Exercise (Instructions): Client-Server Communication using $http

Objectives and Outcomes

In this exercise, you will learn about how to use the built-in Angular $http service to communicate with the server and retrieve the data from the server. At the end of this exercise, you will be able to:

- Use the $http service to retrieve data from a server using the $http.get() method
- Use the retrieved data to render the page in your web application

Updating Services

- Open *services.js* to update it to retrieve data from the server. First add a constant to the Angular module as follows:

```
angular.module('confusionApp')
        .constant("baseURL","http://localhost:3000/")
```

- Next you will do dependency injection into the menuFactory service of the $http service and baseURL constant as follows:

```
        .service('menuFactory', ['$http', 'baseURL', function($http,baseURL) {

            . . .

        }])
```

Make sure to put the closing] at the end of the service function, to close the dependency array.

- Then, you will go into the *menuFactory* service and delete the *dishes* object from there. You will download the dishes information from the server.
- Next, update the two method in the *menuFactory* service to use the $http service as follows:

```
                this.getDishes = function(){
                                    return $http.get(baseURL+"dishes");
                                };
                    this.getDish = function (index) {
                                    return $http.get(baseURL+"dishes/"+index);
```

```
        };
```

- Save the *services.js* file and then open *controllers.js* file.

Updating the Controllers

- Update the code in the *MenuController* to retrieve the data from the service as follows:

```
        $scope.dishes= [];
        menuFactory.getDishes()
        .then(
            function(response) {
                $scope.dishes = response.data;
            }
        );
```

- Similarly update the *DishDetailController* as follows:

```
        $scope.dish = {};
                    menuFactory.getDish(parseInt($stateParams.id,10))
        .then(
            function(response){
                $scope.dish = response.data;
                $scope.showDish=true;
            }
        );
```

- Also update the IndexController to retrieve the data for the dish from the server as follows:

```
                    $scope.dish = {};

                    menuFactory.getDish(0)
                    .then(
                        function(response){
                            $scope.dish = response.data;
                            $scope.showDish = true;
                        }
                    );
```

- Save *controllers.js* and then open *menu.html*
- Update the following code in menu.html as shown below to remove the _ from the id:

```
<a ui-sref="app.dishdetails({id: dish.id})">
```

- Save the changes in *menu.html* and then view the web page in the browser.

Conclusions

In this exercise, we updated the app to retrieve the data from the server using the $http.get() request.