



# ANGULAR

# JS



[www.facebook.com/nunetc](https://www.facebook.com/nunetc)



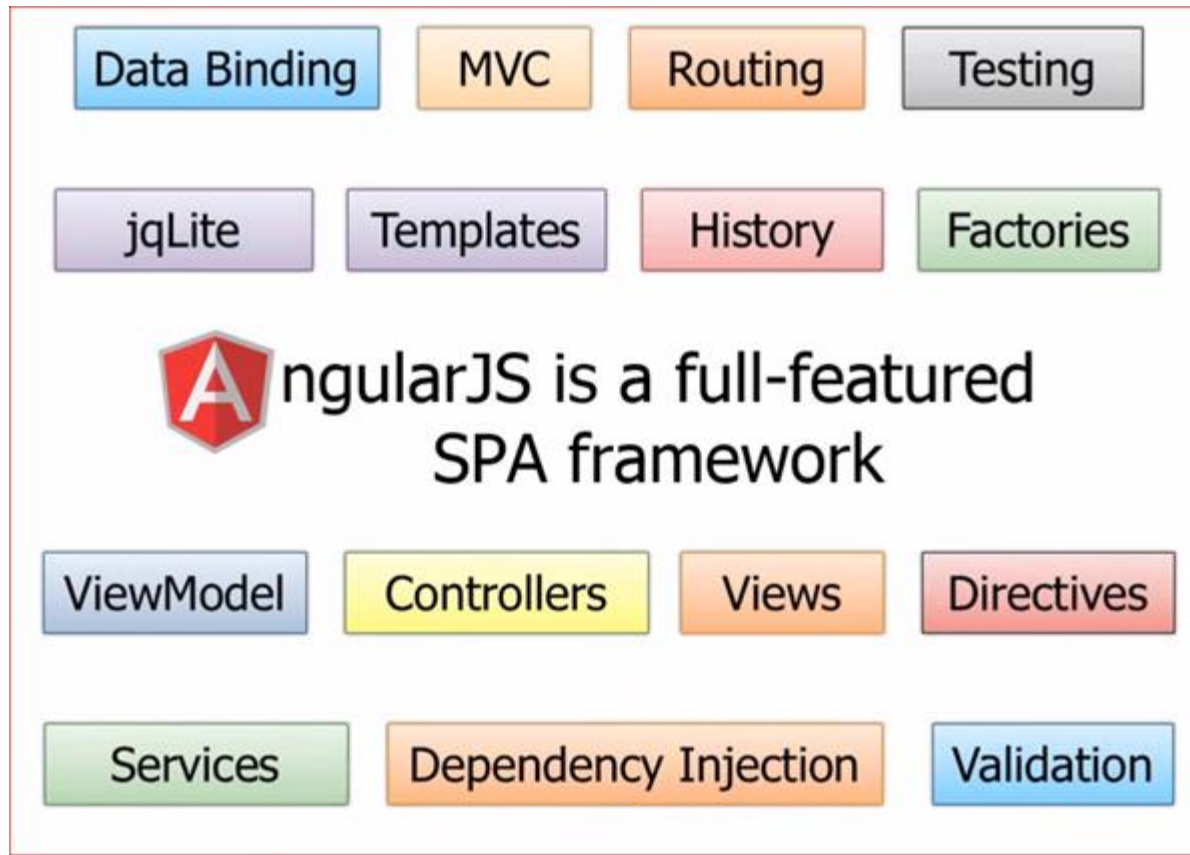
[nunetc](#)

# WHAT IS ANGULARJS ?

Client-side Javascript Framework เพื่อเพิ่มความสามารถของ HTML สำหรับสร้างเว็บสมัยใหม่แบบ Single Page Application พัฒนาจากบริษัท Google



# FEATURE



# WORKSHOP 1

**Hello World**

<http://localhost:8888/workshop1/index.html>

# WORKSHOP 1

Directive

Controller

```
1 <!DOCTYPE html>
2 <html ng-app="myApp">
3 <head>
4   <meta charset="UTF-8">
5   <title>Angular</title>
6 </head>
7 <body>
8   <h1>Workshop1 - Hello World</h1>
9
10  <div ng-controller="HelloController">
11    <input type="text" ng-model="name">
12
13    <p ng-show="name">
14      Hello {{name}}
15    </p>
16  </div>
17
```

Directive

Expression

# WORKSHOP 1

Create Module 'myApp'

Dependency Injection '\$scope' Service

```
19  <script>
20      angular.module('myApp', [])
21
22      .controller('HelloController', ['$scope', function($scope) {
23          $scope.name = 'World';
24      }]);
25  </script>
```

Create Controller 'HelloController'

# CONTROLLER

- ส่วนที่กำหนดการทำงานต่าง ๆ ของแอปพลิเคชัน
- เราจะได้ Logic ต่าง ๆ และการดึงข้อมูลจาก Web Service ที่ส่วนนี้

# SCOPE

- ตัวเชื่อมระหว่าง Controller และ View (HTML)
- สามารถสืบทอดจาก Parent Scope



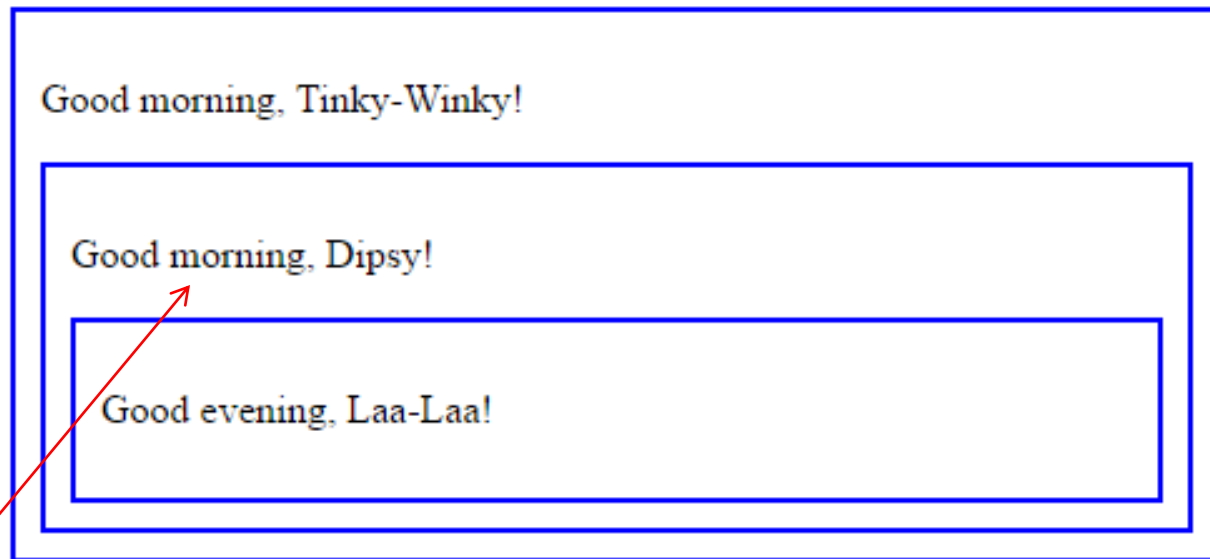


# WORKSHOP 2

## Scope Inheritance

<http://localhost:8888/workshop2/index.html>

# WORKSHOP 2



Inheritance from 'ParentController'

# EXPRESSION

การแทรกโค้ด Javascript อย่างง่ายใน HTML โดยใช้เครื่องหมาย `{{...}}` เช่น

- `{{1+1}}` // 2
- `{{'Hi '+'Bob'}}` // Hi Bob
- `{{user.name}}` // Object
- `{{items[index]}}` // Array
- `{{functionName()}}` // Function Call

# DIRECTIVE

- Tag หรือ Attribute ใน HTML ซึ่งกระตุ้นให้ AngularJS ทำงาน
- AngularJS มาพร้อม built-in directive เช่น ng-repeat, ng-model, ng-bind, a, input, select, textarea, ...
- Directive สามารถเขียนได้หลากหลายแบบ ดังนี้
  - ng-model
  - ng:model
  - ng\_model
  - data-ng-model
  - x-ng-model

# WORKSHOP 3

## Sample Built-in Directives

<http://localhost:8888/workshop3/index.html>

# WORKSHOP 3

```
11 <input type="text" ng-model="name">
12
13 <p ng-show="name">
14   ng-show Hello {{name}}
15 </p>
16 <p ng-if="name">
17   ng-if Hello {{name}}
18 </p>
19
20 <hr>
21 <p>{{name}}</p>
22 <p ng-bind="name"></p>
23
24 <hr>
25 <ul>
26   <li ng-repeat="user in users">{{user.name}}</li>
27 </ul>
28
29 <hr>
30 <button ng-click="save()">Save</button>
```

# FILTER

จัดการแสดงผลค่าของ Expression

สามารถเรียกใช้ได้ ใน HTML, Controller, Service เช่น

- `{{1500.5 | number:2}}` `// 1,500.50`
- `{{'hello' | uppercase}}` `// HELLO`
- `{{user.name | lowercase}}` `// tinky-winky`

# WORKSHOP 4

## Custom Filter

<http://localhost:8888/workshop4/index.html>



# WORKSHOP 4

Create filter 'capitalize'

Optional Parameter

```
3 .filter('capitalize', function() {  
4     return function(input, isCapitalizeWord) {  
5         if (angular.isString(input)) {  
6             if (isCapitalizeWord) {  
7                 return input.replace(/(?:^|\s)\S/g, function(a) { return a.toUpperCase(); });  
8             }  
9             else {  
10                return input.substring(0, 1).toUpperCase() + input.substring(1);  
11            }  
12        }  
13        else {  
14            return '';  
15        }  
16    };  
17 })
```

```
<p>  
  Capitalize -> {{data | capitalize}}  
</p>
```

```
<p>  
  Capitalize Word -> {{data | capitalize:true}}  
</p>
```

Parameter

# ROUTING

AngularJS ใช้ module ngRoute ในการกำหนดการแสดงผลหน้า  
เว็บไซต์แบบ Single Page Application

โดยการจับคู่ระหว่าง Url, Template และ Controller

# WORKSHOP 5

## Routing

<http://localhost:8888/workshop5/index.html>


# WORKSHOP 5

Dynamic Content



```
10 <div ng-view>
11
12 </div>
13
14 <div>
15   <hr>
16   This is static footer
17 </div>
18
19 <script src="/lib/angular/angular.js"></script>
20 <script src="/lib/angular/angular-route.js"></script>
21 <script src="app.js"></script>
```

ngRoute module



# WORKSHOP 5

Config routing

Dependency module

Route parameter

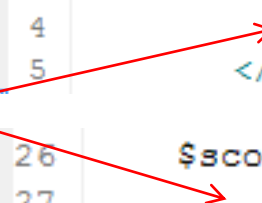
```
1 angular.module('myApp', ['ngRoute'])
2
3 .config(['$routeProvider', function($routeProvider) {
4     $routeProvider
5         .when('/list', {
6             templateUrl: 'list.html',
7             controller: 'ListController'
8         })
9         .when('/detail/:name', {
10             templateUrl: 'detail.html',
11             controller: 'DetailController'
12         })
13         .otherwise({
14             redirectTo: '/list'
15         });
16 } ])
```

# WORKSHOP 5

Change Route

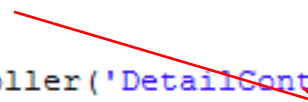
```
30 <li ng-repeat="user in users">
4   <a href="#/detail/{{user.name}}">{{user.name}}</a>
5 </li>

26 $scope.gotoDetail = function(user) {
27   $location.path('detail/'+user.name);
28 }
```



Get route parameter value

```
31 .controller('DetailController', ['$scope', '$routeParams', '$window',
32   $scope.userName = $routeParams.name;
33
34   $scope.back = function() {
35     $window.history.back();
36   }
37 } ])
```



# SERVICE

- Javascript Object ที่สามารถ Dependency Injection ใช้ใน controller และ service อื่น ๆ ได้
- สามารถใช้เพื่อ share data ระหว่าง controller
- Built-in services จะขึ้นต้นด้วย \$ เสมอ เช่น \$scope, \$http
- เป็น Singleton

# \$http

Service หลักของ AngularJS ใช้สำหรับติดต่อกับ HTTP Server ผ่าน XMLHttpRequest

```
$http.get('/mock-webservice/users.json')
    .then(function successCallback(response) {
        // success callback
    }, function errorCallback(response) {
        // error callback
    });

$http.post('/mock-webservice/user', $scope.user)
    .then(function successCallback(response) {
        // success callback
    }, function errorCallback(response) {
        // error callback
    });
```



# \$http

- ทำงานแบบ Asynchronous
- ประกอบด้วย method get, post, put, delete, head, patch, jsonp
- สามารถ caching GET และ JSONP request ได้
- สามารถ config default http headers ได้ผ่าน \$httpProvider, \$http

# WORKSHOP 6

**\$http Service and \$http Interceptor**

<http://localhost:8888/workshop6/index.html>

# WORKSHOP 7

## Custom Service

<http://localhost:8888/workshop7/index.html>



# WORKSHOP 7

Create 'NotifyService' Service

Call 'SweetAlert' Library

```
3  .factory('NotifyService', [function() {  
4      var notify = {};  
5  
6      notify.alert = function(msg) {  
7          swal(msg)  
8      }  
9  
10     notify.success = function(msg) {  
11         swal('Good job!', msg, 'success');  
12     }  
13  
14     notify.error = function(msg) {  
15         swal('Oops...', msg, 'error');  
16     }  
17  
18     return notify;  
19 }])
```


Return Service Object

# WORKSHOP 7

Inject 'NotifyService' Service



```
36 .controller('ListController', ['$scope', '$location', 'NotifyService', function($scope, $location, NotifyService) {
37     $scope.users = [
38         {name: 'Tinky-Winky'},
39         {name: 'Dipsy'},
40         {name: 'Laa-Laa'},
41         {name: 'Po'}
42     ];
43
44     $scope.gotoDetail = function(user) {
45         $location.path('detail/'+user.name);
46     }
47
48     $scope.alert = function(msg) {
49         NotifyService.alert(msg);
50     }
51 }])
```



Call 'NotifyService' Function

