcset="
  ../images/dskids-450w.jpg">
  <source meda="(min-width:800px)" srcset="
  ../images/dsKids-800w.jpg">
  <img src="../images/dsKids-800w.jpg"
  alt="Jordan with kids at recess">
</picture>
<p>
  During this past spring break, Jordan participated
  in a service trip to the Dominician Republic.
</p>
</body>
</html>
```

The media attribute holds a viewport width condition

The srcset attribute holds an image filename

D D M M Y Y Y Y

Page No.

3. Develop an HTML5 table for the following screenshot

Grading Weights for Web Programming I

▶ Help

Homework Assignments			Exams		Ability to stay Awake
1	2	3	1	2	
10%	10%	10%	30%	36%	4%

Ans

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="author" content="John Dean">
<title> grading Weights </title>
<style>
 table, th, td { border: thin solid; }
 th, td {
        text-align: right;
        padding: 20px;
      }
caption { margin-bottom: 15px; }
</style>
</head>
<body>
<table>
  <caption>
  grading Weights for Web programming I
  <details>
```

D D M M Y Y Y Y

Page No.

<summary> help </summary>
The first 3 columns show weights for the 3 homework assignments.
The next 2 columns show weights for the 2 exams
The last column shows the weight for staying awake during class.
</details>
</caption>
<tr>
<th colspan = "3" > homework </th>
<th colspan = "2" > exams </th>
<th rowspan = "2" > ability to </br> stay awake </th>
</tr>
<tr>
<th id = "hw1" headers = "hw" > 1 </th>
<th id = "hw2" headers = "hw" > 2 </th>
<th id = "hw3" headers = "hw" > 3 </th>
<th id = "exams1" headers = "exams" > 1 </th>
<th id = "exams2" headers = "exams" > 2 </th>
</tr>
<tr>
<td headers = "hw hw1" > 10% </td>
<td headers = "hw hw2" > 10% </td>
<td headers = "hw hw3" > 10% </td>
<td headers = "exams exam1" > 30% </td>
<td headers = "exams exam2" > 36% </td>
<td headers = "awake" > 4% </td>
</tr>
</table>
</body>
</html>

D D M M Y Y Y Y

Page No.

4. explain popular controls and the elements of forms used in javascript

Ans. Accessing a form's control values.

- Whenever you pass an argument to a function, you should have an associated parameter in the function's heading. Therefor, to receive the form object passed to the generate Email function, there's a form parameter in the function's heading
  - * function generateEmail (form)
- The form parameter to retrive the text control user inputs.
  - * form.elements['first'].value

eg:-

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="author" content="John Doan">
<title>email address generator</title>
<script>
    // This function generates an email address

    function generateEmail (form) {
        document.getElementById('email').innerHTML =
            form.elements["first ~~name~~"].value + "." +
            form.elements["last"].value + "@park.edu";
    form.reset();
    form.elements["first"].focus();
    } //end generateEmail
</script>
</head>
</html>
```

Parameter that holds an form object

D D M M Y Y Y Y

Page No.

- To access the controls that are within a form, we use the form object's elements property
- The elements property holds a collection of controls, where a collection is a group of items that are the same type
- To access a control within the elements collection, you put quotes around the control's id value and surround the quoted value with [ ]'s
- To get the user's input, you need more than just the control by itself; you need to access the control's value property
- alternative
  * form["first"]

Javascript object properties and HTML element attributes.

- The value property returns the text control's, user-entered value. If there's no user entry, then the value of the text control's value attribute is returned
- There's a parallel world between javascript properties and HTML element attributes.
- Text control element attributes: type, placeholder, size and maxlength, value, autofocus, disabled, readonly.
- Javascript properties for a text control element object: type, placeholder, size, maxLength, value, autofocus, disabled, readOnly,

Control elements innerHTML property

- The innerHTML property accesses the content within the control element's code, not including its start and end tags.

D D M M Y Y Y Y

Page No.

- <p id="email'></p>
- document.getElementById("email").innerHTML =
  forms.elements["fisst"].value + "." +
  form.elements["last"].value + "@park.edu";
- + is string concatenation opuation
- To connect a string to something else (eg, another string, a number), you need to use the concatenation operator, +. The resulting connected value forms a string.
- forms.elements to retrieve the two text controls. As an alternative, we would have used document.getElementById to retrieve the controls (e.g., document.getElementById["fisst"])

5. Develop email address generator webpage using forms controls.

Ans

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="author" content="John Doan">
<title>Email address generator</title>

<script>
  // This function generates an email address.
function generateEmail(form)
{
  document.getElementById("email").innerHTML =
  form.elements["first"].value + "." +
  form.elements["last"].value + "@park.edu";
  form.reset();
  // form.elements["fisst"]oadoa focus();
```

```
}
//end generateEmail
</script>
</head>
<body>
<h3>
enter your first and last names and then click the
button.
</h3>
<form>
 firstName:
<input type = "text" id = "first" size = "15">
<br>
Last Name:
<input type = "text" id = "last" size = "15">
<br><br>
<input type="button" value="generateEmail" onclick =
                "generateEmail(this.form);">
<p id = "email"></p>
</body>
</form>
</html>
```