013 Gulp - angular annotation

Gulp Kata List – Blog Page

[Gulp Kata List on GitHub](https://github.com/robertdunaway/katas-gulp)

# Duration

5 minutes

# Brief

Angular 1x dependency injected is broken by minification if not properly annotated. The ngAnnotate plugin handles this for us.

### For More Information

BING/GOOGLE: “Gulp ”

Book: <http://www.amazon.com/Gulp-Quick-guide-getting-running/dp/151468960X>

# Instructions

Get tutorial folder or the entire katas-angular2 repo.

Open the [before/\*.sln] file and execute the kata.

Feel free to execute this kata multiple times because repetition creates motor memory.

# Github

* Before (start kata with this)
  + https://github.com/robertdunaway/katas-gulp/tree/master/013%20Gulp%20-%20angular%20annotation/before
* After
  + https://github.com/robertdunaway/katas-gulp/tree/master/013%20Gulp%20-%20angular%20annotation/after

# Kata

Using ngAnnotate, modify Angular 1x files in preparation for optimization. Consider when annotation needs to occur and whether or not you should alter your source code or only the distribution code.

# Review

You’ll need gulp-ng-annotate.

npm install gulp-ng-annotate --save-dev  
  
  
Add a reference to here.

, ngAnnotate = require('gulp-ng-annotate')

We will annotate our code before copying it to wwwroot for optimization. This requires the use of run-sequence. One trick to making this work is saving the resulting file back to its original location, overwriting the original file.

Create the annotate task.

var gulp = require('gulp')

, uglify = require('gulp-uglify')

, rename = require('gulp-rename')

, sourcemaps = require('gulp-sourcemaps')

, ngAnnotate = require('gulp-ng-annotate')

, runSequence = require('run-sequence');

gulp.task('annotate', function () {

return gulp.src(['src/\*\*/\*.js', '!src/\*\*/\*.min.js'], { base: 'src/./' })

.pipe(ngAnnotate())

.pipe(gulp.dest('src/./'));

});

gulp.task('copy-to-wwwroot', function () {

return gulp.src(['src/\*\*/\*'])

.pipe(gulp.dest('wwwroot'));

});

gulp.task('minify-js', function () {

return gulp.src(['wwwroot/\*\*/!(\*.min).js', '!wwwroot/lib/\*\*/\*'])

.pipe(sourcemaps.init())

.pipe(uglify())

.pipe(rename({

extname: '.min.js'

}))

.pipe(sourcemaps.write('./'))

.pipe(gulp.dest('wwwroot/./'));

});

gulp.task('default', function () {

runSequence(['annotate', 'copy-to-wwwroot'], 'minify-js');

});

The src/js/main.js file contains an Angular 1x service. This service is NOT properly annotated. In its current state it will execute correctly but the moment the file is optimized it is broken.

### src/js/main.js

Before annotation

(function () {

ngApp.service('coreDataService', function ($http, $q, $log, cacheService) {

'use strict';

return {

getCache: function (cacheName) {

var deferred = $q.defer();

cacheService.getCache(cacheName).then(function (data) {

deferred.resolve(data);

});

return deferred.promise;

}

};

});

});

After annotation

(function () {

ngApp.service('coreDataService', ["$http", "$q", "$log", "cacheService", function ($http, $q, $log, cacheService) {

'use strict';

return {

getCache: function (cacheName) {

var deferred = $q.defer();

cacheService.getCache(cacheName).then(function (data) {

deferred.resolve(data);

});

return deferred.promise;

}

};

}]);

});



# Next

Take a few minutes and imagine more examples.