

Termwork 1

Create Multi Column article using HTML Tags, Integrate social sharing feature. Implement both Web view & Mobile view.

Theory:

- HTML (Hyper text markup language) is the most basic building block of the Web.
- HTML uses "markup" to annotate text, images, and other content for display in a Web browser
- There are some basic tags which has an ending tag, except some tags eg. `
`
- Some basic tags.
 - `<h1></h1>` } used for headings and size varies
 - `<h6></h6>` } from big(`<h1>`) to small(`<h6>`)
 - `<p></p>` → paragraph
 - `<body></body>` → defines body of the document
 - `<html></html>` → root of an HTML document
- Responsive Web designing is about using HTML and CSS to automatically resize, shrink, hide, enlarge websites to make it look good on all devices.
- `<meta>` is used as follows (for eg.)
`<meta name="viewport" content="width=device-width", initial scale=1.0">`
- for images we can use max-width property.
- `<picture>` is also used depending on browser's width
- Text size can be altered & using vw in the end of font-size.

Program :

```
<!DOCTYPE html>
<html>
<head>
    <title> Multi Column Article </title>
    <style>
        .article {
            column-count: 4;
            column-width: 100px;
            column-gap: 40px;
            column-rule-color: black;
            column-rule-width: 1px;
            column-rule-style: solid;
        }
        h1 {
            column-span: all;
        }
        #Share-buttons img {
            width: 15px;
            padding: 5px;
            border: 0;
            box-shadow: 0;
            display: inline;
        }
    </style>
</head>
<body>
    <div class="article">
        <h1> This is spanning all columns. </h1>
    </div>
</body>
```

<p> General Relativity is physicist's Albert Einstein's understanding of how gravity affects the fabric of space-time.

The theory, which Einsteins ...

</p>

<h1> This is spanning all columns. </h1>

<p> To understand general relativity, first, lets start with gravity, the force of attraction that two bodies exert on one another. Sir Issac Newton quantified gravity ...

</p>

</div>

<div id = "share_buttons">

<!-- Twitter -->

<!-- Reddit -->

<!-- Facebook -->


```
<!-- LinkedIn -->
<a href = "http://www.twitter.com" target = "_blank">
    <img src = "https://simplesharebuttons.com/images/somacro/
        linkedin.png" alt = "linkedin" >
</a>

<!-- Pinterest -->
<a href = "http://www.pinterest.com" target = "_blank">
    <img src = "https://simplesharebuttons.com/images/somacro/
        pinterest.png" alt = "pinterest" >
</a>

</div>
</body>
</html>
```

Termwork 1

Name: Vinit Gunaki

USN: 2gi20cs181

Output:

This is Spanning all columns..

lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent eu efficitur tellus. Nam in nisl sed sapien lobortis sodales. Maecenas at eleifend turpis. Duis non maximus ligula. Nunc rutrum diam quis cursus interdum. Nulla viverra dolor et tellus consecetur, et congue

enim blandit. Proin non sem dui. Ineget porta cursus dolor sed laoreet. Pellentesque non turpis ut amet eros imperdiet suscipit. Sed quis efficitur ipsum, eu ornare tellus. Aliquam orci est, ultrices non cursus ut amet, placerat nec nisi. Nulla ligula est, viverra a nibh ac,

accumam bibendum lacus. Quisque ut tagitata lectus. Aliquam tincidunt in nunc eu hendrerit. Duis nulla enim, rutrum eget arcu vel, pharetra cursus lacus. Aliquam ultricies, ex nec iaculis congue, ante elit lacunam metus, eger congue lorem est eu risus. Pellentesque nec

massa turpis. Fusce eu augue non lorem maximus commodo in quis massa. Aenean id viverra nisi, id lacunam ligula. Nunc viverra dolor id turpis sollicitudin, eger cursus turpis tempor

This is Spanning all columns..

lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent eu efficitur tellus. Nam in nisl sed sapien lobortis sodales. Maecenas at eleifend turpis. Duis non maximus ligula. Nunc rutrum diam quis cursus interdum. Nulla viverra dolor et tellus consecetur, et congue

enim blandit. Proin non sem dui. Ineget porta cursus dolor sed laoreet. Pellentesque non turpis ut amet eros imperdiet suscipit. Sed quis efficitur ipsum, eu ornare tellus. Aliquam orci est, ultrices non cursus sit amet, placerat nec nisi. Nulla ligula est, viverra a nibh ac,

accumam bibendum lacus. Quisque ut tagitata lectus. Aliquam tincidunt in nunc eu hendrerit. Duis nulla enim, rutrum eget arcu vel, pharetra cursus lacus. Aliquam ultricies, ex nec iaculis congue, ante elit lacunam metus, eger congue lorem est eu risus. Pellentesque nec

massa turpis. Fusce eu augue non lorem maximus commodo in quis massa. Aenean id viverra nisi, id lacunam ligula. Nunc viverra dolor id turpis sollicitudin, eger cursus turpis tempor



Termwork 2

Implement HTML5 dropdown menu with CSS & Bootstrap.

Theory:

- o CSS is the language we use to style an HTML document.
- o CSS (Cascading Style Sheets) can control the layout of multiple pages all at once.
- o Each tag can have its own styling and user can define it before the body tag.

eg: body {

 background-color : blue;

}

 p {

 font-family : veranda;

 text-align : center;

}

- o Selector { declaration } : syntax.

- o Bootstrap is a free front-end framework for faster and easier web development.

- o Bootstrap includes HTML and CSS based design templates.

- o Bootstrap gives you the ability to easily create responsive design.

Program :

```
<!DOCTYPE html>
<html>
<head>
    <title>dropdown </title>
    <meta name = "viewport" content = "width=device-width,
        initial-scale=1">
    <link rel = "stylesheet" href = "bootstrap.min.css">
    <script src = "jquery.min.js"></script>
    <script src = "bootstrap.min.js"></script>
    <style>
        .dropdown-submenu {
            position: relative;
        }
        .dropdown-submenu .dropdown-menu {
            left: 100%;
            margin: 10px;
        }
    </style>
</head>
<body>
    <div class = "container-fluid">
        <h2> Subjects offered @ 3rd Semester </h2>
        <div class = "dropdown">
            <button class = "btn btn-default" type = "button", data-
                toggle = "dropdown"> Subject Dropdown <span
                class = "caret" ></span>
            </button>
            <ul class = "dropdown-menu">
```

```
* <li><a href="#"> 1th semester </a> </li>
<li><a href="#"> 5th semester </a> </li>
<li class = "dropdown-submenu">
    <a class = "test" href="#"> 3rd semester <span class =
        "caret"></span>
    </a>
    <ul class = "dropdown-submenu">
        <li><a href="#"> DS </a> </li>
        <li><a href="#"> JAVA </a> </li>
        <li><a href="#"> WEB </a> </li>
    </ul>
</li>
</ul>
</div>
<script>
$(document).ready (() => {
    $('.dropdown-submenu a.test').on ('click',
        e => {
            $(this).next ('ul').toggle()
            e.stopPropagation()
            e.preventDefault()
        }
    )
})
</script>
</div>
</body>
</html>
```

Termwork 2

Name: Vinit Gunaki

USN: 2gi20cs181

Output:

Subjects offered @ 3rd Sem

SUBJECTS ▾

7th sem

5th sem

3rd sem ▾

Subjects offered @ 3rd Sem

SUBJECTS ▾

7th sem

5th sem

3rd sem ▾

DS

JAVA

WEB

Termwork 3

Implement HTML5 local storage.

Theory:

- HTML5 local storage is a component of the Web storage application programming interface. It is a method by which Web pages locally store named key/value pairs inside a client's Web browser.
- Mainly used to store and retrieve data in HTML pages from the same domain.
- Even after restarting a browser, the data can be recovered from all windows in the same domain.
- From the JavaScript code, HTML5 local storage may be accessed through a `localStorage` object on the global window object.
- As data is stored on client-side, stored data is based on the browser that is used.

Program :

```
<!DOCTYPE html>
<html>
  <head>
    <title> Web Local Storage </title>
  </head>
  <body>
    <div id = "local"></div>
    <h2> this page was viewed <span id = "count"></span>
      time(s). </h2>
    <p>
```

Capital of India:

```
  <span id = "cIn"></span>
```

```
</p>
<script type = "text/javascript">
```

```
var cIndia = "";
// replace "pageload" with hits
```

```
if (!localStorage.pageloadCount) {
  localStorage.pageLoadCount = 0;
}
```

```
localStorage.pageLoadCount = parseInt(localStorage.pageLoadCount)+1;
```

```
document.getElementById('count').textContent =
  localStorage.pageLoadCount;
```

```
localStorage.setItem("In", "Delhi");
```

```
cIndia = localStorage.getItem("In");
```

```
document.getElementById('cIn').textContent = cIndia;
```

```
//localStorage.clear();
```

```
</script>
```

```
</body>
<script>
    if (typeof(Storage) != "undefined") {
        localStorage.setItem("MyUSN", "2gi20cs181");
        document.getElementById("local").innerHTML =
            localStorage.getItem("MyUSN");
    }
    else {
        document.getElementById("local").innerHTML =
            "Sorry, your Browser does not support." ;
    }
</script>
</html>
```

Termwork 3

Name: Vinit Gunaki

USN: 2gi20cs181

Output:



this page was viewed 4 time(s).

Capital of India : Delhi



this page was viewed 5 time(s).

Capital of India : Delhi

Termwork 4

Form Validation using HTML5, JavaScript, angularJS,
Bootstrap.

Theory :

- JavaScript (JS) is a programming language that is one of the core technologies of the Web.
- A high-level, just-in-time compiled language.
- Java and JavaScript are 2 different languages based on syntax, Standard Libraries & Design.
- AngularJS was a javascript-based, open-source, front-end web framework for developing single page Application.
- It lets you use HTML as your template lang. and lets you extend HTML's syntax to express your application's components clearly and succinctly.
- Data Binding & dependency injection eliminate much of the code you would otherwise have to write.
- can be added to an HTML page using `<script>` tag.
- `ng-app` directive defines AngularJS application
 - `ng-model` directive binds the values of HTML controls to application data
 - `ng-bind` binds application data to HTML view.
 - `ng-init` initializes AngularJS application variables
- expressions are written ~~within~~ inside `{}{exp{}}`

Program:

```
<!DOCTYPE html>
<html>
<head>
<!-- CSS -->
<link rel="stylesheet" href="lib/bootstrapA.min.css">
<style>
body {
    padding-top: 30px;
}
.container {
    width: 50%;
    text-align: center;
    margin: 0 auto;
    padding: 20px;
    background: #f0eb8c;
}
</style>
<!-- JS -->
<script src="codes/angular.js"></script>
<script src="app.js"></script>
</head>
<body ng-app="validationApp" ng-controller="mainController">
<div class="container">
<!-- PAGE HEADER -->
<div class="page-header"><h1>Form validation </h1></div>
<!-- pass in the variable if our form is valid or invalid -->
<form name="userform" ng-submit="submitForm(userform.$valid)" novalidate>
```

```
<!-- Email -->
<div class = "form-group" ng-class = "{'has-error' : userForm.email.$invalid && !userForm.email.$pristine}">
    <label> Email </label>
    <input type = "email" name = "email" class = "form-control" ng-model = "user.email" required>
    <p ng-show = "userForm.email.$invalid && !userForm.email.$pristine" class = "help-block"> Enter a valid email. </p>
    <br>
    Password : <input type = "password" name = "pwd1" />
    Confirm &nbsp ; &nbsp ; <input type = "password" name = "pwd2" />
</div>
<button type = "submit" class = "btn btn-primary" onclick = "checkPass()"> Submit </button>
</form>
</div>
</body>
</html>
```

Termwork 4

Name: Vinit Gunaki

USN: 2gi20cs181

Output:

Form Validation

The image shows a sequence of three screenshots from a web application demonstrating form validation. The first screenshot shows a 'Form Validation' page with fields for Email and Password. The second screenshot shows a success message: 'This page says Fn checkPass: Password matched!' with an 'OK' button. The third screenshot shows another success message: 'This page says Email validated' with an 'OK' button.

Email

vinit.gunaki@gmail.com

Password: Confirm:

Submit

This page says
Fn checkPass: Password matched!

OK

Email

2pm

Password: Confirm:

Submit

This page says
Email validated

OK

Termwork 5

Implement AngularJS Routing and AngularJS controllers.

Theory :

- AngularJS controllers control the data of AngularJS application.
- ng-controller directive defines the application controllers.
- A controller is a JavaScript Object , created by a standard Javascript object constructor.
- ng-controller = "myCtrl" defines a controller myCtrl function is a JavaScript function.
- invoke the controller with \$scope object.
- AngularJS can validate input data.
- AngularJS also holds info. about whether they have been touched, or modified or not.
- There's also a Custom validation where you can manipulate your validation according to your code.
 - use my-directive and refer it by any object.

Program:

```
<!DOCTYPE html>
<script src="lib/angular.min.js"></script>
<script src="lib/angular-route.js"></script>
<body ng-app="myApp">
  <a href="#!/l1"> link1 </a>
  <a href="#!/l2"> link2 </a>
  <a href="#!/l3"> link3 </a>
  <div ng-view></div>
<script>
  var app = angular.module('myApp', ['ngRoute']);
  app.config(function($routeProvider) {
    $routeProvider
      .when('/l1', {
        template: "<h4> I am Big !! </h4>"
      })
      .when('/l2', {
        template: "<h2> I am Bigger !! </h2>"
      })
      .when('/l3', {
        template: "<h1> So what ?? I'm a bigger loser !! </h1>"
      })
    ;
  });
</script>
</body>
</html>
```

Termwork 5

Name: Vinit Gunaki

USN: 2gi20cs181

Output:

Link1 Link2 Link3

I am big!!

Link1 Link2 Link3

I am bigger!!

Link1 Link2 Link3

So what?? I'm a bigger loser!!

Termwork 6.

Implement UI DatePicker using Jquery

Theory :

- Jquery is a javascript library and simplifies Javascript programming
- jquery takes alot of common tasks that require many lines of JavaScript code to accomplish, and wraps them into methods that you can call with a single line of code.
- jquery library contains the following features :
 - HTML / DOM manipulation
 - CSS manipulation
 - HTML event methods
 - Effects and Animations
 - AJAX
 - Utilities
- jquery has plugins for almost any task out there.

Program:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title> JQuery UI Datepicker</title>
    <link href="lib/jquery-ui.css" rel="stylesheet">
    <script src="lib/jquery-1.10.2.js"></script>
    <script src="lib/angular-route.js"></script>
    <script src="lib/jquery-ui.js"></script>
    <script src="lib/angular.min.js"></script>
    <script src="lib/angular.js"></script>
    <script src="lib/bootstrap.min.js"></script>
    <script src="lib/bootstrap.css"></script>
    <script src="lib/bootstrapA.min.css"></script>
    <!-- Javascript -->
    <script>
        $(function() {
            $("#datepicker-1").datepicker({
                showOn: "button",
                buttonImage: "lib/calender-icon.png",
                buttonImageOnly: true
            });
        });
    </script>
    <style>
        body {
            padding-top: 30px;
        }
    </style>
```

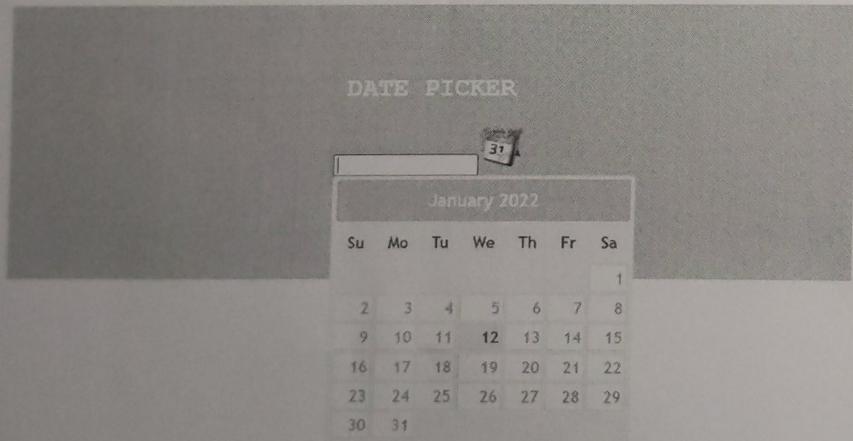
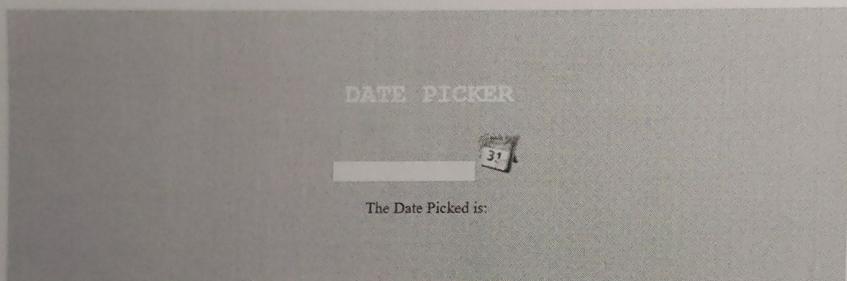
```
.container {  
    width: 50%;  
    text-align: center;  
    margin: auto;  
    padding: 50px;  
    background-color: deepskyblue;  
}  
  
h1 {  
    text-align: center;  
    font-family: monospace;  
    color: #FFFFFF;  
}  
</style>  
</head>  
<body>  
    <div class = "container" ng-app = "DemoApp" ng-controller = "DemoController">  
        <h1> Datepicker </h1>  
        <input type = "text" ng-model = "jDate" id = "datepicker-1">  
        <p> the date picked is </p>  
        <p ng-bind = "jDate"></p>  
    </div>  
    <script>  
        var app = angular.module ('', []);  
        app.controller ("DemoController", function ($scope) {  
            $scope.jDate = "";  
        });  
    </script>  
    </body>  
</html>
```

Termwork 6

Name: Vinit Gunaki

USN: 2gi20cs181

Output:



Teamwork 7

Implement Drag and Drop using HTML5 and Jquery.

Theory:

- Jquery is a javascript library and simplifies javascript programming.
- jquery takes alot of common tasks that require many lines of javascript code to accomplish, and wraps them into methods that you can call with a single line of code.
- jquery library contains the following features.
 - HTML / DOM manipulation
 - CSS Manipulation
 - HTML event methods
 - Effects and Animations
 - AJAX
 - Utilities
- jquery has plugins for almost any task out there.

Program :

```
<!DOCTYPE html>
<html>
<head>
<style>
#div1 {
    width: 250px;
    height: 250px;
    padding: 10px;
    border: 4px dashed #AASS10;
}
h1 {
    font-family: monospace;
    color: #11BB22;
    text-align: justify;
}
</style>
<script>
function allowDrop(ev) {
    ev.preventDefault();
}
function drop(ev) {
    ev.preventDefault();
    var data = ev.dataTransfer.getData("tgt");
    ev.target.appendChild(document.getElementById(data));
}
function drag(ev) {
    ev.dataTransfer.setData("tgt", ev.target.id);
}
</script>
```

```
</script>
</head>
<body>
<h1> Drag & Drop :the image into the Square Box </h1>
<div id = "div1" ondrop = "drop(event)"
      ondragover = 'allowDrop(event)'"></div>
<br>
<img id = "drg1" src = "Chelsea_FC.svg.png" draggable = "true" ondragstart =
      "drag(event)" width = "250", height = "250">
</body>
</html>
```

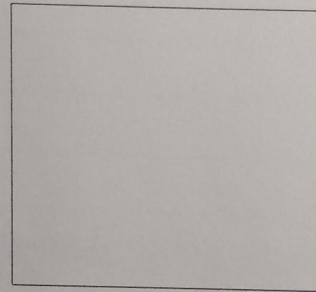
Termwork 7

Name: Vinit Gunaki

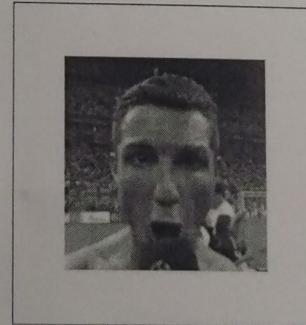
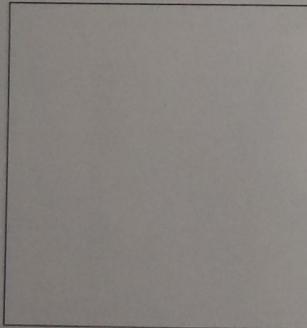
USN: 2gi20cs181

Output:

Drag the heading into the rectangle:



Drag the heading into the rectangle:



Termwork 8

Implement UI Autocomplete using Jquery.

Theory:

- Jquery is a javascript library and simplifies javascript programming
- jquery takes a lot of common tasks that require many lines of Javascript code to accomplish and wraps them into methods that you can call with a single line of code.
- Jquery library contains the following features.
 - HTML / DOM manipulation
 - CSS Manipulation
 - HTML Event methods
 - Effects and Animations
 - AJAX
 - Utilities
- Jquery has plugins for almost any task out there.

Program:

```
<!DOCTYPE html>
<html lang = "en">
<head>
    <meta charset = "utf-8">
    <meta name = "viewport" content = "width=device-width,
        initial-scale=1">
    <title>jQuery UI Autocomplete </title>
    <link href = "https://code.jquery.com/ui/1.10.4/themes/
        lightness/jquery-ui.css" rel = "stylesheet">
    <script src = "https://code.jquery.com/jquery-1.10.2.js"></script>
    <script src = "https://code.jquery.com/ui/1.10.4/jquery-ui.js"></script>
<script>
$(function() {
    var availableTags = [
        "Action Script",
        "AppleScript",
        "Asp",
        "BASIC",
        "C++",
        "C",
        "Clojure",
        "COBOL",
        "coldFusion",
        "Erlang",
        "Fortran",
        "groovy",
        "Haskell",
        "Java"
    ]
})
```

```
"JavaScript"
" Lisp"
" Perl"
" PHP"
"Python"
"Ruby"
"Scala"
"Scheme"
];
$( "#tags" ).autocomplete({
    source: availableTags
});
</script>
</head>
<body>
<center>
<div class = "ui-widget">
<h2> UI Autocomplete Using jQuery </h2>
<small> Type in the input box to get the autocomplete
dropdown </small>
<br />
<label for = "tags" > Tags: </label>
<input id = "tags" >
</div>
</center>
</body>
</html>
```

Termwork 8

Name: Vinit Gunaki

USN: 2gi20cs181

Output:

UI Autocomplete Using jQuery

Type in the input box to get the autocomplete dropdown

Tags:

- ActionScript
- AppleScript
- Asp
- BASIC
- Erlang
- Fortran
- Haskell
- Java
- JavaScript
- Scala

UI Autocomplete Using jQuery

Type in the input box to get the autocomplete dropdown

Tags:

- BASIC
- COBOL
- Ruby

UI Autocomplete Using jQuery

Type in the input box to get the autocomplete dropdown

Tags:

- Groovy
- Python
- Ruby

UI Autocomplete Using jQuery

Type in the input box to get the autocomplete dropdown

Tags: