

JavaScript + Angular

Part 8

Routing, Providers, Forms

Author: Andrey Kucherenko

routing

\$route

```
npm install angular-route --save
```

```
angular.module('ngRouteExample', ['ngRoute'])
```

\$route

```
$routeProvider
  .when('/Book/:bookId', {
    templateUrl: 'book.html',
    controller: 'BookController',
    resolve: {
      // I will cause a 1 second delay
      delay: function($q, $timeout) {
        var delay = $q.defer();
        $timeout(delay.resolve, 1000);
        return delay.promise;
      }
    }
  })
  .when('/Book/:bookId/ch/:chapterId', {
    templateUrl: 'chapter.html',
    controller: 'ChapterController'
  });

$locationProvider.html5Mode(true);
```

\$route

```
<a href="Book/Moby">Moby</a> |  
<a href="Book/Moby/ch/1">Moby: Ch1</a> |  
<a href="Book/Gatsby">Gatsby</a> |  
<a href="Book/Gatsby/ch/4?key=value">Gatsby: Ch4</a> |  
<a href="Book/Scarlet">Scarlet Letter</a><br/>
```

ui-router

```
npm install angular-ui-router --save
```

```
angular.module('myApp', ['ui.router']);
```


ui-router

```
<!-- index.html -->
<body>
  <div ui-view></div>
  <!-- We'll also add some navigation: -->
  <a ui-sref="state1">State 1</a>
  <a ui-sref="state2">State 2</a>
</body>
```

ui-router

```
<!-- partials/state1.html -->
<h1>State 1</h1>
<hr/>
<a ui-sref="state1.list">Show List</a>
<div ui-view></div>

<!-- partials/state2.html -->
<h1>State 2</h1>
<hr/>
<a ui-sref="state2.list">Show List</a>
<div ui-view></div>
```


ui-router

```
//  
// For any unmatched url, redirect to /state1  
$urlRouterProvider.otherwise("/state1");  
//  
// Now set up the states  
$stateProvider  
  .state('state1', {  
    url: "/state1",  
    templateUrl: "partials/state1.html"  
  })  
  .state('state1.list', {  
    url: "/list",  
    templateUrl: "partials/state1.list.html",  
    controller: function($scope) {  
      $scope.items = ["A", "List", "Of", "Items"];  
    }  
  })  
  .state('state2', {  
    url: "/state2",  
    templateUrl: "partials/state2.html"  
  })  
  .state('state2.list', {  
    url: "/list",  
    templateUrl: "partials/state2.list.html",  
    controller: function($scope) {  
      $scope.things = ["A", "Set", "Of", "Things"];  
    }  
  })  
});
```

ui-router

```
<!-- index.html -->
<body>
  <div ui-view="viewA"></div>
  <div ui-view="viewB"></div>
  <!-- Also a way to navigate -->
  <a ui-sref="route1">Route 1</a>
  <a ui-sref="route2">Route 2</a>
</body>
```

ui-router

```
myApp.config(function($stateProvider) {
  $stateProvider
    .state('index', {
      url: "",
      views: {
        "viewA": { template: "index.viewA" },
        "viewB": { template: "index.viewB" }
      }
    })
    .state('route1', {
      url: "/route1",
      views: {
        "viewA": { template: "route1.viewA" },
        "viewB": { template: "route1.viewB" }
      }
    })
    .state('route2', {
      url: "/route2",
      views: {
        "viewA": { template: "route2.viewA" },
        "viewB": { template: "route2.viewB" }
      }
    })
})
```

Tasks

- Create application with tag game, timer and calculator on separate views
- Use multiple views for show statistic and apps

providers

1. `app.config(() => {})`
2. `app.run(() => {})`
3. directive's compile functions
(if they are found in the DOM)
4. `app.controller((Services, Factories) => {})`
5. directive's link functions
(again, if found)

```
app.config(($http) => {});
```

```
angular.js:68 Uncaught Error: [$injector:modulerr] Failed to instantiate module app due to  
Error: [$injector:unpr] Unknown provider: $http
```

```
http://errors.angularjs.org/1.5.8/$injector/unpr?p0=%24http
```

```
at http://192.168.1.103:8080/lib/bundle.js:3543:12
```

```
at http://192.168.1.103:8080/lib/bundle.js:7986:19
```

```
at getService (http://192.168.1.103:8080/lib/bundle.js:8139:39)
```

```
at injectionArgs (http://192.168.1.103:8080/lib/bundle.js:8163:58)
```

```
at Object.invoke (http://192.168.1.103:8080/lib/bundle.js:8185:18)
```

```
at runInvokeQueue (http://192.168.1.103:8080/lib/bundle.js:8086:35)
```

```
at http://192.168.1.103:8080/lib/bundle.js:8095:11
```

```
at forEach (http://192.168.1.103:8080/lib/bundle.js:3796:20)
```

```
at loadModules (http://192.168.1.103:8080/lib/bundle.js:8076:5)
```

```
at createInjector (http://192.168.1.103:8080/lib/bundle.js:7998:19)
```

```
http://errors.angularjs.org/1.5.8/$injector/modulerr?p0=Lesson5App&p1=Error...
```



```
app.run(($translateProvider) => {});
```

```
angular.js:68Uncaught Error: [$injector:unpr]
```

```
Unknown provider: $translateProviderProvider <- $translateProvider
```

```
http://errors.angularjs.org/1.5.8/\$injector/unpr?p0=%24translateProviderProvider
```

```
app.controller('MyController1', function ($translateProvider) {});
```

```
angular.js:13920Error: [$injector:unpr]
```

```
Unknown provider: $translateProviderProvider <- $translateProvider <- MyController1
```

```
http://errors.angularjs.org/1.5.8/\$injector/unpr?p0=%24translateProviderProvider
```

```
    at angular.js:68
```

```
    at angular.js:4511
```

```
    at Object.getService [as get] (angular.js:4664)
```

```
    at angular.js:4516
```

```
    at getService (angular.js:4664)
```

```
    at injectionArgs (angular.js:4688)
```

```
    at Object.invoke (angular.js:4710)
```

```
    at $controllerInit (angular.js:10354)
```

```
    at nodeLinkFn (angular.js:9263)
```

```
    at compositeLinkFn (angular.js:8620)
```

```
app.run(($translateProvider) => {});
```

```
angular.js:68Uncaught Error: [$injector:unpr]
```

```
Unknown provider: $translateProviderProvider <- $translateProvider
```

```
http://errors.angularjs.org/1.5.8/\$injector/unpr?p0=%24translateProviderProvider
```

```
app.controller('MyController1', function ($translateProvider) {});
```

```
angular.js:13920Error: [$injector:unpr]
```

```
Unknown provider: $translateProviderProvider <- $translateProvider <- MyController1
```

```
http://errors.angularjs.org/1.5.8/\$injector/unpr?p0=%24translateProviderProvider
```

```
    at angular.js:68
```

```
    at angular.js:4511
```

```
    at Object.getService [as get] (angular.js:4664)
```

```
    at angular.js:4516
```

```
    at getService (angular.js:4664)
```

```
    at injectionArgs (angular.js:4688)
```

```
    at Object.invoke (angular.js:4710)
```

```
    at $controllerInit (angular.js:10354)
```

```
    at nodeLinkFn (angular.js:9263)
```

```
    at compositeLinkFn (angular.js:8620)
```

\$provide

```
var app = angular.module('app', []);

app.config(($provide) => {
  $provide.constant('CONSTANT_URL', '/test');

  $provide.service('movie', function () {
    this.title = 'The Matrix';
  });
});

app.controller('ctrl', function (CONSTANT_URL, movie) {});
```

\$provide

```
var app = angular.module('app', []);

app.config(($provide) => {
  $provide.value('movieTitle', 'The Matrix');
  $provide.decorator('movieTitle', ($delegate) => {
    return $delegate + ' - starring Keanu Reeves';
  });
});

app.controller('ctrl', function (CONSTANT_URL, movie, movieTitle) {});
```

```
var app = angular.module('app', []);

app.provider('movie', function () {
  var version;
  return {
    setVersion: function (value) {
      version = value;
    },
    $get: function () {
      return {
        title: 'The Matrix' + ' ' + version
      }
    }
  }
});

app.config(function (movieProvider) {
  movieProvider.setVersion('Reloaded');
});

app.controller('ctrl', function (movie) {
  expect(movie.title).toEqual('The Matrix Reloaded');
});
```

Tasks

- Create provider for service calls and configure them and config stage

forms


```
<input type="text" ng-model="firstname">
```

```
var app = angular.module('myApp', []);  
app.controller('formCtrl', function($scope) {  
    $scope.firstname = "John";  
});
```

```
<form>  
  Check to show a header:  
  <input type="checkbox" ng-model="myVar"  
    ng-true-value="yes" ng-false-value="no">  
</form>
```

```
<form>
```

Pick a topic:

```
<input type="radio" ng-model="myVar" value="dogs">Dogs
```

```
<input type="radio" ng-model="myVar" value="tuts">Tutorials
```

```
<input type="radio" ng-model="myVar" value="cars">Cars
```

```
</form>
```

```
<form>
Select a topic:
<select ng-model="myVar">
  <option value="">
  <option value="dogs">Dogs
  <option value="tuts">Tutorials
  <option value="cars">Cars
</select>
</form>
```

```
<div ng-controller="MyController" >
  <form>
    <select ng-model="myForm.car"
      ng-options="car.id as car.name for car in myForm.options">
      <option value="">Please choose a car</option>
    </select>
  </form>

  <div>
    {{myForm.car}}
  </div>
</div>
```

```
angular.module("myapp", [])
  .controller("MyController", function($scope) {
    $scope.myForm = {};
    $scope.myForm.car = "nissan";

    $scope.myForm.options = [
      { id : "nissan", name: "Nissan" }
      , { id : "toyota", name: "Toyota" }
      , { id : "fiat" , name: "Fiat" }
    ];

  } );
```

```
<div ng-controller="MyController" >
  <form>
    <select multiple="true" ng-model="myForm.car"
      ng-options="obj.id as obj.name
        for obj in myForm.options
        group by obj.type">

    </select>
  </form>

  <div>
    {{myForm.car}}
  </div>
</div>

<script>
  angular.module("myapp", [])
    .controller("MyController", function($scope) {
      $scope.myForm = {};
      $scope.myForm.car = ["nissan"];

      $scope.myForm.options = [
        { id : "nissan", name: "Nissan" }
        ,{ id : "toyota", name: "Toyota" }
        ,{ id : "fiat" , name: "Fiat" }
      ];

    } );
</script>
```

```
<form>
  <input type="text" id="name"
        ng-model="myForm.name" ng-minlength="5" ng-maxlength="12"> Name <br/>
  <input type="text" id="name" ng-model="myForm.name" ng-pattern="/^\d+$/"> Name <br/>
  <input type="text" id="name" ng-model="myForm.name" ng-required> Name <br/>
</form>
```



```
<form ng-submit="myForm.submitTheForm( )">
  <input type="text" id="name" ng-model="myForm.name" ng-

  <select ng-model="myForm.car">
    <option value="nissan">Nissan</option>
    <option value="toyota">Toyota</option>
    <option value="fiat">Fiat</option>
  </select>

  <input type="submit" value="Submit Form">
</form>
```

Tasks

- Create tick-tack-toe
- Make user registration for your tick-tack-toe game app
- Registration form should have following fields: name, e-mail, password, re-password, avatar, age, role
- User's data should be saved on server
- Statistic should saved on server



Pet Project