Angular Factory and Service

Jogesh K. Muppala

THE DEPARTMENT OF
COMPUTER SCIENCE & ENGINEERING
計算機科學及工程學系

香港科技大學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

Angular Services

- Substitutable objects wired together using DI
- Allows organizing and sharing code across an app
- Lazily instantiated
- Singletons

Angular’s Built-in Services

- Built-in services always start with $
  - e.g., $http , $scope, $rootScope, $location, $parse, $templateCache, $animate, $injector
  - Inject them using DI

Creating Your Own Services

- Five functions that declare services
  - service()
  - factory()
  - provider()
  - constant()
  - value()

Angular Factory

```
angular.module('confusionApp')
        .factory('menuFactory', function() {


                var menufac = {};
                var dishes = [ … ];


                menufac.getDishes = function(){
                        return dishes;
                };
                menufac.getDish = function (index)
                {
                        return dishes[index];
                };
                return menufac;
        });
```

- Usage:

```
angular.module('confusionApp')
.controller('MenuController', ['$scope',
        'menuFactory', function($scope, menuFactory)
{
        $scope.dishes= menuFactory.getDishes();


}]);
```

Angular Service

```
angular.module('confusionApp')
        .service('menuFactory', function() {
                var dishes = [ … ];

                this.getDishes = function(){
                        return dishes;
                };
                this.getDish = function (index)
                {
                        return dishes[index];
                };
        });
```

- Usage is same as factory:

```
angular.module('confusionApp')
.controller('MenuController', ['$scope',
        'menuFactory', function($scope, menuFactory)
{
        $scope.dishes=  menuFactory.getDishes();

 }]);
```

Exercise: Angular Factory and Service

- Use dependency injection to enable the use of custom services in Angular modules and controllers
- Design custom services using the Angular factory and service approaches