

## UNIT 2 PRACTICAL QUESTIONS

### 1. Uppercase

```
<body ng-app >
<h1>AngularJS Uppercase Filter: </h1>
<div ng-init="firstName='Amit'; lastName='Raj'">
Upper case: {{firstName + ' ' +lastName | uppercase}}
</div>
</body>
</html>
```

### 2. Lowercase

```
<!DOCTYPE html>
<html>
<head>
<script>
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.3.16/angular.min.js"></script>
</head>
<body ng-app >
<h1>AngularJS Lowercase Filter: </h1>
<div ng-init="firstName='John'; lastName='Harry'">
Lower case: {{firstName + ' ' +lastName | lowercase}} </div>
</body> </html>
```

### 3.Currency

```
<body ng-app >
Enter Amount:<input type="number" ng-model=
"price"><br>
The amount is: {{price | currency }}
</body>
</html>
```

### 4. Filter

```
<body>
<div ng-app="myApp" ng-controller="namesCtrl">
<ul>
<li ng-repeat="x in names | filter : 'i'">
{{ x }}
</li>
</ul>
</div>
<script>
angular.module('myApp', []).controller('namesCtrl', function($scope) {
$scope.names = ['Oshin','Iram','Karan','Jay','Anand']; });
```

```

</script>
<p>The names containing the letter "i".</p>
</body>

```

## 6. Date

```

<body>
<div ng-app="gfgApp" ng-controller="dateCntrl">
<p>{{ today | date : "dd.mm.yy" }}</p>
</div>
<script>
var app = angular.module('gfgApp', []);
app.controller('dateCntrl', function($scope) {
$scope.today = new Date();
});
</script> </body>

```

Extra : 15. Using AngularJS display the 10 student details in Table format (using ng-repeat directive use Array to store data)

```

<!DOCTYPE html>
<html>
  <script
src="http://ajax.googleapis.com/ajax/libs/angularjs/1.2.15/angular.min.js"></script>

  <head>
    <title>Students Result</title>
  </head>

  <body>

    <div style="text-align:center;" ng-app="" ng-init="details=[
      {rno:01, name:'Veeha', SGPA:'8.2'},
      {rno:02, name:'Jiya', SGPA:'8.0'},
      {rno:03, name:'Prakash', SGPA:'9.1'},
      {rno:04, name:'Dhriti', SGPA:'9.0'},
      {rno:05, name:'Anjali', SGPA:'9.8'}]">

      <h1>Student Details</h1>

      <table border=3 height="50%" width="50%" align="center">
        <tr>
          <th>Roll no</th>
          <th>Student Name</th>
          <th>SGPA</th>
        </tr>
        <tr ng-repeat="x in details">
          <td>{{x.rno}}</td>
          <td>{{x.name}}</td>
          <td>{{x.SGPA}}</td>
        </tr>
      </table>
    </div>
  </body>
</html>

```

```

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

```

## FORM RELATED QUESTIONS

1. Using AngularJS Create a SPA (form) to take the information of a customer for booking a plan consisting of fields such as name, address, contact no., gender, Date of booking, date of journey, no of passenger, name of passenger etc. Display the e-Ticket.

```

<!-- IU2183830028 -->
<!DOCTYPE html>
<html ng-app="signupApp">
<head>
  <title>Signup Form</title>
  <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>
  <style>
    .container {
      max-width: 400px;
      margin: 0 auto;
      padding: 20px;
      border: 1px solid #ddd;
      border-radius: 5px;
      background-color: #f9f9f9;
    }
    .form-group {
      margin-bottom: 15px;
    }
    .form-group label {
      display: block;
      margin-bottom: 5px;
    }
    .form-group input, .form-group select {
      width: 100%;
      padding: 8px;
      box-sizing: border-box;
      border: 1px solid #ccc;
      border-radius: 4px;
    }
    .form-group input.ng-invalid, .form-group select.ng-invalid {
      background-color: #fdd;
    }
    .form-group input.ng-valid, .form-group select.ng-valid {
      background-color: #dff0d8;
    }
    .btn {
      padding: 10px 15px;

```

```

border: none;
border-radius: 4px;
cursor: pointer;
font-size: 16px;
}
.btn-primary {
background-color: #007bff;
color: #fff;
}
.btn-secondary {
background-color: #6c757d;
color: #fff;
}
</style>
</head>
<body ng-controller="SignupController">

<div class="container">
<form name="signupForm" novalidate>
<div class="form-group">
<label for="username">User Name:</label>
<input type="text" id="username" name="username" ng-model="user.username"
required>
</div>

<div class="form-group">
<label for="email">Email id:</label>
<input type="email" id="email" name="email" ng-model="user.email" required>
</div>

<div class="form-group">
<label for="mobile">Mobile Number:</label>
<input type="tel" id="mobile" name="mobile" ng-model="user.mobile" required>
</div>

<div class="form-group">
<label for="gender">Gender:</label>
<select id="gender" name="gender" ng-model="user.gender" required>
<option value="">Select Gender</option>
<option value="Male">Male</option>
<option value="Female">Female</option>
<option value="Other">Other</option>
</select>
</div>

<button type="reset" class="btn btn-secondary">Reset</button>
<button type="button" class="btn btn-primary"
ng-click="submitForm()">Signup</button>

```

```

    </form>
</div>

<script>
    angular.module('signupApp', [])
        .controller('SignupController', ['$scope', function($scope) {
            $scope.user = {};

            $scope.submitForm = function() {
                if ($scope.signupForm.$valid) {
                    alert('Form submitted successfully!');
                } else {
                    alert('Please fill out all required fields.');
```

2. Create a form for Voter IDs Registration that takes input from users such as Name, gender, contact details and age of a person. Apply any one form validation state. This form should also check whether a user is above 18 years or not. (refer PPT)

```

<html >
<head>
<title>Voter ID Registration</title>
<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>
<style> .error { color: red; }
</style>
</head>
<body ng-app="VoterApp" ng-controller="VoterController" style="text-align: center;">
<h1>Voter ID Registration Form</h1>
<form name="voterForm" ng-submit="submitForm()">
<table border="1%" align="center" style="background-color: bisque;">
<tr>
<th>Name:</th>
<td>
<input type="text" id="name" ng-model="voter.name" required>
</td>
</tr>
<tr>
<th>Contact:</th>
<td>
<input type="text" id="contact" ng-model="voter.contact" required>
</td>
</tr>

```

```

<tr>
<th>Age:</th>
<td>
<input type="number" id="age" ng-model="voter.age" required min="18">
<p class="error" ng-show="voterForm.age.$error.min"> You must be 18 years or older to
register. </p>
</td>
</tr>
<tr>
<th>Gender:</th>
<td>
<select id="gender" ng-model="voter.gender" required>
<option value="male">Male</option>
<option value="female">Female</option>
<option value="other">Other</option>
</select>
</td>
</tr>
<tr>
<td colspan="2">
<input type="submit" value="Register">
</td>
</tr>
</table>
</form>
<div ng-if="submitted">
<p>Thank you for registering, {{ voter.name }}!</p>
</div>

<script> var app = angular.module('VoterApp', []);
app.controller('VoterController', ['$scope', function ($scope) {
  $scope.voter = {};
  $scope.submitted = false;
  $scope.submitForm = function () {
    if ($scope.voterForm.$valid) {
      $scope.submitted = true;
    }
  };
}]);
</script>
</body>
</html>

```

### 3. Create a form to demonstrate ng-model options in the code.

```

<!DOCTYPE html>
<html >
<head>
<title>ng-model Demo</title>

```

```

<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js">
</script>
</head>
<body ng-app="NgModelDemo" ng-controller="DemoController">
<h1>Ng-Model Demo</h1>
<table border="1%" >
<tr>
<th> Two way Binding Example </th>
<td >
Enter your name: <input type="text" ng-model="name"><br> <span>Hello, {{ name
}}!</span>
</td>
</tr>
<tr>
<th> Checkbox Example
</th>
<td>
Subscribe to newsletter: <input type="checkbox" ng-model="subscribe"><br>
<span ng-if="subscribe"> You are subscribed! </span>
</td>
</tr>
<tr>
<th>
Radio buttons Example
</th>
<td>
Choose your favorite color:
<label><input type="radio" ng-model="color" value="red"> Red</label>
<label><input type="radio" ng-model="color" value="blue"> Blue</label>
<label><input type="radio" ng-model="color" value="green"> Green</label>
<br> <span > Your favorite color is {{ color }}. </span>
</td>
</tr>
<tr>
<th> Select Dropdown Example
</th>
<td>Select your country: <select ng-model="selectedCountry">
<option value="USA">United States</option>
<option value="UK">United Kingdom</option>
<option value="India">India</option><br>
</select>
<span > You selected: {{ selectedCountry }} </span>
</td>
</tr>
</table>
<script>
angular.module('NgModelDemo', [])
.controller('DemoController', ['$scope', function ($scope) {

```

```

$scope.name = "";
$scope.subscribe = false;
$scope.color = 'red';
$scope.selectedCountry = 'USA';
$scope.feedback = "";
});
</script>
</body>
</html>

```

### NESTED CONTROLLER RELATED QUESTIONS

4. Create a program to display topics and sub-topics of AngularJS using a nested controller. (refer PPT)

```

<html>
<head>
<title>Nested Controller </title>
<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js">
</script>
</head>
<body ng-app="myApp">
  <div ng-controller="parentController">
    <div ng-controller="childController">
      <div ng-controller="subtopicsController">
        <h1>{{title}}</h1>
        <h2>Courses :</h2>
        <div>
          <ul ng-repeat="x in topics">
            <li>{{x}}</li>
          </ul>
        </div>
        <div>
          <h3>Discription :</h3>
          <ul ng-repeat="y in subtopics">
            <li>{{y}}</li>
          </ul>
        </div>
      </div>
    </div>
  </div>
<script type="text/javascript">
var app = angular.module("myApp", []);
app.controller("parentController", function ($scope) {
  $scope.title = "University "; });
app.controller("childController", function ($scope) {
  $scope.topics = ["MCA ", "MBA", "MCOM"]; });
app.controller("subtopicsController", function ($scope) {
  $scope.subtopics = [ "MCA Course is For IT Students.",

```



```

"MBA Course is for Management Studnts.",
" MCOM Course is for Accounting Students.",
];
});
</script>
</body>
</html>

```

5. Create a program to display courses and subjects available in that course using a nested controller. (Practice Question)

Solution:

```

<html>

<head>
  <title>Angular js PG Topics </title>
  <script src=
"https://ajax.googleapis.com/ajax/libs/angularjs/1.6.9/angular.min.js">
  </script>
</head>

<body ng-app="myApp">
  <div ng-controller="parentController">
    <div ng-controller="childController">
      <div ng-controller="subtopicsController">
        <h1>{{title}}</h1>
        <h2>Courses:</h2>

        <div>
          <ul ng-repeat="x in course">
            <li>{{x}}</li>
          </ul>
        </div>

        <div>
          <h3>Subtopics:</h3>

          <ul ng-repeat="y in subjects">
            <li>{{y}}</li>
          </ul>
        </div>
      </div>
    </div>
  </div>

```

```

        </div>
    </div>
</div>

<script type="text/javascript">
    var app = angular.module("myApp", []);
    app.controller("parentController", function ($scope) {
        $scope.title = "Angularjs";
    });
    app.controller("childController", function ($scope) {
        $scope.course = ["MCA", "MSC-IT", "IMCA", "IMSC"];
    });
    app.controller("subtopicsController", function ($scope) {
        $scope.subjects = [
            "Web Technologies using frameworks",
            "Enterprise java Technology",
            "DWDM",
            "STQA",
        ];
    });
</script> </body> </html>

```

## **FILTER RELATED QUESTIONS**

6. Using AngularJS, display the students' results (name, Total Marks and Rank) in ascending and descending order. (refer PPT)

```

<html>
<head>
<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js">
</script>
<style> table{ font-size: 200%; }
th{ font-size: larger; background-color: bisque; font-family: 'Times New Roman',
Times, serif; }
</style>
<title> Ranking </title>
</head>
<body      ng-app="myapp"      style="text-align:center;background-color:
blanchedalmond;">
<div ng-controller="ctrl">
<div>
<h1>OrderBy Ascending</h1>

```

```

<table border="1% " align="center" >
<tr>
<th> Name </th>
<th> Marks </th>
<th> Rank </th>
</tr>
<tr ng-repeat="stud in Student | orderBy : 'rank'">
<td> {{ stud.name }} </td>
<td> {{ stud.marks }} </td>
<td> {{ stud.rank }} </td>
</tr>
</table>
</div>
<div>
<h1>OrderBy Decending</h1>
<table border="1% " align="center" >
<b>
<tr>
<th> Name </th>
<th> Marks </th>
<th> Rank </th>
</tr></b>
<tr ng-repeat="stud in Student | orderBy : '-rank'">
<td> {{ stud.name }} </td>
<td> {{ stud.marks }} </td>
<td> {{ stud.rank }} </td>
</tr>
</table>
</div>
<script> var app = angular.module('myapp',[]);
app.controller('ctrl',function($scope){
$scope.Student = [ { name : 'sakshi',marks: 456,rank :6 }, { name : 'saki',marks:
500,rank :1 }, { name : 'ishi',marks: 450,rank :5 }, { name : 'isuhu',marks: 433,rank :4 },
{ name : 'nishi',marks: 499,rank :2 },
{ name : 'rahul',marks: 480,rank :3 },
] });
</script>
</body>
</html>

```

7. Write a program to change the name of a candidate in upper case and his email in lowercase.

Solution:

```

<!DOCTYPE html>
<html lang="en" ng-app="myApp">
<head>

```

```

<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>AngularJS Name & Email Transformation</title>
<script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>
</head>
<body ng-controller="CandidateController">

  <h2>Candidate Information</h2>

  <form>
    <div>
      <label for="name">Name:</label>
      <input type="text" id="name" ng-model="candidate.name">
    </div>

    <div>
      <label for="email">Email:</label>
      <input type="email" id="email" ng-model="candidate.email">
    </div>
  </form>

  <h3>Transformed Information</h3>

  <p><strong>Name (Uppercase):</strong> {{ candidate.name | uppercaseFilter }}</p>
  <p><strong>Email (Lowercase):</strong> {{ candidate.email | lowercaseFilter }}</p>

  <script>
    // Define the AngularJS application
    var app = angular.module('myApp', []);

    // Create the controller
    app.controller('CandidateController', function($scope) {
      // Initialize candidate details
      $scope.candidate = {
        name: "",
        email: ""
      };
    });
  </script>

```

```

// Custom filter to convert name to uppercase
app.filter('uppercaseFilter', function() {
  return function(input) {
    if (!input) return "";
    return input.toUpperCase();
  };
});

// Custom filter to convert email to lowercase
app.filter('lowercaseFilter', function() {
  return function(input) {
    if (!input) return "";
    return input.toLowerCase();
  };
});
</script>

</body>
</html>

```

8. Using AngularJS, create a program to display the user's data (name, country and mobile number in json format.

Solution:

```

<!DOCTYPE html>
<html lang="en" ng-app="userApp">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>User Data in JSON</title>
  <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>
  <style>
    .json-output {
      background-color: #f4f4f4;
      padding: 10px;
      border-radius: 5px;
      margin-top: 20px;
    }
  </style>

```

```
</head>
<body ng-controller="UserController">

  <h2>User Information Form</h2>

  <form>
    <div>
      <label for="name">Name:</label>
      <input type="text" id="name" ng-model="user.name">
    </div>

    <div>
      <label for="country">Country:</label>
      <input type="text" id="country" ng-model="user.country">
    </div>

    <div>
      <label for="mobile">Mobile Number:</label>
      <input type="text" id="mobile" ng-model="user.mobile">
    </div>
  </form>

  <h3>Entered Data (in JSON format):</h3>

  <div class="json-output">
    <pre>{{ user | json }}</pre>
  </div>

  <script>
    // Define the AngularJS application
    var app = angular.module('userApp', []);

    // Create the controller
    app.controller('UserController', function($scope) {
      // Initialize user details
      $scope.user = {
        name: "",
        country: "",
        mobile: ""
      };
    });
  </script>
</body>
```

```
});
</script>

</body>
</html>
```

9. Write an AngularJS script to search student names according to the character typed and display details (use array and filter).

Solution:

```
<!DOCTYPE html>
<html lang="en" ng-app="studentApp">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Student Search</title>
  <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>
  <style>
    .student-list {
      margin-top: 20px;
    }
    .student-item {
      background-color: #f4f4f4;
      padding: 10px;
      border-radius: 5px;
      margin-bottom: 10px;
    }
  </style>
</head>
<body ng-controller="StudentController">

  <h2>Search Student Names</h2>

  <!-- Search Box -->
  <input type="text" ng-model="searchText" placeholder="Type student name...">

  <h3>Student List:</h3>
```

```

<!-- Student List Display -->
<div class="student-list">
  <div class="student-item" ng-repeat="student in students | filter:searchText">
    <p><strong>Name:</strong> {{ student.name }}</p>
    <p><strong>Age:</strong> {{ student.age }}</p>
    <p><strong>Class:</strong> {{ student.class }}</p>
  </div>
  <p ng-if="(students | filter:searchText).length == 0">No students found</p>
</div>

<script>
  // Define the AngularJS application
  var app = angular.module('studentApp', []);

  // Create the controller
  app.controller('StudentController', function($scope) {
    // Student array with details
    $scope.students = [
      { name: 'John Doe', age: 15, class: '10th Grade' },
      { name: 'Jane Smith', age: 14, class: '9th Grade' },
      { name: 'David Johnson', age: 16, class: '11th Grade' },
      { name: 'Emily Davis', age: 15, class: '10th Grade' },
      { name: 'Michael Brown', age: 17, class: '12th Grade' },
      { name: 'Sarah Wilson', age: 14, class: '9th Grade' }
    ];

    // Initialize the search text
    $scope.searchText = "";
  });
</script>

</body>
</html>

```

10. Using AngularJS display the Employee details order by salary in Table format (using ng-repeat directive, use Array to store data, use filter)

Solution:

```
<!DOCTYPE html>
```



```
<html lang="en" ng-app="employeeApp">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Employee Details - Sorted by Salary</title>
  <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>
  <style>
    table {
      width: 100%;
      border-collapse: collapse;
      margin-top: 20px;
    }
    table, th, td {
      border: 1px solid black;
    }
    th, td {
      padding: 10px;
      text-align: left;
    }
    th {
      background-color: #f4f4f4;
    }
  </style>
</head>
<body ng-controller="EmployeeController">

  <h2>Employee Details</h2>

  <!-- Employee Details Table -->
  <table>
    <thead>
      <tr>
        <th>Employee Name</th>
        <th>Position</th>
        <th>Salary</th>
      </tr>
    </thead>
    <tbody>
      <tr ng-repeat="employee in employees | orderBy:'salary'">
```

```
<td>{{ employee.name }}</td>
<td>{{ employee.position }}</td>
<td>{{ employee.salary | currency }}</td>
</tr>
</tbody>
</table>

<script>
  // Define the AngularJS application
  var app = angular.module('employeeApp', []);

  // Create the controller
  app.controller('EmployeeController', function($scope) {
    // Employee array with details
    $scope.employees = [
      { name: 'Alice Johnson', position: 'Manager', salary: 75000 },
      { name: 'Bob Smith', position: 'Software Engineer', salary: 60000 },
      { name: 'Charlie Brown', position: 'HR', salary: 50000 },
      { name: 'David Wilson', position: 'Accountant', salary: 55000 },
      { name: 'Eve Davis', position: 'Sales Executive', salary: 45000 },
      { name: 'Frank Moore', position: 'Intern', salary: 30000 }
    ];
  });
</script>

</body>
</html>
```

### UNIT 3 QUESTIONS

1. Create a "About us " web page of any company using ngModule.
- 2. Create a web page about India.**

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>About India</title>

  <style>

    body {

      font-family: Arial, sans-serif;

      background-color: #f3f4f6;

      color: #333;

      margin: 0;

      padding: 0;

    }

    header {

      background-color: #ff9933;

      color: white;

      text-align: center;

      padding: 1em 0;

    }

    section {

      padding: 20px;

      max-width: 800px;

      margin: auto;
```

```
    }
h2 {
    color: #2196F3;
}
ul {
    padding-left: 20px;
}
</style>
</head>
<body>

<header>

    <h1>About India</h1>

</header>

<section>

    <h2>Introduction</h2>

    <p>India, known as Bharat in Hindi, is a country in South Asia. It is the seventh-largest country by land area, the second-most populous country, and the most populous democracy in the world. India's diverse culture, rich history, and vibrant traditions make it unique.</p>

    <h2>Geography</h2>

    <p>India is bordered by Pakistan, China, Nepal, Bhutan, Bangladesh, and Myanmar. It has a vast coastline along the Indian Ocean, the Bay of Bengal, and the Arabian Sea. India's geography includes the Himalayan mountains, the Thar Desert, tropical rainforests, and fertile plains.</p>

    <h2>Culture</h2>
```

<p>India is known for its diverse cultures, languages, and religions. It is home to several major religions, including Hinduism, Buddhism, Jainism, and Sikhism. The country celebrates various colorful festivals such as Diwali, Holi, Eid, and Christmas.</p>

## <h2>Famous Landmarks</h2>

<ul>

<li><strong>Taj Mahal</strong> - A symbol of love and one of the Seven Wonders of the World.</li>

<li><strong>Red Fort</strong> - A historic fort in Delhi that served as the main residence of Mughal emperors.</li>

<li><strong>Gateway of India</strong> - An iconic monument in Mumbai overlooking the Arabian Sea.</li>

<li><strong>Qutub Minar</strong> - A UNESCO World Heritage Site and the tallest brick minaret in the world.</li>

</ul>

## <h2>Languages</h2>

<p>India is a multilingual country with 22 officially recognized languages. Hindi is the most widely spoken language, and English is commonly used for official purposes.</p>

## <h2>Interesting Facts</h2>

<ul>

<li>India is the world's largest producer of milk, spices, and tea.</li>

<li>India has the largest postal network in the world.</li>

<li>Yoga originated in India and has been practiced for over 5,000 years.</li>

<li>The Indian Railways is one of the largest railway networks in the world.</li>

</ul>

</section>

```
</body>
```

```
</html>
```

- 3. Using service, create a program which return information about the location of the current web page (refer PPT)**

```
<!DOCTYPE html>
```

```
<html>
```

```
<script
```

```
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.9/angular.min.js"></script>
```

```
<body>
```

```
<div ng-app="myApp" ng-controller="myCtrl">
```

```
<p>The url of this page is:</p>
```

```
<h3 >{{myUrl}}</h3>
```

```
</div>
```

<p>This example uses the built-in \$location service to get the absolute url of the page.</p>

```
<script>
```

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

```
app.controller('myCtrl', function($scope, $location) {
```

```
    $scope.myUrl = $location.absUrl();
```

```
});
```

```
</script>
```

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

- 4. Create a SPA that creates a service that does square of a given number by the user. (refer PPT)**

```
<html>
```

```
<head>
```

```
<title>Angular JS Services</title>
```

```
<script src = "https://ajax.googleapis.com/ajax/libs/angularjs/1.3.14/angular.min.js">
```

```
</script>
```

```
</head>
```

```
<body>
```

```
<h2>AngularJS Sample Application</h2>
```

```
<div ng-app = "mainApp" ng-controller = "CalcController">
```

```
<p>Enter a number: <input type = "number" ng-model = "number" /></p>
```

```
<button ng-click = "square()">X<sup>2</sup></button>
```

```

    <p>Result: {{result}}</p>
</div>
<script>
    var mainApp = angular.module("mainApp", []);

    mainApp.factory('MathService', function() {
        var fac = {};

        fac.multiply = function(a, b) {
            return a * b;
        }
        return fac;
    });
    mainApp.service('CalcService', function(MathService) {
        this.square = function(a) {
            return MathService.multiply(a,a);
        }
    });
    mainApp.controller('CalcController', function($scope, CalcService) {
        $scope.square = function() {
            $scope.result = CalcService.square($scope.number);
        }
    });
</script>
</body> </html>

```

5. Create a SPA that creates a service that doubles a given number by the user.
- 6. Create a simple service which gets the grocery data as shown below and passes it to our component of a web page.**

## Product Details

Get Products

ID	Name	Price
1	Safola oil	900
1	Bournvita	250
1	Maggie	100

**Solution:**

**Product-list.component.html**

```

<!-- src/app/product-list/product-list.component.html -->
<h2>Product List</h2>
<ul>
  <li *ngFor="let product of products">
    {{ product.name }} - {{ product.price }}
  </li>
</ul>

```

```
</ul>
```

### product-list.component.ts

```
import { Component, OnInit } from '@angular/core';
import { ProductService } from '../product.service'; // Import the ProductService
import { CommonModule } from '@angular/common';

@Component({
  selector: 'app-product-list',
  standalone: true,
  imports: [CommonModule],
  templateUrl: './product-list.component.html',
  styleUrls: ['./product-list.component.css']
})
export class ProductListComponent implements OnInit {
  products: any[] = []; // To hold the product list

  // Inject the ProductService
  constructor(private productService: ProductService) {}

  ngOnInit(): void {
    // Get the products from the service when the component initializes
    this.products = this.productService.getProducts();
  }
}
```

### App.component.html

```
<!-- src/app/app.component.html -->
<h1>Welcome to Product Store</h1>
<app-product-list></app-product-list>
```

### app.component.ts

```
import { Component } from '@angular/core';
import { RouterOutlet } from '@angular/router';
import { ProductListComponent } from '../product-list/product-list.component';
```



```

@Component({
  selector: 'app-root',
  standalone: true,
  imports: [RouterOutlet, ProductListComponent],
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {
  title = 'service';
}

```

### app.module.ts

```

import { NgModule } from '@angular/core';
import { BrowserModule } from '@angular/platform-browser';
import { AppComponent } from './app.component';
import { ProductListComponent } from './product-list/product-list.component'; // Ensure this
import exists

@NgModule({
  declarations: [
    AppComponent,
    ProductListComponent // Declare your ProductListComponent here
  ],
  imports: [
    BrowserModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }

```

### product.service.ts

```

import { Injectable } from '@angular/core';

@Injectable({
  providedIn: 'root',

```

```

    })
    export class ProductService {
        // Sample product data
        private products = [
            { id: 1, name: 'Laptop', price: 1000 },
            { id: 2, name: 'Smartphone', price: 700 },
            { id: 3, name: 'Tablet', price: 300 },
        ];

        //constructor() {}

        // Define the getProducts method
        public getProducts() {
            return this.products;
        }
    }
}

```

Output:

# Welcome to Product Store

## Product List

- Laptop - 1000
- Smartphone - 700
- Tablet - 300

7. Create a simple service which gets the travelling package data to the user and passes it to our component of a web page.
8. Create a web page to access the value of a promise in AngularJS using .then() a method. (refer PPT)

## UNIT 4 NODE JS

1. **Write** a program in node.js to print your name.

**Solution:**

```
var http = require('http');
```

```
http.createServer(function (req, res) {  
  res.writeHead(200, {'Content-Type': 'text/plain'});  
  res.end('Hello! My name is..');  
}).listen(8080);
```

**2. Write a Node.Js program to print sum of 10 numbers using while loop.**

```
(while)  
var sum=0, i=1;  
undefined  
> while(i<=10)  
... {  
... sum =sum+i;  
... console.log(sum);  
... i++;  
... }
```

**Or**

```
(for )  
var sum=0;  
undefined  
> for ( var i=1; i<=10; i++)  
... {  
... sum=sum+i;  
... console.log(sum);  
... }
```

**3. Write a Node.Js program to print a table of 5.**

```
> var n = 5;  
undefined  
> for ( var i=1; i<=10; i++)  
... { console.log(n + " * " + i + "=" + n*i);  
... }  
5 * 1=5  
5 * 2=10  
5 * 3=15  
5 * 4=20  
5 * 5=25  
5 * 6=30  
5 * 7=35  
5 * 8=40  
5 * 9=45  
5 * 10=50
```

**4. Write a Node.Js program to palindrome of a given number.**

```
function palindromeCheck(num) {  
... let numStr = num.toString();  
... let result = numStr.split("").reverse().join("");
```

```

... return numStr === result;
... }
undefined
> console.log(palindromeCheck(121));
true
undefined
> console.log(palindromeCheck(123));
false

```

**5. a. Write a nodejs program to write the content in a file named indus.js about indus university.**

**“Indus University, formerly Indus Institute of Technology and Engineering is a private university established in 2006. In 2012, Indus received university status and was recognized by the University Grants Commission. Indus University is an All India Council for Technical Education approved university.”**

Solution:

Step1:

**//indus.js**

```

var fs = require('fs');
fs.writeFile('H:/file1.txt', 'Indus University, formerly Indus Institute of Technology and Engineering is a private university established in 2006. In 2012, Indus received university status and was recognized by the University Grants Commission. Indus University is an All India Council for Technical Education approved university.', function (err) {
  if (err) throw err;
  console.log('Saved!');
});

```

Step2: Open Node.js command prompt and run the hello.js:

```
node indus.js
```

**b. Write a nodejs program to open & read the content of the file indus.js.**

```

const fs = require("fs");
const buf = new Buffer(1024);

console.log("opening an existing file");
fs.open("H:/file1.txt", "r+", function (err, fd) {
  if (err) {
    return console.error(err);
  }
  console.log("File opened successfully!");
  console.log("reading the file");

  fs.read(fd, buf, 0, buf.length, 0, function (err, bytes) {
    if (err) {
      console.log(err);
    }
    console.log(bytes + " bytes read");
  }
});

```

```
// Print only read bytes to avoid junk.  
if (bytes > 0) {  
    console.log(buf.slice(0, bytes).toString());  
}  
});  
});
```