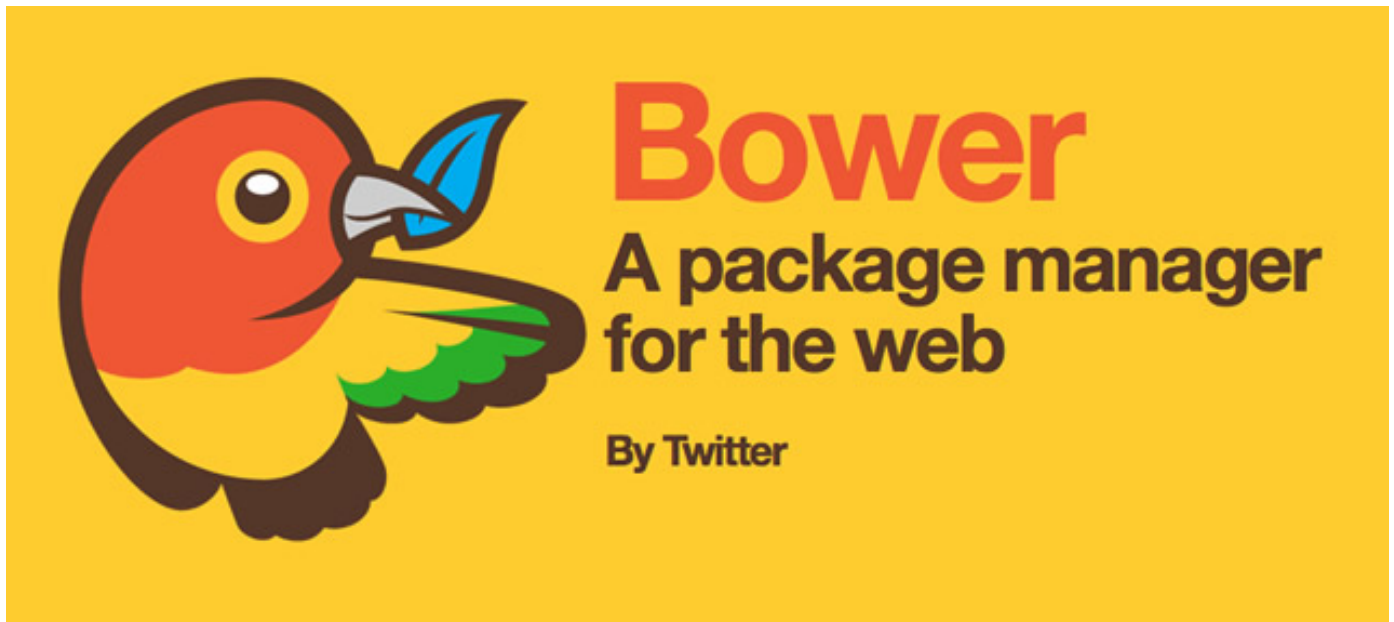


در [قسمت اول](#) گالپ را معرفی کردیم و در [مقاله قبلی](#) به نوشتن اولین تسک با گالپ پرداختیم. در این قسمت می‌خواهیم با نصب bower، پروژه‌ی workflow بوت استرپ راستچین شده را انجام دهیم.

نصب bower



[bower](#) یک مدیریت پکیج سمت Front end است و از مزایای استفاده از آن می‌توان به موارد زیر اشاره کرد:

ساده کردن تعریف وابستگی‌های منابع پروژه با تعریف یک فایل **bower.json**

نیازی به commit کردن وابستگی‌های پروژه نیست.

با ذکر ورژن مربوط به وابستگی یا محدوده‌ی قابل قبول برای آن، به روز رسانی منابع به سادگی با یک دستور انجام می‌شود.

وابستگی‌های وابسته به یک منبع را نیز نصب می‌کند. برای مثال زمانیکه بوت استرپ را به عنوان وابستگی پروژه تعریف می‌کنیم،

وابستگی آن یعنی jquery را چون در فایل **bower.json** بوت استرپ تعریف شده‌است، به صورت خودکار دانلود می‌کند.

در نهایت افراد هم تیمی یا توسعه دهندگان دیگر به راحتی با زدن دستور **bower install** تمام وابستگی‌های پروژه را می‌توانند نصب کنند.

برای نصب آن کافی است دستور زیر را بزنید و بعد از نصب نیز دستور خط دوم را در مسیر پروژه وارد کنید تا یک فایل

bower.json را برایمان بسازد. برای اینکار به سوال‌هایی که می‌پرسد باید جواب دهیم. تنها نکته‌ای که قابل ذکر است، پاسخ به

سوال **what types of modules does this package expose ?** است که باید گزینه‌ی Node را انتخاب کنید.

```
sudo npm install -g bower
bower init
```

حال می‌خواهیم وابستگی‌های پروژه را نصب کنیم که عبارتند از **bootstrap-sass,fontawesome,bootstrap-rtl** :

```
bower install bootstrap-sass-official --save
bower install fontawesome --save
bower install bootstrap-rtl --save
```

نکته : عبارت --save وابستگی مربوطه را به bower.json اضافه می‌کند. اگر نصب با موفقیت صورت گرفته باشد، پکیج‌های مربوطه را می‌توانید در فولدر bower_components در root پروژه مشاهده کنید.

نصب پلاگین‌های مورد نیاز gulp

ما می‌خواهیم که بوت استرپ و نگارش sass آن‌را کامپایل کنیم و همچنین وابستگی‌های bower پروژه را از طریق گالپ نصب کنیم تا نیازی به زدن bower install نباشد و توسعه دهنده‌ی پروژه فقط با زدن npm install، تمام وابستگی‌های پروژه‌ی ما را نصب کند. می‌توان تمام پلاگین‌ها را پشت سر هم با یک دستور نصب کرد و یا به صورت جداگانه این کار را انجام داد.

```
sudo npm install gulp gulp-ruby-sass gulp-notify gulp-bower --save-dev
```

نکته\: پلاگین gulp-notify به منظور نشان دادن خطاها در ترمینال است؛ تا در صورت وجود اشتباه در کامپایل فایل‌های Sass، کل روند گالپ متوقف نشود.

نکته ۲ : برای اینکه کامپایل sass انجام شود نیاز به Ruby دارید. برای ویندوز می‌توانید از [روبی اینستالر](#) استفاده کنید.

نوشتن تسک‌ها برای گالپ

به قسمت مهم و هیجان انگیز کار رسیدیم! همان طور در مقاله قبلی گفتیم، ابتدا باید ماژول‌هایی را که نصب کردیم، include کنیم به این صورت:

```
var gulp = require('gulp'),
    sass = require('gulp-ruby-sass'),
    notify = require('gulp-notify'),
    bower = require('gulp-bower');
```

برای اینکه دسترسی به مسیرهای مهم پروژه آسان‌تر شود، آن‌ها را درون یک شیء نگه داری می‌کنیم.

```
var config = {
  sassPath = './resources/sass',
  bowerDir = './bower_components'
}
```

تسک باور را اضافه می‌کنیم تا کار bower install را خودکار کنیم. مزیت این کار این است اگر یک هم تیمی، پکیج جدیدی را در حین توسعه‌ی پروژه نصب کرد، بدون اینکه لازم باشد تا در جایی از پروژه، بقیه را از آن مطلع کنید، فقط با زدن gulp خیالتان راحت شود که تمام کارهایی که باید انجام دهید، گالپ برایتان انجام می‌دهد.

```
// create a task to do bower install
gulp.task('bower', function() {
  return bower()
    .pipe(gulp.dest(config.bowerDir))
});
```

در گام بعدی، تسک جاوا اسکریپت را اضافه می‌کنیم. یعنی جی کوری و فایل bootstrap.js را به مسیر public/js می‌آوریم. فولدر public برای جدا سازی فایل‌های نهایی از فایل‌های توسعه است و به همین صورت برای فونت آیکن‌های fontawesome.



```
// Copy js files to public folder
gulp.task('js', function() {
  return gulp.src([config.bowerDir + '/bootstrap-sass-official/assets/javascripts/bootstrap.min.js',
    config.bowerDir + '/jquery/dist/jquery.min.js'
  ])
  .pipe(gulp.dest('./public/js'));
});

// Copy fontawesome icons to public/fonts folder
gulp.task('icons', function() {
  return gulp.src(config.bowerDir + '/fontawesome/fonts/**/*.*)
  .pipe(gulp.dest('./public/fonts'));
});
```

برای سی اس اس هم تسک مربوطه اش را به صورت زیر می نویسیم



```
gulp.task('css', function() {
  return sass(config.sassPath + '/style.scss', { // Our custom sass
    style: 'compressed', // minify css
    loadPath: [ // load paths to easily use imports in resources/sass
      './resources/sass',
      config.bowerDir + '/bootstrap-sass-official/assets/stylesheets', // bootstrap sass files
      config.bowerDir + '/fontawesome/scss' // awesome icons sass files
    ]
  });
});
```

حال تعریف می‌کنیم که اگر خطایی در حین کامپایل رخ داد، آن را به ما نشان دهد و در نهایت فایل کامپایل و فشرده شده را در مسیر خروجی قرار می‌دهیم. کدها را به صورت زیر به روز می‌کنیم

```
gulp.task('css', function() {
  return sass(config.sassPath + '/style.scss', { // Our custom sass
    style: 'compressed', // minify css
    loadPath: [ // load paths to easily use imports in resources/sass
      './resources/sass',
      config.bowerDir + '/bootstrap-sass-official/assets/stylesheets', // bootstrap sass files
      config.bowerDir + '/fontawesome/scss' // awesome icons sass files
    ]
  })
  .on('error', notify.onError(function(error) {
    return 'Error: ' + error.message;
  }))
  .pipe(gulp.dest('./public/css'));
});
```

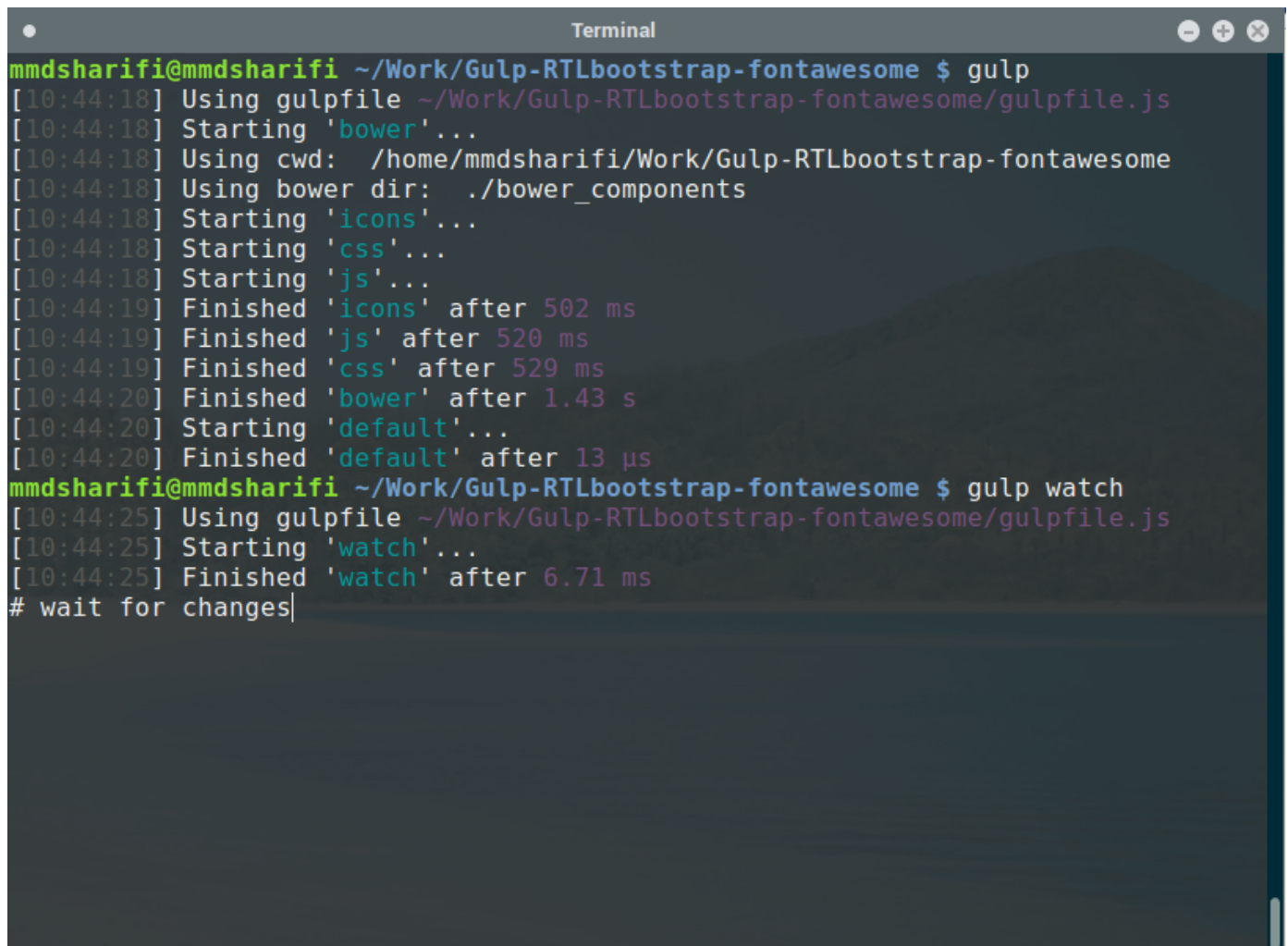
اینجا ما می‌خواهیم که کارها را خودکار سازی کنیم تا با تغییر و ذخیره‌ی مجدد فایل‌های سس، تسک سی اس اس را انجام دهد. برای این کار کدهای زیر را اضافه می‌کنیم

```
// Rerun the task when a file changes
gulp.task('watch', function() {
```

```
gulp.watch(config.sassPath + '/*.scss', ['css']);
});
```

در نهایت برای سادگی می‌توانیم مجموعه‌ای از وظایف را در یک تسک تعریف کنیم تا به راحتی و با زدن تنها یک دستور در ترمینال، کارها خودکار سازی شوند

```
// Run this task with : gulp
// OR gulp default
gulp.task('default', ['bower', 'icons', 'css', 'js']);
```



```
Terminal
mmdsharifi@mmdsharifi ~/Work/Gulp-RTLbootstrap-fontawesome $ gulp
[10:44:18] Using gulpfile ~/Work/Gulp-RTLbootstrap-fontawesome/gulpfile.js
[10:44:18] Starting 'bower'...
[10:44:18] Using cwd: /home/mmdsharifi/Work/Gulp-RTLbootstrap-fontawesome
[10:44:18] Using bower dir: ./bower_components
[10:44:18] Starting 'icons'...
[10:44:18] Starting 'css'...
[10:44:18] Starting 'js'...
[10:44:19] Finished 'icons' after 502 ms
[10:44:19] Finished 'js' after 520 ms
[10:44:19] Finished 'css' after 529 ms
[10:44:20] Finished 'bower' after 1.43 s
[10:44:20] Starting 'default'...
[10:44:20] Finished 'default' after 13 µs
mmdsharifi@mmdsharifi ~/Work/Gulp-RTLbootstrap-fontawesome $ gulp watch
[10:44:25] Using gulpfile ~/Work/Gulp-RTLbootstrap-fontawesome/gulpfile.js
[10:44:25] Starting 'watch'...
[10:44:25] Finished 'watch' after 6.71 ms
# wait for changes
```

بسیار خوب؛ ما توانستیم پایه‌ی ورک فلو یمان را بسازیم. در مقاله‌ی بعدی از پلاگین‌های دیگری برای بهینه سازی کارهایمان کمک خواهیم گرفت
[مخزن گیت هاب](#)

commit : [Update :gulp file and bower.json](#)