



EAST WEST INSTITUTE OF TECHNOLOGY

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING



LABORATORY MANUAL

ANGULAR JS



V SEMESTER

Prepared By:

Ms. K S Smitha

APPROVED BY

**Dr. Suresh M. B
Professor & Head,
Dept. of ISE, EWIT**

**Dr. K. Channakeshavalu
Principal & Director
EWIT, Bangalore.**

Name:

USN: *Branch:*

Section: *Batch:*

PROGRAMMING EXPERIMENTS

1. Develop Angular JS program that allows user to input their first name and last name and display their full name. Note: The default values for first name and last name may be included in the program.
2. Develop an Angular JS application that displays a list of shopping items. Allow users to add and remove items from the list using directives and controllers. Note: The default values of items may be included in the program.
3. Develop a simple Angular JS calculator application that can perform basic mathematical operations (addition, subtraction, multiplication, division) based on user input.
4. Write an Angular JS application that can calculate factorial and compute square based on given user input.
5. Develop AngularJS application that displays a details of students and their CGPA. Allow users to read the number of students and display the count. Note: Student details may be included in the program.
6. Develop an AngularJS program to create a simple to-do list application. Allow users to add, edit, and delete tasks. Note: The default values for tasks may be included in the program.
7. Write an AngularJS program to create a simple CRUD application (Create, Read, Update, and Delete) for managing users.
8. Develop AngularJS program to create a login form, with validation for the username and password fields.
9. Create an AngularJS application that displays a list of employees and their salaries. Allow users to search for employees by name and salary. Note: Employee details may be included in the program.
10. Create AngularJS application that allows users to maintain a collection of items. The application should display the current total number of items, and this count should automatically update as items are added or removed. Users should be able to add items to the collection and remove them as needed. Note: The default values for items may be included in the program.
11. Create AngularJS application to convert student details to Uppercase using angular filters. Note: The default details of students may be included in the program.
12. Create an AngularJS application that displays the date by using date filter parameter

1. Develop Angular JS program that allows user to input their first name and last name and display their full name. Note: The default values for first name and last name may be included in the program.

index.html

```
<!DOCTYPE html>
<html lang="en" ng-app="nameApp">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Full Name App</title>
  <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.7.9/angular.min.js"></script>
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 20px;
    }

    label {
      display: block;
      margin-bottom: 5px;
    }

    input {
      width: 10%;
      padding: 8px;
      margin-bottom: 10px;
      box-sizing: border-box;
    }

    button {
      padding: 10px;
      background-color: #4CAF50;
      color: white;
      border: none;
      border-radius: 4px;
      cursor: pointer;
    }

    button:hover {
      background-color: #45a049;
    }

    p {
      margin-top: 10px;
    }
  </style>
```

```
</head>
<body>
  <div ng-controller="NameController">
    <label for="firstName">First Name:</label>
    <input type="text" id="firstName" ng-model="firstName" />

    <label for="lastName">Last Name:</label>
    <input type="text" id="lastName" ng-model="lastName" />

    <button ng-click="displayFullName()">Display Full Name</button>

    <p ng-if="fullName">Full Name: {{ fullName }}</p>
  </div>
  <script src="p1.js"></script>
</body>
</html>
```

p1.js

```
angular.module('nameApp', [])
.controller('NameController', function ($scope) {

  // Default values for full name

  $scope.firstName = 'John';
  $scope.lastName = 'Doe';
  $scope.fullName = "";

  // Function to display full name

  $scope.displayFullName = function () {
    $scope.fullName = $scope.firstName + ' ' + $scope.lastName;
  };
});
```

Output

First Name:

Last Name:

Full Name: John Doe

2. Develop an Angular JS application that displays a list of shopping items. Allow users to add and remove items from the list using directives and controllers. Note: The default values of items may be included in the program.

index.html

```
<!DOCTYPE html>
<html lang="en" ng-app="shoppingApp">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Shopping List App</title>
  <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.7.9/angular.min.js"></script>
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 20px;
    }

    input {
      width: 10%;
      padding: 8px;
      margin-bottom: 10px;
      box-sizing: border-box;
    }

    h2 {
      color: #333;
    }

    ul {
      list-style-type: none;
      padding: 0;
    }

    li {
      display: flex;
      align-items: center;
      justify-content: space-between;
      border: 1px solid #ccc;
      padding: 10px;
      margin-bottom: 5px;
      width: 300px;
    }

    button {
      background-color: #d9534f;
      color: white;
```

```

        border: none;
        padding: 5px 10px;
        cursor: pointer;
    }

    button:hover {
        background-color: #c9302c;
    }

    input {
        padding: 8px;
        margin-right: 10px;
    }

    input[type="text"] {
        width: 200px;
    }

    input[type="text"],
    button {
        margin-bottom: 10px;
    }
</style>
</head>
<body>
    <div ng-controller="ShoppingController">
        <h2>Shopping List</h2>
        <ul>
            <li ng-repeat="item in shoppingItems">
                {{ item }}
                <button ng-click="removeItem($index)">Remove</button>
            </li>
        </ul>
        <input type="text" ng-model="newItem" />
        <button ng-click="addItem()">Add Item</button>
    </div>
    <script src="p2.js"></script>
</body>
</html>

```

p2.js

```

angular.module('shoppingApp', [])
.controller('ShoppingController', function ($scope) {
    // Default list of items
    $scope.shoppingItems = ['Apple', 'Banana', 'Orange'];
    // Function to add a new item
    $scope.addItem = function () { if ($scope.newItem) {

```

```
$scope.shoppingItems.push($scope.newItem);
$scope.newItem = ""; // Clear the input field after adding
}
};
// Function to remove an item
$scope.removeItem = function (index) {
$scope.shoppingItems.splice(index, 1);
};
});
```

Output:

Shopping List

Apple	Remove
Banana	Remove
Orange	Remove
<input type="text"/>	Add Item

3. Develop a simple Angular JS calculator application that can perform basic mathematical operations (addition, subtraction, multiplication, division) based on user input.

```
<!DOCTYPE html>
<html lang='en' ng-app='myapp'>
<head>
  <title>Calculator App</title>
  <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>
  <style>
    body {
      font-family: Arial, sans-serif;
      background-color: #f4f4f4;
      text-align: center;
      margin: 20px;
    }

    h1 {
      color: #333;
    }

    input[type='text'] {
      padding: 8px;
      font-size: 16px;
      width: 200px;
    }

    input[type='button'] {
      padding: 10px;
      font-size: 16px;
      margin: 5px;
      cursor: pointer;
      background-color: #4caf50;
      color: white;
      border: none;
      border-radius: 5px;
    }

    input[type='button']:hover {
      background-color: #45a049;
    }
  </style>
</head>
<body ng-controller='myctrl'>
  <h1>Simple Calculator</h1>

  First Number<br>
  <input type='text' ng-model='no1'><br>
  Second Number<br>
  <input type='text' ng-model='no2'><br><br>

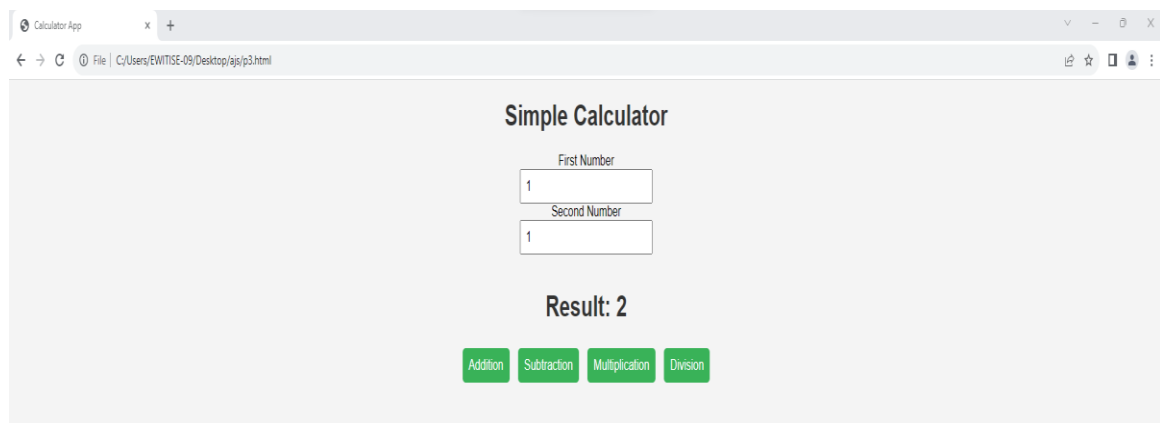
  <h1>Result: {{ result }}</h1>
```



```
<input type='button' value="Addition" ng-click="calculateOp('+')">
<input type='button' value="Subtraction" ng-click="calculateOp('-')">
<input type='button' value="Multiplication" ng-click="calculateOp('*')">
<input type='button' value="Division" ng-click="calculateOp('/')">

<script>
var app = angular.module('myapp', []);
app.controller('myctrl', function ($scope) {
  $scope.calculateOp = function (op) {
    switch (op) {
      case '+':
        $scope.result = parseInt($scope.no1) + parseInt($scope.no2);
        break;
      case '-':
        $scope.result = parseInt($scope.no1) - parseInt($scope.no2);
        break;
      case '*':
        $scope.result = parseInt($scope.no1) * parseInt($scope.no2);
        break;
      case '/':
        $scope.result = parseInt($scope.no1) / parseInt($scope.no2);
        break;
    }
  };
});
</script>
</body>
</html>
```

Output:



4. Write an Angular JS application that can calculate factorial and compute square based on given user input.

```
<!DOCTYPE html>

<html lang="en" ng-app="calculatorApp">

<head>

<meta charset="UTF-8">

<title>AngularJS Calculator</title>

<style>

body {

font-family: Arial, sans-serif;

margin: 20px;

text-align: center;

}

h2 {

color: #333;

}

label {

display: block;

margin-top: 10px;

}

input {

padding: 5px;

margin-bottom: 10px;

}

button {

padding: 8px;

background-color: #4CAF50;

color: white;

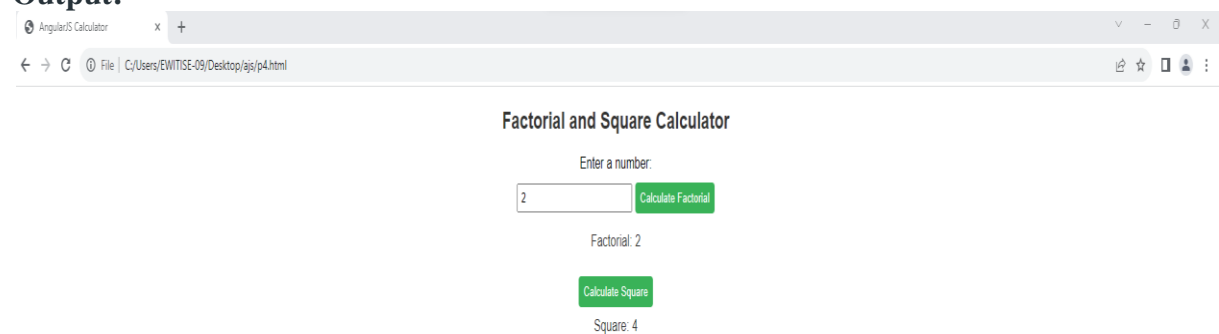
border: none;

border-radius: 4px;
```

```
cursor: pointer;
margin-top: 10px;
}
p {
margin-top: 10px;
color: #333;
}
</style>
<script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.7.9/angular.min.js"></script>
</head>
<body>
<div ng-controller="CalculatorController">
<h2>Factorial and Square Calculator</h2>
<label for="numberInput">Enter a number:</label>
<input type="number" id="numberInput" ng-model="userInput" />
<button ng-click="calculateFactorial()">Calculate Factorial</button>
<p ng-if="factorialResult !== undefined">Factorial: { { factorialResult } }</p>
<button ng-click="calculateSquare()">Calculate Square</button>
<p ng-if="squareResult !== undefined">Square: { { squareResult } }</p>
</div>
<script>
var app = angular.module('calculatorApp', []);
app.controller('CalculatorController', function($scope) {
$scope.userInput = 0;
$scope.factorialResult = undefined;
$scope.squareResult = undefined;
$scope.calculateFactorial = function() {
$scope.factorialResult = factorial($scope.userInput);
};
```

```
$scope.calculateSquare = function() {  
  
$scope.squareResult = square($scope.userInput);  
  
};  
  
function factorial(n) {  
if (n === 0 || n === 1) {  
return 1;  
} else {  
return n * factorial(n - 1);  
}  
}  
  
function square(n) {  
return n * n;  
}  
  
});  
</script>  
</body>  
</html>
```

Output:



AngularJS Calculator x +

File | C:/Users/EWITISE-09/Desktop/ajs/p4.html

Factorial and Square Calculator

Enter a number:

Calculate Factorial

Factorial: 2

Calculate Square

Square: 4

5. Develop AngularJS application that displays a details of students and their CGPA. Allow users to read the number of students and display the count. Note: Student details may be included in the program.

```
<!DOCTYPE html>
<html lang="en" ng-app="studentApp">

<head>
  <meta charset="UTF-8">
  <title>Student Details App</title>
  <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.0/angular.min.js"></script>
  <style>
    body {
      font-family: Arial, sans-serif;
      background-color: #f4f4f4;
      margin: 0;
      padding: 0;
      display: flex;
      align-items: center;
      justify-content: center;
      height: 100vh;
    }

    .container {
      background-color: #fff;
      box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
      padding: 20px;
      border-radius: 5px;
      text-align: center;
    }

    ul {
      list-style-type: none;
      padding: 0;
    }

    li {
      margin-bottom: 10px;
      padding: 10px;
      border: 1px solid #ddd;
      border-radius: 5px;
      background-color: #fff;
    }

    label {
      margin-right: 10px;
    }

    input {
      padding: 8px;
      margin-bottom: 10px;
      border-radius: 5px;
    }
```

```

        border: 1px solid #ddd;
    }
</style>
</head>

<body>
    <div ng-controller="StudentController" class="container">
        <h2>Student Details</h2>
        <label for="studentCount">Enter the number of students:</label>
        <input type="number" id="studentCount" ng-model="numberOfStudents" ng-
change="updateStudentList()">
        <ul>
            <li ng-repeat="student in students | limitTo:numberOfStudents">
                {{ student.name }} - CGPA: {{ student.cgpa }}
            </li>
        </ul>
        <p>Total number of students: {{ students.length }}</p>
    </div>

    <script>
    angular.module('studentApp', [])
        .controller('StudentController', function ($scope) {
            $scope.students = [
                { name: 'John Doe', cgpa: 3.5 },
                { name: 'Jane Smith', cgpa: 4.0 },
                { name: 'Bob Johnson', cgpa: 3.2 },
                // Include more student details as needed
            ];

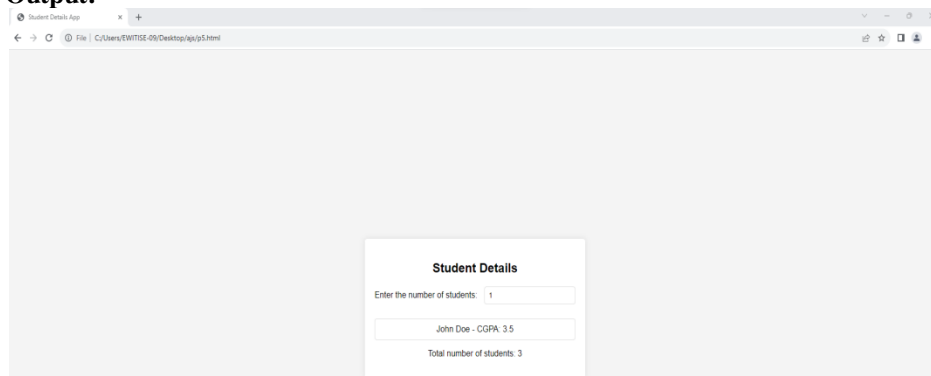
            $scope.updateStudentList = function () {
                $scope.numberOfStudents = Math.max(0, Math.floor($scope.numberOfStudents));
            };

            $scope.updateStudentList();
        });
    </script>
</body>

</html>

```

Output:



6. Develop an AngularJS program to create a simple to-do list application. Allow users to add, edit, and delete tasks. Note: The default values for tasks may be included in the program.

```
<!DOCTYPE html>
<html lang="en" ng-app="todoApp">

<head>
  <meta charset="UTF-8">
  <title>To-Do List App</title>
  <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.0/angular.min.js"></script>
  <style>
    body {
      font-family: Arial, sans-serif;
      background-color: #f4f4f4;
      margin: 0;
      padding: 0;
      display: flex;
      align-items: center;
      justify-content: center;
      height: 100vh;
    }

    .container {
      background-color: #fff;
      box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
      padding: 20px;
      border-radius: 5px;
      text-align: center;
    }

    ul {
      list-style-type: none;
      padding: 0;
    }

    li {
      margin-bottom: 10px;
      padding: 10px;
      border: 1px solid #ddd;
      border-radius: 5px;
      background-color: #fff;
      display: flex;
      justify-content: space-between;
      align-items: center;
    }

    input[type="text"] {
```

```

padding: 8px;
border-radius: 5px;
border: 1px solid #ddd;
}

button {
margin-left: 5px;
background-color: #4caf50;
color: white;
border: none;
border-radius: 3px;
padding: 8px;
cursor: pointer;
}

button.edit {
background-color: #2196F3;
}

button.delete {
background-color: #f44336;
}
</style>
</head>

<body>
<div ng-controller="TodoController" class="container">
<h2>To-Do List</h2>
<input type="text" ng-model="newTask" placeholder="Add a new task">
<button ng-click="addTask()">Add</button>

<ul>
<li ng-repeat="task in tasks">
{{ task.name }}
<button class="edit" ng-click="editTask($index)">Edit</button>
<button class="delete" ng-click="deleteTask($index)">Delete</button>
</li>
</ul>
</div>

<script>
angular.module('todoApp', [])
.controller('TodoController', function ($scope) {
$scope.tasks = [
{ name: 'Task 1' },
{ name: 'Task 2' },
{ name: 'Task 3' },
];

```



```

$scope.addTask = function () {
    if ($scope.newTask) {
        $scope.tasks.push({ name: $scope.newTask });
        $scope.newTask = "";
    }
};

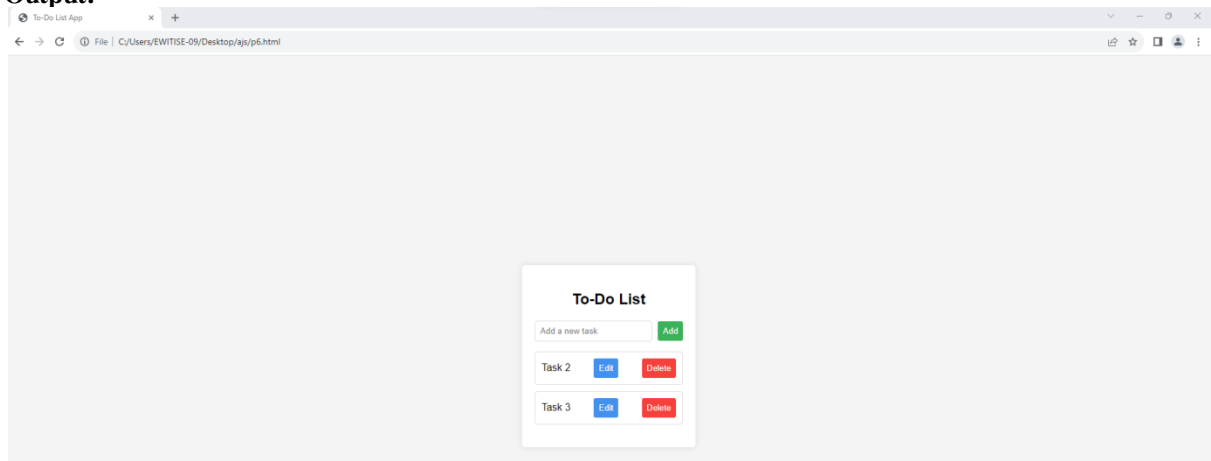
$scope.editTask = function (index) {
    var editedTask = prompt('Edit task:', $scope.tasks[index].name);
    if (editedTask !== null) {
        $scope.tasks[index].name = editedTask;
    }
};

$scope.deleteTask = function (index) {
    var confirmDelete = confirm('Are you sure you want to delete this task?');
    if (confirmDelete) {
        $scope.tasks.splice(index, 1);
    }
};
});
</script>
</body>

</html>

```

Output:



```
<!DOCTYPE html>
<html lang="en" ng-app="crudApp">

<head>
  <meta charset="UTF-8">
  <title>User CRUD App</title>
  <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.0/angular.min.js"></script>
  <style>
    body {
      font-family: Arial, sans-serif;
      background-color: #f4f4f4;
      margin: 0;
      padding: 0;
      display: flex;
      align-items: center;
      justify-content: center;
      height: 100vh;
    }

    h2 {
      text-align: center;
      color: #333;
    }

    ul {
      list-style-type: none;
      padding: 0;
      margin: 0;
    }

    li {
      margin-bottom: 10px;
      padding: 10px;
      border: 1px solid #ddd;
      border-radius: 5px;
      background-color: #fff;
      display: flex;
      justify-content: space-between;
      align-items: center;
    }

    li span {
      flex-grow: 1;
    }
  </style>
</head>

<body>
  <div>
    <h2>User CRUD App</h2>
    <ul>
      <li><span><input type="text" value="Name" /></span><input type="button" value="Add" /></li>
      <li><span><input type="text" value="Email" /></span><input type="button" value="Add" /></li>
      <li><span><input type="text" value="Phone" /></span><input type="button" value="Add" /></li>
      <li><span><input type="text" value="Address" /></span><input type="button" value="Add" /></li>
    </ul>
  </div>
</body>
</html>
```

```

    }

    form {
        margin-top: 20px;
        display: flex;
        flex-direction: column;
        align-items: center;
    }

    label {
        margin-bottom: 5px;
        color: #333;
    }

    input {
        padding: 8px;
        border-radius: 5px;
        border: 1px solid #ddd;
        margin-bottom: 10px;
    }

    button {
        background-color: #4caf50;
        color: white;
        border: none;
        border-radius: 3px;
        padding: 8px;
        cursor: pointer;
        margin-top: 10px;
    }

    button.edit {
        background-color: #2196F3;
    }

    button.delete {
        background-color: #f44336;
    }
</style>
</head>

<body ng-controller="UserController">
    <h2>User CRUD App</h2>
    <!-- Display Users -->
    <ul>
        <li ng-repeat="user in users">
            <span>{{ user.name }} - {{ user.email }}</span>
            <button class="edit" ng-click="editUser(user)">Edit</button>
        </li>
    </ul>

```

```

        <button class="delete" ng-click="deleteUser(user)">Delete</button>
    </li>
</ul>
<!-- Add or Update User Form -->
<form ng-submit="saveUser()">
    <label>Name:</label>
    <input type="text" ng-model="currentUser.name" required>

    <label>Email:</label>
    <input type="email" ng-model="currentUser.email" required>
    <button type="submit">{{ isEditMode ? 'Update' : 'Add' }} User</button>
</form>

<script>
angular.module('crudApp', [])
.controller('UserController', function ($scope) {
    $scope.users = [
        { id: 1, name: 'John Doe', email: 'john@example.com' },
        { id: 2, name: 'Jane Smith', email: 'jane@example.com' },
        // Add more users as needed
    ];
    $scope.currentUser = { };
    $scope.isEditMode = false;

    $scope.editUser = function (user) {
        $scope.currentUser = angular.copy(user);
        $scope.isEditMode = true;
    };

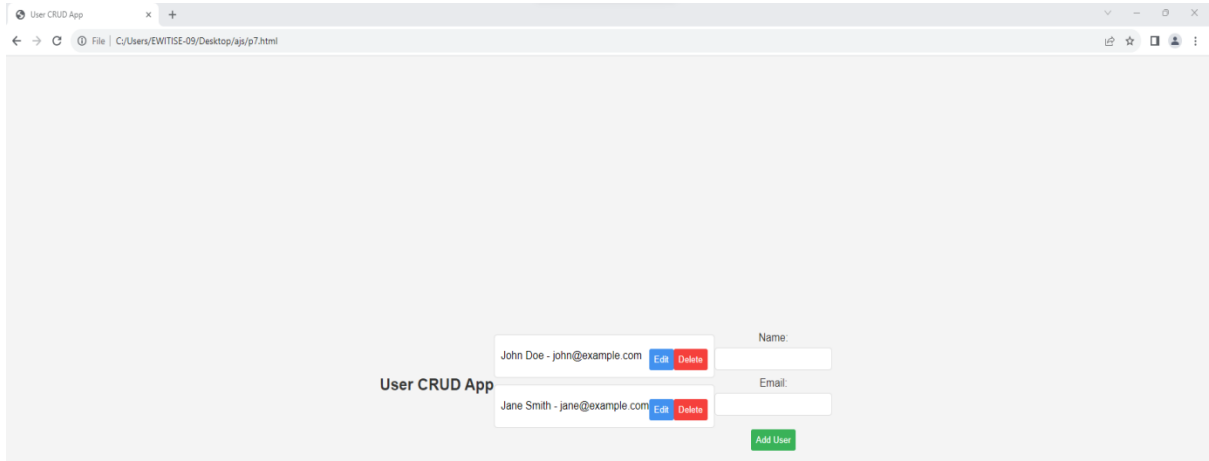
    $scope.saveUser = function () {
        if ($scope.isEditMode) {
            // Update existing user
            var index = $scope.users.findIndex(u => u.id === $scope.currentUser.id);
            $scope.users[index] = angular.copy($scope.currentUser);
        } else {
            // Add new user
            $scope.currentUser.id = $scope.users.length + 1;
            $scope.users.push(angular.copy($scope.currentUser));
        }
        // Clear form and reset mode
        $scope.currentUser = { };
        $scope.isEditMode = false;
    };

    $scope.deleteUser = function (user) {
        var index = $scope.users.indexOf(user);
        $scope.users.splice(index, 1);
    };
});

```

```
});  
</script>  
</body>  
  
</html>
```

Output:



8. Develop AngularJS program to create a login form, with validation for the username and password fields.

```
<!DOCTYPE html>
<html lang="en" ng-app="loginApp">

<head>
  <meta charset="UTF-8">
  <title>Login Form</title>
  <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.0/angular.min.js"></script>
  <style>
    body {
      font-family: Arial, sans-serif;
      background-color: #f4f4f4;
      margin: 0;
      padding: 0;
      display: flex;
      align-items: center;
      justify-content: center;
      height: 100vh;
    }

    .login-container {
      background-color: #fff;
      box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
      padding: 20px;
      border-radius: 5px;
      text-align: center;
    }

    label {
      display: block;
      margin-bottom: 5px;
      color: #333;
    }

    input {
      width: 100%;
      padding: 8px;
      border-radius: 5px;
      border: 1px solid #ddd;
      margin-bottom: 10px;
    }

    button {
      background-color: #4caf50;
      color: white;
      border: none;
      border-radius: 3px;
```

```

        padding: 10px;
        cursor: pointer;
    }

    button:hover {
        background-color: #45a049;
    }

    .error-message {
        color: red;
        margin-top: 10px;
    }
</style>
</head>

<body ng-controller="LoginController">
    <div class="login-container">
        <h2>Login Form</h2>
        <form ng-submit="login()">
            <label for="username">Username:</label>
            <input type="text" id="username" ng-model="user.username" required>

            <label for="password">Password:</label>
            <input type="password" id="password" ng-model="user.password" required>

            <button type="submit">Login</button>

            <p class="error-message" ng-show="loginError">{{ loginError }}</p>
        </form>
    </div>

    <script>
        angular.module('loginApp', [])
            .controller('LoginController', function ($scope) {
                $scope.user = {
                    username: "",
                    password: ""
                };

                $scope.loginError = "";

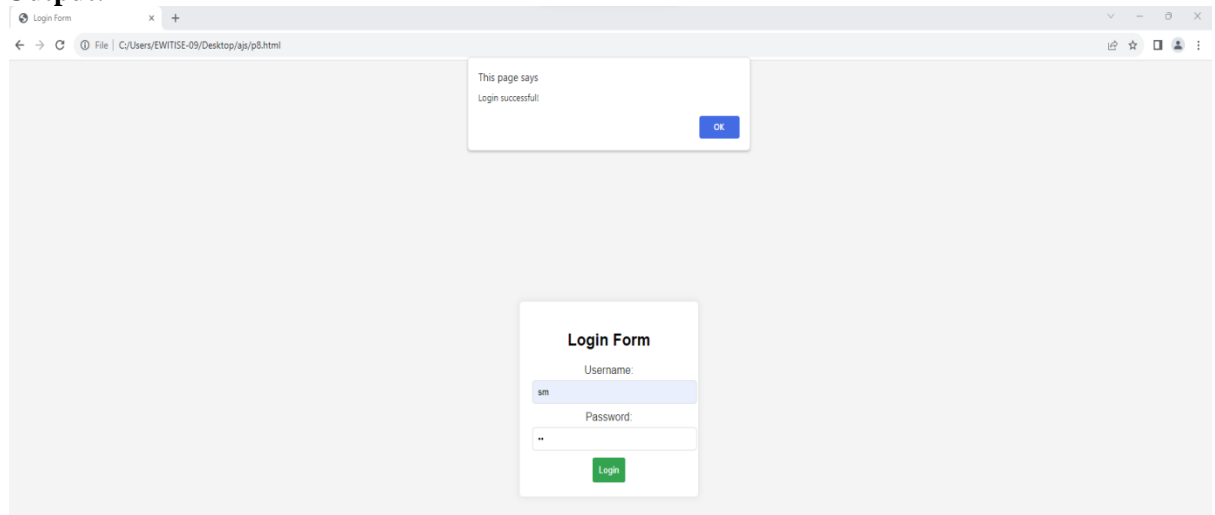
                $scope.login = function () {
                    // Simulate login logic
                    if ($scope.user.username && $scope.user.password) {
                        // Successful login
                        alert('Login successful!');
                        $scope.loginError = "";
                    } else {
                        // Failed login
                        $scope.loginError = 'Both username and password are required';
                    }
                };
            });
    </script>

```

```
</script>
</body>

</html>
```

Output:



- 9. Create an AngularJS application that displays a list of employees and their salaries. Allow users to search for employees by name and salary. Note: Employee details may be included in the program.**

```
<!DOCTYPE html>
<html lang="en" ng-app="employeeApp">

<head>
  <meta charset="UTF-8">
  <title>Employee List</title>
  <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.0/angular.min.js"></script>
  <style>
    body {
      font-family: Arial, sans-serif;
      background-color: #f4f4f4;
      margin: 0;
      padding: 0;
      display: flex;
      align-items: center;
      justify-content: center;
      height: 100vh;
    }

    .employee-container {
      background-color: #fff;
      box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
      padding: 20px;
      border-radius: 5px;
      text-align: center;
    }

    label {
      display: block;
      margin-bottom: 5px;
      color: #333;
    }

    input {
      width: 100%;
      padding: 8px;
      border-radius: 5px;
      border: 1px solid #ddd;
      margin-bottom: 10px;
    }

    ul {
      list-style-type: none;
      padding: 0;
      margin: 0;
    }

    li {
      margin-bottom: 10px;
    }
  </style>
</head>

<body>
  <div class="employee-container">
    <label>Search by name or salary:</label>
    <input type="text"/>
    <ul>
      <li></li>
    </ul>
  </div>
</body>
</html>
```

```

padding: 10px;
border: 1px solid #ddd;
border-radius: 5px;
background-color: #fff;
display: flex;
justify-content: space-between;
align-items: center;
}

li span {
flex-grow: 1;
}

.no-results {
color: #555;
margin-top: 10px;
}
</style>
</head>

<body ng-controller="EmployeeController">
  <div class="employee-container">
    <h2>Employee List</h2>

    <label for="searchName">Search by Name:</label>
    <input type="text" id="searchName" ng-model="searchName">

    <label for="searchSalary">Search by Salary:</label>
    <input type="number" id="searchSalary" ng-model="searchSalary">

    <ul>
      <li ng-repeat="employee in employees | filter: { name: searchName, salary: searchSalary}">
        <span>{{ employee.name }}</span>
        <span>Salary: {{ employee.salary }}</span>
      </li>
    </ul>

    <p class="no-results" ng-show="(employees | filter: { name: searchName, salary:
searchSalary}).length === 0">
      No results found.
    </p>
  </div>

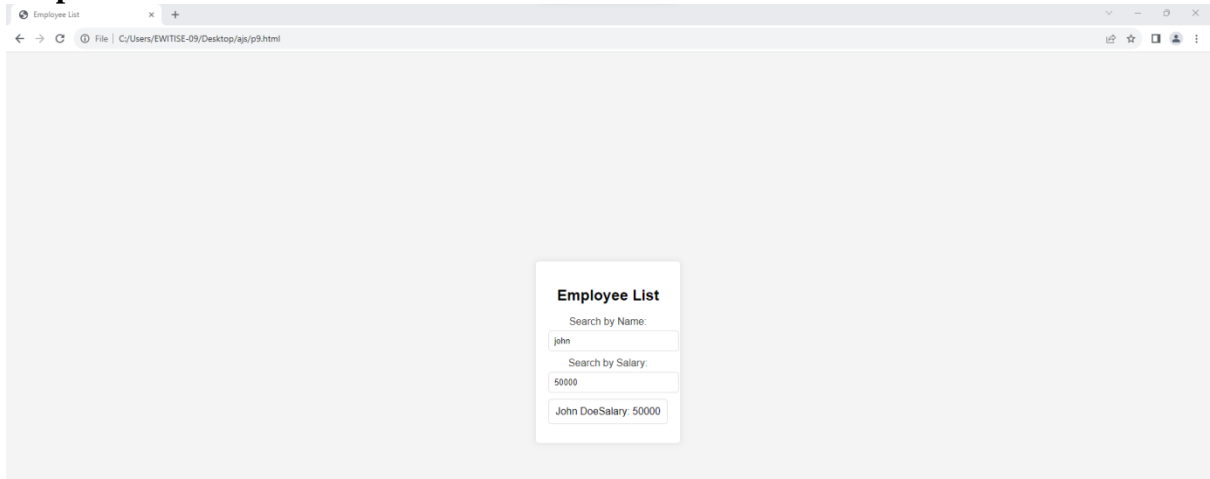
  <script>
    angular.module('employeeApp', [])
      .controller('EmployeeController', function ($scope) {
        $scope.employees = [
          { name: 'John Doe', salary: 50000 },
          { name: 'Jane Smith', salary: 60000 },
          { name: 'Bob Johnson', salary: 55000 },
          // Include more employee details as needed
        ];
      });
  </script>

```

```
</script>
</body>

</html>
```

Output:



- 10. Create AngularJS application that allows users to maintain a collection of items. The application should display the current total number of items, and this count should automatically update as items are added or removed. Users should be able to add items to the collection and remove them as needed. Note: The default values for items may be included in the program.**

```
<!DOCTYPE html>
<html ng-app="itemApp">

<head>
  <title>Item Collection</title>
  <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.7.9/angular.min.js"></script>
  <style>
    body {
      font-family: Arial, sans-serif;
      background-color: #f4f4f4;
      margin: 0;
      padding: 0;
      display: flex;
      align-items: center;
      justify-content: center;
      height: 100vh;
    }
    .container {
      background-color: #fff;
      box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
      padding: 20px;
      border-radius: 5px;
      text-align: center;
    }
    h2 {
      color: #333;
    }
    label {
      display: block;
      margin-bottom: 5px;
      color: #333;
    }
    input {
      width: 70%;
      padding: 8px;
      border-radius: 5px;
      border: 1px solid #ddd;
      margin-bottom: 10px;
    }

    button {
      background-color: #4caf50;
      color: white;
      border: none;
      border-radius: 3px;
      padding: 8px;
      cursor: pointer;
      margin-left: 5px;
    }
    button:hover {
      background-color: #45a049;
    }
    ul {
      list-style-type: none;
      padding: 0;
      margin: 0;
    }
  </style>
</head>

<body>
  <div class="container">
    <h2>Item Collection</h2>
    <div>
      <input type="text" value="Add item here" />
      <button>Add</button>
    </div>
    <div>
      <ul>
        <li>Item 1</li>
        <li>Item 2</li>
        <li>Item 3</li>
        <li>Item 4</li>
        <li>Item 5</li>
      </ul>
      <p>Total items: 5</p>
    </div>
  </div>
</body>
</html>
```

```

li {
    margin-bottom: 10px;
    padding: 10px;
    border: 1px solid #ddd;
    border-radius: 5px;
    background-color: #fff;
    display: flex;
    justify-content: space-between;
    align-items: center;
}

p {
    color: #555;
}
</style>
</head>

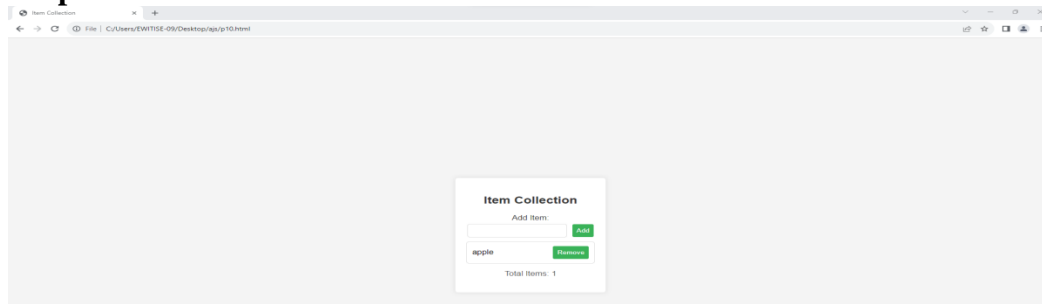
<body ng-controller="ItemController">
<div class="container">
<h2>Item Collection</h2>
<label>Add Item:</label>
<input type="text" ng-model="newItem">
<button ng-click="addItem()">Add</button>
<ul>
<li ng-repeat="item in items">
    {{ item }}
    <button ng-click="removeItem($index)">Remove</button>
</li>
</ul>
<p>Total Items: {{ items.length }}</p>
</div>

<script>
angular.module('itemApp', [])
.controller('ItemController', function ($scope) {
    $scope.items = [];
    $scope.addItem = function () {
        if ($scope.newItem) {
            $scope.items.push($scope.newItem);
            $scope.newItem = ""; // Clear the input field
        }
    };
    $scope.removeItem = function (index) {
        $scope.items.splice(index, 1);
    };
});
</script>
</body>

</html>

```

Output:



11. Create AngularJS application to convert student details to Uppercase using angular filters.

Note: The default details of students may be included in the program.

```
<!DOCTYPE html>
<html lang="en" ng-app="studentApp">
<head>
  <meta charset="UTF-8">
  <title>Student Details Uppercase</title>
  <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.7.9/angular.min.js"></script>
  <style>
    body {
      font-family: Arial, sans-serif;
      background-color: #f4f4f4;
      margin: 0;
      padding: 0;
      display: flex;
      align-items: center;
      justify-content: center;
      height: 100vh;
    }
    .student-container {
      background-color: #fff;
      box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
      padding: 20px;
      border-radius: 5px;
      text-align: center;
    }
    h2 {
      color: #333;
    }
    input {
      width: 70%;
      padding: 8px;
      border-radius: 5px;
      border: 1px solid #ddd;
      margin-bottom: 10px;
    }
    button {
      background-color: #4caf50;
      color: white;
      border: none;
      border-radius: 3px;
      padding: 8px;
      cursor: pointer;
    }
    button:hover {
      background-color: #45a049;
```

```

    }
    p {
      text-align: center;
      color: #555;
    }
  </style>
</head>
<body ng-controller="StudentController">
  <div class="student-container">
    <h2>Student Details Uppercase</h2>

    <label for="inputName">Enter Name:</label>
    <input type="text" id="inputName" ng-model="userInput">

    <button ng-click="convertToUppercase()">Convert to Uppercase</button>

    <p>Uppercase Input: {{ uppercaseResult }}</p>
  </div>

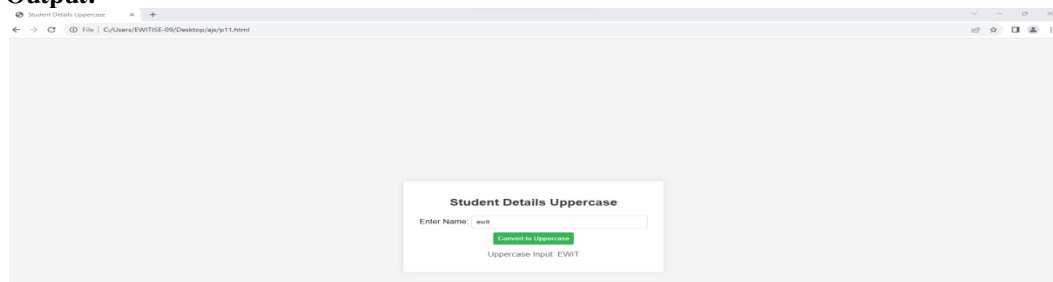
  <script>
    angular.module('studentApp', [])
      .controller('StudentController', function ($scope) {
        $scope.userInput = "";
        $scope.uppercaseResult = "";

        $scope.convertToUppercase = function () {
          $scope.uppercaseResult = $scope.userInput.toUpperCase();
        };
      });
  </script>
</body>

</html>

```

Output:



12. Create an AngularJS application that displays the date by using date filter parameter

```
<!DOCTYPE html>
<html ng-app="myApp">

<head>
  <title>AngularJS Date Display</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      background-color: #f4f4f4;
      margin: 0;
      padding: 0;
      display: flex;
      justify-content: center;
      align-items: center;
      height: 100vh;
    }

    .date-container {
      text-align: center;
    }

    .date {
      font-size: 24px;
      color: #333;
    }
  </style>
</head>

<body>
  <div ng-controller="DateController" class="date-container">
    <h1>Current Date:</h1>
    <div class="date">{{ currentDate | date:'fullDate' }}</div>
  </div>

  <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>
  <script>
    var app = angular.module('myApp', []);

    app.controller('DateController', function ($scope, $interval) {
      // Update the current date every second
      $interval(function () {
        $scope.currentDate = new Date();
      }, 1000);
    });
  </script>
```


</body>

</html>

Output:

Current Date:

Tuesday, January 30, 2024