HTML LAYOUT

Prerequisites

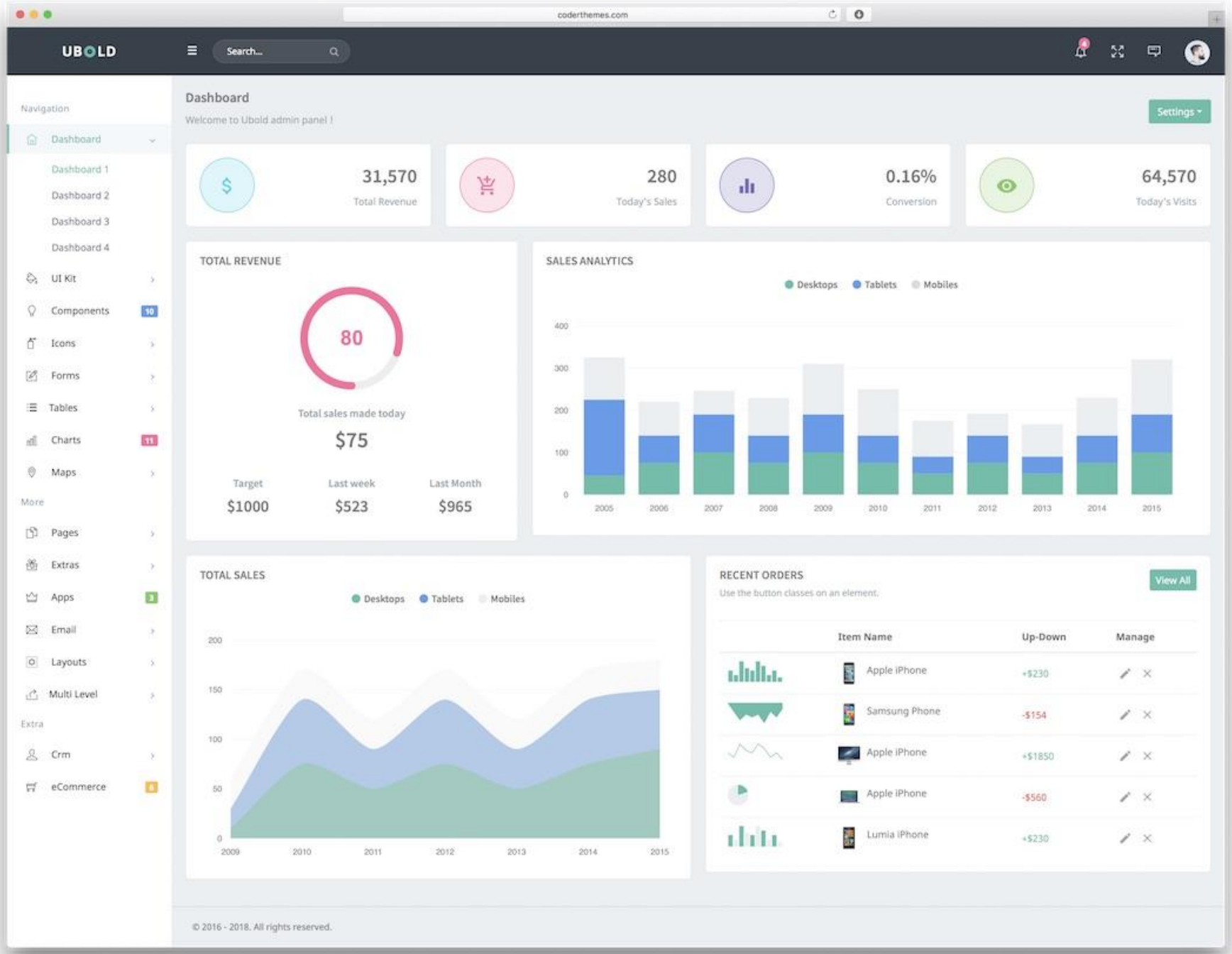
- Basic idea about web development
- Knowledge about HTML Tags and Elements
- Basic concept/idea of CSS.

Learning Objectives

- What is HTML layouts and Why it’s used
- Learn about HTML Layout Techniques
  - *CSS float property*
  - *CSS flexbox*
- What is HTML forms and how to use

HTML layout

- HTML Layout known as “**Semantic Elements**”
- HTML layout refers to the structuring and positioning of various elements on a webpage.
- A proper HTML layout is not just about making it beautiful; it is fundamental in ensuring a website is accessible, user-friendly, SEO-optimized, and easy to maintain and update.


coderthemes.com
UBOLD
Search...
Navigation
Dashboard
Dashboard 1
Dashboard 2
Dashboard 3
Dashboard 4
UI Kit
Components
Icons
Forms
Tables
Charts
Maps
More
Pages
Extras
Apps
Email
Layouts
Multi Level
Extra
Crm
eCommerce
Dashboard
Welcome to Ubold admin panel !
Settings
31,570
Total Revenue
280
Today's Sales
0.16%
Conversion
64,570
Today's Visits
TOTAL REVENUE
80
Total sales made today
$75
Target
$1000
Last week
$523
Last Month
$965
SALES ANALYTICS
Desktops
Tablets
Mobiles
400
300
200
100
0
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
TOTAL SALES
Desktops
Tablets
Mobiles
200
150
100
50
0
2009
2010
2011
2012
2013
2014
2015
RECENT ORDERS
Use the button classes on an element.
View All
Item Name
Up-Down
Manage
Apple iPhone
+$230
Samsung Phone
-$154
Apple iPhone
+$1850
Apple iPhone
-$560
Lumia iPhone
+$230
© 2016 - 2018. All rights reserved.

List semantic elements

- <header>
- <nav>
- <aside>
- <main>
- <article>
- <section>
- <form>
- <mark>
- <footer>

The Header element

- The <header> element is generally found at the top of a document, This element typically contains the website logo, navigation menu, and other important details such as the “search bar”.

```
<header>
    <div class="logo">
        <img src="./unicomtic.jpg" alt="unicom-tic-logo" width="50px" height="50px">
    </div>

    <div class="manu">
        <ul>
            <li>Home</li>
            <li>Our story</li>
            <li>FAQ</li>
            <li>Contactus</li>
        </ul>
    </div>

    <div class="search-form">
        <input type="search" placeholder="Search">
    </div>
</header>
```

The Nav element

- The <nav> element is used to define a section of navigation links.

```
<header>
    <div class="logo">
        <img src="./unicomtic.jpg" alt="unicom-tic-logo" width="50px" height="50px">
    </div>

    <nav>
        <ul>
            <li>Home</li>
            <li>Our story</li>
            <li>FAQ</li>
            <li>Contactus</li>
        </ul>
    </nav>

    <div class="search-form">
        <input type="search" placeholder="Search">
    </div>
</header>
```

The aside element

- It is most commonly used as a sidebar in the document. It does not render anything special in the browser.

```
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
    <style>
        main {
            padding: 10px;
            margin: 10px;
        }

        aside {
            width: 200px;
            border: 1px solid black;
            padding: 10px;
            margin: 10px;
            float: left;
        }
    </style>
</head>
<body>
    <aside>
        <h2>Sidebar</h2>
        <p>This is some content in the sidebar.</p>
    </aside>
    <main>
        <h1>Main Content</h1>
        <p>This is the main content of the page.</p>
    </main>
</body>
```

The Main element

- The HTML <main> tag is a semantic tag used to define the main content of the document. The main content contains the primary information or functionality that the page is meant to provide to the user.
- The document must not contain more than one <main> element.
- The <main> element should not be a child element of an <article>, <footer>, <header>, or <nav> element.

The Article and Section element

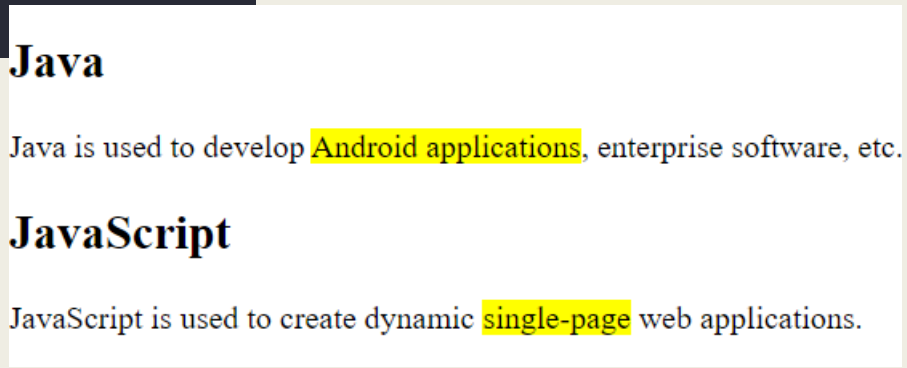
- Both these elements are used for sectioning a content
- Article element defines an independent piece of content, such as a blog post or a news article.
- Section element includes a group of related content. This element has no restrictions related to the type of content, so you can add anything to it – text, pictures, videos, etc.:

The Mark element

- The <mark> HTML element represents text which is marked or highlighted for reference or notation purposes due to the marked passage's relevance in the enclosing context.

```
<main>
    <section>
        <h2>Java</h2>
        <p>Java is used to develop <mark>Android applications</mark>, enterprise software, etc.</p>
    </section>

    <section>
        <h2>JavaScript</h2>
        <p>JavaScript is used to create dynamic <mark>single-page</mark> web applications.</p>
    </section>
</main>
```



The Footer element

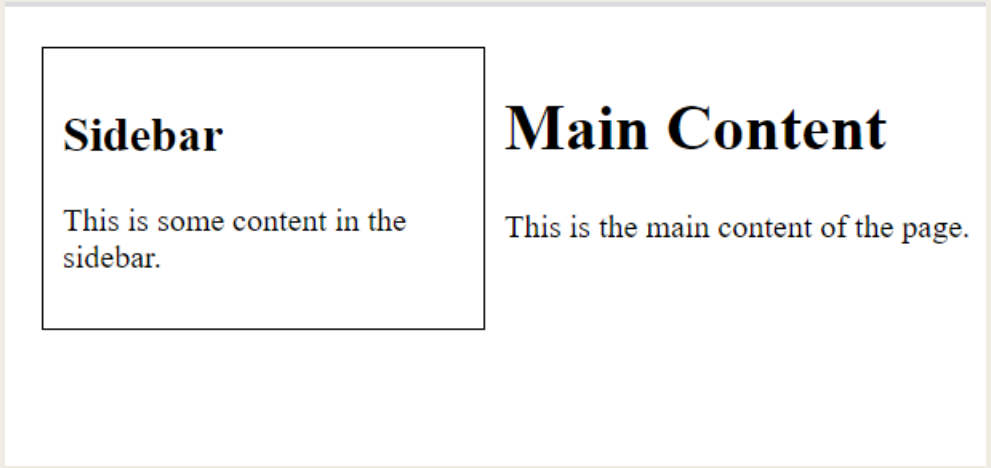
- A <footer> is generally found at the bottom of a document, a section, or an article.
- A footer element also contains authorship information, copyright information, contact information, sitemap, back-to-top links, related documents, etc.

HTML Layout Techniques

- CSS Float Layout - The float property is used for positioning and formatting content

```html
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
    <style>
        main {
            padding: 10px;
            margin: 10px;
        }

        aside {
            width: 200px;
            border: 1px solid black;
            padding: 10px;
            margin: 10px;
            float: left;
        }
    </style>
</head>
<body>
    <aside>
        <h2>Sidebar</h2>
        <p>This is some content in the sidebar.</p>
    </aside>
    <main>
        <h1>Main Content</h1>
        <p>This is the main content of the page.</p>
    </main>
</body>
```



- CSS flexbox - Flexbox is a one-dimensional layout method for arranging items in rows or columns. Items flex (expand) to fill additional space or shrink to fit into smaller spaces.

```
<nav>
    <ul>
        <li>Home</li>
        <li>Out Story</li>
        <li>Student Hub</li>
        <li>Lecturers Hub</li>
        <li>FAQ</li>
        <li>Contact</li>
    </ul>
</nav>
```

```
nav ul {
    margin: 0;
    display: flex;
    justify-content: space-around;
}
```

Form Element

The Form element

- HTML forms are used to get information from users. They are widely used in webpages or apps for surveys or registration processes.
- You can use the <form> element to create an HTML form.
- The HTML <form> element is a container for several HTML form elements. The <form> element can contain the following:
  - *<input>*
  - *<select>*
  - *<textarea>*
  - *<button>*
  - *<option>*

Form Attributes

- Action - The action attributes define the action to be performed when the form is submitted. It is usually the url for the server where the form data is to be sent.
- Method - The method attribute defines the HTTP method to be used when the form is submitted.
- Target - It specifies where to display the response received after the form is submitted. Similar to the target attribute in <a> tags, the target attribute has four possible values.
  - *_self (default): Load the response into the same browser tab.*
  - *_blank: Load the response into a new browser tab.*

- Enctype - It specifies how the form data should be encoded for the request. It is only applicable if we use the POST method.
- Name - It specifies the name of the form. The name is used in Javascript to reference or access this form. (recommended to use id to access the form elements)
- Novalidate - If the novalidate attribute is set, all validations in the form elements are skipped.

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <form action="/login" method="post" novalidate target="_self" enctype="application/x-www-form-urlencoded">
        <label for="email">Email:</label>
        <input type="email" name="email"><br><br>
        <label for="password">Password:</label>
        <input type="password" name="password"><br><br>
        <input type="submit" value="Submit">
    </form>
</body>
</html>
```

How to Use HTML Form Elements

How to Use the HTML <input> Element

- The <input> element is the most commonly used form element. The type of information an <input> element can hold depends on the <type> attribute.
- The <input> element can only accept a particular type of data assigned to it using the <type> attribute.
- The following are the different <type> attributes that can be assigned to an <input> element:

- Text - Allows the user to type in text.
- Number - Allows the user to type in number.
- Email - The user input must follow an email format.
- File - The user input must follow an email format.
- Password - Accepts password from the user. The passwords are masked, usually displayed as asterisks (*) or dots, to protect the privacy of the input.
- Checkbox - The user can select none or many of the displayed checkboxes. Checkboxes can be checked or unchecked.
- Radio - Allows the user to select only one from the multiple-choice radio buttons.
- Range - Creates a range picker from which the user can select the value
- Submit - Enables the user to submit the form.

- Date - Allows the user to picker the date.
- Time - Allows the user to accepts time value
- Week - lets the user pick a week and a year from a calendar
- Month - Creates an input field that lets the user enter month and year
- datetime-local - Creates a date and time picker
- Color - Allows the user to picker the color
- Hidden - Creates an invisible input field

Input element attributes

- <input name=" "> : Assigns the input field a name. The assigned name identifies the input data when the form is submitted.
- <input id=" ">: The identifier creates a unique id for the input field. It is usually associated with CSS for styling and JavaScript for other manipulations.
- <input value=" ">: Used to set the initial value for the input field. The default initial value gives the user an idea of the information required.
- <input placeholder=" ">: A faint pseudo value set to the input field that disappears once the user starts typing. Gives a hint on what data to enter, similar to the value attribute.
- <input required>: Requires that the input field must be filled out before submission. Gives an error message when not filled out.
- <input disabled>: As the name implies, this prevents the user from interacting with the input field. Disables the input field from accepting input. With this attribute, the input field becomes unclickable.
- <input readonly>: The user can only read the initially set value but can't change it. Unlike the disabled attribute, the input field is clickable but can't be modified.

How to Use the HTML <select> Element

- The <select> element creates a drop-down list, that allows users to select one or multiple options from the listed choices.
- The <option> element is contained within the <select> element. The <option> element holds the items to be selected. Each <option> represents one item in the drop-down list.

```html
<form>
    <div>
        <label>Student Name</label>
        <input type="text" placeholder="Enter student name">
    </div>
    <div>
        <label>Student Email</label>
        <input type="email" placeholder="Enter student email">
    </div>
    <div>
        <label>AboutUs</label>
        <textarea rows="4" placeholder="Enter about us"></textarea>
    </div>
    <div>
        <label>Gender</label>
        <input type="radio" value="Male" name="gender"> <span>Male</span>
        <input type="radio" value="Female" name="gender"> <span>Male</span>
    </div>
    <div>
        <label>Skill</label>
        <select>
            <option value="HTML">HTML</option>
            <option value="CSS3">CSS3</option>
            <option value="Angular">Angular</option>
        </select>
    </div>
    <div>
        <label>Is Active</label>
        <input type="checkbox">
    </div>
    <input type="submit" value="Submit">
</form>
```