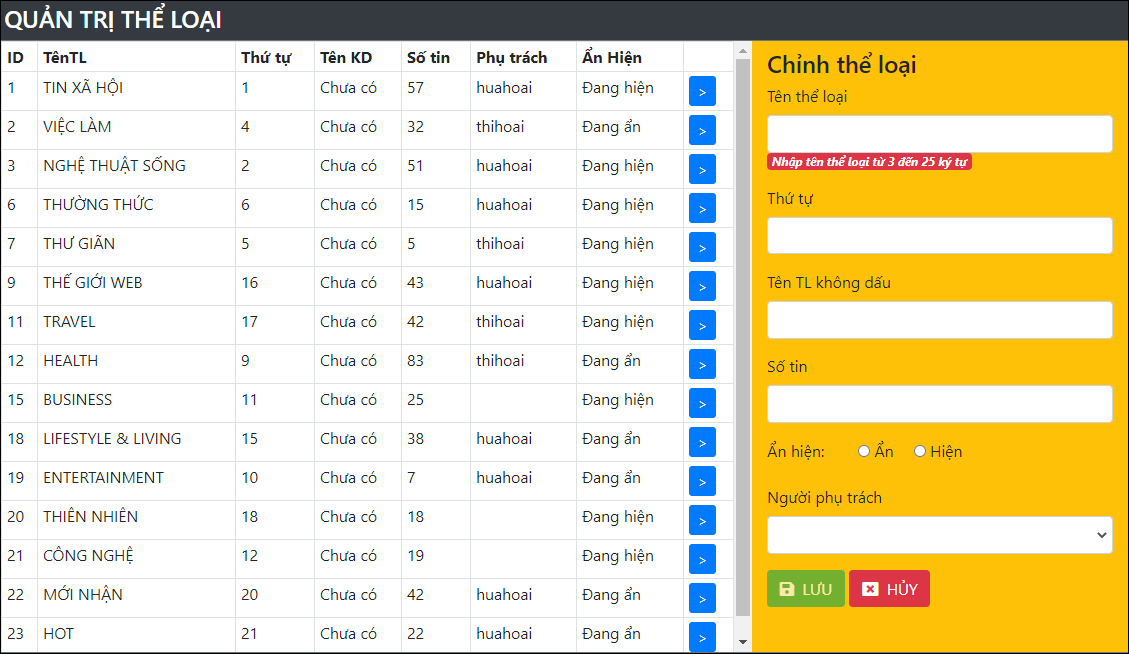
**LAB 6 FRONTEND FRAMEWORK (ANGULARJS)**

Chuẩn bị: tạo folder **<MaSV\_HoTen>\_Lab6** và làm lab trong folder vừa tạo.

### Bài 1: Hiện list dữ liệu có định dạng và form chỉnh dữ liệu



Xây dựng trang web với yêu cầu sau:

1. Giao diện gồm 2 phần: bên trái là danh sách các thể loại, bên phải là form chỉnh thể loại.
2. Chương trình dùng biến: **index**, **listtheloai**, **tlhientai**

* Biến **index**: chỉ số của thể loại hiện tại đang chỉnh sửa trong form.
* Mảng **listheloai** là danh sách các thể loại.
* Đối tượng **tlhientai** là đối tượng chứa thông sinh thể loại hiện hành đang chỉnh sửa trong form

#### Khởi tạo

1. Tạo file **bai1.html**
2. Chèn bootstrap và tạo layout 2 cột :

<body ng-app="myapp" ng-controller="myctrl">

<div class="container">

<header class="row bg-dark p-2">

        <h1 class="text-white h4">QUẢN TRỊ THỂ LOẠI</h1>

    </header>

    <div class="row" style="height:620px">

        <div class="col-8 bg-secondary h-100">Cột trái </div>

        <div class="col-4 bg-warning h-100">Cột phải</div>

    </div>

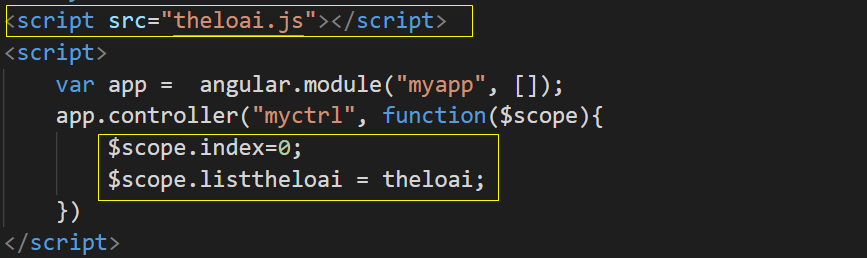
</div>

</body>

Xem thử:

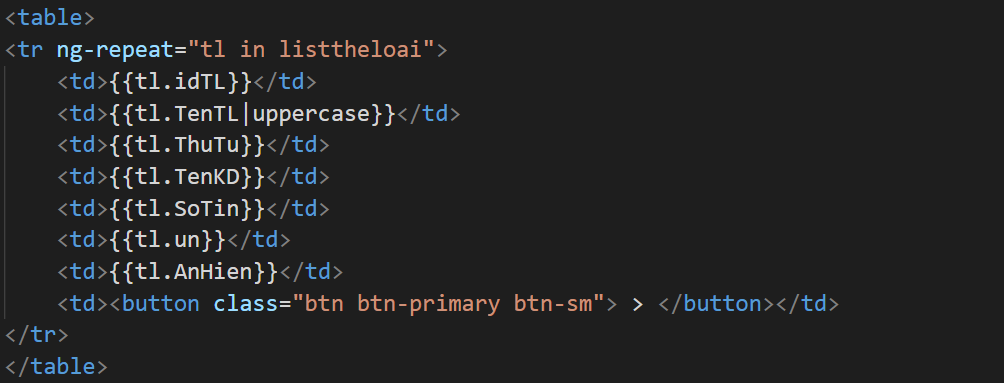


1. Nhúng thư viện angularjs vảo trang, sau đó khai báo app, controller, chèn danh sách thể loại trong file theloai.js, khai báo biến listtheloai và index như sau:



#### Hiện danh sách thể loại ở cột trái

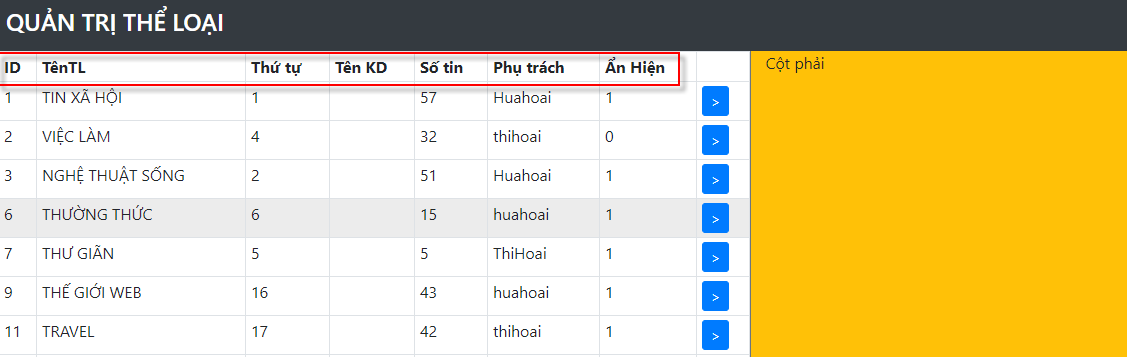
1. Xóa chữ Cột trái và code chèn 1 table là lặp tag tr để hiện các thể loại



Xem thử:



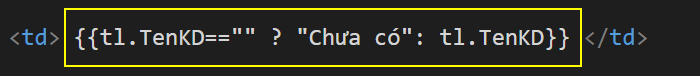
1. Thêm dòng tiêu đề cho table và định dạng table cho đẹp:



#### Chỉnh các hiện dữ liệu cho các cột AnHien, Phụ trách

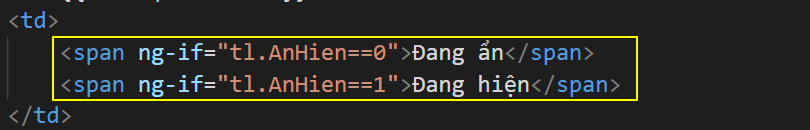
* **Cột TenKD**: nếu giá trị là rỗng thì hiện chữ **Chưa có**

Thay {{tl.TenKD}} bằng code:



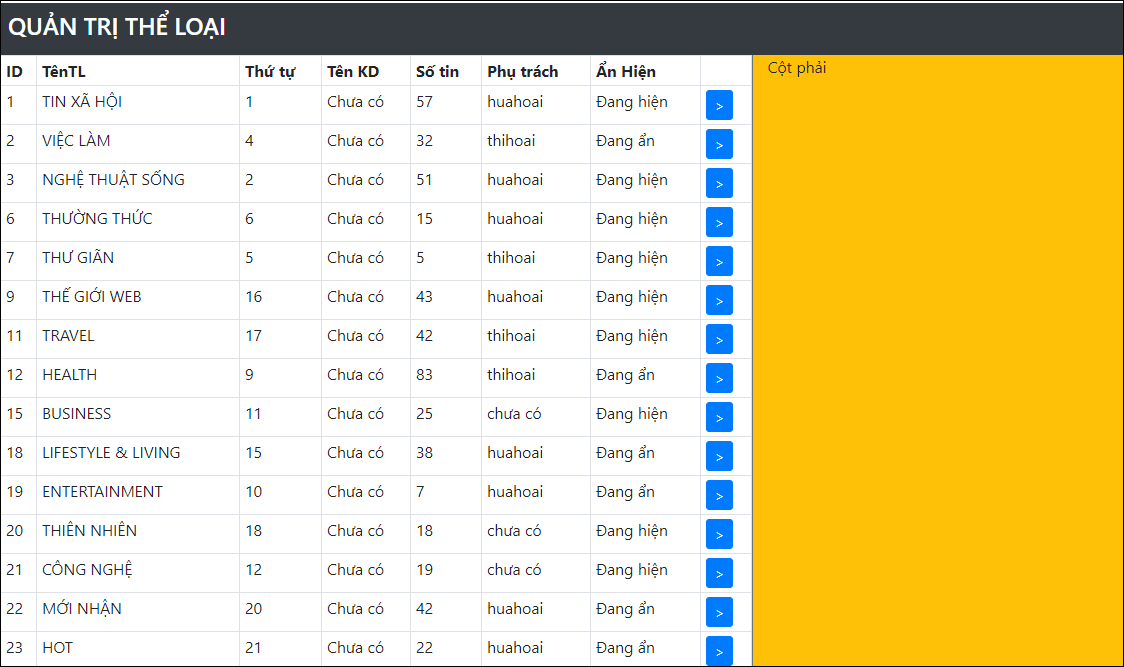
* **Cột Ẩn hiện**: Trong cột Ẩn hiện có các số 0 và số 1 (xem hình trên) , hãy thay thế số 0 thành chữ **Đang ẩn** và số 1 thành chữ **Đang hiện**

Thay code **{{tl.TenKD}}** bằng code mới:



* **Cột Phụ trách**: Nếu giá trị là rỗng thì hiện **Chưa có,** hiện giá trị của cột ở dạng chữ thường

Tham khảo kết quả:

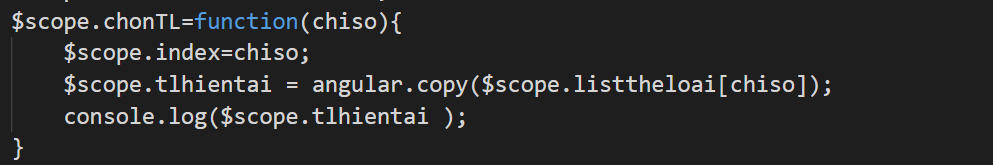


#### Code cho button : ghi nhận thể loại được nhắp vào biến tlhientai

* Code cho button để gọi hàm chonTL và truyền tham số là chỉ số của dòng hiện tại

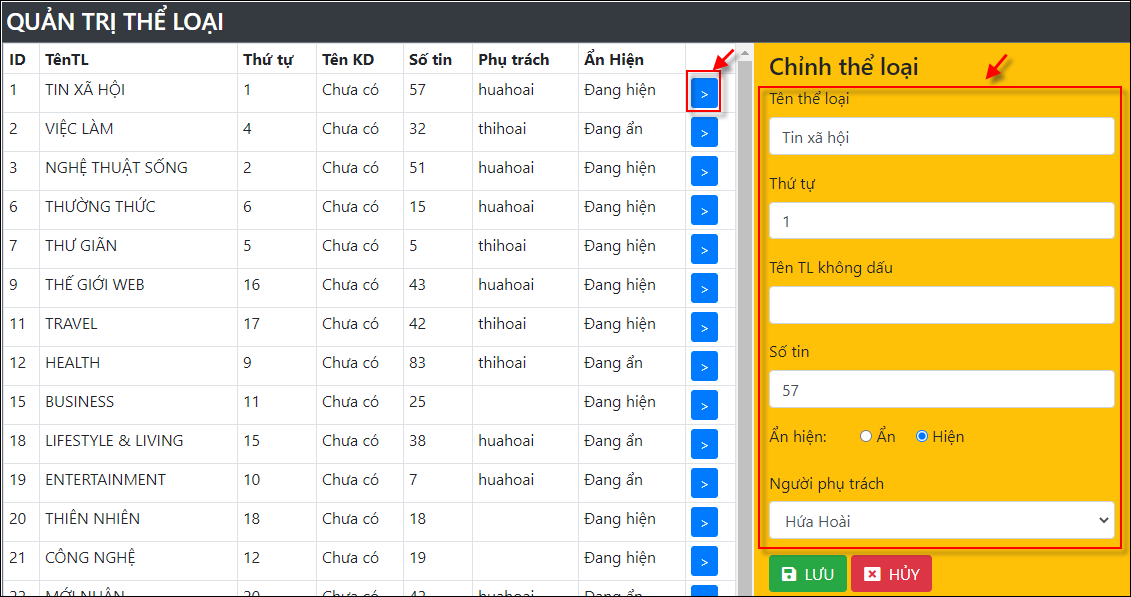


* Định nghĩa hàm chonTL trong $scope

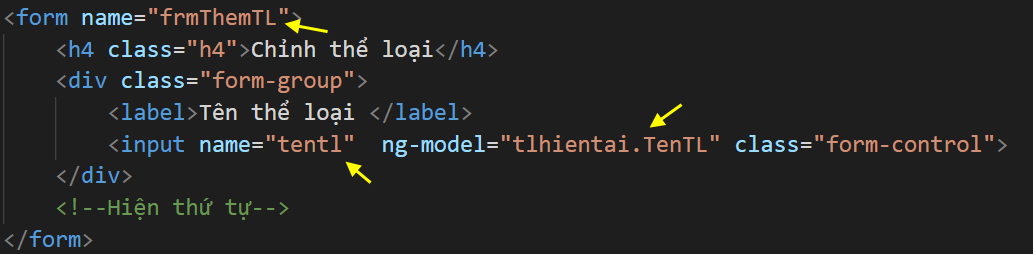


* Xem thử trong console sau mỗi lần nhắp button

#### Tạo form hiện giá trị của biến tlhientai

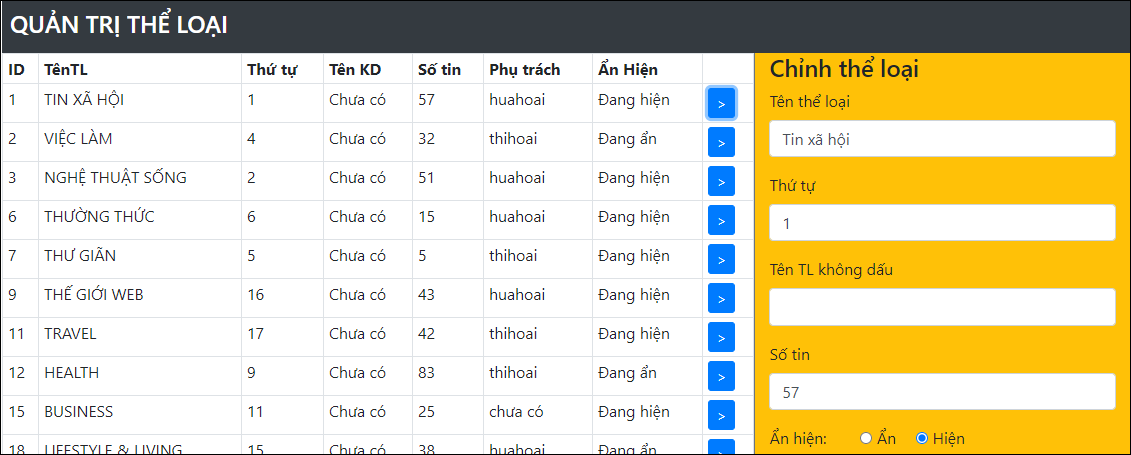


1. Xóa chữ **Cột phải** trong trang web
2. Tại vị trí vừa xóa, code để tạo form có tên **frmThemTL** và 1 textfield có tên là **tentl**, bind vào field **tlhientai.TenTL** :



Test thử bằng cách nhắp các button trong trang, sẽ thấy tên thể loại hiện ra trong form

1. Phía sau Tên thể loại, bạn code để hiện các field Thứ tự, Tên TL không dấu, Số tin, Ẩn hiện để được kết quả sau:



1. Hiện selectbox Người phụ trách:

* Trong $scope, định nghĩa biến listusername

$scope.listusername=[

     {un:"dodatcao", hoten: "Đỗ đạt Cao"},

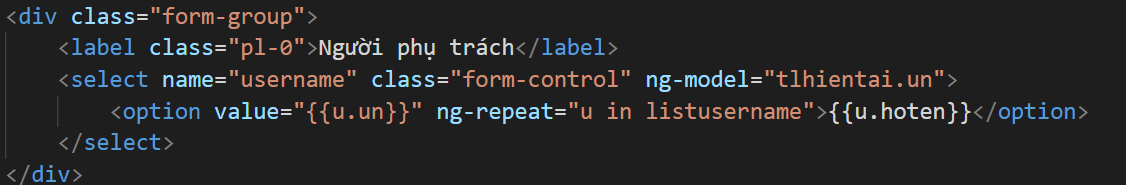
     {un:"huahoai", hoten:"Hứa Hoài"},

     {un:"thihoai", hoten: "Thi Hoài"},

     {un:"phamkyluat", hoten:"Phạm Kỹ Luật"}

];

* Code tag select để lặp qua listusername



1. Hiện 2 button để được kết quả như mẫu



### Bài 2: Kiểm lỗi trong form

#### Tạo file

Save as file **bai1.html** thành **bai2.html**

#### Kiểm lỗi field Tên thể loại

Yêu cầu : field Tên thể loại bắt buộc nhập, dài từ 3 đến 25 ký tự. Hiện thông báo như sau khi user nhập lỗi: **Nhập tên thể loại từ 3 đến 25 ký tự**

**Hướng dẫn thực hiện:**

* Thêm các thuộc tính required , ng-minlength, ng-maxlength vào tag input TenTL
* Thêm tag div hiện lỗi (chỉ hiện khi giá trị trong field TenTL không hợp lệ)



#### Kiểm lỗi field Thứ tự

Yêu cầu : field Thứ tự giá trị nhập số, nhỏ nhất là 1 lớn nhất 23. Hiện thông báo như sau khi user nhập lỗi: **Số tứ tự từ 1 đến 23**

**Hướng dẫn thực hiện:**

* Thêm các thuộc tính ng-min, ng-max vào tag input thứ tự
* Thêm tag div hiện lỗi (chỉ hiện khi giá trị trong field thứ tự không hợp lệ)

#### Kiểm lỗi field Tên thể loại không dấu

Yêu cầu : field Tên thể loại không dấu phải nhập chỉ các ký tự từ a-z, 0-9 và dấu - . Hiện thông báo như sau khi user nhập lỗi: **Tên thể loại không dấu chỉ có chữ, số và dấu -**

**Hướng dẫn thực hiện:**

* Thêm thuộc tính **ng-pattern="/^[a-zA-Z0-9-]\*$/"** vào tag input Tên thể loại không dấu
* Thêm tag div hiện lỗi (chỉ hiện khi giá trị trong field tên thể loại không dấu không hợp lệ)

#### Kiểm lỗi field Số tin

Yêu cầu : field Số tin phải nhập số, nhỏ nhất là 1. Hiện thông báo như sau khi user nhập lỗi: **Số tin là số nguyên >0**

**Hướng dẫn thực hiện:**

* Thêm thuộc tính **min="1"** vào tag input Số tin
* Thêm tag div hiện lỗi (chỉ hiện khi giá trị trong field số tin không hợp lệ)

#### Khóa button lưu theo điều kiện

* Khóa button Lưu khi form chưa có sữa chữa



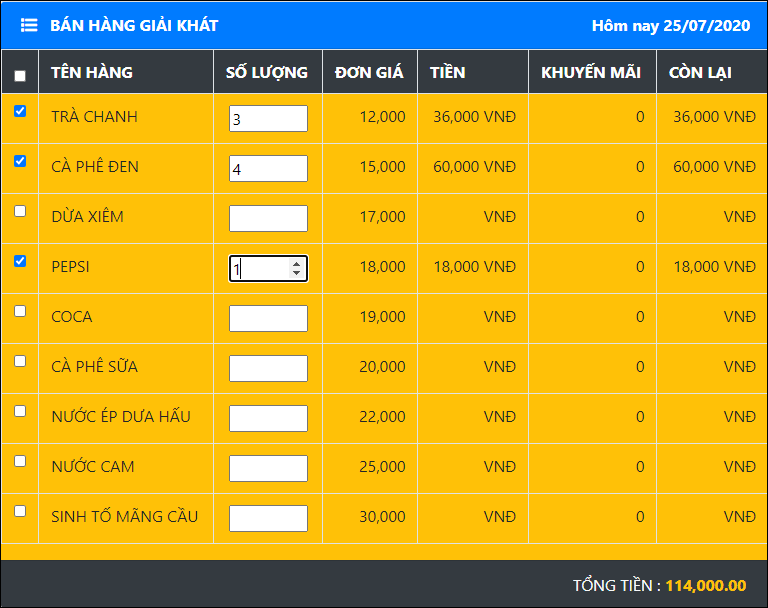
* Khóa nút lưu khi user nhập có lỗi trong form

#### Code cho nút LƯU cà button Cancel

Sinh viên thực hiện giống Lab trước:

* Khi nhắp nút **Lưu** sẽ lưu thông tin trong forrm vào danh sách, còn khi nhắp
* Khi nút **Hủy** thì cho index=-1, và tlhientai={}

### Bài 3: Lặp có sắp xếp, định dạng, chỉnh trạng thái , tính toán, sử dụng dịch vụ



#### Tạo file sanpham.js và nhập nội dung

[

{ "name": "Coca", "price": "19000" },

{ "name": "Pepsi", "price": "18000" },

{ "name": "Sinh tố mãng cầu", "price": "30000" },

{ "name": "Nước ép dưa hấu", "price": "22000" },

{ "name": "Nước cam", "price": "25000" },

{ "name": "Cà phê đen", "price": "15000" },

{ "name": "Cà phê sữa", "price": "20000" },

{ "name": "Trà chanh", "price": "12000" },

{ "name": "Dừa xiêm", "price": "17000" }

]

#### Tạo file bai3.html và chèn bootstrap, angularjs + code

<body ng-app="myapp" ng-controller="myctrl">

<div class="container">

</div>

</body>

<script>

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

app.controller("myctrl", function ($scope) {

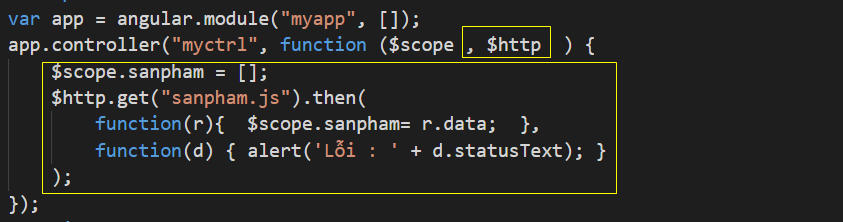
    //code xử lý cho controller

});

</script>

#### Lấy danh sách sản phẩm bằng dịch vụ $http

Lấy danh sách các sản phẩm từ file **sanpham.js** bằng cách khai báo và dùng dịch vụ $http



#### Hiện danh sách sản phẩm

* Chèn component **card** (có header, footer vào trang) của bootstrap vào trang.
* Nhập nội dung và định dạng cho header, body, footer như mẫu



* Trong card-body. xóa chữ danh sách sản phẩm và chèn table + lặp qua dãy sản phẩm. Mổi sản phẩm hiện trên 1 dòng của table , trong đó:

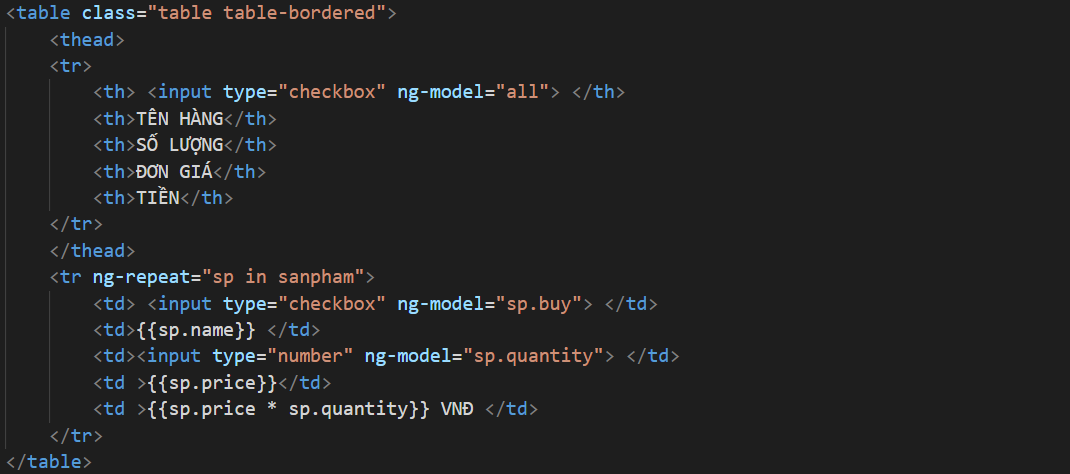
+ Ô 1: Hiện 1 checkbox, bind vào biến buy của sp.

+ Ô 2: Hiện tên sp

+ Ô 3: Hiện textfield bind với quality của sản phẩm.

+ Ổ 4: Hiện giá của sp

+ Ổ 5: Hiện tiền theo công thức giá\*soluong



* Xem thử và định dạng trang cho đẹp



#### Định dạng các cột và sắp xếp

* Định dạng (filter) bằng angular để tên sản phẩm thành chữ hoa.
* Định dạng (filter) bằng angular để cột **Đơn giá** và cột **Tiền** theo dạng **number**
* Sắp xếp các sản phẩm theo giá giảm dần.

#### Hiện ngày hôm nay

* Trong hàm xử lý của controller , khai báo biến homnay



* Hiện biến homnay ở phía sau chữ Hôm nay và định dạng cho đẹp



#### Thêm chức năng khóa/mở textfield số lượng và gán giá trị mặc định và kiểm lỗi

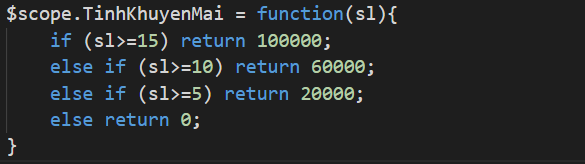
* Yêu cầu: Nếu checkbox ở đầu hàng được check thì textfield quality mới sáng lên.
* Thực hiện như sau: Thêm code sau vào textfield quality:



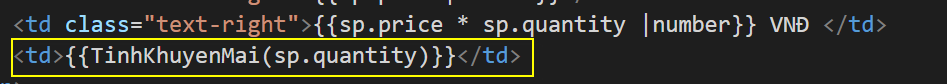
* Khi textfield số lượng sáng lên, gán giá trị o nếu value của nó là không có.
* Chỉ cho nhập số lượng là số nguyên từ 1 đến 25. Ngoài phạm vi sẽ báo lỗi.

#### Thêm cột khuyến mãi

* Trong hàm xử lý của controller, định nghĩa hàm TinhKhuyenMai với tham số là số lượng và tính khuyến mãi theo công thức như sau:



* Trong table, sau td hiện tiền, thêm tag td và gọi hàm TinhKhuyenMai



* Trong thead, thêm tag th chứa chữ **Khuyến mãi** và xem thử



* Định dạng cho cột khuyến mãi thành number và canh phải chữ

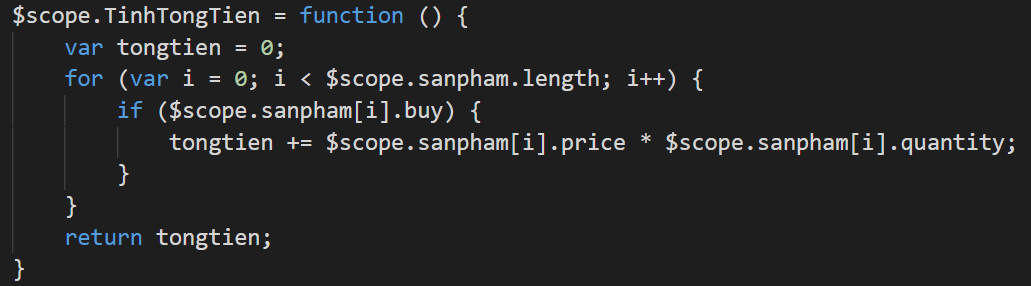
#### Thêm cột Còn lại

Trong table, sau ô hiện Khuyến mãi, thêm tag td và viết công thức tính tiền còn lại = Tiền - Khuyến mãi

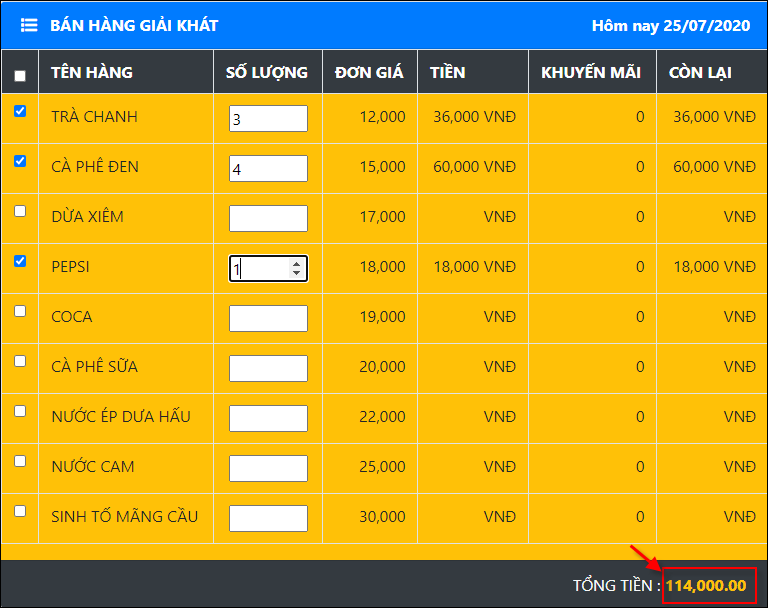


#### Tính tổng tiền

* Định nghĩa hàm tính tổng tiền



* Gọi hàm tính tổng tiền sau chữ Tổng tiền và định dạng cho đẹp

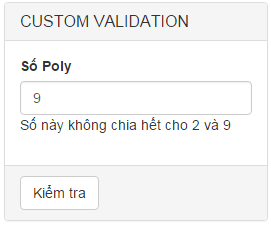


#### Thêm chức năng chọn hết

Yêu cầu: Nhắp vào checkbox ở trước chữ Tên hàng thì tất cả các checkbox bên dưới sẽ được chọn hết . Sinh viên tự thực hiện.

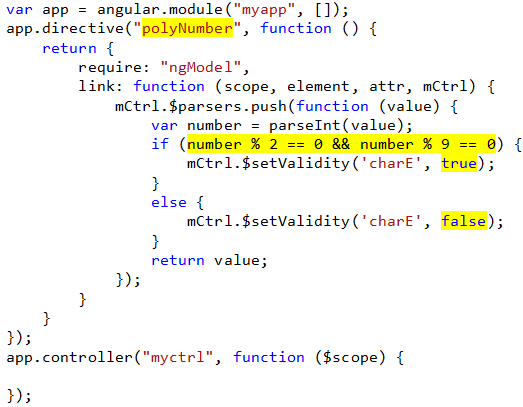
### Bài 4:

Kiểm tra số Poly: (Số Poly là số chẵn chia hết cho 9)



Hướng dẫn: Với yêu cầu này không có thuộc tính kiểm lỗi nào trong HTML5 hỗ  
trợ. Vì vậy bạn cần phải viết một chỉ thị kiểm lỗi cho riêng mình

✓ Xây dựng chỉ thị polyNumber và kiểm lỗi chia hết cho 2 và cho 9 như sau



✓ Sử dụng kiểm lỗi đã xây dựng

