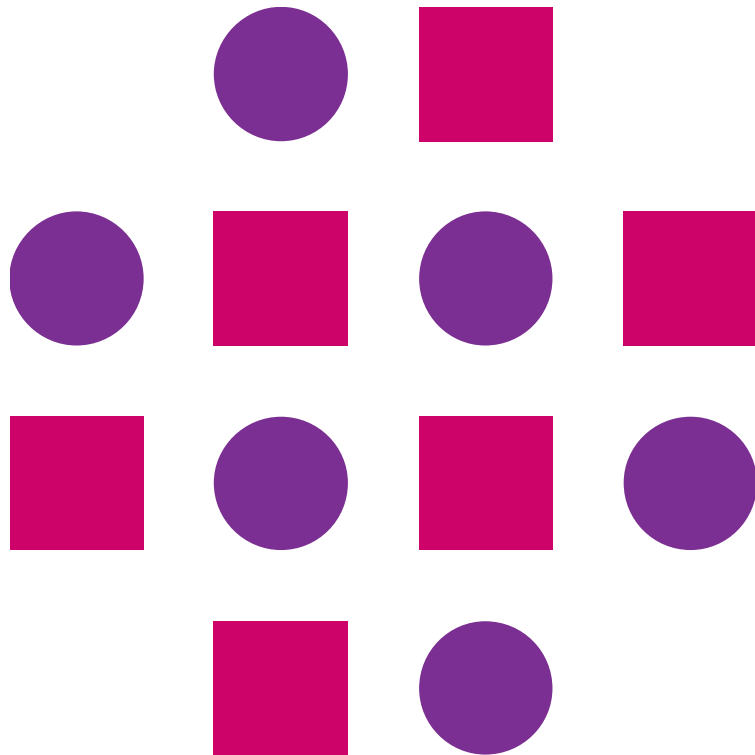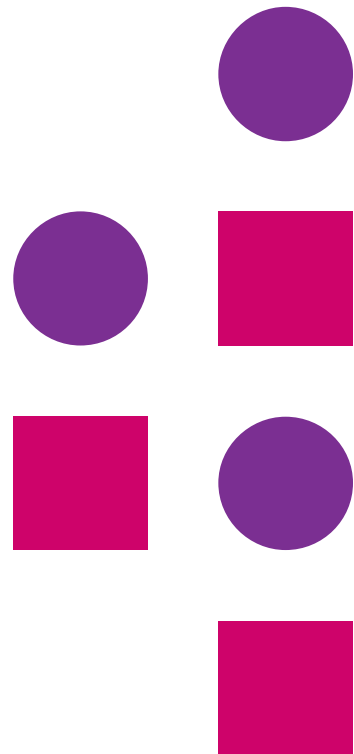# Block, class, layout

Hypertext Markup Language 5 (HTML5)

Table of Content

- Block and inline elements

- Class

- Layout

Block and inline

(Hypertext Markup Language 5)

CSS Display property

- Every element on a web page is a rectangular box

- Every HTML element has a default display value depending on its type

- CSS display property determines how that rectangular box behaves

- The default display value for most elements is block or inline

- **Values** : none | inline | block | inline-block

Syntax :
Style = "display : value"

CSS Display property

Value	Description
inline	Display element as inline container; height and width properties have no effect
block	Display element as block container; height and width properties are effective
inline-block	Displays element as an inline-level block container. The element itself is formatted as an inline element, but height and width properties are effective
none	The element is completely removed

Block elements

- Elements with display property set to "block" are called block elements

- A block element always starts on a new line and takes up the whole width

- Width and height properties are effective on block elements

- Block level elements create larger structures (than inline elements)

Block elements
(Example)

- Division element <div>

- Heading elements <h1>....<h6>

- Paragraph element <p>

- Form element <form>

Div Tag

- The DIV tag is is designed to allow you to define "divisions" of a page (or logical containers)

- The HTML <div> tag is just like a container unit which is used to encapsulate other page elements and divides the HTML documents into sections (block-level)

Div Tag
(Example)

<div> Hello World!!</div>

<p> Welcome to Webstack Academy </p>

<p> The Web Devlopment Institute</p>

<div> M. G. Road</div>

The output

Hello World!!

Welcome to Webstack Academy

The Web Devlopment Institute

M. G. Road

Inline Elements

- Elements with display property set to "inline" are called inline elements

- An inline element does not start on a new line and takes up as much width as necessary

- Width and height properties have no effect on inline elements

- Inline elements create smaller structures (than block level elements)

- The inline elements are treated as a part of the flow of the document and content does not get affected

Inline Elements
(Example)

- Span element `<span>`

- Anchor element `<a>`

- Formatting elements : `<b>`, `<i>`, `<u>`, `<sub>`, `<sup>`, `<em>`, `<strong>` etc

- Line break `<br>`

- Quote `<q>`

HTML span Tag

- Defines a section in a document (inline)

- It can be placed inside a paragraph to define a part of it without affecting the appearance of the text

- However, styling can be applied on the section created by tag

Example :
<p> Welcome to Web Stack Academy </p>

Inline-block elements

- Elements with display property set to "inline-block" are called inline-block elements

- Such elements behave as inline container on which height and width properties can be applied

- Example :

  - Image element <img>

Class

(Hypertext Markup Language 5)

The class Attribute

- Class global attribute is a space-separated "list of classes"

- It specifies one or more class names for an element

- Classes allows CSS and Javascript to select and access specific elements via the class selectors or functions (like the DOM method document.getElementsByClassName)

- A class attribute can be used on one or more HTML element(s)

The class Attribute

- Class name is case sensitive

- Though, specification doesn't put requirements on the name of classes, it is recommended to use names that describe the semantic purpose of the element, rather than the presentation of the element

- Semantic names remain logical even if the presentation of the page changes

The class Attribute

- Coding recommendations for class

  - Use small case letters for class names

  - No camelCase or under_score names

CSS Class and ID selectors

- A class selector is a name preceded by a full stop (".")

- An ID selector is a name preceded by a hash character ("#")

Syntax (class selector) :
.class-name {
   property: value;
}
Syntax (ID selector) :
#id-name {
   property: value;
}

Class selector
(Declaration)

```
<style>
.states-theme {
    background-color: pink;
    color: brown;
    margin: 20px 0 20px 0;
    padding: 20px;
}
</style>
```

Class selector
(Element specific)

```
<style>

div.states-theme {

    background-color: pink;

    color: brown;

    margin: 20px 0 20px 0;

    padding: 20px;

}

</style>
```

Applying class selector

- Example

```
<div class="states-theme">

<h2>Karnataka</h2>

<p>Karnataka  is a state in south western region of India.The capital and

largest city is Bangalore (Bengaluru). Karnataka is bordered by the Arabian

Sea.</p>

</div>
```

Applying class selectors

- Example – using two class selectors

```
<p class = "brown-text pink-bg"> Karnataka  is a state in south western region of India. The capital and largest city is Bangalore (Bengaluru). Karnataka is bordered by the Arabian Sea. </p>
```

ID vs class

- An ID can be used to identify one element
- A class can be used to identify more than one elements
- Recommendation – use class selector

Layout

(Hypertext Markup Language 5)

HTML5 Layout Elements

Tag	Description
<header>	Defines a header for a document or a section
<nav>	Defines a container for navigation links
<section>	Defines a section in a document
<article>	Defines an independent self-contained article
<aside>	• Inside article - defines side content related to article • Outside article – defines side content related to entire page
<footer>	Defines a footer for a document or a section
<details>	Defines additional details
<summary>	Defines a heading for the <details>

HTML5 Layout Elements
<header>

- Header element represents a container for introductory content

- Footer typically contains

    – One or more heading elements (<h1> - <h6>)

    – Logo or icon

    – Authorship information

- A <header> tag cannot be placed within a <footer>, <address> or another <header> element

HTML5 Layout Elements

<footer>

- Defines a footer for a document or section

- Footer typically contains
  - Authorship information
  - Copyright information
  - Contact information
  - Sitemap
  - Back to top links
  - Related documents

HTML5 Layout Elements

<nav>

- Defines a set of navigation links

- The <nav> element is intended only for major block of navigation links

HTML5 Layout Elements

<aside>

- The aside element is a block level element

- Aside element can be used within or outside article element

- When used inside article element - content of the aside element must be relevant to the article content

- When used outside article element - content of aside element must be relevant to entire web page

- Block level and inline elements can be used inside aside element

HTML5 Layout Elements

<section>

- Defines sections in a document, such as chapters, headers, footers, or any other sections of the document

Layout Techniques

- HTML Tables (not recommended to use as layout)

- HTML Float

- CSS Framework

- CSS FlexBox

Layout Techniques

- **HTML Tables** - The purpose of HTML tables is to display data in tabular form

- **CSS Framework** - To create the layout fast we can use css framework like bootstrap, W3.CSS

- **Float –** An element can be floated left or right by using the property float

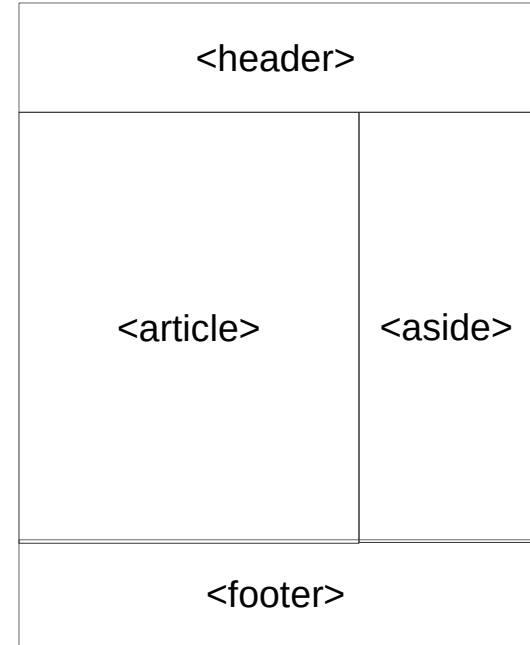- **FlexBox** – The Flexbox is used to redefine layouts in CSS

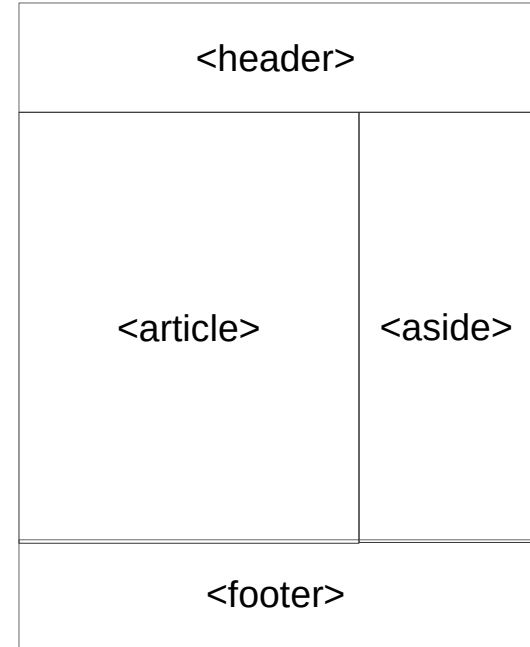# Page Layout

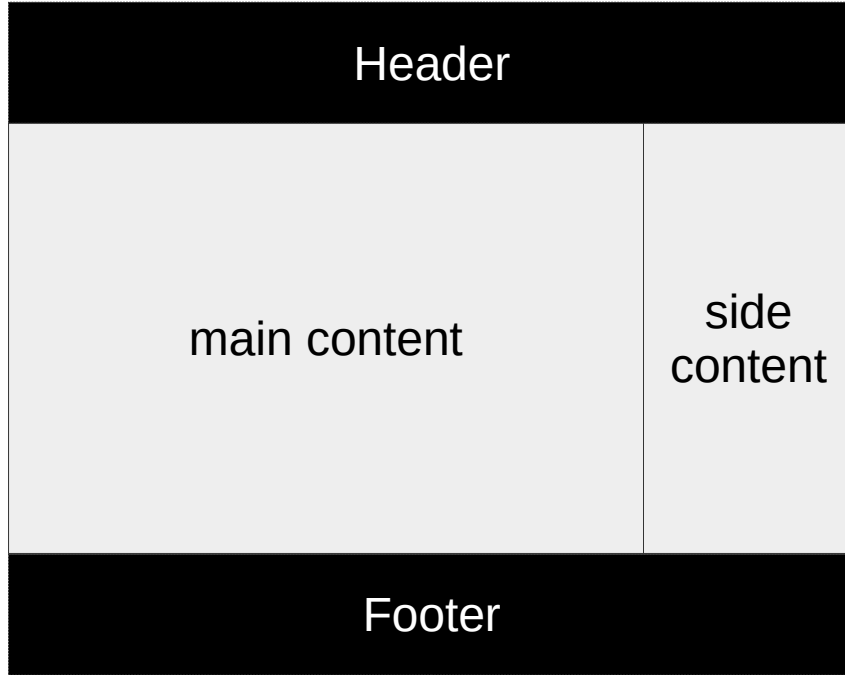Introduction to HTML5
April 27' 2018

A HTML5 is gaining popularity, there is huge demand of HTML5 developers across globe. India is leading software destination for development and maintenance software products.
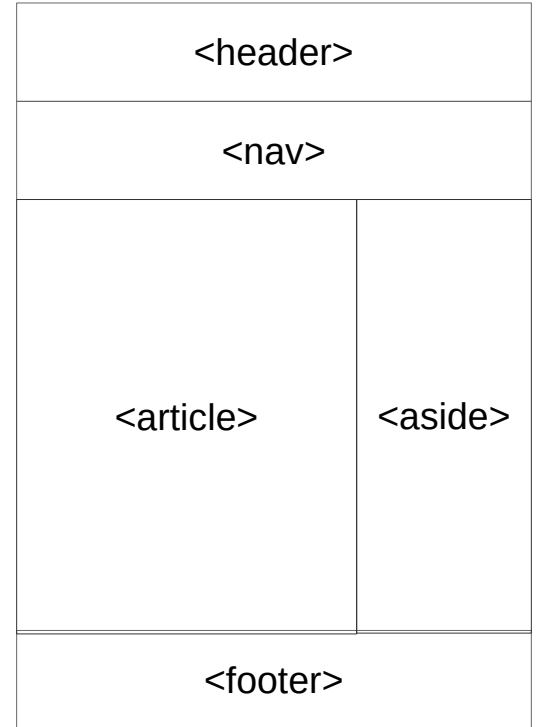
side Content can be placed here for HTML5
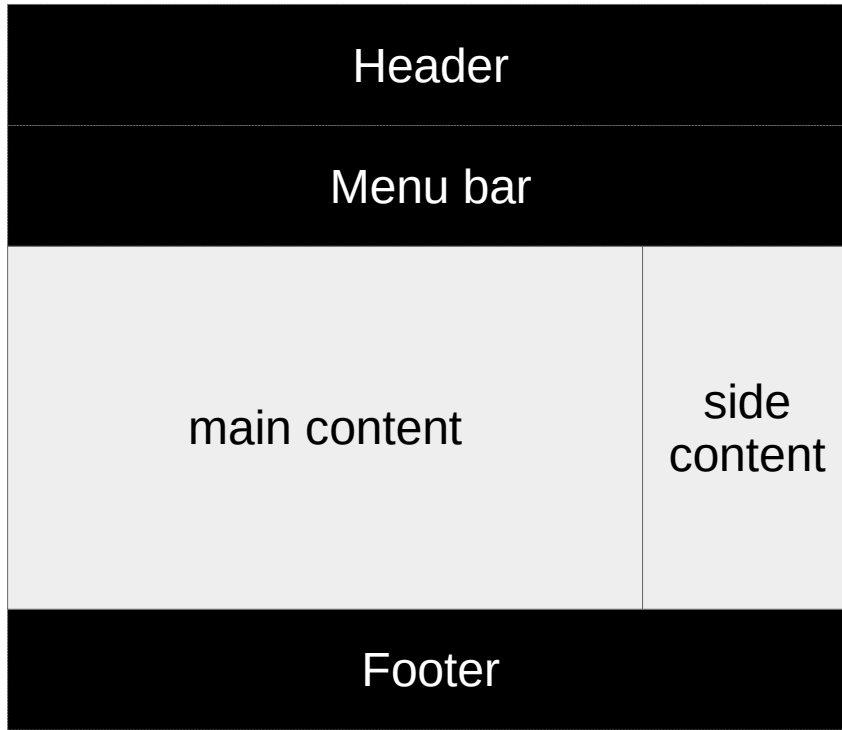
Copyright Webstack Academy

<header>

<article>

<aside>

<footer>

Page Layout - 1

Header	
main content	side content
Footer	

<header>	
<article>	<aside>
<footer>	

Page Layout - 2

Header		
Menu bar		
main content		side content
Footer		

<header>	
<nav>	
<article>	<aside>
<footer>	

WSA | Forward looking IT finishing school

Page Layout - 3

Header
Menu bar

section	side content
main content	

Footer

<header>
<nav>

<section>	<aside>
<article>	

<footer>
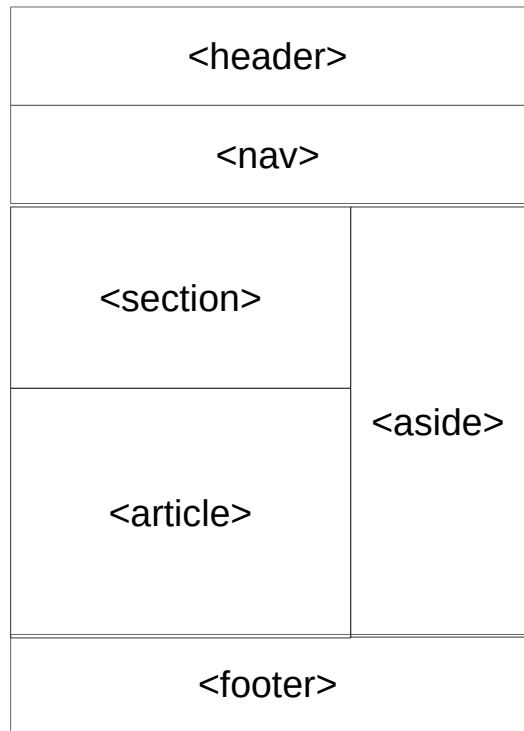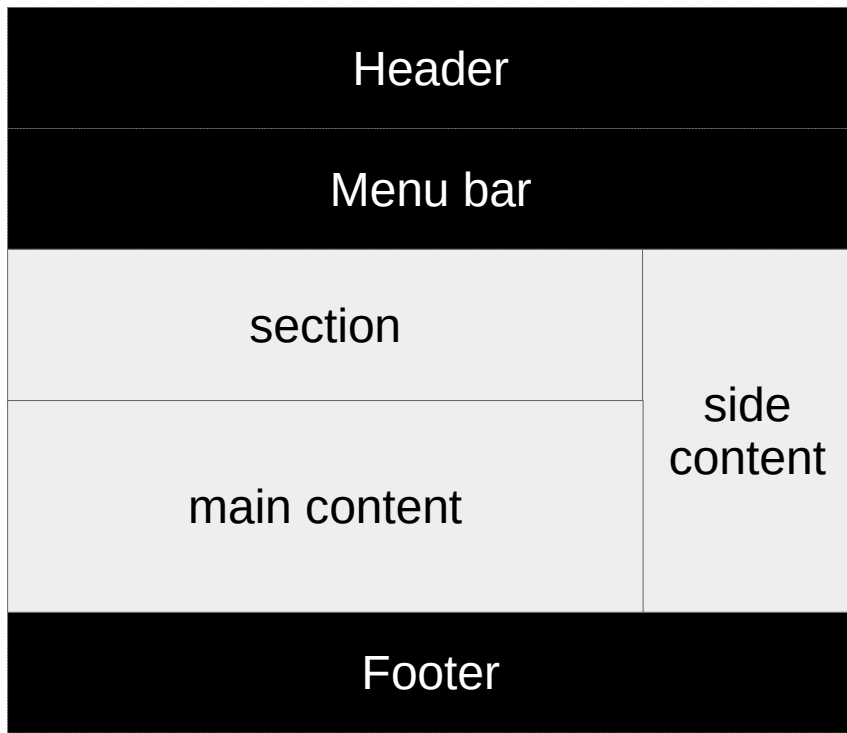
Float property

- The float property is used for positioning and layout on web pages

- The float property values

  - left : element floats to the left of its container

  - right : element floats to the right of its container

  - none (default) : element does not float (will be displayed just where it occurs in the text)

  - inherit : element inherits the float value of its parent

- In its simplest use, float property can be used to wrap text around images

Clear property

- The clear property specifies what elements can float beside the cleared element and on which side

- The clear property values

  - none : Allows floating elements on both sides. This is default
  - left : No floating elements allowed on the left side
  - right : No floating elements allowed on the right side
  - both : No floating elements allowed on either the left or the right side
  - inherit : The element inherits the clear value of its parent

Clear property

- The most common way to use the clear property is after you have used a float property on an element

- Your floated element will continue to float, but the cleared element will appear below it on the web page

Overflow property

- The overflow property specifies what happens if content overflows an element's box

- This property specifies whether to clip content or to add scrollbars when an element's content is too big to fit in a specified area

- The overflow property only works for block elements with a specified height

Syntax :
overflow : value;

Overflow property

Value	Description
visible	The overflow content is not clipped. It renders outside the element's box (default value)
hidden	The overflow is clipped, and rest of the content will be invisible
scroll	The overflow is clipped, but a scroll-bar is added to see rest of the content
auto	If overflow is clipped, a scroll-bar should be added to see the rest of the content
initial	Sets this property to its default value
inherit	Inherits this property from its parent element

![WSA - Forward looking IT finishing school]

Thank you

Web Stack Academy (P) Ltd

#83, Farah Towers,

1st floor,MG Road,

Bangalore – 560001

M: +91-80-4128 9576

T: +91-98862 69112

E: info@www.webstackacademy.com