

## ADVANCED APPLICATION DEVELOPMENT LAB MANUAL

---

➤ **Practical 1 :** Write a program to implement MongoDB data models.

**Soln. :**

### **Embedded Data Model**

```
{
  _id: ,
  Emp_ID: "10025AE336"
  Personal_details :{
    First_Name: "Vrushali",
    Last_Name: "Ghatpande",
    Date_Of_Birth: "1985-09-26"
  },
  Contact : {
    e-mail: "vrushali.123@gmail.com",
    phone: "9568022338"
  },
  Address : {
    city: "Mumbai",
    Area: "Thane",
    State: "Maharashtra"
  }
}
```

### **Normalized Data Model**

Employee :

```
{
  _id: <ObjectId101>,
  Emp_ID: "10025AE336"
}
```

Personal\_details :

```
{
  _id: <ObjectId102>,
  empDocID: " ObjectId101",
}
```

```

    First_Name: "Vrushali",
    Last_Name: "Ghatpande",
    Date_Of_Birth: "1985-09-26"
  }
  Contact :
  {
    _id: <ObjectId103>,
    empDocID: " ObjectId101",
    e-mail: "vrushali.123@gmail.com",
    phone: "9568022338"
  }
  Address :
  {
    _id: <ObjectId104>,
    empDocID: " ObjectId101",
    city: "Mumbai",
    Area: "Thane",
    State: "Maharashtra"
  }

```

➤ **Practical 2 : Write a program to implement CRUD operations on MongoDB.**

Soln. :

**Create a module**

```

db.student.insert({
  regNo: "3014",
  name: "Test Student",
  course: {
    courseName: "MCA",
    duration: "3 Years"
  },
  address: {
    city: "Bangalore",
    state: "KA",
    country: "India"
  }
})

```

Read

db.collection\_name.find()

Update

db.student.update({  
"regNo": "3014"

},

\$set:

{

"name": "Viraj"

})

Delete

db.collection\_name.remove({"fieldname": "value"})

➤ Practical 3 : Write a program to perform validation of a form using AngularJS.

Soln. :

<doctype html>

## ***PRACITACL NO:3***

**Aim : Write a program to perform validation of form using angular js.**

Program:

```
<!DOCTYPE html>
<html>
<head>
  <title>AngularJS Form Validation</title>
  <script src=
"https://ajax.googleapis.com/ajax/libs/angularjs/1.6.9/angular.min.js">
  </script>
  <style>
    body {
      text-align: center;
      font-family: Arial, Helvetica, sans-serif;
    }
    h1 {
      color: green;
    }
  </style>
</head>
<body ng-app="developapp">
```

```

<h1>GeeksforGeeks</h1>

<h3>AngularJS Custom Form Validation</h3>

<form name="form1">

  Username:

  <input name="username" required><br><br>

  Age:

  <input name="userage"

    ng-model="userage" required app-directive>

</form>

<p>The input's valid state is:</p>

<h3>{{form1.userage.$valid}}</h3>

<script>

  var app = angular.module('developapp', []);
  app.directive('appDirective', function () {
    return {
      require: 'ngModel',
      link: function (scope, element, attr, mCtrl) {
        function myValidation(value) {
          if (value >= 18) {
            mCtrl.$setValidity('charE', true);
          } else {

```

```
mCtrl.$setValidity('charE', false);  
    }  
    return value;  
    }  
    mCtrl.$parsers.push(myValidation);  
    }  
};  
});
```

```
</script>
```

```
<p>
```

```
<b>Note:</b>
```

The input field must have  
user age greater than 18 to be  
considered valid for voting.

```
</p>
```

```
</body>
```

```
</html>
```

Output:

**GeeksforGeeks**

**AngularJS Custom Form Validation**

Username:

Age:

The input's valid state is:

**true**

**Note:** The input field must have user age greater than 18 to be considered valid for voting.

## Practical no.4

**Aim : Write a program to create and implement modules and controllers in angular js.**

### ❖ Creating a Module

```
<div ng-app="myApp">...</div>

<script>

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

</script>
```

### ❖ Adding a Controller

```
<!DOCTYPE html>

<html>

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

<body>

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

{{ firstName + " " + lastName }}

</div>
```



```
<script>
var app = angular.module("myApp", []);
app.controller("myCtrl", function($scope) {
    $scope.firstName = "John";
    $scope.lastName = "Doe";
});
</script>
</body>
</html>
```

## Output:



John Doe

### ❖ Modules And Controllers In File

```
<!DOCTYPE html>

<html>

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

<body>

<div ng-app="myApp" ng-controller="myCtrl">
{{ firstName + " " + lastName }}
</div>

<script src="myApp.js"></script>
<script src="myCtrl.js"></script>

</body>

</html>
```

### Output :

John Doe

## PRACITACL NO:5

Aim : Write a program to implement Error Handling in Angular JS

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
    <script src=
```

```
"https://ajax.googleapis.com/ajax/libs/angularjs/1.7.9/angular.min.js">
```

```
    </script>
```

```
    <style>
```

```
        h1 {
```

```
            color: green
```

```
        }
```

```
        button {
```

```
            color: white;
```

```
            background-color: black;
```

```
            height: 30px;
```

```
            width: 160px;
```

```
            padding: 3px;
```

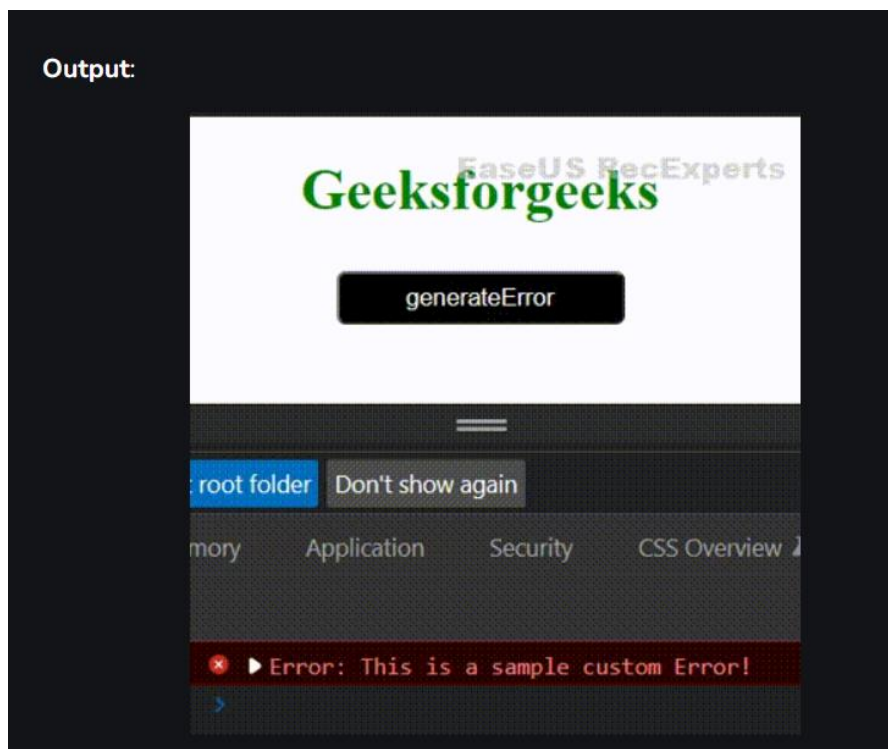
```
            margin: 5px;
```

```

        border-radius: 5px;
    }
</style>
</head>
<body ng-app="myApp">
    <center>
        <h1> Geeksforgeeks</h1>
        <div ng-controller="MyCtrl">
            <button ng-click="generateError()">
                generateError
            </button>
        </div>
    </center>
    <script>
        angular.module('myApp', [])
            .factory
('errorService', function ($exceptionHandler) {
                return function (message) {
                    $exceptionHandler(message);
                }
            })
    </script>

```

```
.controller('MyCtrl', function ($scope, errorService) {  
    $scope.generateError = function () {  
        errorService("Error: This is a sample custom  
Error!");  
    }  
});  
</script>  
</body>  
</html>
```



## Practical no.6

**Aim :** Create an application for Customer / Students records using AngularJS

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
    <title>
```

```
        Angular show hide table element on click
```

```
</title>
```

```
    <script src=
```

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

```
    </script>
```

```
</head>
```

```
<body>
```

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

```
        <button ng-click="showPresent()">
```

```
            Show People Present
```

```
    </button>
```

```
        <button ng-click="showAbsent()">
```

```
            Show People Absent
```

```

</button>

    <table>
        <tr>
            <td>Name</td>
            <td>Age</td>
        </tr>
        <!-- The next two lines are interchangeable-->
        <!-- <tr ng-repeat="p in people" ng-
hide="p.attended!=flag"></tr>-->
        <tr ng-repeat="p in people" ng-show="p.attended==flag">
            <td>{{p.name}}</td>
            <td>{{p.Age}}</td>
        </tr>
    </table>
</div>

<script>
    var app = angular.module("mainApp", []);
    app.controller("MyCtrl", function($scope) {
        $scope.flag = 1;
        $scope.people = [{
            name: "GeekAgashi",
            Age: 12,
            attended: 1

```

```
        }, {
            name: "GeekSatoshi",
            Age: 16,
            attended: 0
        }, {
            name: "GeekNakumato",
            Age: 14,
            attended: 1
        }
    ];

    $scope.showPresent = function() {
        $scope.flag = 1;
    };

    $scope.showAbsent = function() {
        $scope.flag = 0;
    };
});

</script>

</body>

</html>
```





## Practical no.7

Aim : Write a program to create a simple web application using Express, Node JS and Angular JS

```
<!DOCTYPE html>

<html>

<link rel="stylesheet" href="/w3css/4/w3.css">

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

<body ng-app="myApp" ng-controller="userCtrl">
<div class="w3-container">
<h3>Users</h3>
<table class="w3-table w3-bordered w3-striped">
  <tr>
    <th>Edit</th>
    <th>First Name</th>
    <th>Last Name</th>
  </tr>
  <tr ng-repeat="user in users">
```

```

<td>

    <button class="w3-btn w3-ripple" ng-
click="editUser(user.id)">&#9998; Edit</button>

</td>

<td>{{ user.fName }}</td>

<td>{{ user.lName }}</td>

</tr>

</table>

<br>

<button class="w3-btn w3-green w3-ripple" ng-
click="editUser('new')">&#9998; Create New User</button>

<form ng-hide="hideform">

    <h3 ng-show="edit">Create New User:</h3>

    <h3 ng-hide="edit">Edit User:</h3>

    <label>First Name:</label>

    <input class="w3-input w3-border" type="text" ng-
model="fName" ng-disabled="!edit" placeholder="First Name">

    <br>

    <label>Last Name:</label>

```

```
<input class="w3-input w3-border" type="text" ng-model="lName"
ng-disabled="!edit" placeholder="Last Name">
```

```
<br>
```

```
<label>Password:</label>
```

```
<input class="w3-input w3-border" type="password" ng-
model="passw1" placeholder="Password">
```

```
<br>
```

```
<label>Repeat:</label>
```

```
<input class="w3-input w3-border" type="password" ng-
model="passw2" placeholder="Repeat Password">
```

```
<br>
```

```
<button class="w3-btn w3-green w3-ripple" ng-disabled="error ||
incomplete">&#10004; Save Changes</button>
```

```
</form>
```





```
</div>
```

```
<script src= "myUsers.js"></script>
```

```
</body>
```

```
</html>
```

Users

Edit	First Name	Last Name
 Edit	Hege	Pege
 Edit	Kim	Pim
 Edit	Sal	Smith
 Edit	Jack	Jones
 Edit	John	Doe
 Edit	Peter	Pan

 Create New User

Create New User:

First Name:

First Name

Last Name:

Last Name

Password:

Password

Repeat:

Repeat Password

 Save Changes

## Practical no.8

Aim : Create a simple HTML “Hello World” Project using AngularJS Framework and apply ng-controller, ng-model and expressions

```
<!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="personCtrl">

First Name: <input type="text" ng-model="firstName"><br>
Last Name: <input type="text" ng-model="lastName"><br>
<br>
Full Name: {{fullName()}}

</div>

<script>
var app = angular.module('myApp', []);
app.controller('personCtrl', function($scope) {
    $scope.firstName = "John";
    $scope.lastName = "Doe";
```

```
$scope.fullName = function() {  
    return $scope.firstName + " " + $scope.lastName;  
};  
});  
</script>  
</body>  
</html>
```

**Output:**

First Name:	<input type="text" value="John"/>
Last Name:	<input type="text" value="Doe"/>

Full Name: John Doe

## Practical no.9

Aim : Create an app using Flutter for User Authentication

```
import 'package:flutter/material.dart';

void main() {
  runApp(const MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      title: 'Geeks For Geeks Passing Data',
      theme: ThemeData(

        primarySwatch: Colors.green,
      ),
      PassDataDemo(),
    );
  }
}
```



```
}
```

```
}
```

```
class PassDataDemo extends StatefulWidget {  
  const PassDataDemo({Key? key}) : super(key: key);  
  @override  
  State<PassDataDemo> createState() => _PassDataDemoState();  
}
```

```
class _PassDataDemoState extends State<PassDataDemo> {  
  final myController = TextEditingController();  
  @override  
  void dispose() {  
    myController.dispose();  
    super.dispose();  
  }  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  

```

```
title: Text("Geeks For Geeks"),
    ),
    body: Center(
      child: Column(
        children: [
          SizedBox(
            height: 50,
          ),
          Container(
            height: 50,
            width: 300,
            margin: EdgeInsets.all(8),
            decoration: BoxDecoration(
              borderRadius: BorderRadius.circular(10),
              border: Border.all(color: Colors.black, width: 1),
            ),
            child: TextField(
              controller: myController,
              decoration: InputDecoration(
```

```
        labelText: 'Enter The Data',
        border: InputBorder.none,
        floatingLabelBehavior: FloatingLabelBehavior.never,
      ),
    ),
  ),
  SizedBox(
    height: 30,
  ),
  // Step 1 <-- SEE HERE
  ElevatedButton(
    onPressed: () {
      // Step 3 <-- SEE HERE
      Navigator.push(
        context,
        MaterialPageRoute(
          builder: (context) =>
            DetailScreen(title: myController.text),
        ),
      ),
    ),
  ),
),
```

```

        );
    },
    child: const Text(
        'Pass Data To Next Screen',
        style: TextStyle(fontSize: 24),
    ),
),
),
],
),
),
),
);
}
}

```

```

class DetailScreen extends StatefulWidget {
  const DetailScreen({Key? key, required this.title}) : super(key: key);
  final String title;
  @override
  State<DetailScreen> createState() => _DetailScreenState();
}

```

```
class _DetailScreenState extends State<DetailScreen> {  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  
        title: Text("Screen Two"),  
      ),  
      body: Center(  
        child: Column(  
          children: [  
            SizedBox(  
              height: 50,  
            ),  
            Text(  
              '${widget.title}',  
              style: TextStyle(fontSize: 54),  
            ),  
          ],  
        ),  
      ),  
    );  
  }  
}
```

```
        ElevatedButton(  
          onPressed: () {  
            Navigator.pop(context);  
          },  
          child: Text("Write New Text"),  
        )  
      ],  
    ),  
  ),  
);  
}  
}
```

