Karma Jasmine

**Tools:**

1. karma-cli

2. jasmine testing framework.

**Environment Installation:**

Install karma : npm install karma-cli -g

Install jasmine → npm install -g jasmine

Install karma-jasmine(local) → npm install karma-jasmine –save-dev

Install karma-jasmine(global) → npm install -g karma-jasmine

Install karma-firefox-launcher(plugin)→ npm install karma-firefox-launcher –save-dev

**\*need node version 10+**

**karma config file(karma.config.js):**

**\***create **karma.config.js** file in the root directory of the project

module.exports = function (config) {

config.set({

basePath: '',

frameworks: ['jasmine'],

plugins: [

"karma-firefox-launcher",

"karma-jasmine"

],

files: [

'bower\_components/angular/angular.js',

'bower\_components/angular-mocks/angular-mocks.js',

'bower\_components/bower-angular-route/angular-route.js',

'bower\_components/bower-angular-route/angular-route.min.js',

'/home/jyoti/jyoti/crud/crud.js',

'\*\*Test.js'

],

////////'/home/jyoti/jyoti/crud/crud.js'→ controller file

/////'\*\*Test.js'→ all test cases file

exclude: [

],

preprocessors: {

},

reporters: ['progress'],

port: 9876,

colors: true,

logLevel: config.LOG\_INFO,

autoWatch: true,

browsers: ['Firefox'],

singleRun: false,

concurrency: Infinity

})

}

**Test cases:**

Test.js→

|  |
| --- |
| describe("test manmon", function(){  it("should be false",function(){  expect(false).toBe(false);  })  }); |

CtrlTest.js →

|  |
| --- |
| (function(){  describe("app module",function(){  beforeEach(module("crudApp"));  describe("crudCtrl",function(){  var scope,controller;  beforeEach(inject(function($rootScope, $controller){  scope = $rootScope.$new();  controller = $controller;  }));  it("should be hello manmon",function(){  controller("crudCtrl",{$scope: scope});  expect(scope.message).toBe("manmon is ready")  })  });  });  })(); |

Test run→ go to project root directory and open terminal → **karma start**