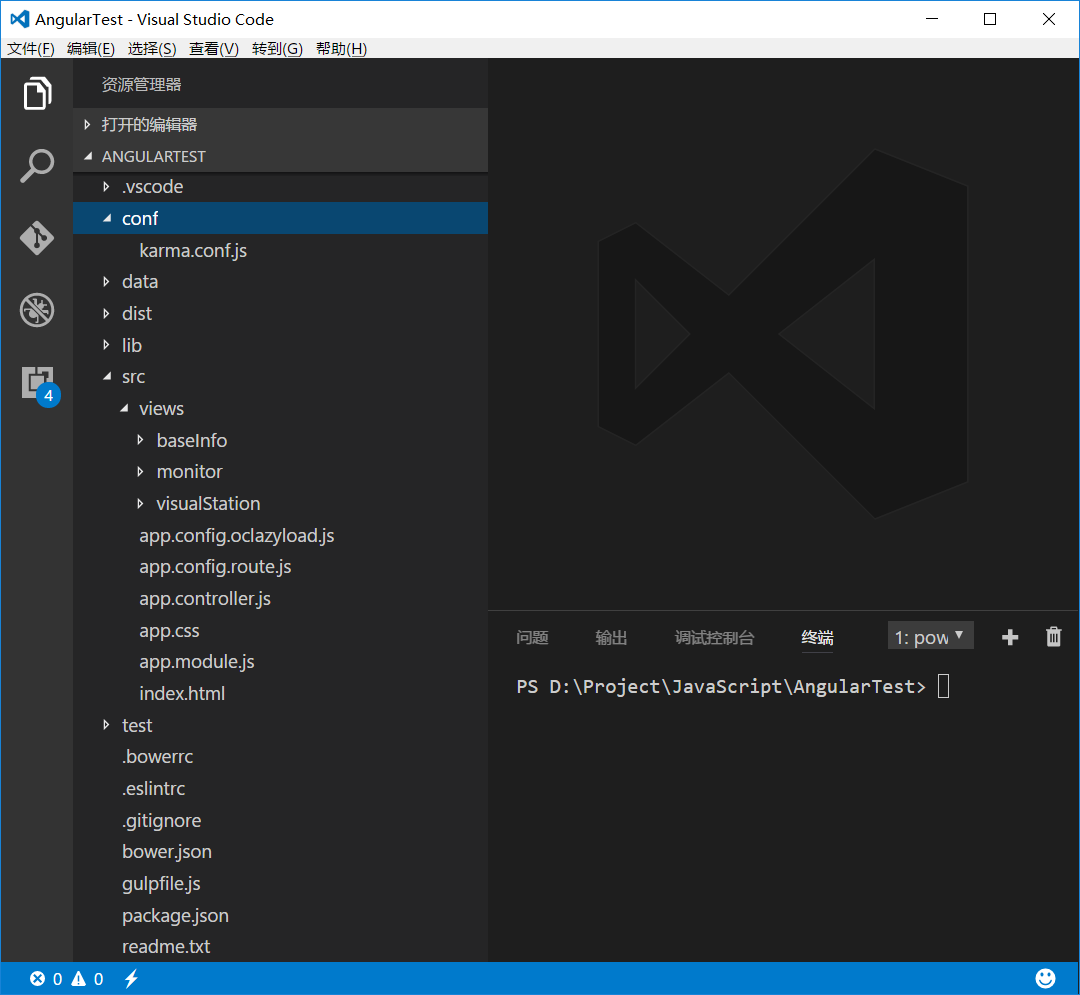
# 前端框架搭建

## 目录结构



## 前端编译

### js文件的格式与语法检查

gulp.task('eslint', function () {

    return gulp.src('./src/\*\*/\*.js')

        .pipe(eslint())

        .pipe(eslint.format())

        .pipe(eslint.failAfterError())

        .pipe(gulp.dest('./dist'))

        .pipe(connect.reload());

});

### styl文件的编译

// stylus

gulp.task('css', function () {

    return gulp.src('./src/\*\*/\*.styl')

        .pipe(stylus())

        .pipe(gulp.dest('./dist'))

        .pipe(connect.reload());

});

### html文件的输出

// html

gulp.task('html', function () {

    gulp.src('./src/views/\*\*/\*.html')

    .pipe(gulp.dest('./dist/views'))

        .pipe(connect.reload());

});

### 将CSS与JS的依赖注入HTML中

#### HTML文件设置

<!DOCTYPE html>

<html lang="zh-cn">

<head>

  <meta charset="utf-8">

  <meta name="description" content="">

  <meta http-equiv="X-UA-Compatible" content="IE=edge">

  <meta name="viewport" content="width=device-width, inital-scale=1.0">

  <title>WebGIS</title>

  <!-- build:css css/vendor.min.css -->

  <!-- bower:css -->

  <!-- endbower -->

  <!-- endbuild -->

  <!-- build:css css/style.min.css -->

  <!-- inject:css -->

  <!-- endinject -->

  <!-- endbuild -->

</head>

<body ng-app="app" ng-controller="appController as $ctrl">

  <nav class="navbar navbar-default">

    <div class="container-fluid">

      <ul class="nav navbar-nav">

        <li ng-class="{'active': $ctrl.selectedItem == 'monitor'}">

          <a href="#" ui-sref="monitor">monitor</a>

        </li>

        <li ng-class="{'active': $ctrl.selectedItem == 'baseInfo'}">

          <a href="#" ui-sref="baseInfo">baseInfo</a>

        </li>

        <li ng-class="{'active': $ctrl.selectedItem == 'visualStation'}">

          <a href="#" ui-sref="visualStation">visualStation</a>

        </li>

        <li class="dropdown">

          <a href="#" class="dropdown-toggle" data-toggle="dropdown" role="button" aria-haspopup="true" aria-expanded="false">

            Dropdown <span class="caret"></span>

          </a>

          <ul class="dropdown-menu">

            <li><a href="#">Action</a></li>

            <li><a href="#">Another action</a></li>

            <li><a href="#">Something else here</a></li>

            <li role="separator" class="divider"></li>

            <li><a href="#">Separated link</a></li>

            <li role="separator" class="divider"></li>

            <li><a href="#">One more separated link</a></li>

          </ul>

        </li>

      </ul>

    </div>

  </nav>

  <div class="main-container">

    <ui-view></ui-view>

  </div>

  <!-- build:js js/vendor.min.js -->

  <!-- bower:js -->

  <!-- endbower -->

  <!-- endbuild -->

  <!-- build:js js/app.min.js -->

  <!-- inject:js -->

  <!-- endinject -->

  <!-- endbuild -->

</body>

</html>

#### Bower管理的库文件的注入

// wiredep

gulp.task('wiredep', function () {

    gulp.src('./dist/index.html')

        .pipe(wiredep({

            directory: './lib',

            exclude: [/requirejs/]

        }))

        .pipe(gulp.dest('./dist'));

});

#### 自定义文件的注入

// inject

gulp.task('inject', ['wiredep'], function () {

    var utilStream = gulp.src(['./dist/\*.js', '!./dist/app.module.js'], {

        read: false

    });

    var appStream = gulp.src(['./dist/app.module.js'], {

        read: false

    });

    gulp.src('./dist/index.html')

        .pipe(inject(series(appStream, utilStream), {

            relative: true

        }))

        .pipe(inject(

            gulp.src(['./dist/\*.css'], {

                read: false

            }), {

                relative: true

            }

        ))

        .pipe(gulp.dest('./dist'));

});

#### 注入后生成的HTML文件

<!DOCTYPE html>

<html lang="zh-cn">

<head>

  <meta charset="utf-8">

  <meta name="description" content="">

  <meta http-equiv="X-UA-Compatible" content="IE=edge">

  <meta name="viewport" content="width=device-width, inital-scale=1.0">

  <title>WebGIS</title>

  <!-- build:css css/vendor.min.css -->

  <!-- bower:css -->

  <link rel="stylesheet" href="../lib/bootstrap/dist/css/bootstrap.css" />

  <link rel="stylesheet" href="../lib/bootstrap/dist/css/bootstrap-theme.css" />

  <link rel="stylesheet" href="../lib/font-awesome/css/font-awesome.css" />

  <link rel="stylesheet" href="../lib/angular/angular-csp.css" />

  <link rel="stylesheet" href="../lib/angular-motion/dist/angular-motion.css" />

  <link rel="stylesheet" href="../lib/angular-ui-grid/ui-grid.css" />

  <link rel="stylesheet" href="../lib/angular-ui-layout/src/ui-layout.css" />

  <link rel="stylesheet" href="../lib/angular-bootstrap/ui-bootstrap-csp.css" />

  <link rel="stylesheet" href="../lib/angular-block-ui/dist/angular-block-ui.css" />

  <link rel="stylesheet" href="../lib/openlayers/dist/ol.css" />

  <!-- endbower -->

  <!-- endbuild -->

  <!-- build:css css/style.min.css -->

  <!-- inject:css -->

  <link rel="stylesheet" href="app.css">

  <!-- endinject -->

  <!-- endbuild -->

</head>

<body ng-app="app" ng-controller="appController as $ctrl">

  <nav class="navbar navbar-default">

    <div class="container-fluid">

      <ul class="nav navbar-nav">

        <li ng-class="{'active': $ctrl.selectedItem == 'monitor'}">

          <a href="#" ui-sref="monitor">monitor</a>

        </li>

        <li ng-class="{'active': $ctrl.selectedItem == 'baseInfo'}">

          <a href="#" ui-sref="baseInfo">baseInfo</a>

        </li>

        <li ng-class="{'active': $ctrl.selectedItem == 'visualStation'}">

          <a href="#" ui-sref="visualStation">visualStation</a>

        </li>

        <li class="dropdown">

          <a href="#" class="dropdown-toggle" data-toggle="dropdown" role="button" aria-haspopup="true" aria-expanded="false">Dropdown <span class="caret"></span></a>

          <ul class="dropdown-menu">

            <li><a href="#">Action</a></li>

            <li><a href="#">Another action</a></li>

            <li><a href="#">Something else here</a></li>

            <li role="separator" class="divider"></li>

            <li><a href="#">Separated link</a></li>

            <li role="separator" class="divider"></li>

            <li><a href="#">One more separated link</a></li>

          </ul>

        </li>

      </ul>

    </div>

  </nav>

  <div class="main-container">

    <ui-view></ui-view>

  </div>

  <!-- build:js js/vendor.min.js -->

  <!-- bower:js -->

  <script src="../lib/jquery/dist/jquery.js"></script>

  <script src="../lib/bootstrap/dist/js/bootstrap.js"></script>

  <script src="../lib/angular/angular.js"></script>

  <script src="../lib/angular-route/angular-route.js"></script>

  <script src="../lib/angular-resource/angular-resource.js"></script>

  <script src="../lib/angular-animate/angular-animate.js"></script>

  <script src="../lib/angular-mocks/angular-mocks.js"></script>

  <script src="../lib/angular-ui-grid/ui-grid.js"></script>

  <script src="../lib/angular-ui-mask/dist/mask.js"></script>

  <script src="../lib/angular-ui-event/dist/event.js"></script>

  <script src="../lib/angular-ui-validate/dist/validate.js"></script>

  <script src="../lib/angular-ui-layout/src/ui-layout.js"></script>

  <script src="../lib/angular-ui-indeterminate/dist/indeterminate.js"></script>

  <script src="../lib/angular-ui-scroll/dist/ui-scroll.js"></script>

  <script src="../lib/angular-ui-scrollpoint/dist/scrollpoint.js"></script>

  <script src="../lib/angular-sanitize/angular-sanitize.js"></script>

  <script src="../lib/angular-cookies/angular-cookies.js"></script>

  <script src="../lib/angular-bootstrap/ui-bootstrap.js"></script>

  <script src="../lib/angular-bootstrap/ui-bootstrap-tpls.js"></script>

  <script src="../lib/angular-block-ui/dist/angular-block-ui.js"></script>

  <script src="../lib/angular-ui-router/release/angular-ui-router.js"></script>

  <script src="../lib/ui-router-extras/release/ct-ui-router-extras.js"></script>

  <script src="../lib/oclazyload/dist/ocLazyLoad.js"></script>

  <script src="../lib/d3/d3.js"></script>

  <script src="../lib/openlayers/dist/ol.js"></script>

  <!-- endbower -->

  <!-- endbuild -->

  <!-- build:js js/app.min.js -->

  <!-- inject:js -->

  <script src="app.module.js"></script>

  <script src="app.config.oclazyload.js"></script>

  <script src="app.config.route.js"></script>

  <script src="app.controller.js"></script>

  <!-- endinject -->

  <!-- endbuild -->

</body>

</html>

### 将HTML中CSS与JS压缩

// useref

gulp.task('compress', function () {

    gulp.src('./dist/index.html')

        .pipe(useref())

        .pipe(gulpif('\*.js', uglify({

            mangle: true

        })))

        .pipe(gulpif('\*.css', minifyCss()))

        .pipe(gulp.dest('./dist'));

});

### 去掉没有应用的CSS

// uncss

gulp.task('uncss', function () {

    gulp.src('./dist/app.css')

        .pipe(uncss({

            html: ['./dist/index.html']

        }))

        .pipe(gulp.dest('./dist'));

});

### 复制文件

// copy

//gulp.task('copy', ["clean"], function () {

gulp.task('copy', function () {//

    // gulp.src('./lib/bootstrap/fonts/\*\*')

    //     .pipe(gulp.dest('./dist/fonts'));

    // gulp.src('./lib/font-awesome/fonts/\*\*')

    //     .pipe(gulp.dest('./dist/fonts'));

    //

    gulp.src('./src/index.html')

        //.pipe(useref())

        .pipe(gulp.dest('./dist'));

});

### 清理输出目录

gulp.task("clean", function () {

    gulp.src('./dist/\*')

        .pipe(clean());

});

### 文件监控

// watch

gulp.task('watch', function () {

    gulp.watch(['./src/\*\*/\*.html'], ['html']);

    gulp.watch(['./src/\*\*/\*.styl'], ['css', 'inject']);

    gulp.watch(['./src/\*\*/\*.js', './gulpfile.js'], ['eslint', 'inject']);

    //gulp.watch(['./bower.json'], ['wiredep']);

});

### 服务

// server

gulp.task('server', function () {

    connect.server({

        root: './',

        hostname: '0.0.0.0',

        port: 8000,

        livereload: true

    });

});

### 集成

gulp.task('build', ['copy', 'uncss']); // 'compress', "clean",

gulp.task('default', ['server', 'css', 'eslint', 'html', 'index', 'watch']); // 'server', 'eslint', , 'watch'

## AngularJS与UI Route和ocLazyLoad集成

### 启动模块（app.module.js）

(function (angular) {

  'use strict';

  angular.module('app', ['ui.router', 'ct.ui.router.extras', 'oc.lazyLoad']);

})(angular);

### 配置模块加载（app.config.oclazyload.js）

(function (angular) {

  'use strict';

  angular.module('app').config(['$ocLazyLoadProvider', function ($ocLazyLoadProvider) {

    $ocLazyLoadProvider.config({

      debug: true,

      event: true,

      modules: [{

        name: 'baseInfo',

        serie: true,

        files: ['views/baseInfo/baseInfo.module.js', 'views/baseInfo/baseInfo.service.js', 'views/baseInfo/baseInfo.component.js']

      }, {

        name: 'baseInfoItem',

        serie: true,

        files: ['views/baseInfo/components/baseInfoItem/baseInfoItem.module.js', 'views/baseInfo/components/baseInfoItem/baseInfoItem.component.js']

      }, {

        name: 'monitor',

        serie: true,

        files: ['views/monitor/monitor.module.js', 'views/monitor/monitor.service.js', 'views/monitor/monitor.controller.js']

      }, {

        name: 'visualStation',

        serie: true,

        cache: 'false',

        files: ['views/visualStation/visualStation.module.js', 'views/visualStation/visualStation.service.js', 'views/visualStation/visualStation.component.js', 'views/visualStation/visualStation.css']

      }, {

        name: 'gis',

        serie: true,

        cache: 'false',

        //rerun: true,

        //reconfig: true,

        files: ['views/visualStation/components/gis/gis.module.js', 'views/visualStation/components/gis/gis.service.js', 'views/visualStation/components/gis/gis.component.js', 'views/visualStation/components/gis/gis.css']

      }, {

        name: 'openlayers',

        serie: true,

        //rerun: true,

        //reconfig: true,

        files: ['lib/openlayers/dist/ol.js']

      }]

    });

  }]);

})(angular);

### 配置UI Route的路由（app.config.oclazyload.js）

(function (angular) {

  'use strict';

  angular.module('app').config(['$stateProvider', '$urlRouterProvider', '$futureStateProvider', function ($stateProvider, $urlRouterProvider, $futureStateProvider) {

    //

    var lazyDeferred;

    $urlRouterProvider.otherwise('/monitor');

    $stateProvider

      .state('monitor', {

        url: '/monitor',

        templateUrl: 'views/monitor/monitor.html',

        controller: 'MonitorController',

        resolve: {

          load: function ($templateCache, $ocLazyLoad, $q) {

            var deferred = $q.defer();

            $ocLazyLoad.load('monitor').then(function (name) {

              deferred.resolve();

            }, function () {

              deferred.reject();

            });

            return deferred.promise;

            /\*

            $ocLazyLoad.load({

              serie: true,

              files: ['views/monitor/monitor.controller.js']

            }).then(function (name) {

              deferred.resolve();

            }, function () {

              deferred.reject();

            });

            return deferred.promise;

            \*/

          }

        }

      })

      .state('baseInfo', {

        url: '/baseInfo',

        template: '<base-info></base-info>',

        resolve: {

          load: function ($templateCache, $ocLazyLoad, $q) {

            var deferred = $q.defer();

            $ocLazyLoad.load('baseInfo').then(function (name) {

              deferred.resolve();

            }, function () {

              deferred.reject();

            });

            return deferred.promise;

            /\*

            $ocLazyLoad.load({

              serie: true,

              files: ['views/baseInfo/baseInfo.component.js']

            }).then(function (name) {

              deferred.resolve();

            }, function () {

              deferred.reject();

            });

            return deferred.promise;

            \*/

          }

        }

      })

      .state('visualStation', {

        url: '/visualStation',

        cache: 'false',

        template: '<visual-station></visual-station>',

        resolve: {

          load: function ($templateCache, $ocLazyLoad, $q) {

            var deferred = $q.defer();

            $ocLazyLoad.load('visualStation').then(function (name) {

              deferred.resolve();

            }, function () {

              deferred.reject();

            });

            return deferred.promise;

          }

        }

      });

    //$locationProvider.html5Mode(true);

  }]);

})(angular);

### 控制器（app.controller.js）

(function (angular) {

    'use strict';

    function AppController($scope, $state, $log) {

        var self = this;

        self.selectedItem = 'baseInfo';

        $scope.$on('$stateChangeSuccess', function (event, toState, toParams, fromState, fromParams) {

            $log.debug('successfully changed states');

            $log.debug('event', event);

            $log.debug('toState', toState);

            $log.debug('toParams', toParams);

            $log.debug('fromState', fromState);

            $log.debug('fromParams', fromParams);

            self.selectedItem = toState.name;

        });

    }

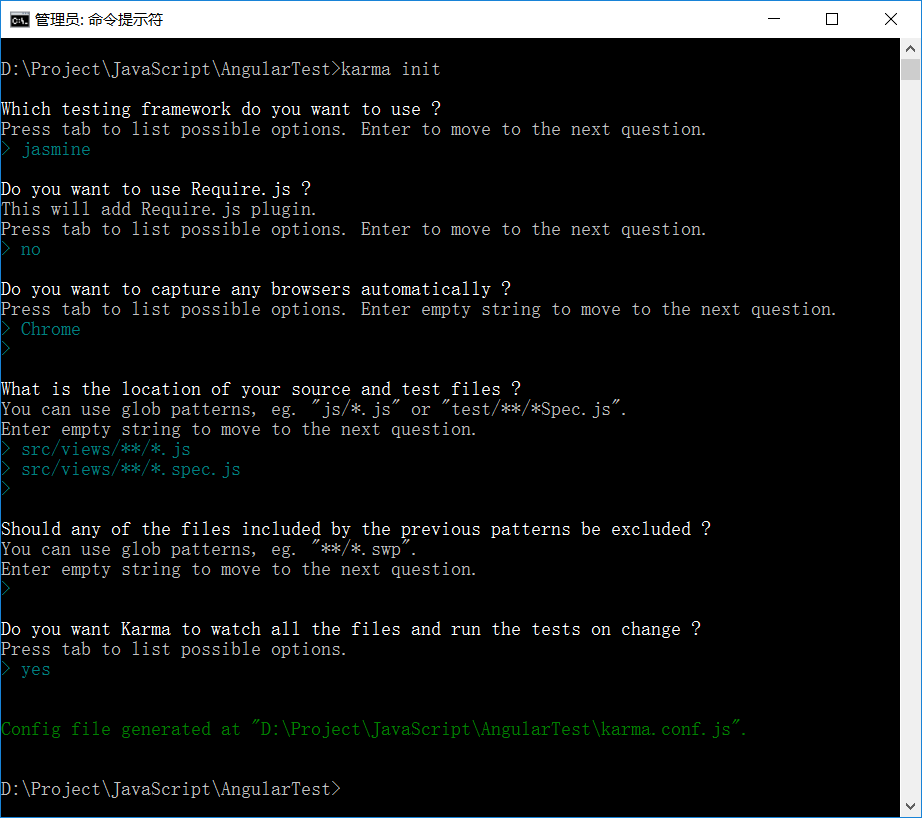
    angular.module('app').controller('appController', ['$scope', '$state', '$log', AppController]);

})(angular);

## Karma

### 生成karma配置文件

karma init



D:\Project\JavaScript\AngularTest>karma init

Which testing framework do you want to use ?

Press tab to list possible options. Enter to move to the next question.

> jasmine

Do you want to use Require.js ?

This will add Require.js plugin.

Press tab to list possible options. Enter to move to the next question.

> no

Do you want to capture any browsers automatically ?

Press tab to list possible options. Enter empty string to move to the next question.

> Chrome

>

What is the location of your source and test files ?

You can use glob patterns, eg. "js/\*.js" or "test/\*\*/\*Spec.js".

Enter empty string to move to the next question.

> src/views/\*\*/\*.js

> src/views/\*\*/\*.spec.js

>

Should any of the files included by the previous patterns be excluded ?

You can use glob patterns, eg. "\*\*/\*.swp".

Enter empty string to move to the next question.

>

Do you want Karma to watch all the files and run the tests on change ?

Press tab to list possible options.

> yes

Config file generated at "D:\Project\JavaScript\AngularTest\karma.conf.js".

// Karma configuration

// Generated on Sun Mar 26 2017 22:48:28 GMT+0800 (中国标准时间)

module.exports = function (config) {

  config.set({

    // base path that will be used to resolve all patterns (eg. files, exclude)

    basePath: '../',

    // frameworks to use

    // available frameworks: https://npmjs.org/browse/keyword/karma-adapter

    frameworks: ['jasmine'],

    // list of files / patterns to load in the browser

    files: [

      'lib/angular/angular.js',

      'lib/angular-mocks/angular-mocks.js',

      'src/views/\*\*/\*.module.js',

      'src/views/\*\*/\*.js'

    ],

    // list of files to exclude

    exclude: [],

    // preprocess matching files before serving them to the browser

    // available preprocessors: https://npmjs.org/browse/keyword/karma-preprocessor

    preprocessors: {},

    // test results reporter to use

    // possible values: 'dots', 'progress'

    // available reporters: https://npmjs.org/browse/keyword/karma-reporter

    reporters: ['progress'],

    // web server port

    port: 9876,

    // enable / disable colors in the output (reporters and logs)

    colors: true,

    // level of logging

    // possible values: config.LOG\_DISABLE || config.LOG\_ERROR || config.LOG\_WARN || config.LOG\_INFO || config.LOG\_DEBUG

    logLevel: config.LOG\_INFO,

    // enable / disable watching file and executing tests whenever any file changes

    autoWatch: true,

    // start these browsers

    // available browser launchers: https://npmjs.org/browse/keyword/karma-launcher

    browsers: ['Chrome'],

    // Continuous Integration mode

    // if true, Karma captures browsers, runs the tests and exits

    singleRun: false,

    // Concurrency level

    // how many browser should be started simultaneous

    concurrency: Infinity,

    //

    plugins: [

      'karma-chrome-launcher',

      'karma-jasmine'

    ]

  })

}

### 编写单元测试用例

'use strict';

describe('component: gis', function () {

  var $componentController;

  beforeEach(module('common.gis'));

  var $ctrl, $scope;

  beforeEach(inject(function (\_$rootScope\_, \_$componentController\_) {

    $scope = \_$rootScope\_.$new();

    $ctrl = \_$componentController\_('gis', {

      $scope: $scope

    });

  }));

  it('$onInit should be defined', function () {

    expect($ctrl.$onInit).toBeDefined();

  });

});

### 运行Karma 进行单元测试

set CHROME\_BIN="D:\GreenSoftware\Chrome55\chrome.exe"

karma start ./conf/karma.conf.js

