<!DOCTYPE html>

<html ng-app="myApp">

<head>

<title>Getting Started with AngularJS</title>

<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>

</head>

<body bgcolor="black" text="White">

<h1> <marquee> Program #1 </marquee> </h1>

<center><h2> Sri Venkateshwara College of Engineering </h2>

<h3>Dept. of InformationScience and Engineering</h3>

 <h4>Angular JS code to Display a Message on a Web Page</h4>

 </center> <hr>

<div ng-controller="MyCtrl">

<h1>Welcome to AngularJS!</h1>

<p>Dynamic Content Binding Example:</p>

<p>Enter your First name: <input type="text" ng-model="fname"></p>

<p>Enter your  Second name: <input type="text" ng-model="lname"></p>

<p>Hello, {{ fname }} {{lname}}!</p>

</div>

<script> angular.module('myApp', [])

.controller('MyCtrl', function($scope) {

// Initialize scope variables

$scope.fname = "User";

$scope.lname = "User";

});

</script>

</body>

</html>

2 .

<!DOCTYPE html>

<html ng-app="shoppingApp">

<head>

  <title>Shopping List App</title>

  <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>

  <style>

    input[type="text"] {

      width: 200px;

      margin-bottom: 10px;

    }

  </style>

</head>

<body>

<div ng-controller="ShoppingController">

  <h2>Shopping List</h2>

  <input type="text" ng-model="newItem" placeholder="Enter new item" />

  <button ng-click="addItem()">Add Item</button>

  <ul>

    <li ng-repeat="item in shoppingList">

      {{ item }}

      <button ng-click="removeItem($index)">Remove</button>

    </li>

  </ul>

</div>

<script>

  var app = angular.module('shoppingApp', []);

  app.controller('ShoppingController', function($scope) {

    $scope.shoppingList = ["Apples", "Bananas", "Milk", "Bread"];

    $scope.newItem = "";

    $scope.addItem = function() {

      if ($scope.newItem !== "") {

        $scope.shoppingList.push($scope.newItem);

        $scope.newItem = "";

      }

    };

    $scope.removeItem = function(index) {

      $scope.shoppingList.splice(index, 1);

    };

  });

</script>

</body>

</html>

3. Angular JS code to Perform calculation on a Web Page

<!DOCTYPE html>

<html ng-app="calculatorApp">

<head>

  <title>AngularJS Calculator</title>

  <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>

  <style>

    input[type="text"] {

      width: 200px;

      margin-bottom: 10px;

    }

  </style>

</head>

<body>

<h1> <marquee> Program #3 </marquee> </h1>

<center><h2> Sri Venkateshwara College of Engineering </h2>

<h3>Dept. of InformationScience and Engineering</h3>

 <h4>Angular JS code to Perform calculation on a Web Page</h4>

 </center> <hr>

<div ng-controller="CalculatorController">

  <h2>Simple Calculator</h2>

 Enter Number 1: <input type="text" ng-model="operand1" placeholder="Enter operand 1" />

  <br>

  Enter Number 2:<input type="text" ng-model="operand2" placeholder="Enter operand 2" />

  <br>

  <button ng-click="add()">Add</button>

  <button ng-click="subtract()">Subtract</button>

  <button ng-click="multiply()">Multiply</button>

  <button ng-click="divide()">Divide</button>

  <br>

  <strong>Result: {{ result }}</strong>

</div>

<script>

  var app = angular.module('calculatorApp', []);

  app.controller('CalculatorController', function($scope) {

    $scope.operand1 = 0;

    $scope.operand2 = 0;

    $scope.result = 0;

    $scope.add = function() {

      $scope.result = Number($scope.operand1) + Number($scope.operand2);

    };

    $scope.subtract = function() {

      $scope.result = $scope.operand1 - $scope.operand2;

    };

    $scope.multiply = function() {

      $scope.result = $scope.operand1 \* $scope.operand2;

    };

    $scope.divide = function() {

      if ($scope.operand2 !== 0) {

        $scope.result = $scope.operand1 / $scope.operand2;

      } else {

        alert("Cannot divide by zero!");

      }

    };

  });

</script>

</body>

</html>

4. Angular JS code to Compute Factorial and Square of the number on a Web Page

<!DOCTYPE html>

<html ng-app="calculatorApp">

<head>

  <title>AngularJS Calculator</title>

  <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>

  <style>

    input[type="number"] {

      width: 200px;

      margin-bottom: 10px;

    }

  </style>

</head>

<body>

<h1> <marquee> Program #4 </marquee> </h1>

<center><h2> Sri Venkateshwara College of Engineering </h2>

<h3>Dept. of InformationScience and Engineering</h3>

 <h4>Angular JS code to Compute Factorial and Square of the number on a Web Page</h4>

 </center> <hr>

<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" placeholder="Enter a number" />

  <br>

  <button ng-click="calculateFactorial()">Calculate Factorial</button>

  <button ng-click="calculateSquare()">Calculate Square</button>

  <p ng-show="result !== undefined">Result: {{ result }}</p>

</div>

<script>

  var app = angular.module('calculatorApp', []);

  app.controller('CalculatorController', function($scope) {

    $scope.userInput = 0;

    $scope.result = undefined;

    $scope.calculateFactorial = function() {

      $scope.result = undefined;

      if ($scope.userInput >= 0) {

        $scope.result = getFactorial($scope.userInput);

      } else {

        alert("Factorial is undefined for negative numbers!");

      }

    };

    $scope.calculateSquare = function() {

      $scope.result = $scope.userInput \* $scope.userInput;

    };

    function getFactorial(number) {

      if (number === 0 || number === 1) {

        return 1;

      }

      return number \* getFactorial(number - 1);

    }

  });

</script>

</body>

</html>

5. Angular JS code to display the details of the students with CGPA on a Web Page

<!DOCTYPE html>

<html ng-app="studentApp">

<head>

  <title>Student Details</title>

  <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>

</head>

<body>

<h1> <marquee> Program #5 </marquee> </h1>

<center><h2> Sri Venkateshwara College of Engineering </h2>

<h3>Dept. of Information Science and Engineering</h3>

 <h4>Angular JS code to display the details of the students with CGPA on a Web Page</h4>

 </center> <hr>

<div ng-controller="StudentController">

  <h2>Student Details</h2>

  <label for="numberOfStudents">Enter the number of students:</label>

  <input type="number" id="numberOfStudents" ng-model="numberOfStudents" min="1" required>

  <button ng-click="generateStudentDetails()">Generate Details</button>

  <div ng-show="studentDetails.length">

    <h3>Student Count: {{ studentDetails.length }}</h3>

    <ul>

      <li ng-repeat="student in studentDetails">

        <strong>{{ student.name }}</strong> - CGPA: {{ student.cgpa }}

      </li>

    </ul>

  </div>

</div>

<script>

  var app = angular.module('studentApp', []);

  app.controller('StudentController', function($scope) {

    $scope.studentDetails = [];

    $scope.numberOfStudents = 0;

    // Static student details (for demonstration purposes)

    var staticStudentDetails = [

      { name: 'John Doe', cgpa: '3.8' },

      { name: 'Jane Smith', cgpa: '3.5' },

      { name: 'Bob Johnson', cgpa: '3.2' },

      // Add more student details as needed

    ];

    $scope.generateStudentDetails = function() {

      $scope.studentDetails = [];

      // Use either static or dynamically generated student details

      var detailsToUse = staticStudentDetails;

      for (var i = 0; i < $scope.numberOfStudents && i < detailsToUse.length; i++) {

        $scope.studentDetails.push(detailsToUse[i]);

      }

    };

  });

</script>

</body>

</html>

6. Angular JS to create to-do list

<!DOCTYPE html>

<html ng-app="todoApp">

<head>

  <title>AngularJS Todo List</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: 20px;

    }

    h2 {

      text-align: center;

      color: #333;

    }

    ul {

      list-style-type: none;

      padding: 0;

    }

    li {

      background-color: #fff;

      margin: 10px 0;

      padding: 10px;

      border-radius: 4px;

      border: 1px solid #ddd;

      display: flex;

      justify-content: space-between;

    }

    .task-input {

      width: 200px;

      margin-right: 10px;

    }

    .edit-input {

      width: 150px;

    }

    .button-group {

      display: flex;

    }

    .button-group button {

      margin-left: 5px;

    }

  </style>

</head>

<body bgcolor="white" text="Black">

  <h1> <marquee> Program #6 </marquee> </h1>

  <center><h2> Sri Venkateshwara College of Engineering </h2>

  <h3>Dept. of InformationScience and Engineering</h3>

   <h4>Angular JS to create to-do list</h4>

   </center> <hr>

<body ng-controller="TodoController">

  <h2>AngularJS Todo List</h2>

  <form ng-submit="addTask()">

    <input type="text" ng-model="newTask" class="task-input" placeholder="Add a new task..." required />

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

  </form>

  <ul>

    <li ng-repeat="task in tasks">

      <span ng-show="!task.editing">{{ task.name }}</span>

      <input ng-show="task.editing" ng-model="task.updatedName" class="edit-input" />

      <div class="button-group">

        <button ng-click="editTask(task)">Edit</button>

        <button ng-click="deleteTask(task)">Delete</button>

      </div>

    </li>

  </ul>

  <script>

    var app = angular.module('todoApp', []);

    app.controller('TodoController', function ($scope) {

      // Default tasks

      $scope.tasks = [

        { name: 'Task 1', editing: false },

        { name: 'Task 2', editing: false },

        { name: 'Task 3', editing: false }

      ];

      $scope.newTask = '';

      $scope.addTask = function () {

        if ($scope.newTask) {

          $scope.tasks.push({ name: $scope.newTask, editing: false });

          $scope.newTask = '';

        }

      };

      $scope.editTask = function (task) {

        if (task.editing) {

          task.name = task.updatedName;

        }

        task.editing = !task.editing;

      };

      $scope.deleteTask = function (task) {

        var index = $scope.tasks.indexOf(task);

        $scope.tasks.splice(index, 1);

      };

    });

  </script>

</body>

</html>

7. Write an AngularJS program to create a simple CRUD application.

<!DOCTYPE html>

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

<head>

  <meta charset="UTF-8">

  <title>AngularJS CRUD Example</title>

  <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.7.9/angular.min.js"></script>

  <style>

    table {

      width: 100%;

      border-collapse: collapse;

      margin-top: 20px;

    }

    th, td {

      border: 1px solid #ddd;

      padding: 8px;

      text-align: left;

    }

  </style>

</head>

<body>

  <body bgcolor="White" text="Black">

    <h1> <marquee> Program #7 </marquee> </h1>

    <center><h2> Sri Venkateshwara College of Engineering </h2>

    <h3>Dept. of InformationScience and Engineering</h3>

     <h4>Angular JS to create CRUD application</h4>

     </center> <hr>

<div ng-controller="userController">

  <h2>Users</h2>

  <form ng-submit="addUser()">

    <label for="name">Name:</label>

    <input type="text" id="name" ng-model="newUser.name" required>

    <label for="email">Email:</label>

    <input type="email" id="email" ng-model="newUser.email" required>

    <button type="submit">Add User</button>

  </form>

  <table>

    <tr>

      <th>Name</th>

      <th>Email</th>

      <th>Action</th>

    </tr>

    <tr ng-repeat="user in users">

      <td>{{ user.name }}</td>

      <td>{{ user.email }}</td>

      <td>

        <button ng-click="editUser(user)">Edit</button>

        <button ng-click="deleteUser(user)">Delete</button>

      </td>

    </tr>

  </table>

</div>

<script>

  var app = angular.module('crudApp', []);

  app.controller('userController', function ($scope) {

    $scope.users = [

      { name: 'John Doe', email: 'john@example.com' },

      { name: 'Jane Doe', email: 'jane@example.com' },

      // Add more users as needed

    ];

    $scope.newUser = {};

    $scope.addUser = function () {

      $scope.users.push(angular.copy($scope.newUser));

      $scope.newUser = {};

    };

    $scope.editUser = function (user) {

      $scope.newUser = angular.copy(user);

      $scope.deleteUser(user);

    };

    $scope.deleteUser = function (user) {

      var index = $scope.users.indexOf(user);

      if (index !== -1) {

        $scope.users.splice(index, 1);

      }

    };

  });

</script>

</body>

</html>

8. Angular JS to create Login Form

\

<!DOCTYPE html>

<html ng-app="loginApp">

<head>

  <title>Login Form</title>

  <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>

  <style>

    .error {

      color: red;

    }

  </style>

</head>

<body>

  <body bgcolor="white" text="black">

    <h1> <marquee> Program #8 </marquee> </h1>

    <center><h2> Sri Venkateshwara College of Engineering </h2>

    <h3>Dept. of InformationScience and Engineering</h3>

     <h4>Angular JS to create Login Form</h4>

     </center> <hr>

<div ng-controller="LoginController">

  <h2>Login Form</h2>

  <form name="loginForm" ng-submit="submitForm()" novalidate>

    <label for="username">Username:</label>

    <input type="text" id="username" name="username" ng-model="user.username" required>

    <span class="error" ng-show="loginForm.username.$dirty && loginForm.username.$error.required">Username is required.</span>

    <br>

    <label for="password">Password:</label>

    <input type="password" id="password" name="password" ng-model="user.password" required>

    <span class="error" ng-show="loginForm.password.$dirty && loginForm.password.$error.required">Password is required.</span>

    <br>

    <button type="submit" ng-disabled="loginForm.$invalid">Login</button>

  </form>

  <div ng-show="isSubmitted">

    <h3>Form Data:</h3>

    <pre>{{ user | json }}</pre>

  </div>

</div>

<script>

  var app = angular.module('loginApp', []);

  app.controller('LoginController', function($scope) {

    $scope.user = {};

    $scope.isSubmitted = false;

    $scope.submitForm = function() {

      // Perform login logic here

      // For demonstration purposes, we're just displaying the form data

      $scope.isSubmitted = true;

    };

  });

</script>

</body>

</html>

9. Angular JS code to list of employee and their salaries

<!DOCTYPE html>

<html ng-app="employeeApp">

<head>

  <title>Employee Management</title>

  <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>

</head>

<h1> <marquee> Program #9 </marquee> </h1>

<center><h2> Sri Venkateshwara College of Engineering </h2>

<h3>Dept. of InformationScience and Engineering</h3>

 <h4>Angular JS code to list of employee and their salaries</h4>

 </center> <hr>

<body ng-controller ="EmployeeController as ctrl">

  <h2>Employee List</h2>

  <label>Search by Name: </label>

  <input type="text" ng-model="ctrl.searchName" />

  <label>Search by Salary: </label>

  <input type="text" ng-model="ctrl.searchSalary" />

  <ul>

    <li ng-repeat="employee in ctrl.filteredEmployees">

      {{ employee.name }} - {{ employee.salary | currency }}

    </li>

  </ul>

</body>

</html>

<script>

  var app = angular.module('employeeApp', []);

   app.controller('EmployeeController', function () {

   var ctrl = this;

  ctrl.employees = [

    { name: 'John Doe', salary: 50000 },

    { name: 'Jane Smith', salary: 60000 },

    { name: 'Bob Johnson', salary: 75000 },

    // Add more employee details as needed

  ];

  ctrl.filteredEmployees = ctrl.employees;

  ctrl.$watchGroup(['searchName', 'searchSalary'], function () {

    ctrl.filterEmployees();

  });

  ctrl.filterEmployees = function () {

    ctrl.filteredEmployees = ctrl.employees.filter(function (employee) {

      var nameMatch = !ctrl.searchName || employee.name.toLowerCase().includes(ctrl.searchName.toLowerCase());

      var salaryMatch = !ctrl.searchSalary || employee.salary.toString().includes(ctrl.searchSalary);

      return nameMatch && salaryMatch;

    });

  };

});

</script>

10. Angular JS code to maintain a collection of items

<!DOCTYPE html>

<html ng-app="itemApp">

<head>

  <title>Item Collection</title>

  <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>

  <script src="app.js"></script>

</head>

<h1> <marquee> Program #10 </marquee> </h1>

<center><h2> Sri Venkateshwara College of Engineering </h2>

<h3>Dept. of InformationScience and Engineering</h3>

 <h4>Angular JS code to maintain a collection of items</h4>

 </center> <hr>

<body ng-controller="ItemController as ctrl">

  <h2>Item Collection</h2>

  <label>Add Item: </label>

  <input type="text" ng-model="ctrl.newItem" />

  <button ng-click="ctrl.addItem()">Add</button>

  <ul>

    <li ng-repeat="item in ctrl.items">

      {{ item }}

      <button ng-click="ctrl.removeItem($index)">Remove</button>

    </li>

  </ul>

  <p>Total Items: {{ ctrl.items.length }}</p>

</body>

</html>

<script>

  var app = angular.module('itemApp', []);

app.controller('ItemController', function () {

  var ctrl = this;

  ctrl.items = [];

  ctrl.addItem = function () {

    if (ctrl.newItem && ctrl.items.indexOf(ctrl.newItem) === -1) {

      ctrl.items.push(ctrl.newItem);

      ctrl.newItem = '';

    }

  };

n

  ctrl.removeItem = function (index) {

    ctrl.items.splice(index, 1);

  };

});

</script>

11. Angular JS code to maintain to convert the student details to uppercase

<!DOCTYPE html>

<html ng-app="studentApp">

<head>

  <title>Student Details</title>

  <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>

  <script src="app.js"></script>

</head>

<h1> <marquee> Program #11 </marquee> </h1>

<center><h2> Sri Venkateshwara College of Engineering </h2>

<h3>Dept. of InformationScience and Engineering</h3>

 <h4>Angular JS code to maintain to convert the student details to uppercase</h4>

 </center> <hr>

<body ng-controller="StudentController as ctrl">

  <h2>Student Details</h2>

  <table>

    <tr>

      <th>Name</th>

      <th>Roll Number</th>

    </tr>

    <tr ng-repeat="student in ctrl.students">

      <td>{{ student.name | uppercase }}</td>

      <td>{{ student.rollNumber | uppercase }}</td>

    </tr>

  </table>

</body>

</html>

<script>

  var app = angular.module('studentApp', []);

app.controller('StudentController', function () {

  var ctrl = this;

  ctrl.students = [

    { name: 'John Doe', rollNumber: 'A123' },

    { name: 'Jane Smith', rollNumber: 'B456' },

    // Add more student details as needed

  ];

});

</script>

12. Angular JS code to Display the date

<!DOCTYPE html>

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

<head>

  <meta charset="UTF-8">

  <title>AngularJS Date Filter</title>

</head>

<body bgcolor="white" text="black">

  <h1> <marquee> Program #12 </marquee> </h1>

  <center><h2> Sri Venkateshwara College of Engineering </h2>

  <h3>Dept. of InformationScience and Engineering</h3>

   <h4>Angular JS code to Display the date</h4>

   </center> <hr>

<body ng-controller="MainController">

  <label for="startDate">Start Date:</label>

  <input type="text" id="startDate" ng-model="startDate" placeholder="YYYY-MM-DD">

  <label for="endDate">End Date:</label>

  <input type="text" id="endDate" ng-model="endDate" placeholder="YYYY-MM-DD">

  <ul>

    <li ng-repeat="item in filteredData">{{ item.name }} - {{ item.date | date: 'yyyy-MM-dd' }}</li>

  </ul>

  <script>

    var app = angular.module('myApp', []);

    app.controller('MainController', function ($scope) {

      $scope.data = [

        { name: 'Item 1', date: '2023-01-01' },

        { name: 'Item 2', date: '2023-02-15' },

        { name: 'Item 3', date: '2023-03-20' },

        // Add more data as needed

      ];

      $scope.filterByDate = function (item) {

        if (!$scope.startDate && !$scope.endDate) {

          return true;

        }

        var startDate = $scope.startDate ? new Date($scope.startDate) : new Date(0);

        var endDate = $scope.endDate ? new Date($scope.endDate) : new Date();

        var itemDate = new Date(item.date);

        return itemDate >= startDate && itemDate <= endDate;

      };

      $scope.$watchGroup(['startDate', 'endDate'], function () {

        $scope.filteredData = $scope.data.filter($scope.filterByDate);

      });

    });

  </script>

</body>

</html>