HTML is used to Structure content on webpage.

CSS to style html content.

<!DOCTYPE html> defines to the browser the file is a html file.

<head> </head> this is the extra information- meta information like title. It doesnt get rendered to the page.

<body> </body> this is the stuff that get rendered to the browser

<strong> Makes this text bold </strong>

<em> this text is in italics </em>

<small> make this word a little small </small>

6 different type of heading tags. <h1> to <h6>

<ul>

<li>

</ul>

<ul> tags are wrappers for <li> tag. each <li>tag is a list item. <ul> is an unordered list. They are just bullet numbers.

<ol> </ol> is ordered list which gives numbers.

<div></div> stands for divisions. divide content into sections or groups. Will be helpful for css.

<span> </span> are like <p> tags but they are used for adding js hooks.

<hr> is a line which separates content. No closing tag for this just like <br>

An attribute is inside the tag which gives more information about the tag to the browser.

<img src= ‘path to image to use’ alt= ‘a description of picture for browsers which dont load images’>

<a href= ‘link to a website’>The text which is hyperlinked</a>

img tag is a single tag.

<blockquote cite= ‘link to the site from which we sited the text’>

we are all in the gutter but some of us look at the stars

</blockquote>

The above tag is used for citing the source of our text. However it wont do anything but indent the text a little to the right.

<p style= “color:organge;”>this text will be in orange </p>

HTML 5 SEMANTIC TAGS:

When we say <p> This is a paragraph </p> browser doesnt really know what is inside those tags. To make things more meaningful, html5 introduced some semantic tags which describe the data within.

<main>

For the main content of that webpage. The content which is only in that webpage.

<section>

Certain section of the blog

<article>

<aside>

Defines content related to something else Ex:similar blogs

<header>

For header of the page. Things which are common for all the webpages. Navigation bar, title, logo etc.

<footer>

**JAVA SCRIPT DOM**

With the help of DOM we can

1) change/remove html elements in the DOM

2) change/add CSS styles

3) Read and Change element attributes.

Use one.html file

We add javascript source file to an html page with the tags:

<script src= “something.js”> </script>

document object is provided by the browser.

document.getElementById(‘id’)

id’s should be unique. If they arent then the browser/js will give unexpected results.

class isnt unique. you can name different sections as same class.

document.getElementByClassName(‘class\_name’); will be used.



html5 attributes in the tags must be written in all lower case. Some browsers correct the mistakes but some dont. They may lead to errors.

2 types of html elements: (Refer four.html)

1)Inline elements: line up next to each other. img, a, span, em etc are inline tags. Dont take more space than the content requires.

2) Block elements: takes up whole width of the page. starts in new line. p,div, h1,h2,h3,ul,li etc..

We can nest inline elements inside block level elements. We cant do the opposite Ex: we cant nest p tag in a tag.

**CSS**

A stylesheet consist of set of css rules.

Selectors are the elements we want to target on the web page.Tags, classes, attributes etc are selectors. all of them can be selectors.

Declarations are key value pairs.

The following is a rule or rule set:

selector{

declaration1;

declaration2;

}

color : red;

margin: 20px;

font-weight: bold;

Adding css:

1) Add directly with style tag

We add the style tag in the head section.

<style>

</style>

Use two.html

2) External style sheet

Use three.html and styles.css files. styles.css has different style rules. Refer this.

Use the link tag.

<link rel="stylesheet" href="styles.css">

Using web safe font is important because the user might not have the font which you have selected.

For colors we can use hexcodes instead of colors.

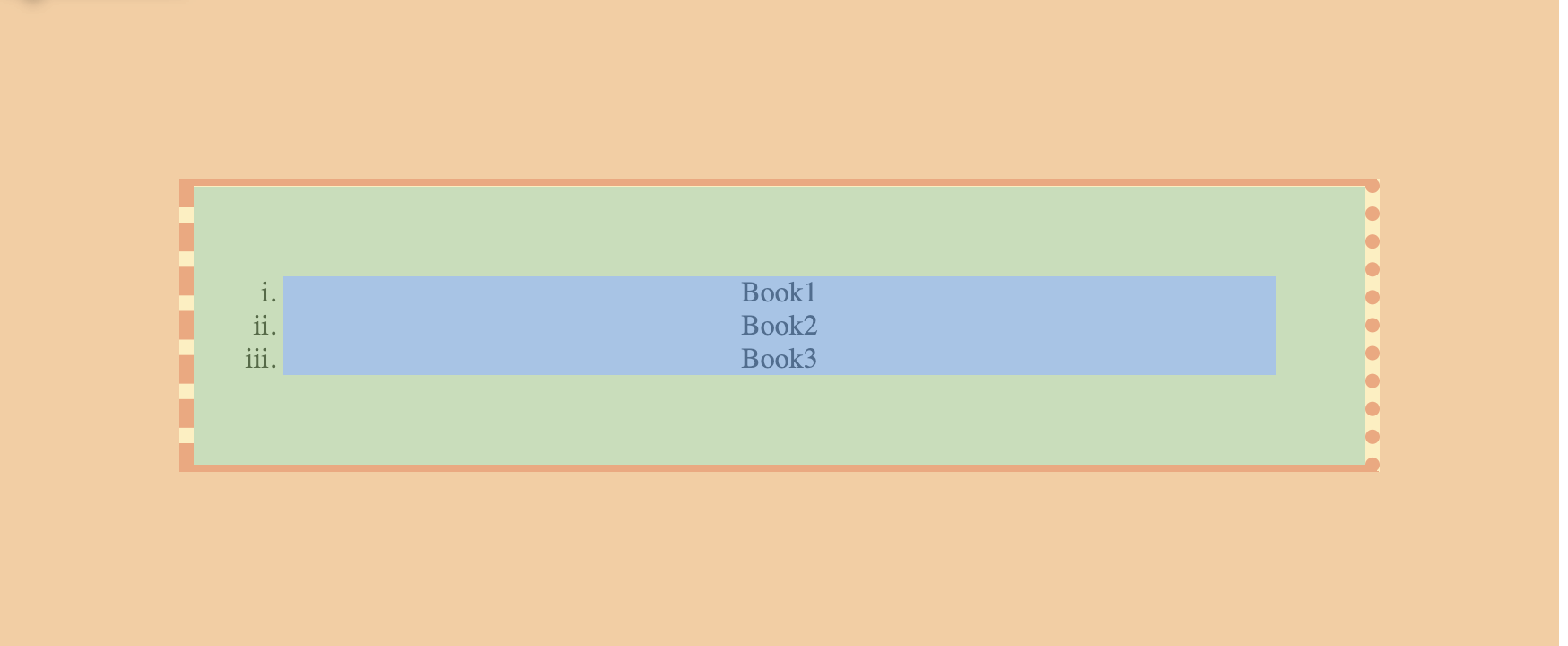
Also we can convert an inline element into block level element. Refer four.html with styles.css.

Block level elements can have margin on all 4 sides.

But for inline level elements margin can be only on left and right. only padding can be on four sides.

/\*margin is outside of the element border\*/

/\*padding is for internal space\*/



Orange: margin

Green:padding

Dark orange: border

**CLASSES**

Use five.html and styles2.css

In css say we dont want to target all the <p> tags?

We use class attribute in html.

Same class to many elements of the page.

Different types of tags can have the same class. However, if you want to style only one type of tag under a class we do it the following way:

tag\_type.class\_name{

declaration1;

key:value;

}

For example:

Say in a html <div> and <ul> both tags have same class named “error”. But I want to style only div tags with that class, we use the following syntax:

div.error{

}

What if there are many divs but we want to style only some divs? then we give two classes to those div tags:

<div class= “success feedback” id= “div1”>

This div section belongs to two classes. Two classes are separated with a space. Two classes are success and feedback. This will have color crimson.

</div>

<div class= “success” id= “div2”>

This div section belongs only to one class. This will have color green.

</div>

In styles.css

div.success.feedback{ *\*this will style the first div with id div1\**

color: crimson;

background-color: black;

}

div.success{

*\*this will style div with id div2\**

color:green;

}

Even IDs can be used for CSS hooks.

All IDs in a particular page must be unique. Only one tag must have a particular id.

**DESCENDENT SELECTORS**

Use six.html and styles3.css. Refer these files for detailed notes.

So till now we used

1) class

2) id

3) descendent selectors

to hook html elements in css. IDs lets use mostly in javascript applications.

Now lets see how to use attributes as selectors.

**ATTRIBUTE SELECTORS**

Use seven.html and styles4.css

tag\_name[attribute\_name]{

declaration1;

declaration2;

}

The above rule will style a particular tag having an attribute

Ex:

a[href]{}

We can also select a tag with a particular attribute with a particular value.

a[href= “google.co.in”]{

}

So till now we used the following ways to seleect an html attribute to style with css

1) class

2) id

3) descendent selectors

4) attributes

**CASCADE**

HTML elements can inherit css rules that apply to their parent elements. For example the style of div will be inherited by <p> the child and <span> the grand-child

<div>

<p>adafdf<span>lol</span></p>

<div>

Font, text styles, colors are inherited. Background colors, margins etc arent.

Didnt listen to the class. Left it here.

**PSEUDO CLASSES**

For styling elements when they are in a particular state Ex: hovering over an element, when element is first child of parent element. We use pseudo classes by tacking them to our selectors.

use eight.html and styles5.css

There are different pseudo classes. hover, focus

Stopped the video in the middle.

**FORMS**

Input type=text,password,email etc.

textarea

radio

select

When we click the submit buttion, the browser takes the value inside the action attribute of the form that will be a reference to a backend script. It will send all the data to the user script.