ASSIGNMENT NO. 1

Case study :-

Before coding of the Website, planning is important. Students should visit different Websites for different client projects & note down the evaluation results for the Websites.

Objective :-

Understand the features required to evaluate a Website.

Theory :-

For evaluation of Website following points can be considered.

1). **Usability :-**

> One of the most important aspect of Web design is actually making the site usable for the average user.

2). **Speed :-**

> Website speed can make or break your entire company.
> Despite the insane technology involved with connecting people to various pages all around the world.

3). Aesthetics :-
> The look & feel of your page is just as important as the functionality.
> Don't overdo it with color schemes & crazy images like 1990's, just have simple, sharp & decent looking Webpages.

4). Content :-
> Users are very picky.
> Even if your Webpage is easily accessible functions well, works quickly, & looks great, they still won't be pleased unless you have compelling & engaging content on your site.

5). Contact info :-
> Nothing makes customers angrier than not being able to contact a business when they want to.

6). Website Maintenance :-
> Every page on your site should work 100% of the time.

7). Mobile friendly :-
> Just like your company should be online, your Webpage should function on mobile devices.

Conclusion :- We have learned the points required to evaluate any Website.

ASSIGNMENT NO. 2

Title :- HTML, CSS.

Objectives :-

1. Understand about basic concept of html.
2. Understand the basic concept of css.

Problem statement :-

> Implement a Web page index.htm for any client Website.

Outcome :-

Students will be able to,

1. Design static Webpage using html.
2. Apply css to HTML pages.

Theory concepts :-

HTML :-

> HTML is the standard markup language for creating Web pages.
> HTML stands for HyperText MarkUp Language.
> It describes the structure of Web pages.
> Its elements are building blocks of pages.
> HTML elements are represented by tags.
> HTML tags label pieces of content such as 'heading', paragraph, table & so on.

HTML Versions :-

HTML	1991
HTML 2.0	1995
HTML 3.2	1997
HTML 4.01	1999
HTML	2000
HTML 5	2014.

CSS :-

> CSS stands for cascading style sheets.
> It is nothing, but design language intended to process of making it presentable
> CSS handles the feel & look part of web page, by using css, one can control the layout design.

Advantages of CSS :-

> Saves time
> Easy maintenance.
> Multiple device compatibility.
> Global web standards.
> Pages load faster.
> Offline browsing.
> Platform independence

CSS3 Modules :-

> Box Model
> Background
> Text effects
> 2D/3D Transformation.
> Trng vals & replaced content.
> selector
> Border
> Animations.
> User Interface.
> Multiple col Layout.

Technology / Tools :-

> The <!DOCTYPE html> declaration defines this document to be HTML5.
> The <html> element is the root element.
> <head> contains meta info. about the document.
> <title> specifies a title for the document.
> <body> contains the visible page content.
> <h1> defines a large heading.
> <p> defines a paragraph.
> HTML tags & elements are surrounded by angle brackets.

3 Ways to add CSS in HTML.

(1). Inline :-

> By using the style attribute in html elements
> It is used to apply a unique style to a single html doc.
> eg :-

```
<h1 style = "color:blue;" >
This is blue heading </h1>
```

(2). Internal :-

> By using <style> element in the <head> section.
> An internal css is used to define a style for a single html page.
> eg -

```
<style>
    body { background-color : yellow; }
    h1 { color : blue; }
</style>
```

③. External :-
> By using external css file.
> An external style sheet is used to define the style for many HTML pages.
> eg- <link rel = "stylesheet"
href = "style.css">

Design / Execution steps :-
1) Write html code in notepad & save with .html
2) Write the css code in notepad save with .css
3) Import css file in html page.
4) Open html page in the browser.

Conclusion :-
Hence, We have designed static Web pages using html & css.

ASSIGNMENT NO. 3.

Title :- XML, DTD & CSS/XSL.

Problem Statement :-
Design the XML document to store the information of the employees of any business organization & demonstrate the use of :-
(a) DTD
(b) XML Schema.
And display the content in by using CSS/XSL.

Outcomes :-
Students will be able to :-
> Design static Webpage using XML.
> Apply DTD to XML pages.
> Apply CSS/XSLT to XML pages.

Theory :-
> XML stands for extensible Markup Langua
> It is nothing but the text-based markup language. Which is derived from standard Generalized Markup Language.
> XML is not going to replace html in the near feature, but it introduces new possibilities by adopting many successful features of HTML.

There are 3 important characteristics of XML
1. XML is extensible.
2. XML carries the data, does not present it.
3. XML is public standard.

Technology / Tool :-
> The XML doc have an XML declaration, but it is optional, & it is written as-
<?XML version = "1.0" encoding = "UTF-8"?>
Where version is nothing but the version of an XML document & UTF specifies the character encoding used in the document.

XML attributes :-
An XML element can have one or more attributes-
eg-

XML Tutorial
> Here href is the attribute name & http://www.google.com/ is attribute value.

DTD :-
> A DTD is a Document Type Defination.
> A DTD defines the structure & the legal elements & attributes of an XML document.
> An application can use a DTD to verify that XML data is valid.

An internal DTD declaration :-

```
<? xml version = "1.0" ?>
<!DOCTYPE note [
<!ELEMENT note (to, from, heading, body)>
<!ELEMENT to (#PCDATA)
<!element from (#PCDATA)
<!element heading (#PCDATA)
]>
<note>
<to> neha </to>
<from> amit </from>
<heading> Reminder </heading>
</note>
```

XSLT :-

- XSL - eXtensible Stylesheet Language) is a styling language for XML.
- XSLT stands for XSL Transformation.
- Note :- <XSL: stylesheet & <XSL: transform> are completely synonymous & either can be used.

Design / Execution steps :-

1. Write the XML code in notepad & ~~execute~~. save with .xml extension.
2. Write the DTD in notepad & save it with .dtd extension.
3. Write XSLT code in notepad & save with .xsl.
4. Open XML page in browser.

Test Cases :-

> Manual testing is used to check whether XSLT gets applied or not.
> Eclipse validates function used to check whether DTD gets applied or not.

Conclusion :-

Hence, we have designed static web pages using XML, XSLT / CSS & DTD.