STES's

NBN SINHGAD TECHNICAL INSTITUTES CAMPUS, AMBEGAON (BK), PUNE

Department of Information Technology

Sinhgad Institutes

LABORATORY MANUAL

(2019Course)

Web Application Development

TE-INFORMATION TECHLOGY

SEMESTER-II

Subject Code: 31448

TEACHING SCHEME	EXAMINATION SCHEME
Practical: 4Hrs/Week	Term Work: 50 Marks

Department of Information Technology,
NBN Sinhgad School of Engineering, Ambegaon (Bk.), Pune

List of Laboratory Assignments

<table>
<tr><th>Sr. No.</th><th>Title of Assignments</th></tr>
<tr><td>1.</td><td>Case study:
Before coding of the website, planning is important, students should visit different websites (Min. 5) for the different client projects and note down the evaluation results for these websites, either good website or bad website in following format:

From the evaluation, students should learn and conclude different website designissues, which should be considered while developing a website.

<table><tr><td>Sl. No</td><td>Website</td><td>Purpose</td><td>Things linked</td><td>Things disliked</td><td>Overall valuation of Website</td></tr></table>e evaluation, students should learn and conclude differe t website design</td></tr>
<tr><td>2.</td><td>Implement a web page index.htm for any client website (e.g., a restaurant website project) using following:
a. HTML syntax: heading tags, basic tags and attributes, frames, tables, images, lists, links for text and images, forms etc.
b. Use of Internal CSS, Inline CSS, External CSS</td></tr>
<tr><td>3.</td><td>Implement an application in Java Script using following:
a) Design UI of application using HTML, CSS etc.
b) Include Java script validation
c) Use of prompt and alert window using Java Script
e.g., Design and implement a simple calculator using Java Script for operations like addition, multiplication, subtraction, division, square of number etc.
a) Design calculator interface like text field for input and output, buttons for numbers and operators etc.
b) Validate input values
c) Prompt/alerts for invalid values etc.</td></tr>
<tr><td>4</td><td>Design an application using Angular JS.
e.g., Design registration (first name, last name, username, and password) and login page using Angular JS.</td></tr>
<tr><td>5</td><td>Mini Project: Design and implement a dynamic web application for any business functionality by using web development technologies that you have learnt in the above given assignments.</td></tr>
</table>

Assignment No.01

Title :

Case study:

Before coding of the website, planning is important, students should visit different websites (Min. 5) for the different client projects and note down the evaluation results for these websites, either good website or bad website in following format:

Sr. No. Website	Website	Purpose of Website	Thinks liked in the Website	Things disliked in the Website	Over all valuation of the Website

From the evaluation, students should learn and conclude different website design issues, which should be considered while developing a website.

Theory :

Different types of websites:

1. Blogs

Blogs have been a most loved and preferred choice of websites for individuals and families who wish to archive the huge events of their lives, for example, weddings, babies or their precious moments. Nowadays, the cutting option; blogging has gotten on the high tide and has become incredibly well known. Blogs are also great for presenting your personal style, sharing your favorite recipes or showcasing your hobbies or anything in that case to the world. Blogs are normally updated habitually, and more seasoned posts can be seen through archives.

2. Business/Corporate Websites

A business site is any site that is committed to depicting a particular business. It ought to be branded like the company (a similar logo and positioning) and convey the sorts of items as well asservices the company/ business offers. At this point, each company out there, whether big or small, ought to have a site. It's the need of the hour. Each potential client you come across will simply expect that in the event that they Google your business searching for more data, they'll discover a site. What's more, in case they don't, it makes the business look less professional or

legitimate.

3. NGO /Non-Profit Websites

Just like brands and businesses, Non –profit organizations and NGOs also require a website. Similarly, that organizations need sites to be their online nearness, not-for-profits do too. A nonprofit website is the most straightforward path for some potential contributors to give donations and to know more about an Organization to make their mind about whether or not willthey make the donation.

4. E-Commerce Websites

The websites where users can shop and makes purchases are called E-Commerce Websites. All of would have used these sites a myriad of times, and they have become a must-have for every selling business. With the advancement in website development technology, it has become relatively easy to build an E-Commerce site and start selling.

5. Educational Websites

The sites of educational organizations and those providing online courses fall into the class of educational sites. These sites have the essential objective of either giving study materials to visitors or giving data on an educational organization to them. Some educational sites will have ads like entertainment and media sites do. Some offer membership models or educational items for procurement. What's more, some fill in as the digital presence for an educational organization.

6. Entertainment Website

Entertainment sites contain consistently refreshed substance on current undertakings, climate, sports and amusement. News destinations like CNN and entertainment biggies like ELLE both fall into this classification. The content on media locales is dynamic, which means it's refreshed normally. Enormous media locales will distribute numerous articles and recordings each and every day. The majority of these sites do mean to make cash like business and e-commerce sites do, yet for the most part through the ads that appear on the page as opposed to through selling

explicit items or services.

7. Portfolio Website

Portfolio Websites are great for showcasing your work; everyone from photographers to models are utilizing such sites to flaunt their work to potential clients. Essentially utilized by those in the creative business, a portfolio site can be utilized like a CV, showing your abilities so as to dazzle customers, clients, or future prospects.

8. Social Media Websites

We are all aware of the giants Instagram and Facebook; who doesn't know of them? However, social media comes in various forms. These websites are normally made to let individuals share contemplations, pictures or thoughts, or just associate with others corresponding to a specific topic. Social media websites are progressively turning into the go-to goal for individuals to find out about the news, shop and connect with people. Social media is creating a substantial impact in every domain.

9. Forums

Network or Social Forums are an incredible method to connect and share thoughts with individuals with similar interests or from comparative foundations through the internet. This intelligent element can be the fundamental focal point of the site or only a piece of it with the goalthat guests can talk about what the site is about.

10. Web Portals

Online interfaces or Web Portals are sites intended for internal purposes at a business, association or an organisation. They gather data in various configurations from various sources into one spot to make all crucial data open to the individuals who need to see it. They frequently include a login and customised views for various users that guarantee the data that is open is generally helpful to their specific needs.

Following is a Survey of different Websites:

Sr. No. Website	Website	Purpose of Website	Thinks liked in the Website	Things disliked in the Website	Over all valuation of the Website

Conclusion : Hence, we have studied the different types of websites, understand purpose of website and also understand different websites design issues.

Assignment No. 02

Title :	Implement a web page index.htm for any client website (e.g., a restaurant website project) using following: a. HTML syntax: heading tags, basic tags and attributes, frames, tables, images, lists, links for text and images, forms etc. b. Use of Internal CSS, Inline CSS, External CSS
Objectives :	1. Understand about basic concepts of HTML 2. Understand the basic concepts of CSS
Problem Statement :	Design website for restaurant website project using html and css (inline, internal and external)
Outcomes:	*Students will be able to,* 1. Design static webpage using HTML. 2. Apply CSS to HTML pages.
Software Requirements:	Editor: Notepad++/vscode/sublime/atom etc., Browser: chrome/Firefox/IE
Theory:	**<u>HTML: HTML is the standard markup language for creating Web pages.</u>** ● HTML stands for Hyper Text Markup Language ● HTML describes the structure of Web pages using markup ● HTML elements are the building blocks of HTML pages ● HTML elements are represented by tags ● HTML tags label pieces of content such as "heading", "paragraph", "table", and so on ● Browsers do not display the HTML tags, but use them to render the content of the page ● HTML Versions: HTML - 1991 HTML 2.0 - 1995 HTML 3.2 - 1997 HTML 4.01 - 1999 XHTML - 2000 HTML 5 – 2014 **<u>CSS:</u>** CSS stands for Cascading Style Sheet. It is nothing, but design language intended to simplify the process of making web pages presentable.CSS handles the feel and look part of a web page. By using CSS, one can control the color of text, style of fonts, spacing between paragraphs, layout designs. CSS is easy to learn, easy to understand and it provides powerful

control on presentation of an HTML document.

Advantages of CSS:

It saves the time, Pages load faster**,** Easy maintenance, Superior styles to HTML, Multiple Device Compatibility, Global web standards**,** Offline Browsing, Platform Independence.

CSS3 Modules:

CSS3 Modules are having old CSS specifications as well as extension features.

- Box Model
- Selectors
- Background
- Border
- Image Values and Replaced Content
- Text Effects
- Animations
- 2D/3D Transformations
- Multiple Column Layout
- User Interface

HTML Document Structure:

The <!DOCTYPE html> declaration defines this document to be HTML5

The <html> element is the root element of an HTML page

The <head> element contains meta information about the documentThe

<title> element specifies a title for the document

The <body> element contains the visible page content

The <h1> element defines a large heading

The <p> element defines a paragraph

HTML tags are element names surrounded by angle brackets:
<tagname>content goes here...</tagname>

How to add CSS to HTML?

CSS can be added to HTML elements in 3 ways:

1. Inline - by using the style attribute in HTML elements. An inline CSS is used to apply a unique style to a single HTML element.
   Ex. <h1 style="color:blue;">This is a Blue Heading</h1>

2. Internal - by using a <style> element in the <head> section. An internal CSS is used to define a style for a single HTML page. An internal CSS is defined in the <head>section of an HTML page, within a <style> element.

Example:<style>

body {background-color: powderblue;}

h1 {color: blue;}

p {color: red;}

</style>

3. External - by using an external CSS file. An external style sheet is used to define the style for many HTML pages. With an external style sheet, you can change the look of an entireweb site, by changing one file!To use an external style sheet, add a link to it in the <head> section of the HTML page.

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

- Use the HTML **<head>**element to store <style> and <link> elements
- Use the CSS **color** property for text colors
- Use the CSS **font-family** property for text fonts
- Use the CSS **font-size** property for text sizes
- Use the CSS **border** property for borders
- Use the CSS **padding** property for space inside the border
- Use the CSS **margin** property for space outside the border

Conclusion :	Hence, we have designed web pages using HTML and CSS.
Questions :	1. What is the difference between HTML and HTML5? 2. What is the difference between html elements and tags? 3. What is marquee 4. What are tags? 5. What are some of the common lists that can be used when designing a page? 6. How do you insert a comment in HTML? 7. What is an image map? 8. How do you create links to sections within the same page? 9. What are style sheets? 10. State bullet types available in HTML 11. What are the limitations of CSS ? 12. What are the advantages of CSS ? 13. What are the differences between HTML and XML? 14. Which tag is used to find the version of XML and the syntax? 15. What is XML DOM Document? 16. What is an attribute? 17. What are the basic rules while writing XML?

	18. What is External style sheet? What are the advantages and disadvantages? 19. What is CSS selector? 20. What are the components of CSS style? 21. What is browser safe color? 22. Explain table tag with rowspan and colspan?

Lab Experiment 1

a. Create a responsive web page which shows the ecommerce/college/exam admin dashboard with sidebar and statistics in cards using HTML, CSS and Bootstrap.

Assignment No. 03

Title :	Implement an application in Java Script using following: a) Design UI of application using HTML, CSS etc. b) Include Java script validation c) Use of prompt and alert window using Java Script e.g., Design and implement a simple calculator using Java Script for operations like addition, Multiplication, subtraction, division, square of number etc. a) Design calculator interface like text field for input and output, buttons for numbers and Operators etc. b) Validate input values c) Prompt/alerts for invalid values etc.
Objectives :	1. Understand the basic concepts of JavaScript
Problem Statement :	Design and implement a simple calculator using Java Script for operations like addition, multiplication, subtraction, division, square of number etc.
Outcomes:	*Students will be able to,* 1. Design calculator using JavaScript 2. Validate input values 3. Prompt/alerts for invalid values etc.
Software Requirements:	Notepad, Any Browser

Theory:	**JavaScript**: JavaScript is a programming language of HTML as well web. It is preferred for creating network-centric applications. It is integrated and complimentary with Java. As JavaScript is integrated with HTML it is very easy to implement. It is open as well as cross-platform. **Advantages**: The advantages of using JavaScript are − ▪ It requires less server interaction ▪ Immediate feedback to the visitors ▪ Increased interactivity Richer interfaces JavaScript can be implemented using JavaScript statements that are placed within the <script>. You can place the <script> tags, containing your JavaScript, anywhere within your web page, but it is normally recommended that you should keep it within the

	<head> tags. The script tag takes two important attributes: • **Language** − This attribute specifies what scripting language you are using. Typically, its value will be JavaScript. Although recent versions of HTML (and XHTML, its successor) have phased out the use of this attribute. • **Type** − This attribute is what is now recommended to indicate the scripting language in use and its value should be set to "text/JavaScript" **JQuery:** jQuery is a lightweight, "write less, do more", JavaScript library. The purpose of jQuery is to make it much easier to use JavaScript on your website. 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 also simplifies a lot of the complicated things from JavaScript, like AJAX calls and DOM manipulation. The jQuery library contains the following features: • HTML/DOM manipulation • CSS manipulation • HTML event methods • Effects and animations • AJAX • Utilities
Conclusion :	Hence, we have designed calculator using JavaScript.

Questions :	1. Name some Java Script features. 2. How to define anonymous function? 3. What is callback? 4. What is the difference between undefined and not-defined in JavaScript? 5. What is 'closure' In JavaScript? 6. What are JavaScript data types? 7. How to create function in JavaScript? 8. What are the JavaScript data types? 9. How to write html code dynamically using JavaScript? 10. How to create objects in JavaScript? 11. What is jQuery & Why jQuery is needed? 12. Whether jQuery HTML work for both HTML and XML documents?

Lab Exercise 1b):

1b) Write a JavaScript Program to get the user registration data and push to array/local storage with AJAX POST method and data list in new page.

Assignment No. 04

Title :	Design an application using Angular JS. e.g., Design registration (first name, last name, username, password) and login page using Angular JS.
Objectives :	1. Understand the design of single-page applications and how AngularJS facilitates their development 2. Properly separate the model, view, and controller layers of your application and implement them using AngularJS 3. Master AngularJS expressions, filters, and scopes 4. Build Angular forms 5. Elegantly implement Ajax in your AngularJS applications 6. Write AngularJS directives
Problem Statement :	Design registration (first name, last name, username, password) and login page usingAngular JS.
Outcomes:	1. Implement the effective client side implementation. 2. Solve the complex problem of development using MVC framework
Software Requirements:	Eclipse / Netbeans IDE/ Notepad/ Notepad++, Modern Web browser

Theory:

AngularJS: is an open-source web application framework. It was initially created in 2009 by MiskoHevery and Adam Abrons. It is presently kept up by Google. Its most recent adaptation is 1.2.21. "AngularJS is an auxiliary system for dynamic web applications. It gives you a chance to utilize HTML as your layout dialect and gives you a chance to stretch out HTML's linguistic structure to express your application parts plainly and compactly. Its information official and reliance infusion take out a significant part of the code you as of now need to compose. Also, everything occurs inside the program, making it a perfect band together with any server innovation".

General Features

- AngularJS is a productive system that can make Rich Internet Applications (RIA).
- AngularJS gives designers a choices to compose customer side applications utilizing
- JavaScript in a spotless Model View Controller (MVC) way.
- Applications written in AngularJS are cross-program agreeable. AngularJS consequently
- Handles JavaScript code reasonable for every program.
- AngularJS is open source, totally free, and utilized by a great many engineers the world
- over. It is authorized under the Apache permit version2.0.
- ·By and large, AngularJS is a system to assemble expansive scale, elite, and simple take-up web applications.

Core Features:

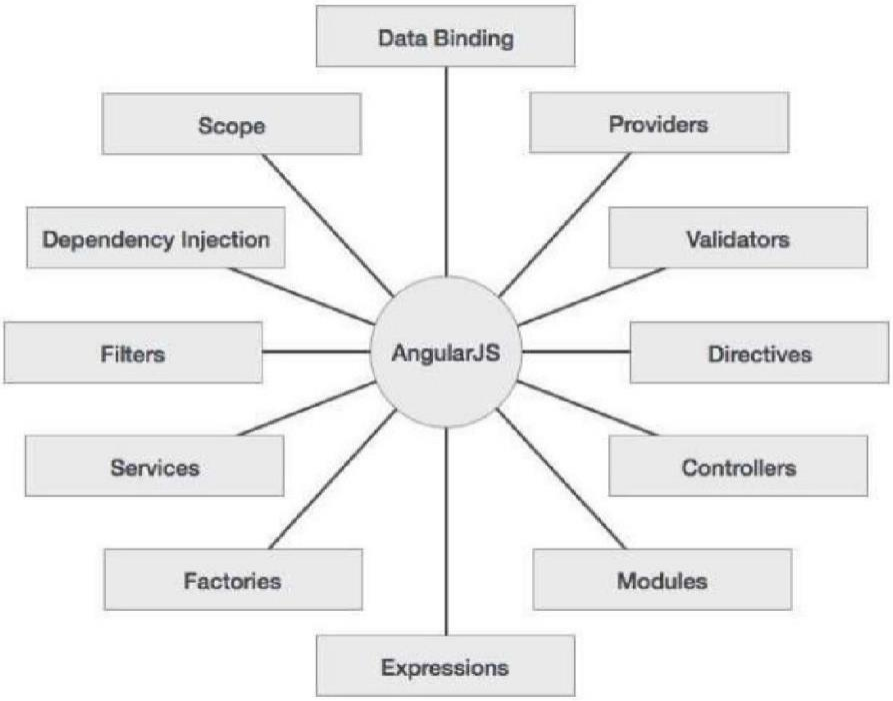


Figure.7: Architecture of AngularJS

1. **Data-authoritative:** It is the programmed synchronization of information amongst model and view parts.
2. **Scope:** These are objects that allude to the model. They go about as paste amongst controller and view.
3. **Controller:** These are JavaScript capacities bound to a specific degree.
4. **Services:** AngularJS accompanies a few implicit administrations, for example, $http to make aXMLHttpRequests. These are singleton objects which are instantiated just once in application.
5. **Filters:** These select a subset of things from a cluster and restore another exhibit.
6. **Directives:** Directives are markers on DOM components, for example, components, characteristics, css, and that's only the tip of the iceberg. These can be utilized to make custom HTML labels that fill in as new, custom gadgets. AngularJS has worked in mandates, for example, ngBind, ngModel, and so on.
7. **Templates:** These are the rendered see with data from the controller and model. These can be a solitary record, (for example, index.html) or different perspectives in a single page utilizing partials.
8. **Routing:** It is idea of exchanging sees**.**
9. **Model View Whatever:** MVW is an outline design for isolating an application into various parts called Model, View, and Controller, each with unmistakable obligations. AngularJS does not actualize MVC in the conventional sense, yet rather something nearer to MVVM (Model-View-ViewModel). The Angular JS group alludes it cleverly as Model View Whatever.
10. **Deep Linking:** Deep connecting permits to encode the condition of use in the URL with the goal that it can be bookmarked. The application would then be able to be reestablished from the URL to a similar state.
11. **Dependency Injection:** AngularJS has a worked in reliance infusion subsystem that encourages the designer to make, comprehend, and test the applications effectively.

Advantages of AngularJS

- It gives the ability to make Single Page Application in a spotless and viable way.
- It gives information restricting ability to HTML. Along these lines, it gives client a rich and responsive experience.
- AngularJS code is unit testable.
- AngularJS utilizations reliance infusion and make utilization of partition of concerns.
- AngularJS gives reusable segmentsWith AngularJS, the engineers can accomplish greater usefulness with short code.
- In AngularJS, sees are unadulterated html pages, and controllers written in JavaScript do the business handling.

Model View Controller

Model View Controller or MVC as it is famously called, is a product configuration design for creating web applications. A Model View Controller design is comprised of the accompanying three sections.

·Model − It is the most minimal level of the example in charge of looking after information.

□ View − It is in charge of showing all or a part of the information to the client.

□ Controller − It is a product Code that controls the connections between the Model and View.

MVC is mainstream since it secludes the application rationale from the UI layer and backings detachment of concerns. The controller gets all solicitations for the application and afterward works with the model to set up any information required by the view.

Model

The model is in charge of overseeing application information. It reacts to the demand from see and to thedirections from controller to refresh itself.

The View

An introduction of information in a specific arrangement, activated by the controller's choice to exhibit theinformation. They are content based layout frameworks, for example, JSP, ASP, PHP and simple to incorporatewith AJAX innovation.

The Controller

The controller reacts to client enter and performs communications on the information show objects. The controllergets input, approves it, and afterward performs business operations that alter the condition of the informationdemonstrate.

AngularJS is a MVC based structure.

- **An AngularJS application comprises of following three essential parts −ng-app − This directive**defines and links an AngularJS application to HTML.
- **ng-model** − This directive binds the values of AngularJS application data to HTML input controls.
- **ng-bind** − This directive binds the AngularJS Application data to HTML tags.

Algorithm:

Steps for AngularJS:

1. When a link https://angularjs.org/ is opened, there are two options to download AngularJS library –

- **View on GitHub** − Click on this button to go to GitHub and get all of the latest scripts.
- **Download AngularJS 1** − Or click on this button, a screen as below would be seen −

This screen gives various options of using Angular JS as follows:

- **Downloading and hosting files locally**

1. There are two different options **legacy** and **latest**. The names itself are selfdescriptive. **Legacy** has version less than 1.2.x and **latest** has 1.5.x version.
2. We can also go with the minified, uncompressed or zipped version.

- **CDN access** − You also have access to a CDN. The CDN will give you access around the world to regional data centers that in this case, Google host. This means using CDN moves the responsibility of hosting files from your own servers to a series of external ones. This also offers an advantage that if the visitor to your webpage has already downloaded a copy of AngularJS from the same CDN, it won't have to be redownloaded.

Example:

Now let us write a simple example using AngularJS library. Let us create an HTML file *myfirstexample.html* as below –

```
<!doctype html>
<html>
<head>
<scriptsrc="https://ajax.googleapis.com/ajax/libs/angularjs/1.5.2/angular.min.js"></script>
</head>
<bodyng-app="myapp">
<divng-controller="HelloController">
<h2>Welcome {{helloTo.title}} to the world of Tutorialspoint!</h2>
</div>
<script>
```

```
angular.module("myapp",[])
.controller("HelloController",function($scope){
$scope.helloTo={};
$scope.helloTo.title="AngularJS";
});
</script>
</body>
</html>
```

Following sections describe the above code in detail:

1. Include AngularJS

We have included the AngularJS JavaScript file in the HTML page so we can use AngularJS −

```
<head>
<scriptsrc="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
</head>
```

To update into latest version of Angular JS, use the following script source.

```
<head>
<scriptsrc="https://ajax.googleapis.com/ajax/libs/angularjs/1.5.2/angular.min.js"></script>
</head>
```

2. Point to AngularJS app

Next we tell what part of the HTML contains the AngularJS app. This done by adding the *ngapp* attribute to the root HTML element of the AngularJS app. You can either add it to *html* element or *body* element as shown below −

```
<bodyng-app="myapp">
</body>
```

3. View

The view is this part −

```
<divng-controller="HelloController">
<h2>Welcome {{helloTo.title}} to the world of Tutorialspoint!</h2>
</div>
```

ng-controller tells AngularJS what controller to use with this view. *helloTo.title*tells AngularJS to write the "model" value named helloTo.title to the HTML at this location.

4. Controller

The controller part is −

```
<script>
angular.module("myapp",[])
.controller("HelloController",function($scope){
$scope.helloTo={};
$scope.helloTo.title="AngularJS";
});
</script>
```

This code registers a controller function named *HelloController* in the angular module named *myapp*. The controller function is registered in angular via the angular.module(...).controller(...) function call.
The $scope parameter passed to the controller function is the *model*. The controller function adds a *helloTo* JavaScript object, and in that object it adds a *title* field.

5. Execution
Save the above code as *myfirstexample.html* and open it in any browser.
Output as below:
Welcome AngularJS
· At the point when the page is stacked in the program, following things happen −
· HTML archive is stacked into the program, and assessed by the program. AngularJS JavaScript document is stacked, the precise worldwide question is made. Next, JavaScript which registers controller capacities is executed.

Next AngularJS look over the HTML to search for AngularJS applications and perspectives. When see is found, it associates that view to the comparing controller work.
· Next, AngularJS executes the controller capacities. It at that point renders the perspectives with information from the model populated by the controller. The page is presently prepared.

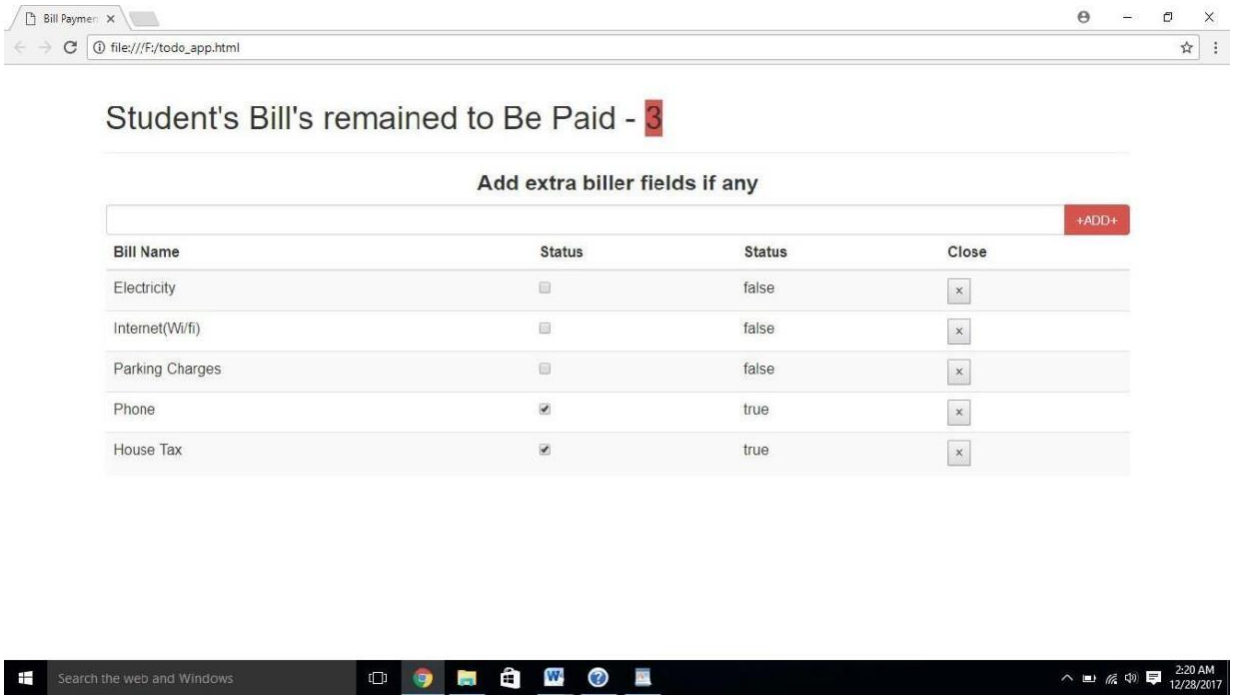
6. How AngularJS integrates with HTML
· ng-app directive indicates the start of AngularJS application.
· ng-model directive then creates a model variable named "name" which can be used with the html page and within the div having ng-app directive.
· ng-bind then uses the name model to be displayed in the html span tag whenever user input something in the text box.
· Closing tag indicates the end of AngularJS application.
AngularJS directives are used to extend HTML. These are special attributes starting with ngprefix.
We're going to discuss following directives −
· **ng-app** − This directive starts an AngularJS Application.
· **ng-init** − This directive initializes application data.
· **ng-model** − This directive binds the values of AngularJS application data to HTML input controls.
· **ng-repeat** − This directive repeats html elements for each item in a collection.
Deploy the Html program run test the result for dynamic implementation of AngularJS.

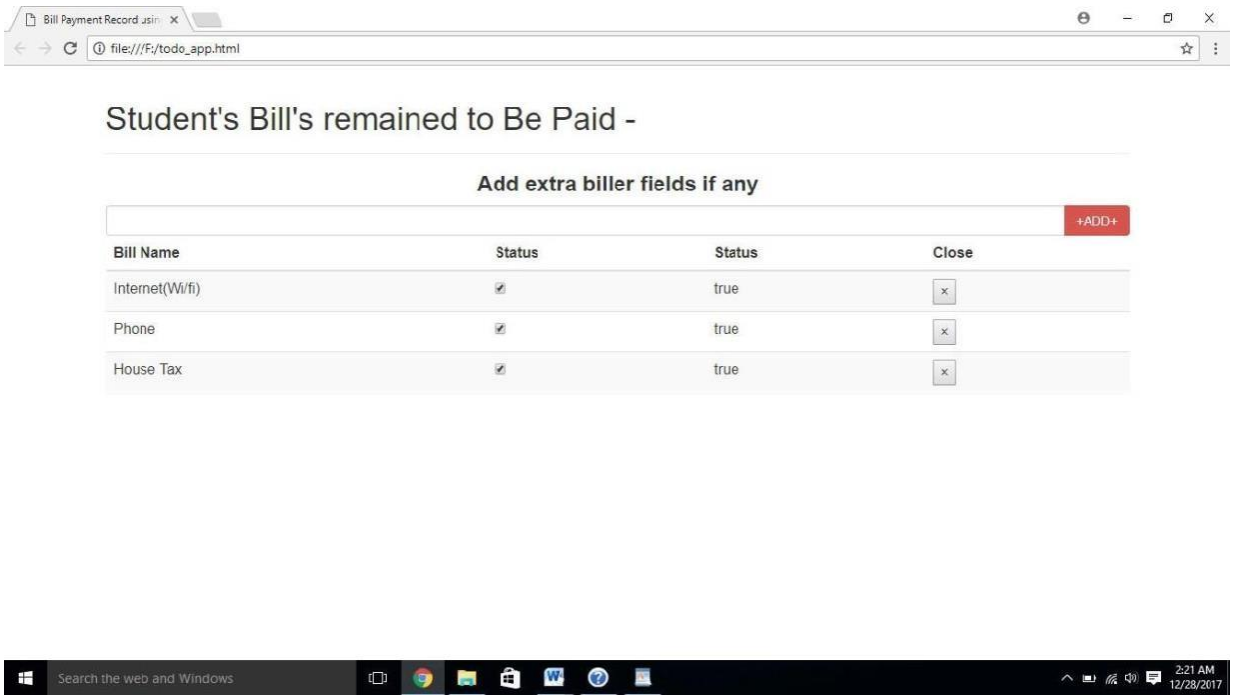
```
<html ng-app="billpayApp">
<!-- SCRIPTS TO BE ADDED IN HEAD TAG -->
<head>
<title>Bill Payment Record using angular and bootstram
framework</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<!-- ACCESSING ANGULARJS BY CDN METHOD-->
```

```
<script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.4/angular.mi
n
.js"></script>
<!-- ACCESSING STYLESHEET FOR DESIGN [OPTIONAL PART CAN BE SKIP]-->
<link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.m
i
n.css">
<!-- MODEL PART-->
<script>
var model = {
customer: "Student",
items: [{
bill: "Electricity",
status: false
},
{
bill: "Internet(Wi/fi)",
status: false
},
{
bill: "Parking Charges",
status: false
},
{
bill: "Phone",
status: true
},
{
bill: "House Tax",
status: true
}
]
}
varbillpayApp = angular.module("billpayApp", []);
billpayApp.controller("billpayctrl", function($scope) {
$scope.billpay = model;
$scope.dueBills = function() {
var items = $scope.billpay.items;
var counter = 0;
items.forEach((item) => {
if (!item.status) {
counter++;
}
})
return counter;
}
$scope.redFlag = function() {
return $scope.dueBills() <= 2 ? "label-success" : "label-danger";
```

```
}
$scope.addBills = function(billName) {
obj = {
bill: billName,
status: false
}
$scope.billpay.items.push(obj);
}
$scope.removeBills = function(rmvBills) {
$scope.billpay.items.splice($scope.billpay.items.indexOf(rmvBills),
1);
}
});
</script>
</head>
<!-- HTML BODY PART-->
<body ng-controller="billpayctrl">
<div class="container">
<div class="page-header">
<h1>{{billpay.customer}}'s Bill's remained to Be Paid -
<span class="lable" ng-class="redFlag()" ng-hide="dueBills()==0">
{{dueBills()}}
</span>
</h1>
</div>
<h3><center><b>Add extra biller fields if any</center></b></h3>
<div class="panel">
<div class="input-group">
<input class="form-control" ng-model="billName" />
<span class="input-group-btn">
<button class="btnbtn-danger" ngclick="
addBills(billName)">+ADD+</button>
</span>
</div>
<table class="table table-striped">
<thead>
<tr>
<th>Bill Name</th>
<th>Status</th>
<th>Status</th>
<th>Close</th>
</tr>
</thead>
<tbodyng-model="rmvBills">
<trng-repeat="item in billpay.items" ng-model="item">
<td>{{item.bill}}</td>
<td><input type="checkbox" ng-model="item.status" /></td>
<td>{{item.status}}</td>
<td>
<button type="button" ng-click="removeBills(item)">&times;</button>
```

```
</td>
</tr>
</tbody>
</table>
</div>
</div>
</div>
</body>
</html>
```





Conclusion :	With the help of this assignment it is helpful to understand features of AngularJS. MVC model structure and its use in advanced web programming is studied.
Questions :	1. What is AngularJS and what are some of its advantages? 2. What is the Model View Controller (MVC)? 3. What is data binding in AngularJS? How does it relate to the MVC architecture? 4. Explain the concept of scope. How does scope inheritance work in AngularJS? 5. Explain the difference between a factory and a service in AngularJS. 6. Explain why there are two "destroy" events associated with the termination of a scope in AngularJS. 7. What is dependency injection and how does it work? 8. What are directives? Can you explain the functions of the following directives? 9. Explain the role of $routeProvider in AngularJS. As Per Syllabus: (Here Assignment means Lab Exercises to be implemented and get it correction with the Execution Status) **Assignment 2** **a.** Create version control account on GitHub and using Git commands to create repository and push your code to GitHub. **b.** Create Docker Container Environment (NVIDEIA Docker or any other). **c.** Create an Angular application which will do following actions: Register User, Login User, Show UserData on Profile Component **Assignment 3** **a.** Create a Node.JS Application which serves a static website. **b.** Create four API using Node.JS, ExpressJS and MongoDB for CURD Operations on assignment 2.C. **Assignment 4** **a.** Create a simple Mobile Website using jQuery Mobile. **b.** Deploy/Host your web application on AWS VPC or AWS Elastic Beanstalk. Mini Project

Assignment No. 5

Title :	**Mini Project**: Design and implement a dynamic web application for any business functionality by using web development technologies that you have learnt in the above given assignments.

Note:	Attach your Mini Project Report Here. (Page 1) Develop a web application using full stack development technologies in any of the following domains: **1.** Social Media **2.** ecommerce **3.** Restaurant **4.** Medical **5.** Finance **6.** Education Any other

i

Note:	Attach your Mini Project Report Here. (Page 2)

Note:	Attach your Mini Project Report Here. (3)

As Per Syllabus: (Here Assignment means Lab Exercises to be implemented and get it correction with the Execution Status)

Assignment 2

a. Create version control account on GitHub and using Git commands to create repository and pushyour code to GitHub.

b. Create Docker Container Environment (NVIDEIA Docker or any other).

c. Create an Angular application which will do following actions: Register User, Login User, Show UserData on Profile Component

Assignment 3

a. Create a Node.JS Application which serves a static website.

b. Create four API using Node.JS, ExpressJS and MongoDB for CURD Operations on assignment 2.C.

Assignment 4

a. Create a simple Mobile Website using jQuery Mobile.

b. Deploy/Host your web application on AWS VPC or AWS Elastic Beanstalk. Mini Project