

Lab 7: Sử dụng AngularJS

Bài 1

Hiển thị thông tin sinh viên gồm họ tên, ngày sinh, giới tính, hình ảnh và điểm trung bình. Các thông tin này được lưu trong \$scope.



Hướng dẫn:

1. Tạo trang bai1.html
2. Code script chèn angularjs vào trang

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

3. Code html cho trang

```
<body>
<div class="container" >

</div>
</body>
```

4. Tạo ứng dụng tên b1

Thêm thuộc tính sau vào tag body

```
ng-app="b1"
```

5. Tạo controller tên hienthi

Thêm thuộc tính sau vào tag div

```
ng-controller="hienthi"
```

6. Code khởi tạo angularjs (để tạo app, gắn controller vào hàm xử lý)

```
<script>
  app= angular.module("b1",[]);
  app.controller("hienthi",xuly);
  function xuly($scope){
    //code xử lý
  }
</script>
```

Kết quả phải như sau:

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

<body ng-app="b1">
<div class="container" ng-controller="hienthi">
  <!-- hiện dữ liệu-->
</div>
</body>
</script>
<script>
  app= angular.module("b1",[]);
  app.controller("hienthi",xuly);
  function xuly($scope){
    //code xử lý
  }
</script>
```

7. Code tạo và gán giá trị các biến (viết trong hàm xử lý)

```
function xuly($scope){
  $scope.hoten = "Nguyễn Văn Tèo";
  $scope.ngaysinh = "24/3/2004";
  $scope.gender = "Nam";
  $scope.photo = "teo.jpg";
  $scope.mark = 8.5;
}
```

(Chú ý: phải thay giá trị của hoten, ngaysinh, gender thành thông tin của mình)

8. Code hiện giá trị các biến (viết trong div container)

```
<div class="caption">
  <h4>Thông tin sinh viên</h4>
  
</div>
<ul>
<li><span>Họ tên : </span> {{hoten}} </li>
<li><span>Ngày sinh : </span> {{ngaysinh}} </li>
<li><span>Phái: </span> {{gender}} </li>
<li><span>Điểm: </span> {{mark}} </li>
</ul>
```

- Tạo 1 folder tên **images** trong folder lab đang làm, sau đó chụp hình selfie chính mình và lưu vào folder images vừa tạo, đặt tên file hình là gì vậy ta?
- Định dạng cho đẹp: width, border, màu chữ, cách canh, margin....

Bài 2

Tạo form nhập chiều dài và chiều rộng của hình chữ nhật. Viết hàm tính chu vi và diện tích của hình chữ nhật như mô tả của hình sau.

TÍNH DIỆN TÍCH CHU VI HÌNH CHỮ NHẬT

Chiều dài

Chiều rộng

Chu vi

Diện tích

- Tạo trang bai2.html

2. Code script chèn angularjs và bootstrap vào trang

```
<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.7.9/angular.min.js"> </script>
<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css">
```

3. Code html cho trang

```
<body>
<div class="container">
  <form>
    <!-- vùng hiển thị-->
  </form>
</div>
</body>
```

4. Tạo ứng dụng tên b2

Thêm thuộc tính sau vào tag body

```
ng-app="b2"
```

5. Tạo controller tên hienthi

Thêm thuộc tính sau vào tag div

```
ng-controller="hienthi"
```

6. Code khởi tạo angularjs (để tạo app, gắn controller vào hàm xử lý)

```
<script>
  app= angular.module("b2",[]);
  app.controller("hienthi",xuly);
  function xuly($scope){
    //code xử lý
  }
</script>
```

Kết quả phải như sau:

```
<body ng-app="b2">
<div ng-controller="hienthi" class="container">
  <form>
    <!-- vùng hiển thị-->
  </form>
</div>
</body>
<script>
  app= angular.module("b1",[]);
  app.controller("hienthi",xuly);
  function xuly($scope){
    //code xử lý
  }
</script>
```

11. Code tạo tính toán các biến (viết trong hàm xử lý)

```
function xuly($scope){
  $scope.tinhtoan= function(){
    var d = parseFloat($scope.dai);
    var r = parseFloat($scope.rong);
    $scope.chuvi = (d+r)*2;
    $scope.dientich = (d*r);
  }
}
```

12. Code hiện form (viết trong tag form)

```

<form>
  <div class="form-group">
    <label>Chiều dài</label>
    <input name="dai" ng-model="dai" class="form-control">
  </div>
  <div class="form-group">
    <label>Chiều rộng</label>
    <input name="rong" ng-model="rong" class="form-control">
  </div>
  <div class="form-group">
    <label>Chu vi </label>
    <span class="form-control">{{chuvi}}</span>
  </div>
  <div class="form-group">
    <label>Diện tích</label>
    <span class="form-control">{{dientich}}</span>
  </div>
  <div class="form-group">
    <button class="btn btn-warning" ng-click="tinhthuan()">Tính toán</button>
  </div>
</form>

```

Xem thử, nhập dài rộng rồi nhấp Tính toán sẽ hiện ra chu vi và diện tích

13. Định dạng

- Cho form nhỏ lại, canh giữa trang, border, màu chữ, màu nền...
- Thêm vào chữ Tính diện tích chu vi hình chữ nhật và định dạng cho đẹp.

Bài 3:

Tạo form và kiểm tra lỗi trong form

FORM NHẬP SẢN PHẨM

TÊN SP

Nhập tên sản phẩm từ 2 đến 30 ký tự

SỐ LƯỢNG

NHÀ SẢN XUẤT

Hãy chọn 1 nhà sx

TÌNH TRẠNG

☐ Hàng cũ ☐ Hàng mới

Hàng cũ hay mới?

MÀU SẮC

☐ Đỏ ☐ Đen ☐ Trắng

Màu gì vậy?

Nhập sản phẩm

1. Chuẩn bị:

- Tạo file bai3.html

```
<body>
<form name="frm" class="col-5 mx-auto border border-primary">
  <div class="form-group">
    <label>Tên SP</label>
    <input name="tensp" class="form-control bg-primary">
  </div>
  <div class="form-group">
    <label>Số lượng</label>
    <input name="soluong" class="form-control bg-warning" >
  </div>
  <div class="form-group">
    <label>Nhà sản xuất</label>
    <select name="nsx" class="form-control bg-secondary text-white">
      <option value="0">Chọn NSX</option>
      <option value="1">Samsung</option>
      <option value="2">Toshiba</option>
      <option value="3">HTC</option>
    </select>
  </div>
  <div class="form-group">
    <label>Tình trạng</label>
    <fieldset class="border border-secondary p-3">
      <input name="tinhtrang" type="radio" value="1"> Hàng cũ &nbsp;  
      <input name="tinhtrang" type="radio" value="2"> Hàng mới
    </fieldset>
  </div>
  <div class="form-group">
    <label>Màu sắc</label>
    <fieldset class="border border-secondary p-3">
      <input value="do" type="checkbox"> Đỏ &nbsp;  
      <input value="den" type="checkbox"> Đen &nbsp;  
      <input value="trang" type="checkbox"> Trắng
    </fieldset>
  </div>
  <div class="form-group">
    <button class="btn btn-info">Nhập sản phẩm</button>
  </div>
</form>
</body>
```

- Chèn thư viện angular và css bootstrap vào đầu trang

```
<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.7.9/angular.min.js"> </script>
<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css">
```

- Tạo ứng dụng tên **b3**
Thêm thuộc tính sau vào tag body

```
ng-app="b3"
```

- Tạo controller tên **ctr1**
Thêm thuộc tính sau vào tag body

```
ng-controller="ctr1"
```

- **Code khởi tạo angularjs** (để tạo app, gắn controller vào hàm xử lý)

```
<script>
  app= angular.module("b3",[]);
  app.controller("ctr1",xuly);
  function xuly($scope){
    //code xử lý
  }
</script>
```

2. Validate với tensp

- Bind giá trị của tensp vào scope :

Thêm thuộc tính sau vào tag input có name là tensp

```
ng-model="tensp"
```

- Thêm quy tắc nhập liệu cho tensp: bắt buộc nhập, từ 2 đến 30 ký tự
Bổ sung vào tag input có name là tensp các thuộc tính sau:

```
required minlength="2" maxlength="30"
```

- Hiện thông báo lỗi

Code sau tag input có name là tensp

```
<em ng-if="frm.tensp.$invalid" class="text-  
danger h6">Nhập tên sản phẩm từ 2 đến 30 ký tự</em>
```

- Test

3. Validate với soluong

- Bind giá trị của soluong vào scope :

Thêm thuộc tính sau vào tag input có name là soluong

```
ng-model="soluong"
```

- Thêm quy tắc nhập liệu cho soluong: kiểu số, nhỏ nhất là 1
Bổ sung vào tag input có name là soluong các thuộc tính sau:

```
type="number" min="1"
```

- Hiện thông báo lỗi khi số lượng nhập ngoài giá trị 1-100

Code sau tag input có name là soluong

```
<em ng-if="frm.soluong.$invalid || soluong>100" class="text-  
danger h6">Nhập số lượng từ 1 đến 100</em>
```

- Test

4. Validate với nhà sản xuất (selectbox)

- Bind giá trị của nsx vào scope :

Thêm thuộc tính sau vào tag select có name là nsx

```
ng-model="nsx"
```

- Hiện thông báo lỗi khi user không chọn mục nào hoặc chọn mục đầu tiên trong selectbox
Code sau tag select có name là nsx

```
<em ng-if="!nsx || nsx==0" class="text-danger h6">Hãy chọn 1 nhà sx</em>
```

- Test

5. Validate với tinhtrang (radio button)

- Bind giá trị của tình trạng vào scope :

Thêm thuộc tính sau vào **2 tag input** có name là tình trạng

```
ng-model="tinhtrang"
```

- Hiện thông báo lỗi khi không chọn cái nào trong radio

Code sau 2 tag input có name là tình trạng

```
<em ng-if="!tinhtrang" class="text-danger h6"><br>Hàng cũ hay mới?</em>
```

- Test

6. Validate với màu sắc (checkbox)

- Bind giá trị của tình trạng vào scope :

Thêm thuộc tính sau vào tag input có value là **do**

```
ng-model="maudo"
```

Thêm thuộc tính sau vào tag input có value là **den**

```
ng-model="mauden"
```

Thêm thuộc tính sau vào tag input có value là **trang**

```
ng-model="mautrang"
```

- Hiện thông báo lỗi khi không chọn cái nào trong các checkbox

Code sau 3 tag input có checkbox

```
<em ng-if="!maudo && !mauden && !mautrang " class="text-  
danger h6"><br>Màu gì vậy?</em>
```

- Test

7. Thông báo thành công

Nếu user nhập đúng tất cả thì hiện dòng chữ **Tôi là <ho tên sinh viên>! Bạn nhập dữ liệu đã đúng yêu cầu.**

8. Định dạng

- Thêm dòng **Form nhập sản phẩm** và định dạng như hình mẫu ở đầu bài
- Định dạng màu nền trang, độ lớn font chữ.....
- Nhập mã sinh viên và họ tên vào tag title của trang (để cho hiện trên thanh tiêu đề của trang)