# Practical 1

**Implementation of Searching algorithm like linear search , Binary search.**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Arithmetic operations and list</title>

    <script src="angular.min.js"></script>

    <style>

        \* {

            margin: 0;

            padding: 0;

            box-sizing: border-box;

        }

        body {

            padding: 20px 0;

            display: flex;

            flex-direction: column;

            align-items: center;

            justify-content: center;

            gap: 10px;

        }

        div {

            width: 700px;

            display: flex;

            flex-direction: column;

            padding: 20px;

            background-color: aqua;

        }

    </style>

</head>

<body ng-app="">

    <div>

        <h1>Arithmetic Operations  </h1>

        <br> Amount = {{2+3}} <br> division = {{10/5}}

    </div>

    <div ng-init="marks = [60,70,80,90,100]">

        <h1>List of Marks </h1>

        Subject 1 = {{marks [0]}} <br>

        Subject 2 = {{marks [1]}} <br>

        Subject 3 = {{marks [2]}} <br>

        Subject 4 = {{marks [3]}} <br>

        Subject 5 = {{marks [4]}} <br>

    </div>

    <div ng-init="people=2; reg=20">

        <h1>Arithmetic Operations </h1>

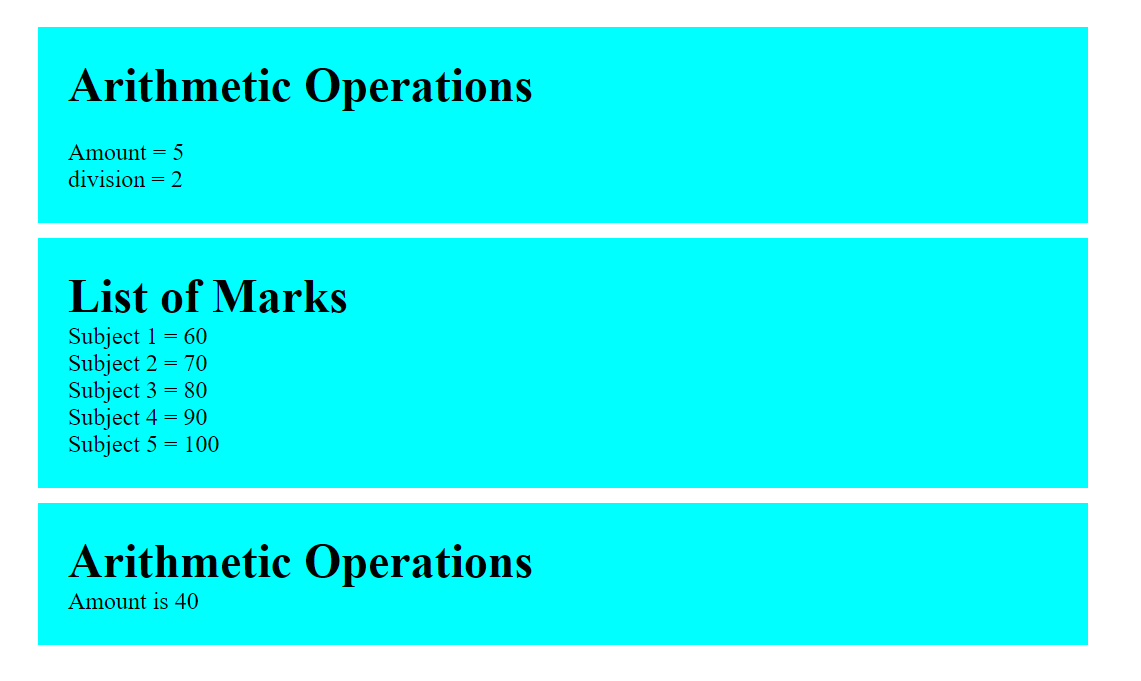
        Amount is {{people\*reg}}

    </div>

</body>

</html>

**Output :-**



# Practical 2

**Create an application in angular.js to calculate registration fees if the number of people and registration amount is given by the user**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Calculate registration fees</title>

    <style>

        \* {

            margin: 0;

            padding: 0;

            box-sizing: border-box;

        }

        body {

            padding: 20px 0;

            display: flex;

            flex-direction: column;

            align-items: center;

            justify-content: center;

            gap: 10px;

        }

        div {

            width: 700px;

            display: flex;

            flex-direction: column;

            padding: 20px;

            background-color: aqua;

        }

        input {

            padding: 10px 20px;

        }

    </style>

    <script src="angular.min.js"></script>

</head>

<body ng-app>

    <div>

        <h1>Calculate registration fees </h1> <br>

        Enter the number of People <input type="number" ng-model="pop"> <br>

        Enter the registration fess <input type="number" ng-model="rege"><br>

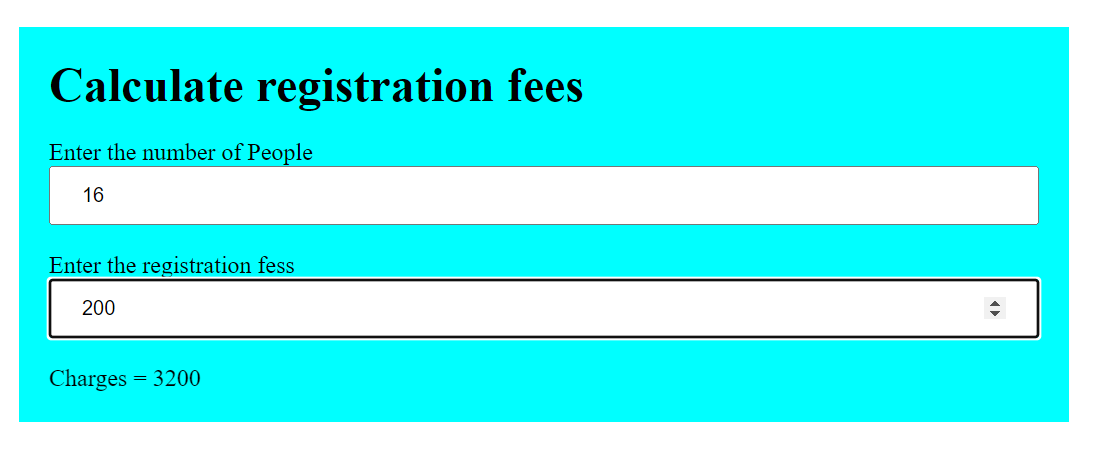
        Charges = {{pop\*rege}}

    </div>

</body>

</html>

**Output :**



# Practical 3

**Create an application in angular.js to calculate simple interest take appropriate input from the user**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Calculate Simple Interest</title>

    <style>

        \*{

            margin: 0;

            padding: 0;

            box-sizing: border-box;

        }

        body {

            padding: 20px 0;

            display: flex;

            flex-direction: column;

            align-items: center;

            justify-content: center;

            gap: 10px;

        }

        div {

            width: 700px;

            display: flex;

            flex-direction: column;

            padding: 20px;

            background-color: aqua;

        }

        input {

            padding: 5px 20px;

        }

    </style>

    <script src="angular.min.js"></script>

</head>

<body ng-app>

    <div>

        <h1>Calculate Simple Interest</h1><br>

        <span>Principle = <input type="number" ng-model="prin"></span><br>

        <span>Rate of Interest= <input type="number" ng-model="roi"></span><br>

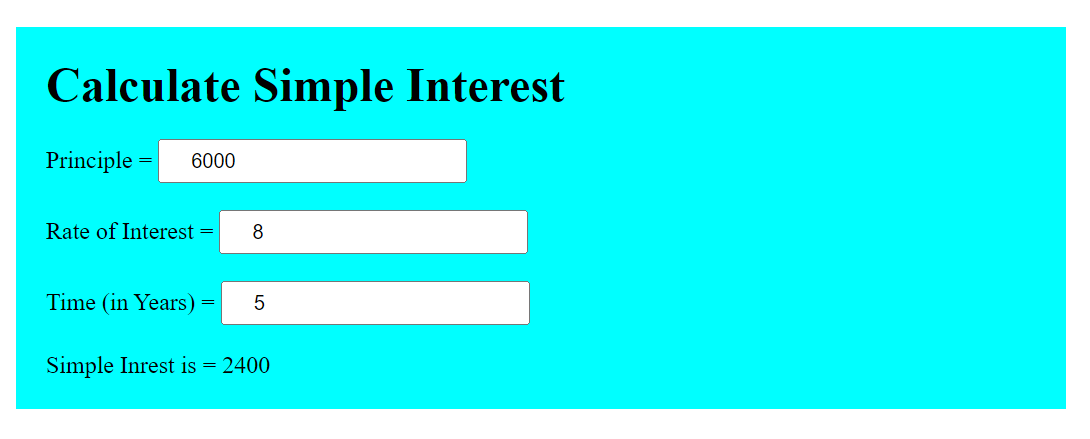
        <span>Time (in Years) = <input type="number" ng-model="t"></span><br>

        <span>Simple Interest is = {{(prin\*roi\*t)/100}} </span>

    </div>

</body>

**Output :-**



# Practical 4

**Write an application in angular.js to create an array of names and display all the names which has letter “i” using controller**

<!DOCTYPE html>

<html>

<script src="angular.min.js"></script>

<title>Display names which

    has letter “i”</title>

<body>

<h1>Display names which

    has letter “i”</h1>

    <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 = [

                'Jani',

                'Carl',

                'Margareth',

                'Hege',

                'Joe',

                'Gustav',

                'Birgit',

                'Mary',

                'Kai'

            ];

        });

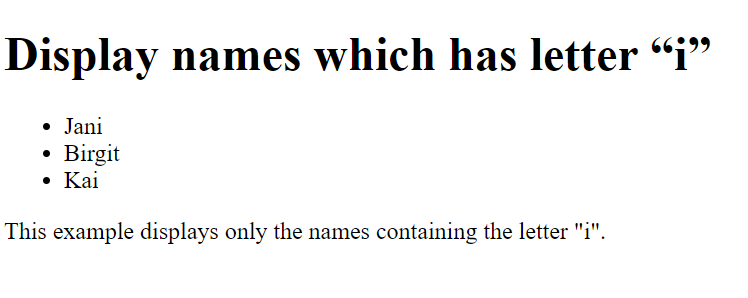
    </script>

    <p>This example displays only the names containing the letter "i".</p>

</body>

</html>

**Output:-**



# Practical 5

**Create an application in angular.js to demonstrate the use of filters**

<!DOCTYPE html>

<html>

<head>

    <script src="angular.min.js"></script>

    <title>Demonstrate the use of filters</title>

</head>

<body ng-app="myApp" ng-controller="myController">

    <div>

        <h1>Demonstrate the use of filters</h1>

        Default currency:{{person.salary | currency}}<br />

        Custom currency identifier:{{person.salary|currency:'Rs.'}}<br />

        No Fraction:{{person.salary|currency:'Rs.':0}}<br />

        Fraction2:<span ng-bind="person.salary|currency:'GBP':2"></span>

        <br>

    </div>

    <div>

        <h1>Name and Number Filter</h1>

        FirstName:{{person.firstName|lowercase}}<br />

        LastName:{{person.lastName|uppercase}}<br />

        Number Filter={{person.salary|number:2}}<br />

        Long date:{{DOB|date:'longDate'}}<br />

        Year:{{DOB|date:"yyyy"}}<br>

        Month:{{DOB|date:"MMMM"}}

    </div>

    <div>

        <h1>Limit to filter</h1>

        LimitTo Get Elements from Beginning:{{limitarr|limitTo:3}}<br />

        LimitTo Get Elements from Ending:{{limitarr|limitTo:-3}}<br />

        Get Elements from String:{{limitarr|limitTo:4}}<br />

        <script>

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

            myApp.controller("myController", function ($scope) {

                $scope.person = {

                    firstName: 'Sagar',

                    lastName: 'Londhe',

                    salary: 47000.67

                }

                $scope.DOB = new Date();

                $scope.limitarr = [20, 10, 43, 5, 1, 4, 6]

            });

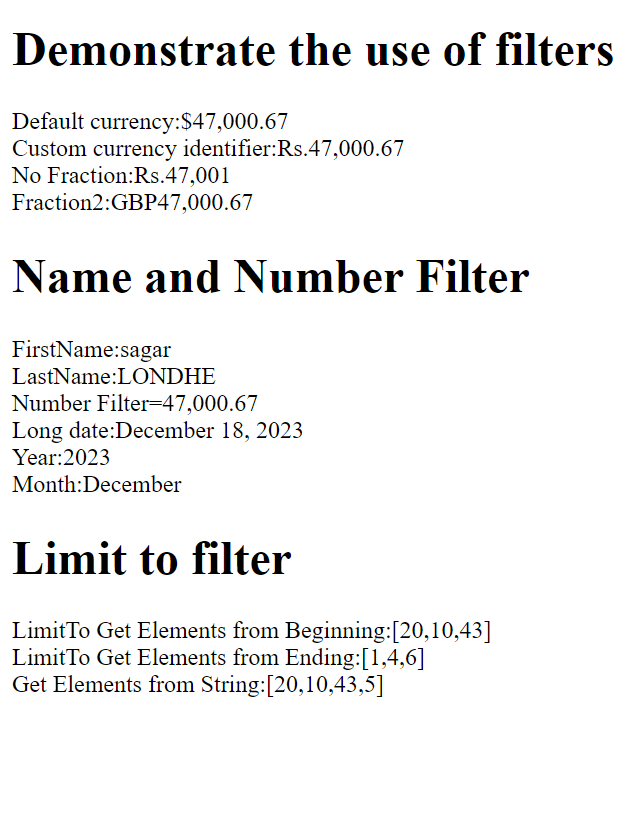
        </script>

    </div>

</body>

</html>

**Output :-**



# Practical 6

**Create an application in angular.js to change the background color as the user changes input in the textbox**

<!DOCTYPE html>

<html lang="en">

<head>

    <title>Change the background color</title>

</head>

<style>

    \* {

        margin: 0;

        padding: 0;

        box-sizing: border-box;

    }

    body {

        height: 100vh;

        display: flex;

        justify-content: center;

        align-items: center;

    }

    input {

        width: 30em;

        height: 3em;

        padding: 20px;

        border: 2px solid black;

        border-radius: 10px;

        cursor: pointer;

    }

</style>

<body>

    <h1 class="head"> Type color Name </h1>

    <input type="text" id="clr">

</body>

<script>

    const bgclr = document.getElementById("clr");

    const headingg = document.querySelector(".head");

    bgclr.addEventListener("input", () => {

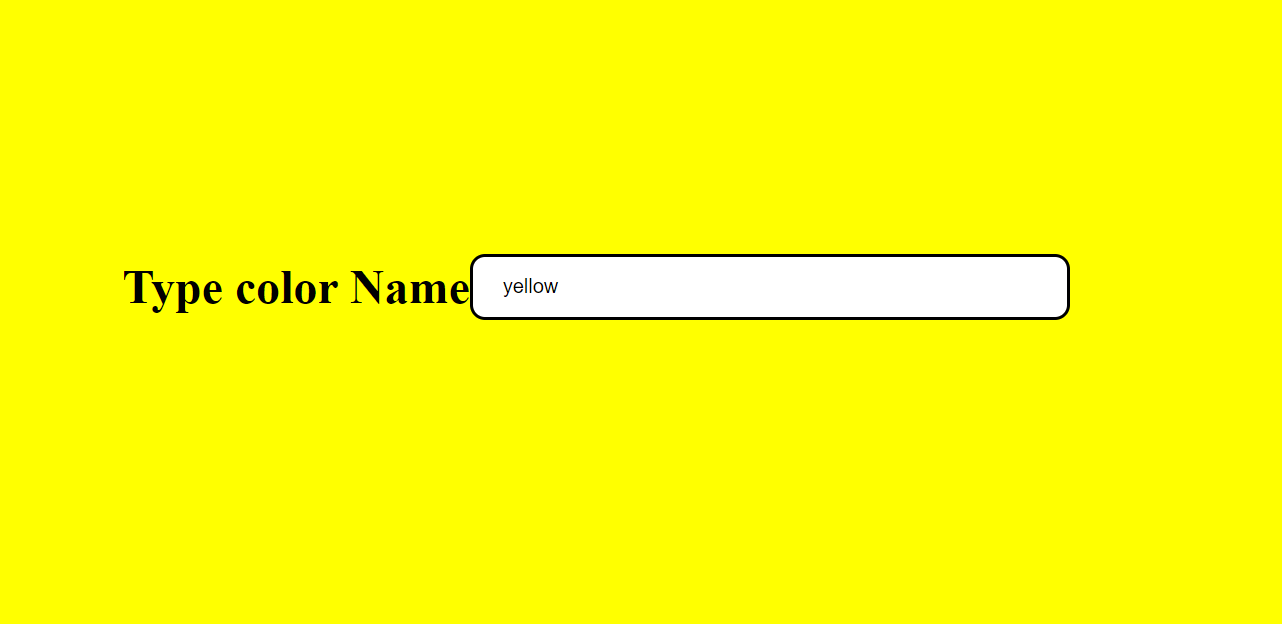
        document.body.style.backgroundColor = bgclr.value;

    });

</script>

</html>

**Output :-**



# Practical 7

**Create an application in angular.js to demonstrate to display text in alert box**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Demonstrate to display text in alert box</title>

    <script src="angular.min.js"></script>

    <style>

        \* {

            margin: 0;

            padding: 0;

            box-sizing: border-box;

        }

        body {

            display: flex;

            align-items: center;

            justify-content: center;

            height: 100vh;

        }

        button {

            padding: 10px 20px;

            background: aqua;

            border-radius: 8px;

        }

        input {

            padding: 10px 20px;

            border-radius: 8px

        }

        div {

            display: flex;

            flex-direction: column;

            width: 500px;

        }

    </style>

</head>

<body ng-app="myApp">

    <div ng-controller="myController" class="">

        <h1>Demonstrate to display text in alert box</h1><br />

        Enter Password: <input type="password" ng-model="password" /> <br />

        <button ng-click="DisplayMessage(password)">Show Password</button>

    </div>

    <script>

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

        myApp.controller("myController", function ($scope, $window) {

            $scope.DisplayMessage = function (value) {

                $window.alert(value);

            }

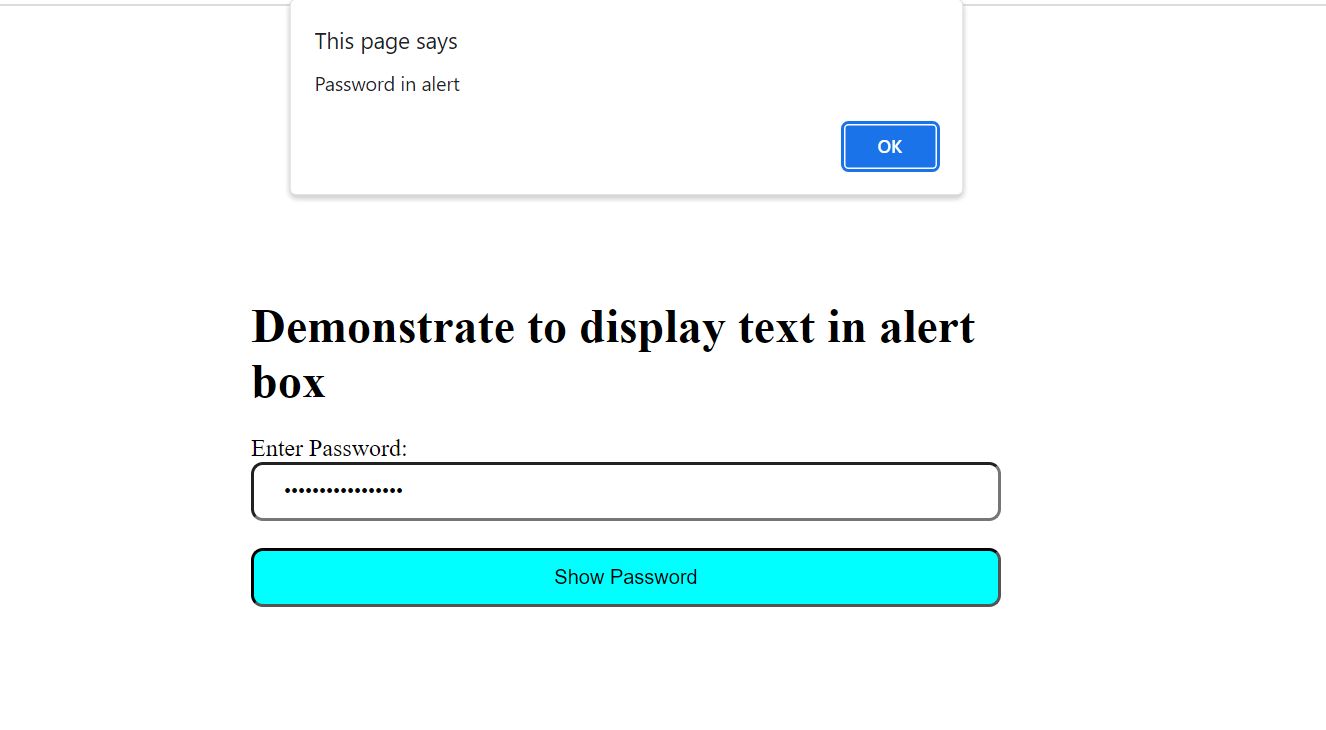
        });

    </script>

</body>

</html>

**Output :-**



# Practical 8

**Create an application in angular.js to demonstrate the use of ng-if, ng-disabled and ng-readonly**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Demonstrate the use of ng-if, ng-disabled and ng-readonly</title>

    <script src="angular.min.js"></script>

    <style>

        div {

            width: 100%;

            height: 20px;

            margin: 15px 0 0 10px;

        }

    </style>

</head>

<body ng-app ng-init="checked=true">

    <h1>Demonstrate the use of ng-if, ng-disabled and ng-readonly</h1>

    <div>Click me for new Textbox:<input type="checkbox" ng-model="checked"></div>

    <div>New Textbox: <input type="text" ng-if="checked"> </div>

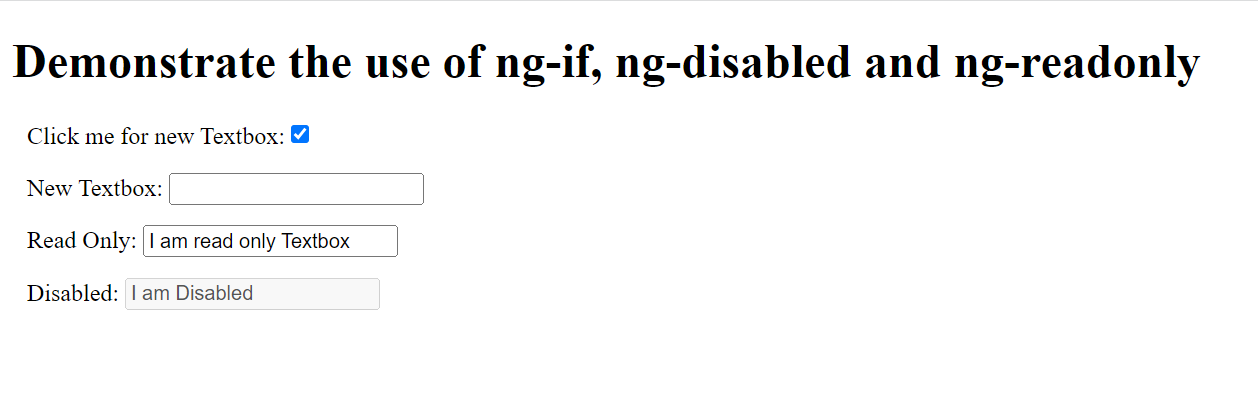
    <div>Read Only: <input type="text" ng-readonly="checked" value="I am read only Textbox"></div>

    <div>Disabled: <input type="text" ng-disabled="checked" value="I am Disabled"></div>

</body>

</html>

**Output :-**



# Practical 9

**Create an application in angular.js to demonstrate use of mouse- enter and mouse-leave event**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Demonstrate use of mouse-enter and mouse-leave

        event</title>

    <script src="angular.min.js"></script>

    <style>

        .redDiv {

            width: 100px;

            height: 100px;

            background-color: red;

            padding: 20px;

        }

        .yellowDiv {

            width: 100px;

            height: 100px;

            background-color: yellow;

            padding: 20px

        }

    </style>

</head>

<body ng-app>

    <h1>Demonstrate use of mouse-enter and mouse-leave

        event</h1>

 <div ng-class="{redDiv:enter,yellowDiv:leave}" ng-mouseenter="enter=true;leave=false" ng-mouseleave="enter=false;leave=true”>

 Mouse

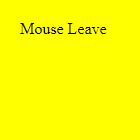
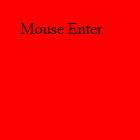
 <span ng-show="enter">Enter</span>

 <span ng-show="leave">Leave</span>

 </div>

</body>

**Output :-**



# Practical 10

**Write an application in angular js to display options using select tag as user chooses the color option the respective color and content should change**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <title> display options using select tag</title>

    <script src="angular.min.js"></script>

</head>

<body ng-app="">

    <div>

        <form>

            Select Color:

            <select ng-model="myVar">

                <option value="pink">pink

                <option value="skyblue">Sky Blue

                <option value="lav">lavender

            </select>

        </form>

    </div>

    <div ng-switch="myVar">

        <div ng-switch-when="">

        </div>

        <div ng-switch="myVar">

            <div ng-switch-when="pink" style="background-color: pink;">

                <h1>Pink</h1>

                <p>Pink Color</p>

            </div>

        </div>

        <div ng-switch="myVar">

            <div ng-switch-when="skyblue" style="background-color: skyblue;">

                <h1>Sky Blue</h1>

                <p>Sky blue color</p>

            </div>

        </div>

        <div ng-switch="myVar">

            <div ng-switch-when="lav" style="background-color: lavender;">

                <h1>Lavender</h1>

                <p>lavender color</p>

            </div>

        </div>

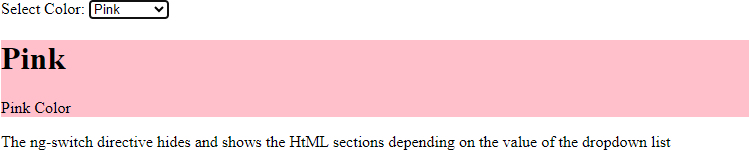
        <p>The ng-switch directive hides and show HTML section depending on the value

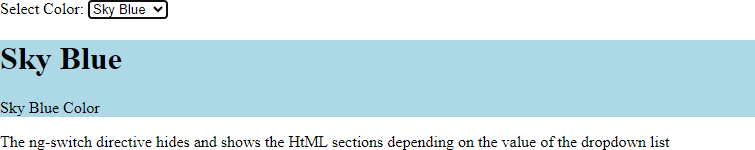
            of the dropdown list</p>

</body>

</html>

**Output :-**





# Practical 11

**Write an Angular JS code to display a Registration form for Student applying for a new Course. Display all the values entered by the students.**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <title>Registration form for Student</title>

    <script src="angular.min.js"></script>

</head>

<body ng-controller="studentController" bgcolor="lightgreen">

    <h1>College admission form:</h1>

    <form ng-submit="submitStudentForm()" form action="page2.html">

        <label for="firstName">First Name: </label><br />

        <input type="text" id="firstName" ng-model="student.firstName" /> <br />

        <label for="lastName">Last Name</label><br />

        <input type="text" id="lastName" ng-model="student.lastName" /> <br />

        <label for="Email">Email</label><br />

        <input type="text" id="Email" ng-model="student.Email" /> <br />

        <label for="dob">DoB</label><br />

        <input type="date" id="dob" ng-model="student.DoB" /> <br /><br />

        <label for="gender">Gender</label> <br />

        <select id="gender" ng-model="student.gender">

            <option value="male">Male</option>

            <option value="female">Female</option>

        </select><br /> <br />

        <span> Fees Payment Via:</span><br />

        <label><input value="online" type="radio" name="training" ng-model="student.trainingType" />Online</label><br />

        <label><input value="offline" type="radio" name="training" ng-model="student.trainingType" />Offline</label>

        <br /><br />

        <span>Course</span><br />

        <label><input type="checkbox" ng-model="student.MCA" />MCA</label> <br />

        <label><input type="checkbox" ng-model="student.MBA" />MBA</label> <br />

        <label><input type="checkbox" ng-model="student.Pharmacy" />Pharmacy</label><br />

        <br />

        <input type="submit" value="Submit " onclick="submitForm()" />

        <input type="reset" ng-click="resetForm()" value="Reset" />

    </form>

    <script>

        function submitForm() {

            alert("Form submitted ");

        }

        //1. create app module

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

        //2. create controller

        studentApp.controller("studentController", function ($scope, $http) {

            //3. attach originalStudent model object

            $scope.originalStudent = {

                firstName: 'sagar',

                lastName: 'londhe',

                DoB: new Date('15/05/2000'),

                gender: 'male',

                trainingType: 'online',

                MCA: false,

                MBA: true,

                Pharmacy: true,

                Email: 'sdl9764177106@gmail.com'

            };

            $scope.student = angular.copy($scope.originalStudent);

            $scope.submitStudentForm = function () {

                var onSuccess = function (data, status, headers, config) {

                    alert('Student saved successfully.');

                };

                var onError = function (data, status, headers, config) {

                    alert('Error occured.');

                }

                $http.post('/student/submitData', {

                        student: $scope.student

                    })

                    .success(onSuccess)

                    .error(onError);

            };

            click.

            $scope.resetForm = function () {

                $scope.student = angular.copy($scope.OriginalStudent);

            };

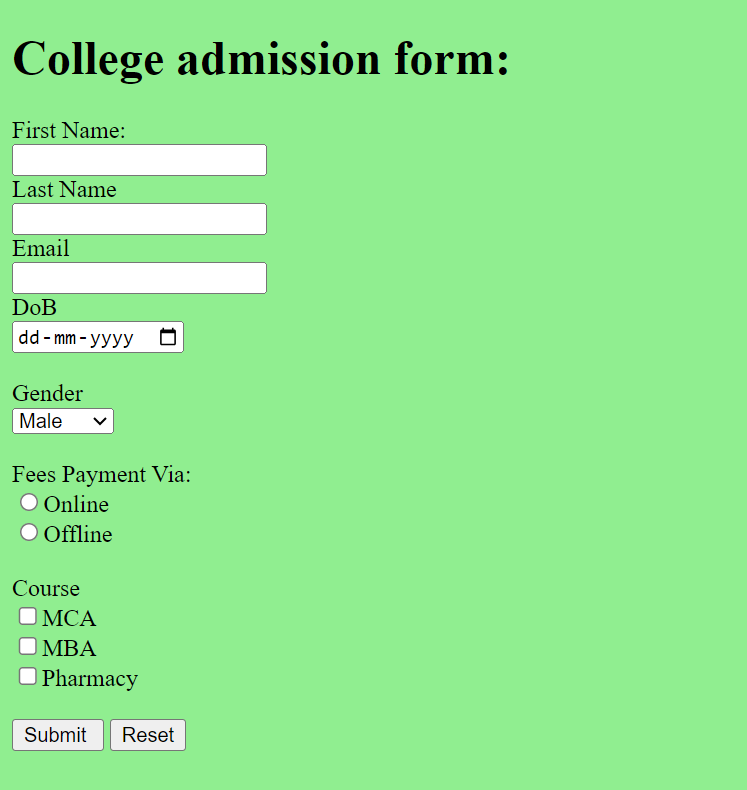
        });

    </script>

</body>

</html>

**Output :**



# Practical 12

**To demonstrate the use of regular expressions for validating input fields in a form**

<!DOCTYPE html>

<html>

<head>

    <meta charset="utf-8">

    <title>Regular Expression for input field</title>

    <script src="angular.min.js">

    </script>

</head>

<body ng-app="app" style="text-align:center">

    <h1 style="color:green;">

        Regular Expression for input field

    </h1>

    <div ng-controller="reg">

        <ng-form name="num">

            Input Number:

            <input type="text" ng-model="number" name="number" ng-pattern="re" /><br />

            <span ng-show="num.number.$error.pattern" style="color:red"> Input is not

                valid.

            </span>

        </ng-form>

    </div>

    <script>

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

        app.controller('reg', ['$scope', function ($scope) {

            $scope.re = /^[0-9]{1,6}$/;

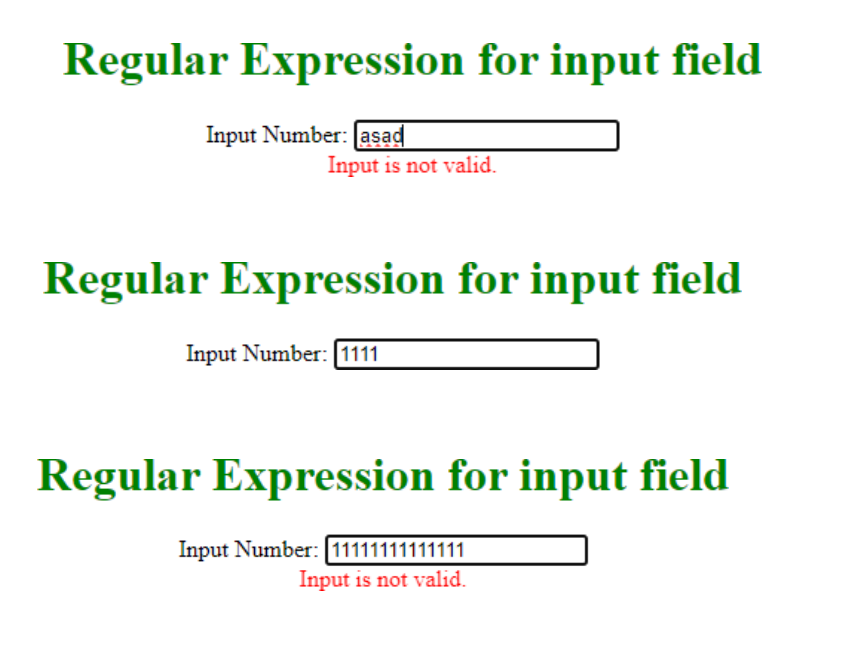
        }]);

    </script>

</body>

</html>

**Output :**



# Practical 13

**To demonstrate use of validation directives.**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>use of validation directives</title>

    <script src='angular.min.js'></script>

</head>

<body ng-app>

    <h1>Use of Validation Directives</h1>

    <form name="studentForm" novalidate>

        <label for="firstName">First Name: </label> <br />

        <input type="text" name="firstName" ng-model="student.firstName" ng-required="true" />

        <span ng-show="studentForm.firstName.$touched &&

   studentForm.firstName.$error.required">First name is required.</span><br /><br />

        <label for="lastName">Last Name</label><br />

        <input type="text" name="lastName" ng-minlength="3" ng-maxlength="10" ng-model="student.lastName" />

        <span ng-show="studentForm.lastName.$touched &&

   studentForm.lastName.$error.minlength">min 3 chars.</span>

        <span ng-show="studentForm.lastName.$touched &&

   studentForm.lastName.$error.maxlength">Max 10 chars.</span><br /><br />

        <label for="dob">Email</label><br />

        <input type="email" id="email" ng-model="student.email" name="email" />

        <span ng-show="studentForm.email.$touched && studentForm.email.$error.email">Please enter

            valid email id.</span><br /><br />

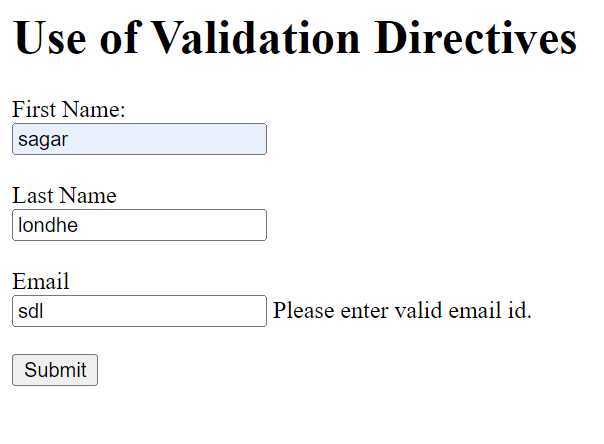
        <input type="submit" value="Submit" />

    </form>

</body>

</html>

**Output:**





# Practical 14

**To demonstrate the state properties of form fields**

<!DOCTYPE html>

<html>

<head>

    <script src="angular.min.js"></script>

</head>

<body ng-app>

    <form name="studentForm" novalidate>

        <h2>Example: State Properties </h2>

        <p>

            First Name Status: <br />

            Pristine: {{studentForm.firstName.$pristine}} <br />

            Touched: {{studentForm.firstName.$touched}}<br />

            Untouched: {{studentForm.firstName.$untouched}}<br /> Valid:

            {{studentForm.firstName.$valid}} <br />

            Invalid: {{studentForm.firstName.$invalid}} <br />

            Dirty: {{studentForm.firstName.$dirty}} <br />

            Error: {{studentForm.firstName.$error}} <br />

        </p>

        <label for="firstName">First Name: </label> <br />

        <input type="text" name="firstName" ng-model="student.firstName" ng required="true" />

        <span ng-show="studentForm.firstName.$touched

&& studentForm.firstName.$error.required">First name is required.</span><br /><br /> <label for="lastName">Last

            Name</label><br />

        <input type="text" name="lastName" ng-minlength="3" ng-maxlength="10" ng-model="student.lastName" /> <br />

        <span ng-show="studentForm.lastName.$error.minlength">min 3

            chars.</span> <span ng-show="studentForm.lastName.$error.maxlength">Max

            10 chars.</span> <br />

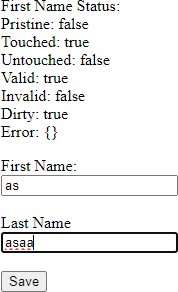
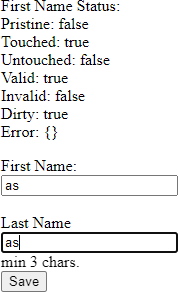
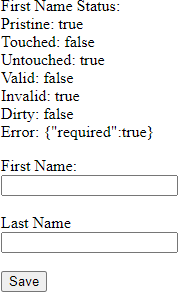
        <input type="submit" value="Save" />

    </form>

</body>

</html>

**Output :**



# Practical 15

**To demonstrate the use of a Single Page Application (SPA)**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Single Page Application</title>

    <script src='angular.min.js'></script>

</head>

<body ng-app>

    <form name="studentForm" novalidate>

        <label for="firstName">First Name: </label> <br />

        <input type="text" name="firstName" ng-model="student.firstName" ng-required="true" />

        <span ng-show="studentForm.firstName.$touched && studentForm.firstName.$error.required">First name

            is required.</span><br /><br />

        <label for="lastName">Last Name</label><br />

        <input type="text" name="lastName" ng-minlength="3" ng- maxlength="10" ng- model="student.lastName" /> <br />

        <span ng-show="studentForm.lastName.$error.minlength">min 3 chars.</span>

        <span ng-show="studentForm.lastName.$error.maxlength">Max 10 chars.</span> <br />

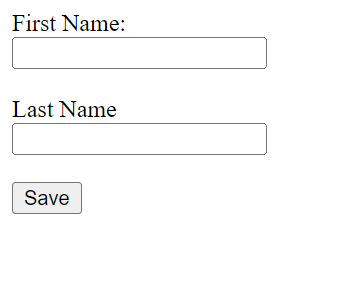
        <input type="submit" value="Save" />

    </form>

</body>

</html>

**Output :**



# Practical 16

**To demonstrate the use of routing in AngularJS. Create a simple application with two routes - Registration and Login**

 <!DOCTYPE html>

 <html lang="en">

 <head>

     <meta charset="UTF-8">

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

     <title>Angular Routing Demo</title>

     <script src="angular.min.js"></script>

     <script src="angular.route.js"></script>

 </head>

 <body ng-app="ngRoutingDemo">

     <h1>

         <center>Angular Routing Demo</center>

     </h1>

     <div>

         <a href="#!/reg">Registration</a>

         <a href="#!/login">Login</a>

     </div>

     <div ng-view align="center"></div>

     <script>

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

         app.config(function ($routeProvider) {

             $routeProvider

                 .when("/reg", {

                     templateUrl: "reg.html"

                 })

                 .when("/login", {

                     templateUrl: "login.html"

                 })

         });

     </script>

 </body>

</html>

**Reg.html**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Registration</title>

    <style>

        @import url('https://fonts.googleapis.com/css?family=PT+Sans');

        body {

            background: #fff;

            font-family: 'PT Sans', sans-serif;

        }

        h2 {

            padding-top: 1.5rem;

        }

        a {

            color: #333;

        }

        a:hover {

            color: #da5767;

            text-decoration: none;

        }

        .card {

            border: 0.40rem solid #f8f9fa;

            top: 10%;

        }

        .form-control {

            background-color: #f8f9fa;

            padding: 20px;

            padding: 10px 15px;

            margin-bottom: 1.3rem;

        }

        .form-control:focus {

            color: #000000;

            background-color: #ffffff;

            border: 3px solid #da5767;

            outline: 0;

            box-shadow: none;

        }

        .btn {

            padding: 0.6rem 1.2rem;

            background: #da5767;

            border: 2px solid #da5767;

        }

        .btn-primary:hover {

            background-color: #df8c96;

            border-color: #df8c96;

            transition: .3s;

        }

    </style>

</head>

<body>

    <div class="container">

        <div class="row justify-content-center">

            <div class="col-md-5">

                <div class="card">

                    <h2 class="card-title text-center">Register <a href="http://opensnippets.com">open snippets</a></h2>

                    <div class="card-body py-md-4">

                        <form \_lpchecked="1">

                            <div class="form-group">

                                <input type="text" class="form-control" id="name" placeholder="Name">

                            </div>

                            <div class="form-group">

                                <input type="email" class="form-control" id="email" placeholder="Email">

                            </div>

                            <div class="form-group">

                                <input type="password" class="form-control" id="password" placeholder="Password">

                            </div>

                            <div class="form-group">

                                <input type="password" class="form-control" id="confirm-password"

                                    placeholder="confirm-password">

                            </div>

                            <div class="d-flex flex-row align-items-center justify-content-between">

                                <a href="#">Login</a>

                                <button class="btn btn-primary">Create Account</button>

                            </div>

                        </form>

                    </div>

                </div>

            </div>

        </div>

    </div>

</body>

</html>

**Login.html**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Login</title>

<style>

    \*{

        padding: 0;

        margin: 0;

        box-sizing: border-box;

    }

    main {

        width: 600px;

        min-height: 400px;

        display: flex;

        flex-direction: column;

        padding: 50px 40px;

        border: 1px solid black;

        gap: 20px;

        background-color: #2D9596;

        box-shadow: 5px 5px 2px 1px rgba(0, 0, 255, .2);

    }

    main div {

        display: flex;

        flex-direction: column;

        gap: 5px;

    }

input {

        padding: 10px 20px;

        box-shadow: 5px 5px 2px 1px rgba(0, 0, 255, .2);

    }

    span {

        font-weight: bold;

        text-transform: uppercase;

    }

    button {

        padding: 10px 20px;

        background-color: #ECF4D6;

        border: none;

        outline: none;

        transition: 1s ease-in-out;

    }

    button:hover {

        transform: scale(1.1);

    }

</style>

</head>

<body>

    <main>

        <div>

            <span>Login</span>

            <input type="text" placeholder="Enter Your username">

        </div>

        <div>

            <span>Password</span>

            <input type="password" placeholder="Enter Your Password">

        </div>

        <div>

            <button>Login</button>

        </div>

    </main>

</body>

</html>

**Output :**



