

# **UNIT VI: MENUS, NAVIGATION AND WEB PAGE PROTECTION**

# Status Bar



- Status bar is present **at the bottom of browser window.**
- It enhances the **readability of the text present on the web page,** when user scroll over it.

# Builds A Static Message



- We can build a **static message** which is displayed on the **status bar**.
- This **message remains permanently present** in the status bar.

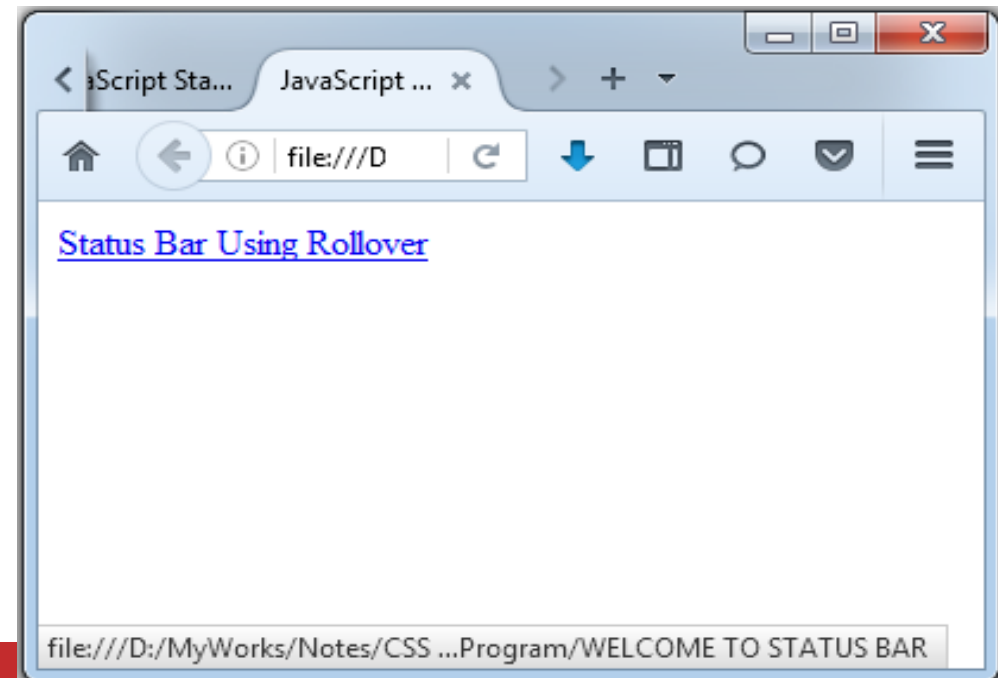
```
<html>
<head>
<title>JavaScript Status Bar</title>
<script>
window.status="WELCOME TO STATUSBAR AT BOTTOM OF WEB
PAGE...";
</script>
</head>
</body>
</html>
```

# Changing The Message Using Rollover



- We will use **onMouseOver** & **onMouseOut** events of a **hyper link** to display the message.
- We can use **window.status** on **onMouseOver** event to **change the status** in the status bar.
- It can display a JavaScript status bar message **whenever users overs your hyperlink.**

```
<html>
<head>
<title>JavaScript Status Bar</title></head>
<body>
  <a href="WELCOMETOSTATUSBAR"
    onMouseOver="window.status='Status Bar Using Rollover';return
true"
    onMouseOut="window.status="";return true">
Status Bar Using Rollover
  </a>
</body>
</html>
```



# Moving The Message Along The Status Bar



- ❑ For moving the message in the status bar we **need to increment the current position of the text one character ahead in a loop.**
- ❑ This will give us the **effect of moving the text.**
- ❑ This will be displayed by **using setInterval function**, which display text after some milliseconds.

```
<html>
<head>
<title>Scrolling Text</title>
<script language="JavaScript">
var scrollPos = 0
var maxScroll = 100
var blanks = ""
function scrollText(text, milliseconds) {
window.setInterval("displayText("+text
+")", milliseconds)
}
function displayText(text) {
  window.defaultStatus = blanks + text
  ++scrollPos
  blanks += ""

```

```
if(scrollPos > maxScroll)
{
  scrollPos = 0
  blanks = ""
}
}
</script>
</head>
<body onload="scrollText(Text
Moving on Status Bar....!!!, 300)">
<p>Watch the text scroll at the
bottom of this window!</p>
</body>
</html>
```



# Banner- Loading & Displaying Banner Advertisement



- It is typically **rectangular advertisement placed on a web site either above, below or on the sides** of the web site's main content & is linked to the advertiser's own web site.
- Banners ads may content **text or graphics images**.
- **Banner ads can be static** or animated.
- On the other hand, you can create some animated effect using JavaScript, like rotating static **banner ads at a certain time interval**.

# Banner- Loading & Displaying Banner Advertisement

- The JavaScript starts by declaring an array to store the banner images using the **new Array keywords**, as follows:
- **MyBanners=new  
Array('banner1.jpg','banner2.jpg','banner3.jpg','banner4.jpg')**

```
<html>
<head>
<script language="Javascript">
MyBanners=new Array(
CProg.jpg','C++Prog.jpg','Java.jpg',
Python.jpg','Fortran.jpg','dotnet.jpg')
banner=0
function ShowBanners()
{
if (document.images)
{
banner++
if (banner==MyBanners.length) {
banner=0}
document.ChangeBanner.src=MyBanner
s[banner]
setTimeout("ShowBanners()",4000)
}}
```

```
</script>
<body onload="ShowBanners()">
<center>
<h1> BANNER IN JAVASCRIPT
</h1>

</center> </body> </html>
```



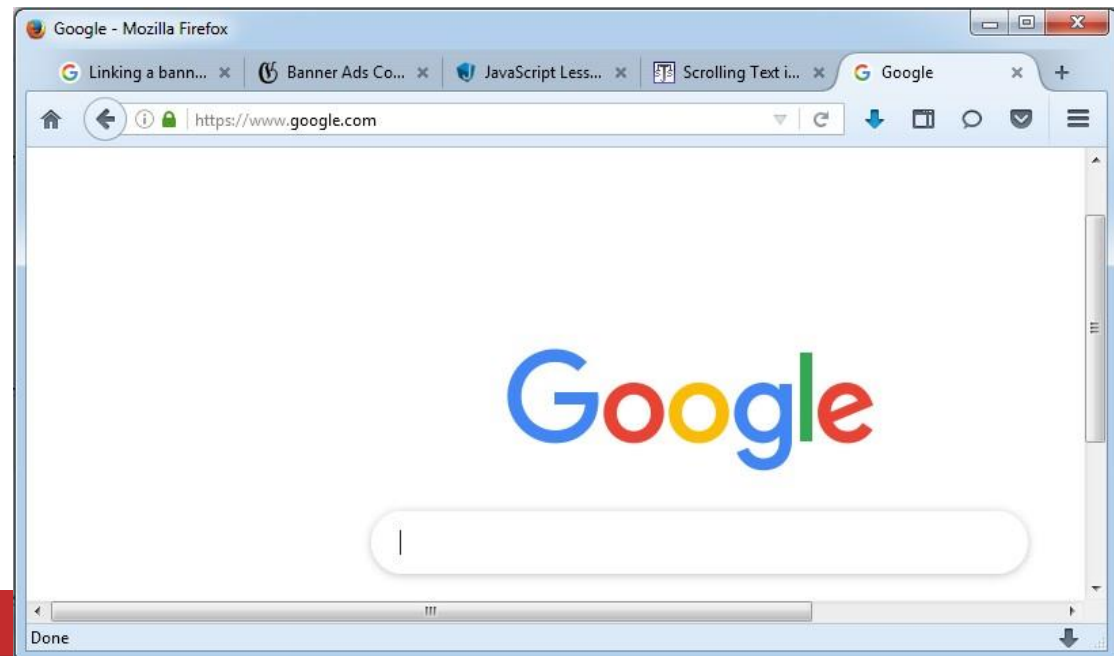
# Linking A Banner Advertisement To Url.



- ❑ In any web site the banner appear mainly for **advertising purpose**. It is essential with their corresponding web site.
- ❑ Creating rotating banner images will provide the visitor to your **webpage with some basic information**. However, if you want the visitor to get more information by clicking on the **banner images**, you need to create rotating banner ads that contain **URL links**.
- ❑ The script is basically the same as the previous one but we need to add another array that comprises the links, as follows:
- ❑ **MyBannerLinks=new Array('URL1','URL2','URL3/','URL4/')**

```
<html>
<head>
<script language="Javascript">
MyBanners=new Array('CProg.jpg','C++Prog.jpg','Java.jpg','Python.jpg')
MyBannerLinks=new Array('www.facebook.com/','www.google.com/','
www.gmail.com/','javascript-tutor.net/')
banner=0
function ShowLinks()
{
document.location.href="https://" + MyBannerLinks[banner]
}
function ShowBanners()
{
if (document.images)
{
banner++
if (banner==MyBanners.length)
```

```
{  
  banner=0}  
  document.ChangeBanner.src=  
  MyBanners[banner]  
  setTimeout("ShowBanners()",4000)  
}  
}  
</script>  
<body onload="ShowBanners()">  
<center>  
<a href="javascript: ShowLinks()">  
<h1> BANNER IN JAVASCRIPT</h1>  
  
</center>  
</body>  
</html>
```



# Slide Show



- A slide show is a **presentation of a series of still images on a projection screen** typically in a pre arranged sequence.
- In JavaScript we can create a slide show by using the **array of image**.

# Creating A Slide Show



- For creating of slide show we **need to have some image files.**
- In JavaScript the slide show is created & the **transition from one slide to another can be possible using the Back & Forward button.**



```
<html>
<head>
<script language="Javascript">
MySlides=new
Array('CProg.jpg','C++Prog.jpg','Java.jpg','Python.jpg','Fortran.jpg','dotnet.jpg')
Slide=0
function ShowSlides(SlideNumber)
{
{
    Slide=Slide+SlideNumber
    if (Slide>MySlides.length-1)
    {
        Slide=0
    }
    if (Slide<0) {
        Slide=MySlides.length-1
    }
    document.DisplaySlide.src=MySlides[Slide]
} }
}
```

```
</script>
</head>
<body>
<div>
<center>
<h1> SLIDE SHOW IN JAVASCRIPT</h1>
<p>
<input type="button" value="Back" onclick="ShowSlides(-1)">
<input type="button" value="Forward" onclick="ShowSlides(1)">
</div>
</center>
</body>
</html>
</head>
```

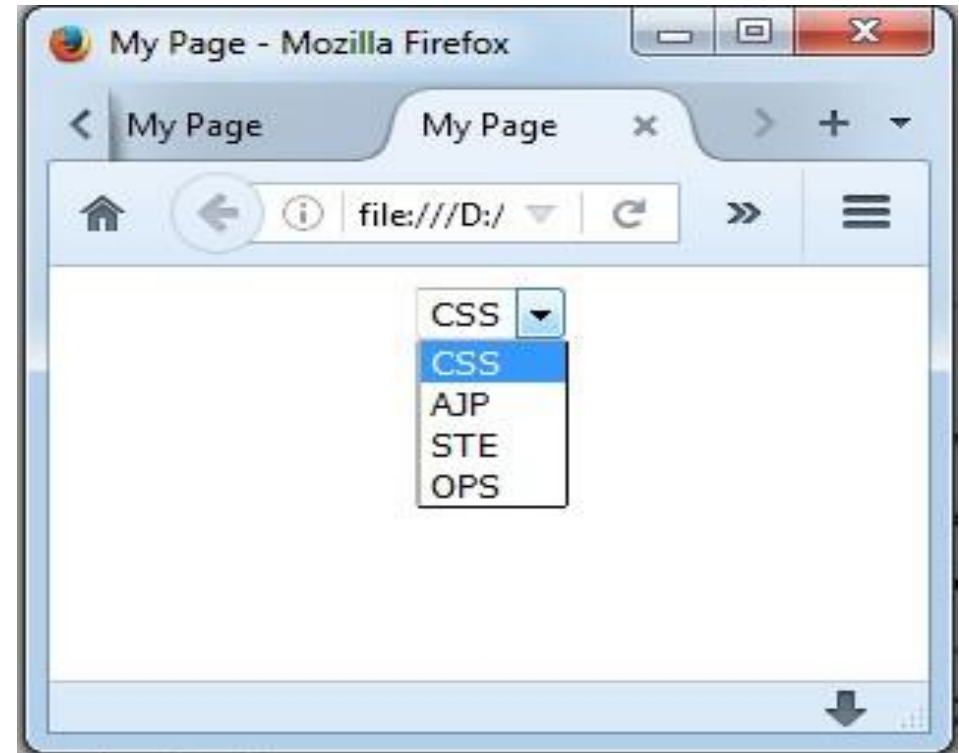


# Menus- Creating A Pulldown Menu



- ❑ A web site is normally a collection of various web pages. A **visitor** to this site **navigates from one page to another**.
- ❑ If a menu of these web pages is created then it becomes **easy for a visitor to select appropriate web pages**.
- ❑ The **<select >** element is used to **create a pulldown menu**.
- ❑ The **<option>** tages inside the **<select>** element define the available **option in the list**.

```
<html>
<head>
<title>My Page</title>
</head>
<body>
<form name="myform" action="Demo" method="POST">
<div align="center">
<select name="mydropdown">
<option value="Subject">CSS</option>
<option value="Subject">AJP</option>
<option value="Subject">STE</option>
<option value="Subject">OPS</option>
</select>
</div>
</form>
</body>
</html>
```



# Dynamically Changing A Menu



- Dynamically changing menu means the **items present in the menu are changing automatically.**

# Validating Menu Selection



- ❑ Many time user can't remember how many item are selected from menu. So JavaScript solve this problem.
- ❑ JavaScript provide the item are **selected from menu or not**.

```
<html>
<script language="Javascript">
function validate()
{
if(document.form.city.selectedIndex=="")
{
alert ("Please select city!");
return false;
}
var sel = document.getElementById("city");
var selectedText = sel.options[sel.selectedIndex].text;
alert("You have selected : "+selectedText);
return true;
}
</script>
```

```
<form name="form" method="post" onSubmit="return  
validate0"><pre>
```

```
<center>
```

```
Select your City <select name="city" id="city">
```

```
<option value="Select">Select</option>
```

```
<option value="Mumbai">Mumbai</option>
```

```
<option value="Nashik">Nashik</option>
```

```
<option value="Delhi">Delhi</option>
```

```
<option value="Bangalore">Bangalore</op
```

```
<option value="Pune">Pune</option>
```

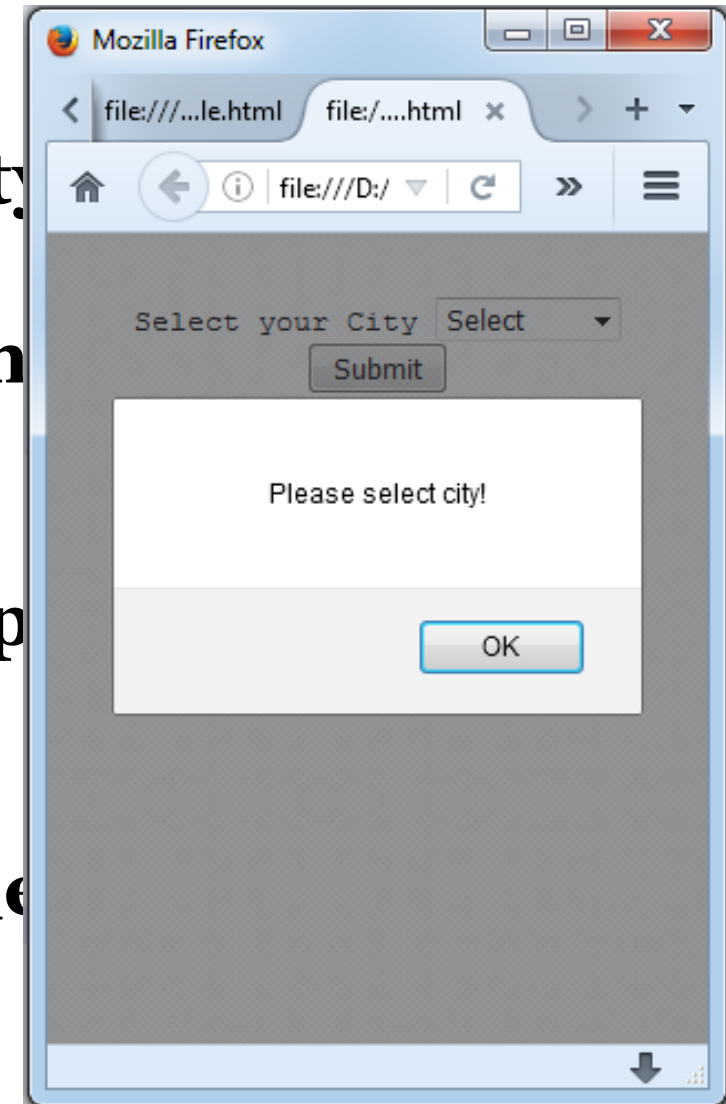
```
</select>
```

```
<input type="submit" name="Submit" value
```

```
</pre>
```

```
</center></form>
```

```
</html>
```





# Floating Menu



- Many web pages are too long to fit on most users' screens. The visitors have to scroll to read the article on the page; however such scrolling hides the navigation menus usually located at the top of the page.
- The JavaScript shown here allows to **create dynamic menus** which move along with scrolling. Such **floating menu will be always visible on screen**. The effect is achieved by moving an **absolutely-positioned or relatively-positioned DIV box** containing the menu markup.

```
<!DOCTYPE html>
<title>Example</title>
<style>
  body {
    background-image: url('JS.jpg');
  }
  main {
    margin-bottom: 200%;
  }
  .floating-menu {
    font-family: sans-serif;
    background: yellowgreen;
    padding: 5px;;
    width: 130px;
    z-index: 100;
    position: fixed;
  }
```

```
.floating-menu a,
  .floating-menu h3 {
  font-size: 0.9em;
  display: block;
  margin: 0 0.5em;
  color: white;
}
</style>
```

```
<main>
```

<p>Scroll down and watch the menu remain fixed in the same position, as though it was floating.</p>

```
<nav class="floating-menu">
```

```
<h3>Floating Menu</h3>
```

```
<a href="/css/">CSS</a>
```

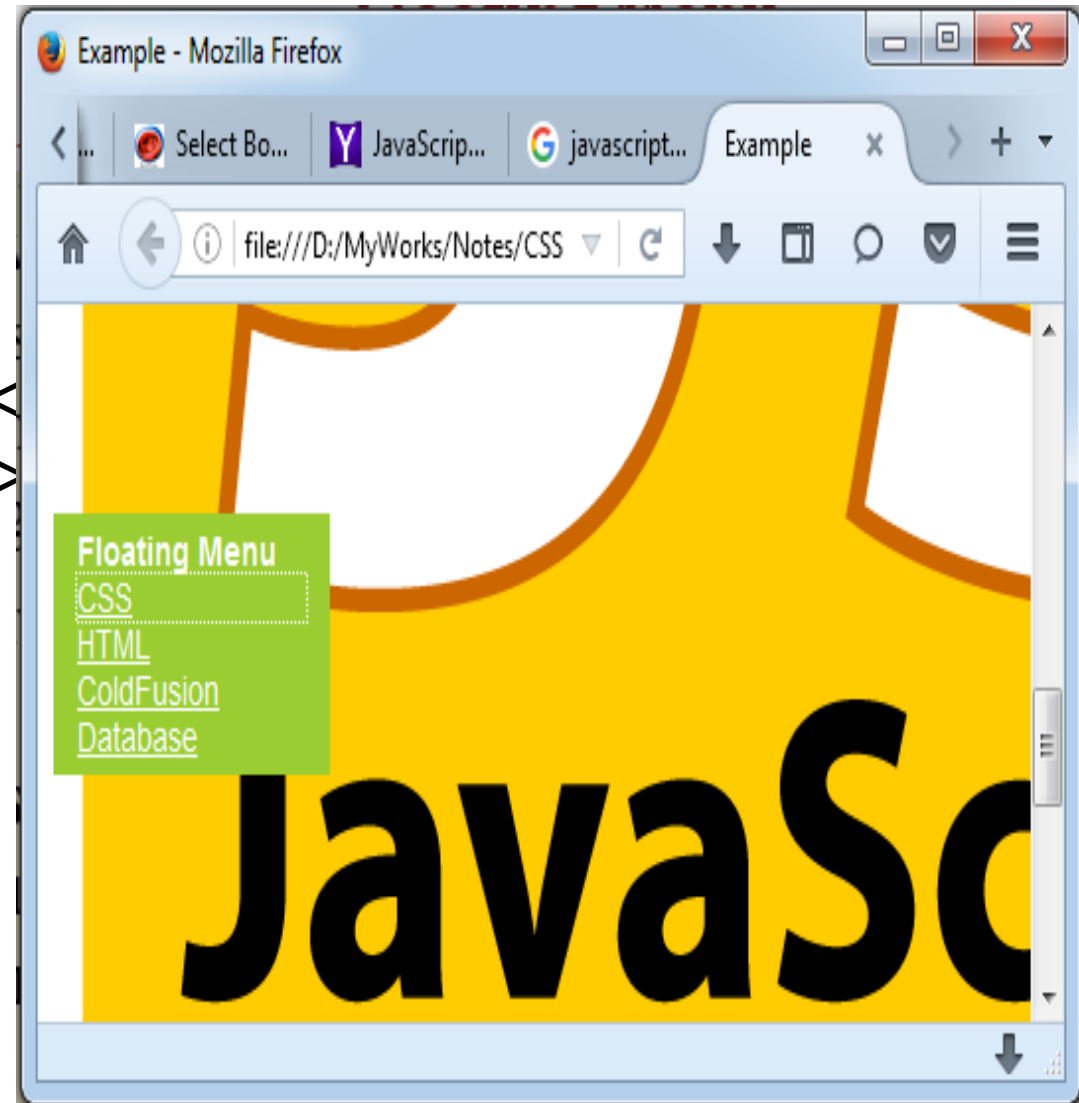
```
<a href="/html/">HTML</a>
```

```
<a href="/coldfusion/">ColdFusion</a>
```

```
<a href="/database/">Database</a>
```

```
</nav>
```

```
</main>
```



# Chain Select Menu



- Chain Select Menu is a kind of menu in which **three or more than one set of menus** & option selected from the **first pulldown menu determines the options for second pulldown menu** & option of second pulldown menu determine the option for third pulldown menu.

Vehicles ▼ Cars ▼ Submit

Cars  
Trucks  
Blkes  
Train

# Tab Menu



- Tab menu display a **one or two word description of the menu option within a tab.**
- A more complete description is displayed below the tab bar as the visitor moves cursor over the tab.

## About

Home

About

Clients

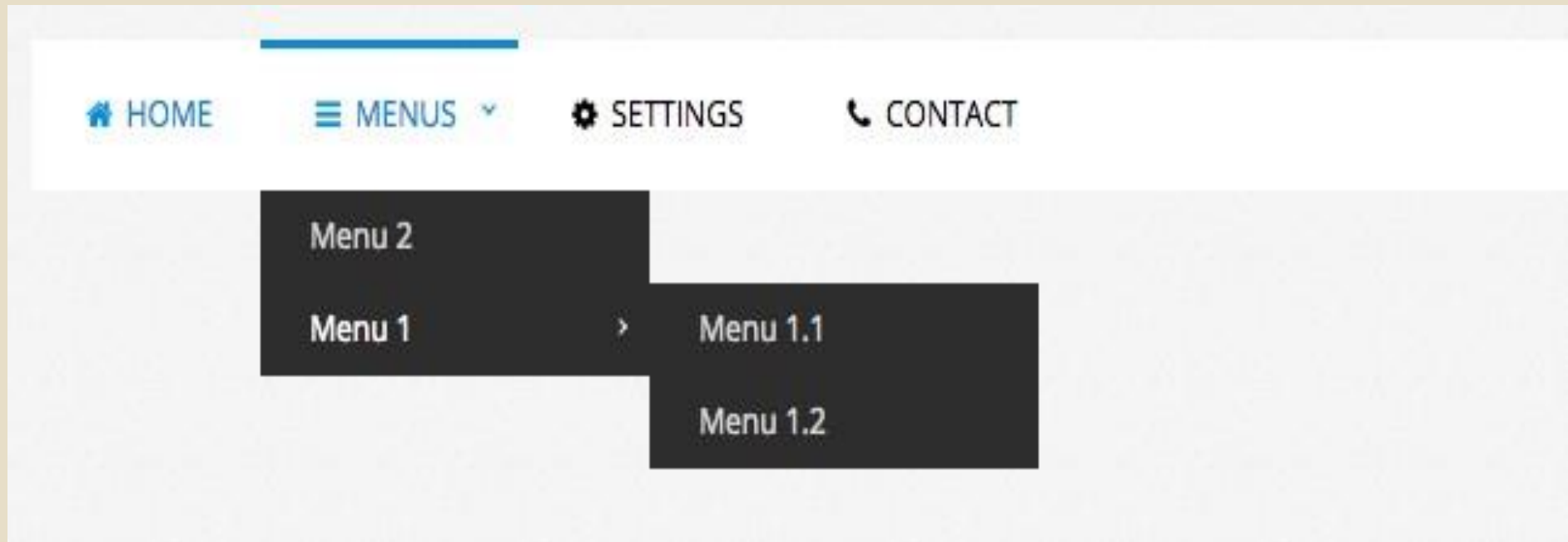
Contact Us

The above navigation clearly indicates the "current" page. All four of the menu options is a link, but notice the current navigation link doesn't behave like one.

# Popup Menu



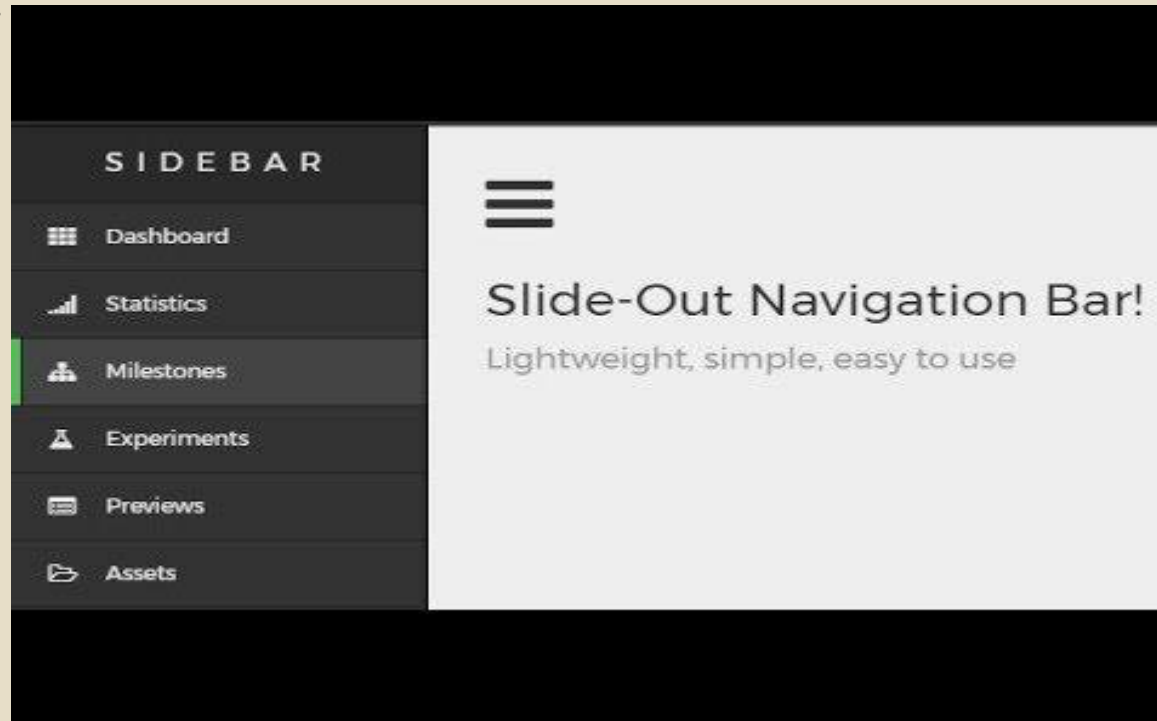
- The pop up menu **contains lower level menu items that are associated with the top level menu item.** The popup menu appears as you move mouse over each item.



# Sliding Menu



- These menus are basically **off screen elements that slide into view when you click or tap on something that looks like an arrow**, a hamburger icon or something else that indicates a menu will appear.



# Highlighted Menu

- We can improve the user experience highlighting the active menu. This makes navigation easier because the visitor will know at the first glance which is the current page.
- This strategy also increases the amount of visited pages because the visitor will more likely click the next page when it's easier to find that.

Home page  
Programs  
Rubik's Cube  
Twisty Puzzles

## Funny images »

Page 2

Page 3

Page 4

Page 5

Page 6

## Rubik's Programs »

Cube solver

Timer

Simulator

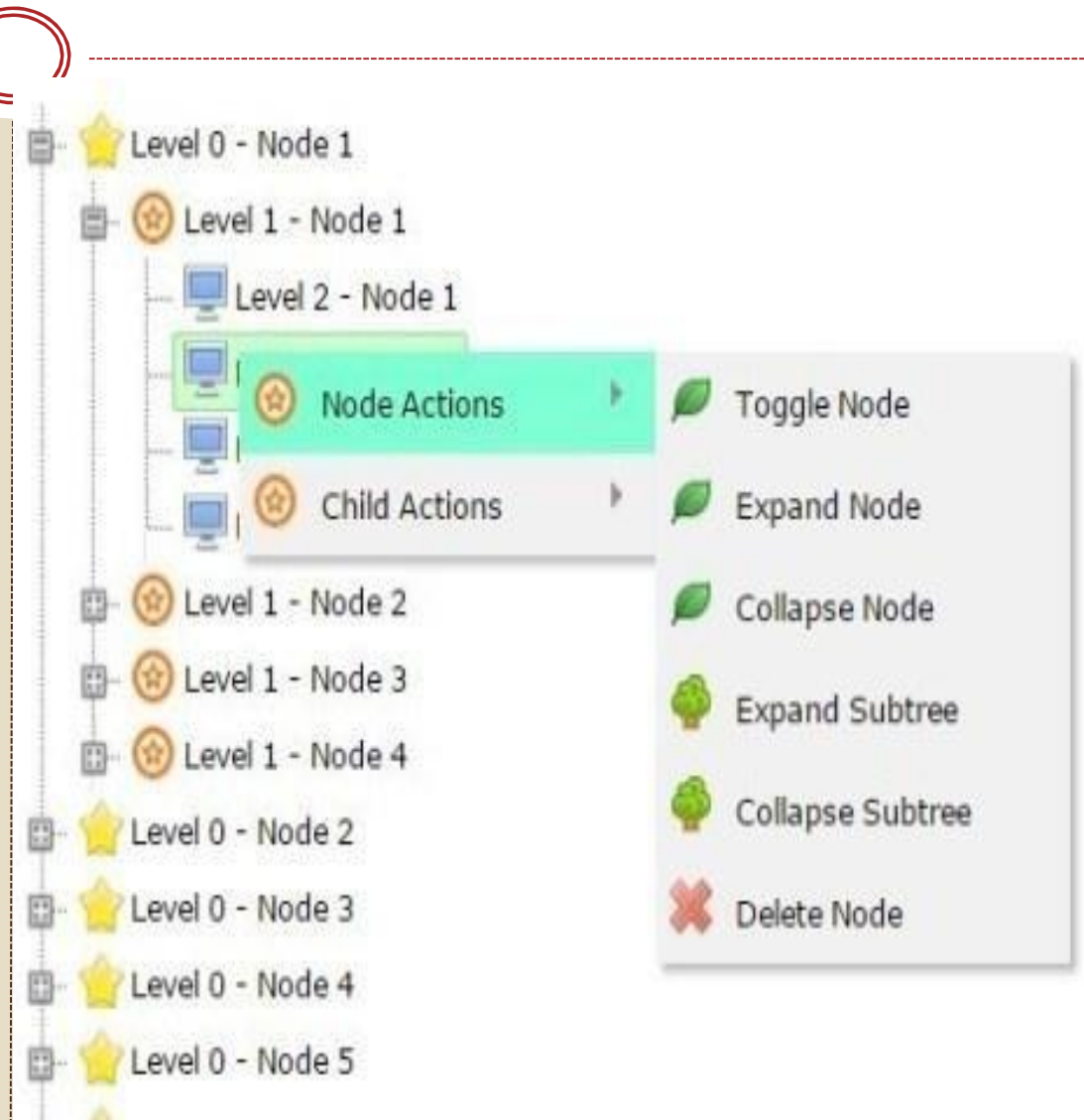
Widget

Scrambler



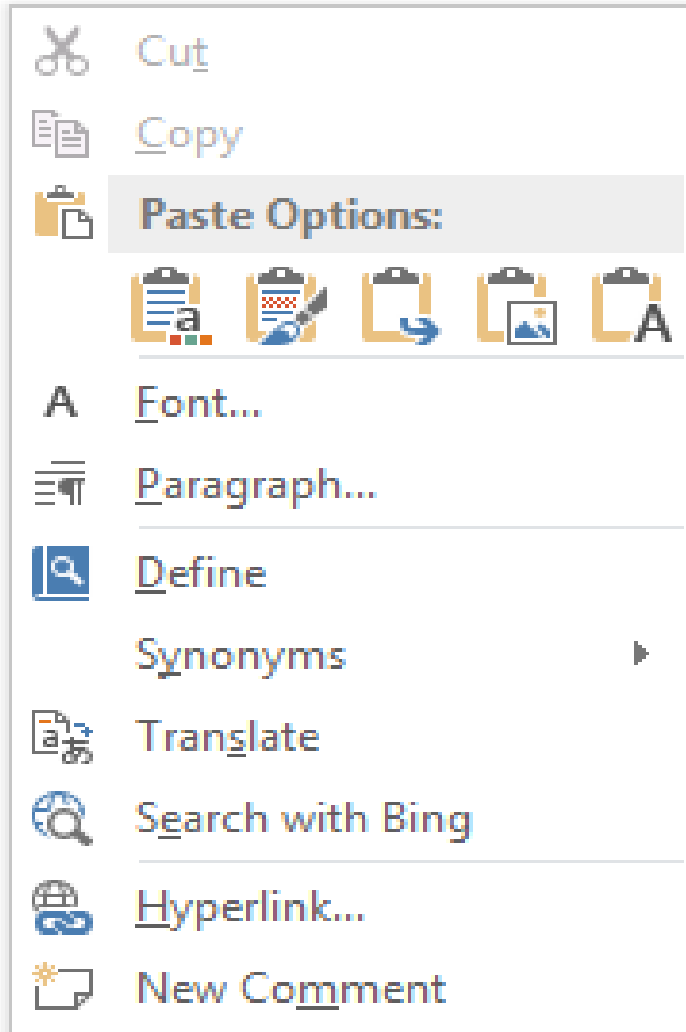
# Folding A Tree Menu

- A menu which contains menu & sub menus contain the sub menu in the hierarchy like tree of menu is created.
- The menu can be **folded by hiding the sub menus & unfolded displaying the submenu** of a menu is known as **foldable tree menu**.



# Context Menu

- The context menu is a menu that **pops up when user clicks right mouse button.**
- The location of the context menu is **determined by the position of mouse.**



# Scrollable Menu

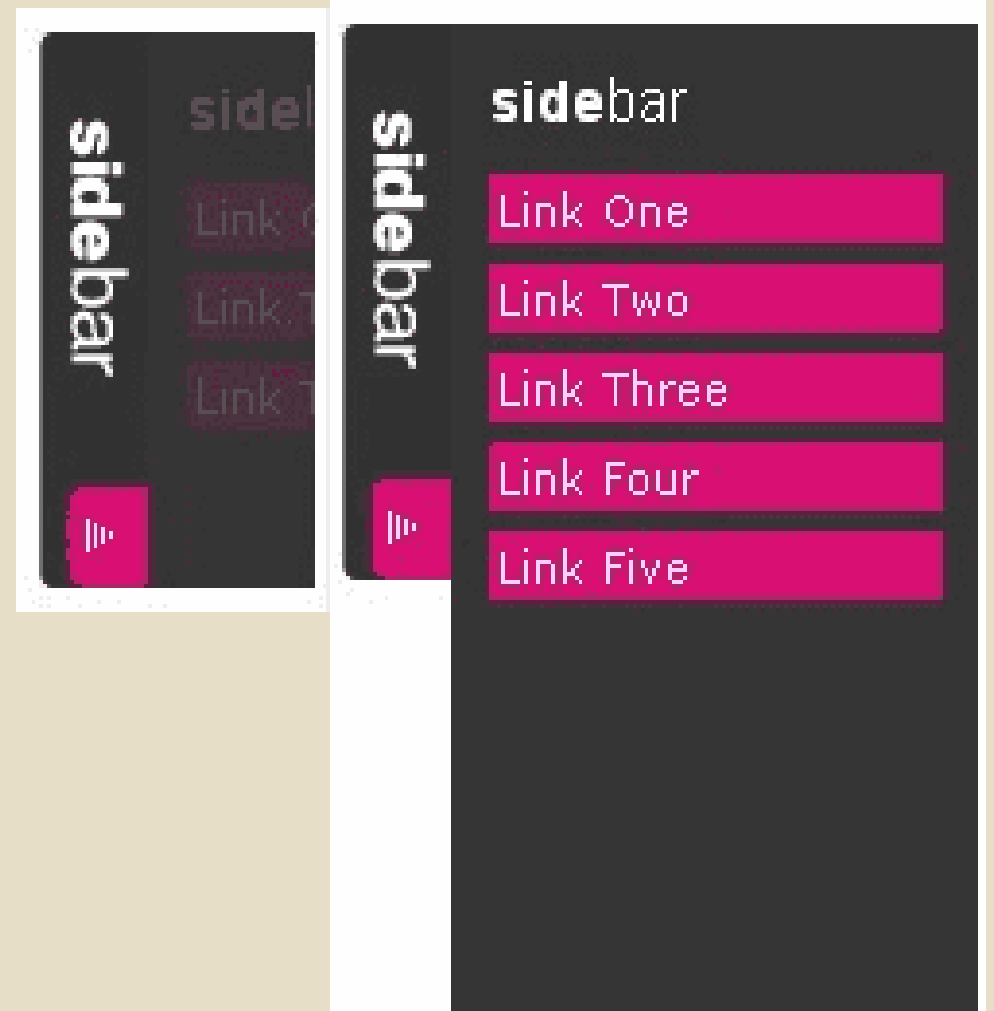


- Sometimes there is **limited space on the web page for displaying all the menu options**.
- Then such a case only **limited menu options are displayed & remaining options can be accessed by scrolling left or right**.



# Slide Bar Menu

- The side bar menu displays a menu on the **side of the web page**. Options on this menu can be **linked to other web pages**.
- Visitors can link to other **menus by moving the mouse cursor over a menu item**. The menu that is associated with that item pops onto the screen.



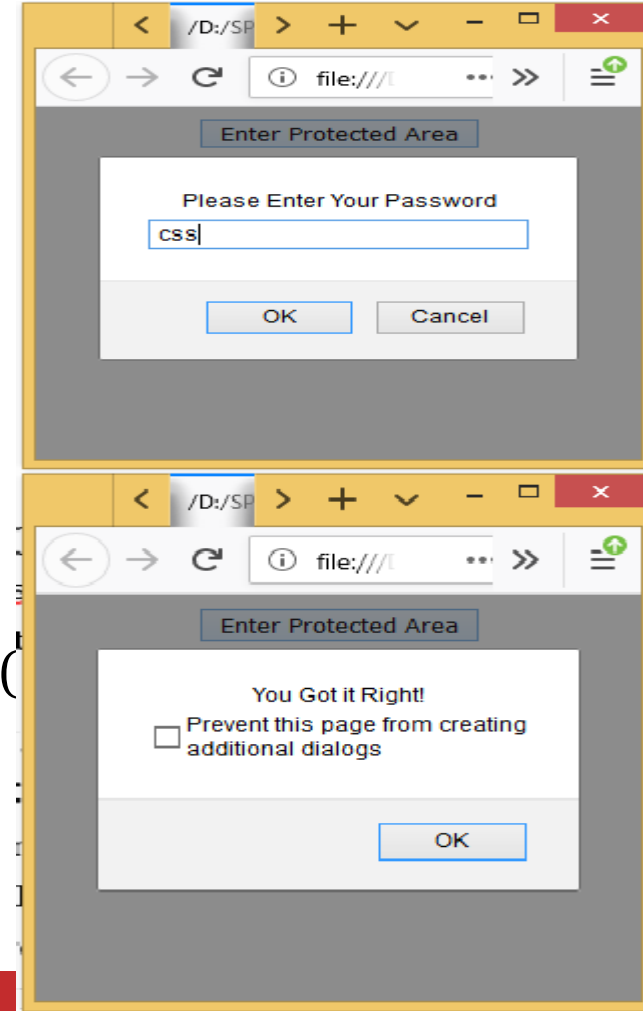
# Protecting Web Page



- There are many reasons to **password protect a web page** you control sensitive personal information, snooping co-workers, or content not intended for children.
- Web data security is an **ever-evolving field and attackers are extremely sophisticated.**
- **Using password field** we can protect web page. If user enter right password then redirect to next page otherwise show error.

```
<html> <SCRIPT>
function passWord() {
var testV = 1;
var pass1 = prompt('Please Enter Your
Password,');
while (testV < 3) {
if (!pass1)
history.go(-1);
if (pass1.toLowerCase() == "css"){
alert('You Got it Right!');
window.open('www.wikihow.com');
break;
}
testV+=1;
var pass1 =
```

```
prompt('Access Denied - Password Incorrect, Please
Try Again.','Password');
}
if (pass1.toLowerCase()!="password" & testV ==3)
history.go(-1);
return " ";
}
</SCRIPT>
<CENTER> <FORM>
<input type="button"
value="Enter Protected
Area" onClick="passWord(
</FORM> </CENTER>
</html>
```



# Hiding Your Code



- ❑ The source code of the web pages can be viewed by clicking right mouse button on the webpage.
- ❑ Anyone can visit to the web page can simply view the source code of the webpage.
- ❑ View a source code by users is not safe.
- ❑ We can hide the **source code of the web page y disabling the right click on webpage.**
- ❑ Hiding source code is **protecting the source code by viewing other users.**

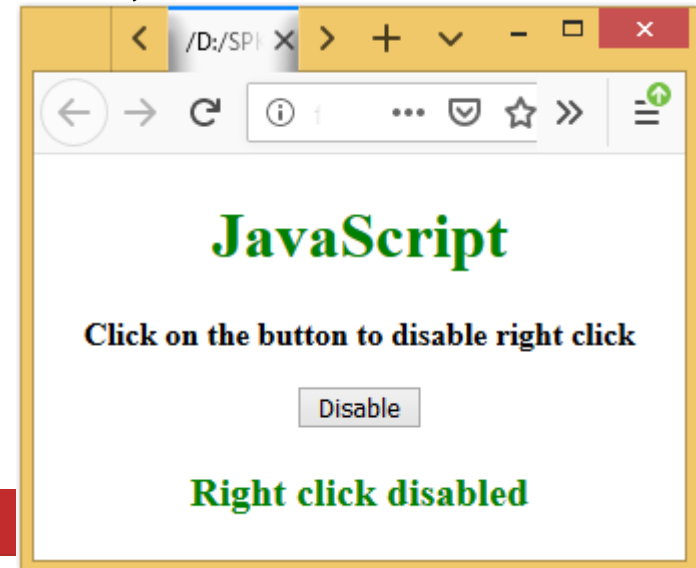
# Disabling The Right Mouse Button In JS



- ❑ JavaScript methods are used to disable the right click on the page. The used methods are listed below:
- ❑ **Syntax:        document.addEventListener(event,        function, useCapture)**
- ❑ **PreventDefault() Event Method** This method cancels the event if it can be cancelled, meaning that it stops the default action that belongs to the event.
- ❑ **Syntax:    event.preventDefault()**



```
<!DOCTYPE HTML> <html>
<body style = "text-align:center;">
<h1 style = "color:green;" >JavaScript</h1>
<p id = "GFG_UP" style = "font-size: 16px; font-weight: bold;"></p>
<button onclick = "Run()"> Disable </button>
<p id = "GFG_DOWN" style = "color:green; font-size: 20px; font-weight: bold;">
</p>
<script>
var up = document.getElementById("GFG_UP");
var down = document.getElementById("GFG_DOWN");
up.innerHTML = "Click on the button to disable right click";
function Run()
{
document.addEventListener('contextmenu', event =>
event.preventDefault());
down.innerHTML = "Right click disabled";
}
</script> </body> </html>
```



# Concealing Email Address



- Adding `<code>` around the @ sign will put most spam bots at bay. `<code>` as defined in HTML puts any text in between it in font.
- The easiest way is to replace the @ sign in your email address with the "&#64;" but not the most secure. Link: `<a href="mailto:news&#64;mydomain.com">Email Us!</a>` Text email: Email us at: news&#64;mydomain.com
- This JavaScript is for providing a clickable email address but hides the email address from spambots.
- The Window `atob()` method is used for decoding a base-64 encoded string. It is used to decode a string of data which has been encoded using the `btoa()` method.

```
<!DOCTYPE HTML> <html>
<a id="contact-link" href="#" data-
secret="aGVsbG9AbXlkb21haW4uY29t">[Javascript required]</a>...
<script>
window.onload = function()
{
var contact = document.getElementById("contact-link");
contact.textContent = atob(contact.dataset.secret);
contact.href = atob("bWFpbHRvOg==") + atob(contact.dataset.secret);
};
</script>
</html>
```

**Output:**

**hello@mydomain.com**

# Frameworks Of JavaScript & Its Application.



- ❑ JavaScript framework is an application framework written in JavaScript where the **programmers can manipulate the functions and use them for their convenience.**
- ❑ Frameworks are more **adaptable for the designing of websites.**
- ❑ JavaScript frameworks are a type of tool that makes **working with JavaScript easier and smoother.**
- ❑ These frameworks also make it possible for the **programmer to code the application as a device responsive.**

# Frameworks Of JavaScript & Its Application.

□ Here's a list of frameworks :

1. **Angular**
2. **React**
3. **Vue.js**
4. **Ember.js**
5. **Meteor**
6. **Mithril**
7. **Node.js**
8. **Polymer**
9. **Aurelia**
10. **Backbone.js**



# Frameworks Of JavaScript & Its Application.



□ Here's a list of frameworks :

- 1. Angular:** This framework is operated by Google and is basically implemented to use for **developing Single Page Application (SPA)**. It extends the **HTML into the application and interprets the attributes in order to perform data binding.**
- 2. React:** It is used to develop and operate the **dynamic User Interface of the web pages with high incoming traffic.** It makes the use of a virtual DOM and hence, the integration of the same with any application is easier.
- 3. Vue.js:** Its **dual integration mode** is one of the most attractive features for the creation of **high-end SPA or Single Page Application.** It is a much reliable platform for developing cross-platform.

# Frameworks Of JavaScript & Its Application.

- 4. Ember.js:** It support **two-way data binding** and hence, establish a **reliable platform for handling the complicated User Interfaces**. E.g.: LinkedIn, Netflix, Nordstrom and many more websites.
- 5. Meteor:** **Back-end development, management of the database, business logic and rendering of the front-end** part of the websites are a major area where Meteor framework is used.
- 6. Mithril:** It's **small (< 8kb gzip), fast and provides routing and XHR utilities** out of the box. It has a few features similar to React.
- 7. Node.js:** It works in the JavaScript **Runtime environment and hence shows similar properties of JAVAlike threading, packages, forming of loops**.

# Frameworks Of JavaScript & Its Application.



8. **Polymer:** It supports both, **one-way and two-way data binding**, hence making the application area wider.
9. **Aurelia:** It can extend the HTML for various purposes, including data binding. Also, its modern architecture ensures that the tool can be used for **interpretation of both client-side and server-side at a time.**
10. **Backbone.js:** This framework was made by keeping in mind the idea that all the **server-side functions must flow through an API, which would help in achieving complex functionalities** by writing less code.





*Thank you*