

Module 2: CSS [30 Hours]

Contents to be covered:

- Introduction CSS
- Applying CSS to HTML
- Selectors, Properties and Values
- CSS Colors and Backgrounds
- CSS Box Model
- CSS Margins, Padding, and Borders
- CSS Text and Font Properties

Assessment Strategy:

1. MCQ based on concepts.
2. Assignments and Online Test.
3. Mini Project after module completion.
4. GitHub Submission of Assignments and Project for assessment and profile building.
5. Sample Interview Questions.

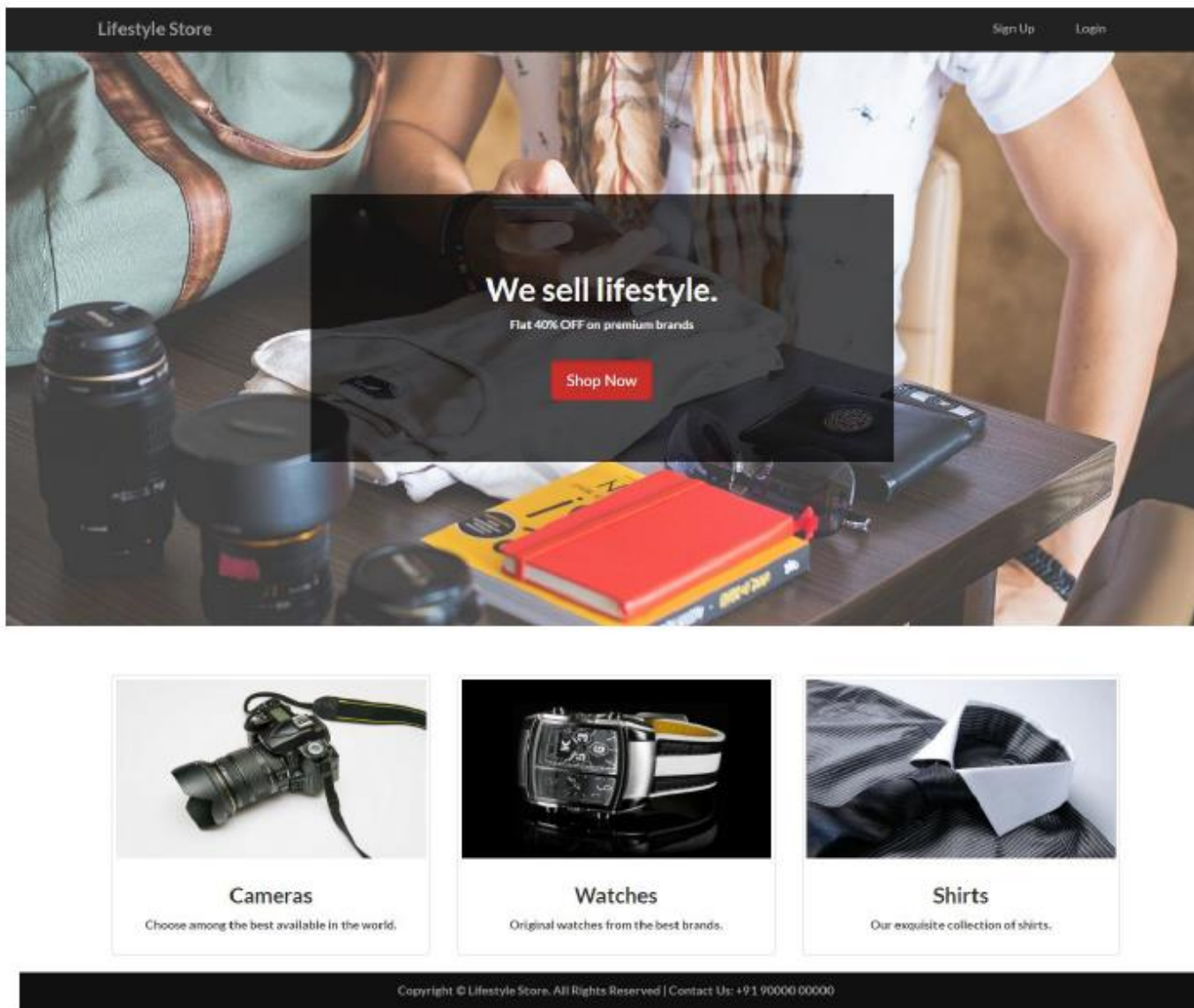
Assignments and Projects:

Problem Statement

Hey! Now that you're familiar with HTML & CSS, you will create an index page for the e-commerce website.

Description: The index page is usually the first page that a user sees when he/she visits a website.

This is how our index page should look like:



This is a web page contains three sections:

1. Header
2. Content
3. Footer

Header: This is the first section of the webpage and it contains the “The Lifestyle Store” as a website name and navigation links to login and signup page.

Content: This section has two parts banner image with a Shop Now button over it and item category

listing section.

Footer: The footer contains text “Copyright @ Lifestyle Store. All Rights Reserved | Contact Us: +91 90000 00000”.

Approach:

Here is a starter template:

```
Index.html
<!DOCTYPE html>
<head>
<!-- The page has a title Lifestyle Store-->
<title>Lifestyle Store</title>
<!-- External css file index.css placed in the folder css is linked-->
<link href="css/index.css" rel="stylesheet" type="text/css"/>
</head>
<body>
</body>
</html>
```

In the above template, the page title is mentioned as “Lifestyle Store” and an external CSS file index.css is linked.

The basic CSS in index.php would be

```
body{
width: 100%;
height: 100%;
margin: 0;
}
a{
text-decoration: none;
background-color: transparent;
color:#ededed;
}
```

1. Inside the <body> create a div (using <div> tag) with class header assigned to it.

2. Class header would contain the following CSS.

```
background-color:#000;
color:#fff;
border-color: #080808;
min-height: 50px;
border: 1px solid transparent;
```

3. Inside the div with class header, create a div with class inner-header which will free some space on the sides of the header.

4. Class inner-header would contain the following CSS.

```
width:80%;
```

margin:auto;

5. Div with class inner-header will contain three div elements.

6. First div element will have class logo with content “Lifestyle Store” which will be a link.

7. Class logo will contain the following CSS.

```
float: left;
height: 50px;
padding: 15px;
font-size: 20px;
font-weight: bold;
```

8. Second div element will have content Sign up and third div element will have content Login with same class header-link to both div elements. Both Signup and login will be links.

9. Class header-link will contain the following CSS.

```
float:right;
font-size:14px;
height: 50px;
padding: 15px 15px;
font-size:16px;
font-weight: bold;
```

10. This marks the end to inner-header div element and header div element. Now, you have made the header of the web page.

11. Start div tag with class content.

12. Class content will have the following CSS.

min-height: 600px;

13. Div element with class content will contain div element with class banner-image.

14. Class banner-image will have the following CSS.

```
padding-bottom: 50px;
margin-bottom: 20px;
text-align: center;
color: #f8f8f8;
background: url(image_name) no-repeat center;
background-size: cover;
```

15. Div element with class banner-image will have one div element with class inner-bannerimage.

16. Class inner-banner-image will have the following CSS.

```
padding-top: 12%;  
width:80%;  
margin:auto;
```

17. Start a div element inside the div element with class inner-banner-image with class banner_content.

18. Class banner_content will have the following CSS.

```
position: relative;  
padding-top: 6%;  
padding-bottom: 6%;  
overflow:hidden;  
margin-bottom: 12%;  
background-color: rgba(0, 0, 0, 0.7);  
max-width: 660px;
```

19. Add H1 tag inside this div element with content “We sell lifestyle.”.

20. Add p tag with content “Flat 40% OFF on premium brands”.

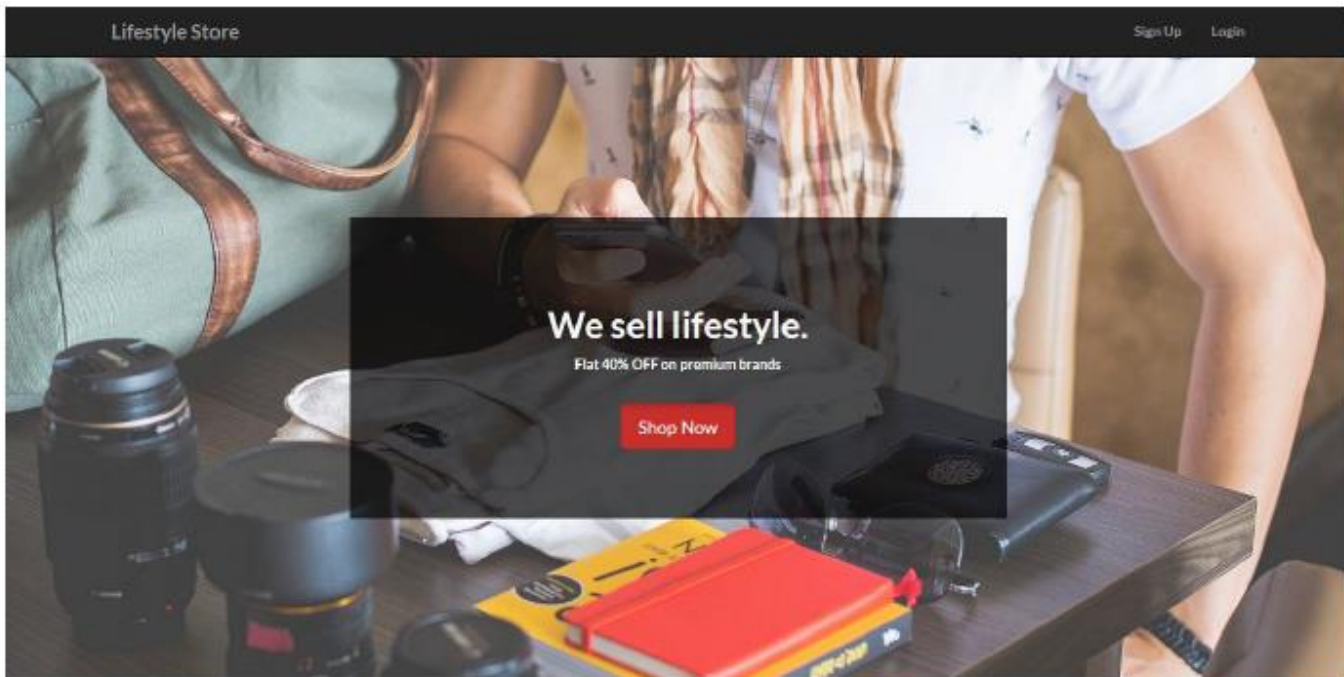
21. Add a link after p tag. Link will contain class button.

22. Class button will have the following CSS.

```
color: #fff;  
background-color: #c9302c;  
border-color: #ac2925;  
box-shadow: inset 0 3px 5px rgba(0, 0, 0, .125);  
padding: 10px 16px;  
font-size: 18px;  
border-radius: 6px;
```

23. Close all the div elements till the div element with class banner_image. Now, you have made the banner.

24. After writing the above HTML and CSS code, your page should look like this:



25. Start a div element with class container. This class will enclose the images and their div elements.

26. Class container will contain the following CSS.

```
width:90%;  
margin:auto;  
overflow:hidden;
```

27. Start a div element inside the container div element with class items.

28. Class items will have the following CSS.

```
width:30%;  
display: block;  
padding: 4px;  
margin-bottom: 20px;  
line-height: 1.42857143;  
background-color: #fff;  
border: 1px solid #ddd;  
border-radius: 4px;  
float:left;  
margin-left:1%;
```

29. Start the anchor tag. All the next HTML elements will be inside the anchor tag.

30. Start the image tag. Class thumbnail is assigned to img tag.

31. Class thumbnail will contain the following CSS.

```
display: block;  
max-width: 100%;  
height: auto;
```

32. Start a div element after image tag with class caption.

33. Class caption will contain the following CSS.

```
color:#000;  
padding:0px 10px 10px;  
font-weight: bold;  
text-align: center;
```

34. Add H2 tag with content “Watches” and paragraph tag with content “Original watches from the best brands.”.

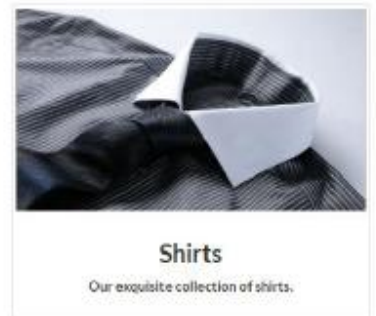
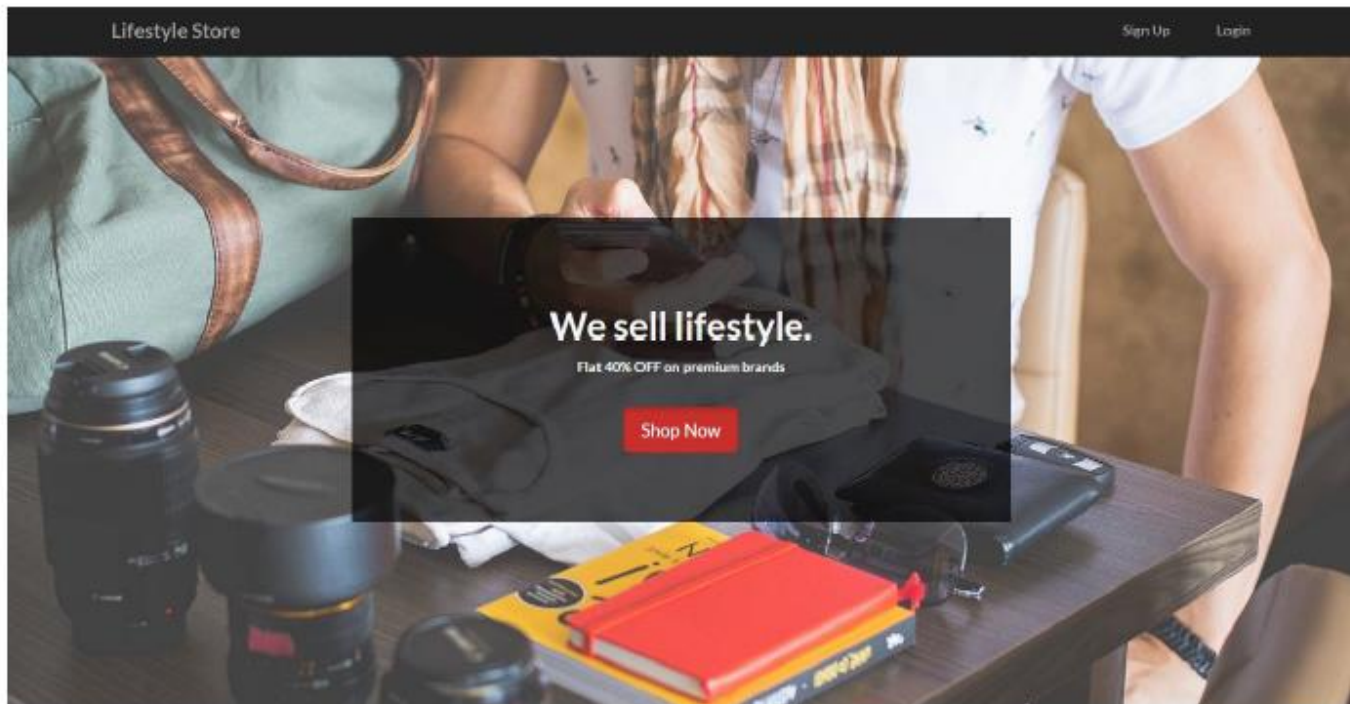
35. Close the caption div element, anchor tag and items tag. The HTML code will be:

```
<div class="items">  
  <a href="#" >  
      
    <div class="caption">  
      <h2>Cameras</h2>  
      <p>Choose among the best available in the world.</p>  
    </div>  
  </a>  
</div>
```

36. Replicate this div element with class items and HTML content inside it two times. IT will create three div elements. Change the content and paragraph for other two elements.

37. Close the container div element and content div element.

38. After writing the above HTML and CSS code, your page should look like this:



39. Start the footer tag.

40. Footer element will have the following CSS.

```
background-color: #000;  
color: #fff;  
font-size: 14px;
```

41. Start a div element with the class container. This class is already used before.

42. Add a paragraph tag with content in center. Content is “Copyright © Lifestyle Store.
All Rights Reserved | Contact Us: +91 90000 00000”.

43. Close the container div element, footer tag.

44. Close the body tag and HTML tag.

45. Hence, your web page is ready.

Mini Project:

Mini Project 1 and 2 from Module 1, should be extended by applying CSS features. Submit the updated projects in GitHub Repository for assessment.

Mini Project:

1. Change the layout of a form

For this project, you have to take up a website that includes forms (inquiry/survey/registration form) and examine whether the form was recently created. Generally, web forms that have been created a while back tend to have container DIVs and float-based layouts that don't bode well on small screens (mobile devices). Also, such forms may contain unnecessary JavaScript elements as well.

The best tool for this project is CSS Grid. It will enable you to remove all the unnecessary markup. You can create bulletproof responsive layouts without relying on media queries.

2. Improve the speed of a website

If your website isn't fast, you will lose visitors. A webpage requires an average 2MB download that takes 20 seconds to load on a mobile screen. With CSS, you can create seven files, each of 65KB. This can make a huge difference in your page loading speed.

Scan an existing website and identify the opportunities for optimization. It could be image replacement/elimination and changing fonts and JavaScript effects. As you make these changes with CSS, the site's weight will be optimized, enhancing its performance.

Sample Interview Questions:

1. What do you understand by the universal selector?
2. What are the elements of the CSS Box Model?
3. How can CSS be integrated into an HTML page?
4. What was the purpose of developing CSS?
5. What are the benefits of CSS Sprites?
6. Name media types allowed by CSS.
7. Explain responsive web design.

8. When should you use translate () instead of absolute positioning?
9. What are mixins?
10. Differentiate between the use of ID selector and class selector.
