# Integrated Mobile Applications Development

# Lab 5 AngularJS Controllers, Filters

## Part 1 - Controllers

1. Write an AngularJS application that has a Controller which is associated with an element.

The Controller should contain code to print out “Hello from Controller” in an alert box.

### Answer

HTML

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

</body>

JS

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

app.controller("MainController", function() {

alert("Hello from Controller");

});

1. Write an Angular JS application that has 2 Controllers – MainController and SecondController. Each Controller displays an alert saying “Hello from *ControllerName*” i.e. MainController or SecondController.

The View (HTML) has 2 div elements - one of which is associated with the MainController and one which is not associated with any Controller.

What alerts are shown and why?

### Answer

Only the “Hello from MainController” is shown, because the SecondController is not associated with any element in the view.

HTML

<body ng-app="myApp">

<div ng-controller="MainController"></div>

<div></div>

</body>

JS

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

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

alert("Hello from Main Controller");

});

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

alert("Hello from Second Controller");

});

1. Using the AngularJS application from the previous question associate the 2nd div element with SecondController.

Now add a new div element which is also associated with SecondController.

What alerts are shown and why?

### Answer

“Hello from MainController” is shown first because that is the first element on the page, so it’s Controller is executed first.

Next “Hello from SecondController” is shown because that is the second element on the page, so it’s Controller is executed second.

Finally, “Hello from SecondController” is again, because the same Controller is associated with two elements in the view.

### Answer

HTML

<body ng-app="myApp">

<div ng-controller="MainController"></div>

<div ng-controller="SecondController"></div>

<div ng-controller="SecondController"></div>

</body>

JS

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

app.controller("MainController", function() {

alert("Hello from Main Controller");

});

app.controller("SecondController", function() {

alert("Hello from Second Controller");

});

1. Write an AngularJS application that has the a $scope variable called myName which is your name.

In the view show the value of myName and its length.

Add another $scope variable called age which is your age.

In the view show the age you were last year by using an expression.

### Answer

HTML

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

Name: {{ myName }} Length: {{ myName.length }}

<br>

Age Last Year {{ age-1 }}

</body>

JS

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

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

$scope.myName="Gerard";

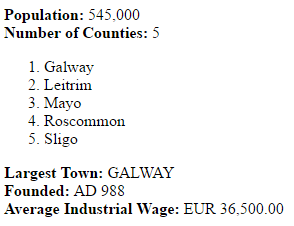
$scope.age = 18;

});

1. Write an AngularJS application that has a $scope object called Connaught, which has the following properties:

* population: 545000
* numCounties: 5
* counties: An array consisting of "Galway", "Leitrim", "Mayo", "Roscommon" and "Sligo".
* largestTown: Galway
* founded: November 12th 988
* avgIndWage: 36500.00

In the View, the following should be displayed:



only text in **bold** should be hard-coded in the view.

### Answer

HTML

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

<h3>Connaught</h3>

<span class="hc">Population: </span><span>{{ connaught.population | number}}</span>

<br>

<span class="hc">Number of Counties: </span><span>{{ connaught.numCounties }}</span>

<ol>

<li ng-repeat="county in connaught.counties">{{ county }}</li>

</ol>

<span class="hc">Largest Town: </span><span>{{ connaught.largestTown | uppercase}}</span>

<br>

<span class="hc">Founded: </span><span>{{ connaught.founded | date : 'GG y'}}</span>

<br>

<span class="hc">Average Industrial Wage: </span><span>{{ connaught.avgIndWage | currency : 'EUR ' }}</span>

</body>

JS

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

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

$scope.connaught = {

population : 545000,

numCounties : 5,

counties : ["Galway", "Leitrim", "Mayo", "Roscommon", "Sligo"],

largestTown : "Galway",

founded: new Date("988-11-12"),

avgIndWage: 36500.00

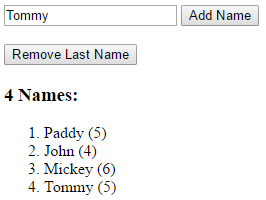
}

});

1. Write an AngularJS application that allows the user to enter a name. Each time the “Add Name” button is pressed, the name just entered is added to an array of names which is then displayed in the view along with the number of characters in the name, as an ordered list.

The heading of the list should state the number of names in the array.

A “Remove Last Name” button should remove the last item in the array when pressed.



### Answer

HTML

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

<input ng-model="name"></input>

<button ng-click="add()">Add Name</button>

<br>

<br>

<button ng-click="remove()">Remove Last Name</button>

<h3>{{ listOfNames.length }} Names:</h3>

<ol>

<li ng-repeat="name in listOfNames track by $index">{{ name }} ({{ name.length }}) </li>

</ol>

</body>

JS

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

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

$scope.name = "";

$scope.listOfNames = [];

$scope.add = function() {

$scope.listOfNames.push($scope.name);

};

$scope.remove = function() {

var x = $scope.listOfNames.pop();

if (typeof x === "undefined") {

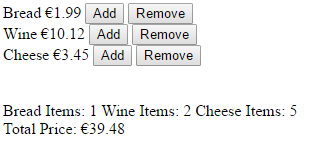
alert("Array Empty");

}

}

});

1. Write an AngularJS Shopping Basket Application that looks as follows:



When the *Add* button is pressed, the relevant product is added to the shopping basket, the item count for that item increases by one, and the total price of items in the shopping basket is also increased by the price of the product.

When the *Remove* button is pressed, the relevant product is removed from the shopping basket – but only if at least one such product already exists in the shopping basket, the item count for that item decreases by one, and the total price of items in the shopping basket is also decreased by the price of the product.

The Product Name and Price can be hardcoded in the JS (not the HTML) part of the application.

### Answer

HTML

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.5/angular.min.js"></script>

</head>

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

{{ Bread.name }} {{ Bread.price | currency : "€" }}

<button ng-click="add(Bread.name)">Add</button>

<button ng-click="remove(Bread.name)">Remove</button>

<br/>

{{ Wine.name }} {{ Wine.price | currency : "€" }}

<button ng-click="add(Wine.name)">Add</button>

<button ng-click="remove(Wine.name)">Remove</button>

<br/>

{{ Cheese.name }} {{ Cheese.price | currency : "€" }}

<button ng-click="add(Cheese.name)">Add</button>

<button ng-click="remove(Cheese.name)">Remove</button>

<br/>

<br/><br/>

{{ Bread.name }} Items: {{ shoppingList.breadItems }}

{{ Wine.name }} Items: {{ shoppingList.wineItems }}

{{ Cheese.name }} Items: {{ shoppingList.cheeseItems }}

<br/>

Total Price: {{ shoppingList.totalPrice | currency : "€" }}

</body>

</html>

JS

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

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

$scope.Bread = {

name : "Bread",

price : 1.99

};

$scope.Wine = {

name : "Wine",

price : 10.12

};

$scope.Cheese = {

name : "Cheese",

price : 3.45

}

$scope.shoppingList = {

breadItems : 0,

wineItems : 0,

cheeseItems : 0,

totalPrice : 0

};

$scope.add = function(name) {

switch(name) {

case $scope.Bread.name:

$scope.shoppingList.breadItems ++;

$scope.shoppingList.totalPrice += $scope.Bread.price;

break;

case $scope.Wine.name:

$scope.shoppingList.wineItems ++;

$scope.shoppingList.totalPrice += $scope.Wine.price;

break;

case $scope.Cheese.name:

$scope.shoppingList.cheeseItems ++;

$scope.shoppingList.totalPrice += $scope.Cheese.price;

break;

}

}

$scope.remove = function(name) {

switch(name) {

case $scope.Bread.name:

if ($scope.shoppingList.breadItems > 0) {

$scope.shoppingList.breadItems --;

$scope.shoppingList.totalPrice -= $scope.Bread.price;

}

break;

case $scope.Wine.name:

if ($scope.shoppingList.wineItems > 0) {

$scope.shoppingList.wineItems --;

$scope.shoppingList.totalPrice -= $scope.Wine.price;

}

break;

case $scope.Cheese.name:

if ($scope.shoppingList.cheeseItems > 0) {

$scope.shoppingList.cheeseItems --;

$scope.shoppingList.totalPrice -= $scope.Cheese.price;

}

break;

}

}

});

1. Write an AngularJS Application that looks as follows:



And has the following behaviour:

* The *Get Name* button, when pressed, allows the user to enter their name/email or any string, which is then displayed next to the button.
* The First character of the string entered, along with its length, is displayed.
* An input field allows the user to enter the characters of the name to be printed e.g. Every 1st, 2nd 3rd etc. character.
* If an invalid/negative number is entered the entire string is displayed.
* If a valid number is entered, e.g. 3, every 3rd character of the string should be displayed.

### Answer

HTML

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.5/angular.min.js"></script>

</head>

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

<button ng-click="getName()">Get Name</button>

{{ name }}

<p>First Char: {{ name.charAt(0) }}</p>

<p>Length: {{ name.length }}</p>

<p>Show every <input ng-model="charsToShow"/> character(s) <button ng-click="showChars()">Go</button></p>

{{ shortenedName }}

</body>

</html>

JS

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

app.controller('MainController', function($scope) {

$scope.name = "";

$scope.shortenedName = "";

$scope.charsToShow = 0;

$scope.getName = function() {

$scope.name = prompt("Enter your name");

$scope.shortenedName = "";

$scope.charsToShow = 0;

}

$scope.showChars = function() {

var charsToShowInt = parseInt($scope.charsToShow);

if (isNaN(charsToShowInt) || charsToShowInt < 0) {

$scope.shortenedName = $scope.name;

return;

}

if (charsToShowInt === 0) {

$scope.shortenedName = "";

} else {

var tempName = "";

for (var i=charsToShowInt-1; i<$scope.name.length; i+=charsToShowInt) {

tempName = tempName.concat($scope.name.charAt(i));

}

$scope.shortenedName = tempName;

}

}

});