



gulpjs.com

“Automate and enhance your workflow”

What we'll cover



- Why use Gulp? (Pre-demo)
- Examples of useful plugins
 - * HTML (gulp-file-include)
 - * CSS (gulp-sass)
 - * JavaScript (browserify)
- Installation
- Setup
 - * Organization
 - * Tasks & Dependencies
 - * Task Groups
 - * Watch (build on save)
- Browsersync
- More demo, Q&A

Why use Gulp?



Slides and examples
github.com/durkes

Why use Gulp?

Plugins

Tasks

Automate



Pre-demo

...then more detail



Slides and examples
github.com/durkes

HTML Example

Plugin: gulp-file-include



Slides and examples
github.com/durkes

index.html + head.html + navbar.html + body_end.html = index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  @@include("_html/head.html", {"title": "Gulp Demo"})
</head>
<body>
  @@include("_html/navbar.html", {"index": "active"})

  <div class="container">
    <div class="jumbotron">
      <h1>Hello, world!</h1>
      <p>This is a basic Bootstrap template for the gulp demo.</p>
    </div>

    <a href="http://www.gulpjs.com/"></a>
  </div>

  @@include("_html/body_end.html")
</body>
</html>
```

index.html + head.html + navbar.html + body_end.html = index.html

```
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1">
<title>@@title</title>

<link rel="stylesheet" href="//maxcdn.bootstrapcdn.com/bootstrap/3.3.4/css/bootstrap.min.css">
<link rel="stylesheet" href="//maxcdn.bootstrapcdn.com/bootstrap/3.3.4/css/bootstrap-theme.min.css">
<link rel="stylesheet" href="/css/global.css">
```


index.html + head.html + navbar.html + body_end.html = index.html

```
<div class="container">
  <nav class="navbar navbar-inverse">
    <div class="container-fluid">
      <div class="navbar-header">
        <button type="button" class="navbar-toggle collapsed"
          data-toggle="collapse" data-target="#bs-example-navbar-collapse-1">

          <span class="sr-only">Toggle navigation</span>
          <span class="icon-bar"></span>
          <span class="icon-bar"></span>
          <span class="icon-bar"></span>
        </button>
        <a class="navbar-brand" href="/">Gulp Demo</a>
      </div>

      <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
        <ul class="nav navbar-nav">
          <li class="@@index"><a href="/">Home</a></li>
          <li class="@@page2"><a href="/page2.html">Page 2</a></li>
          <li class="@@page3"><a href="/page3.html">Page 3</a></li>
        </ul>
      </div>
    </div>
  </nav>
</div>
```

index.html + head.html + navbar.html + body_end.html = index.html

```
<script src="//ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"></script>  
<script src="//maxcdn.bootstrapcdn.com/bootstrap/3.3.4/js/bootstrap.min.js"></script>  
<script src="/js/global.js"></script>
```

index.html + head.html + navbar.html + body_end.html = index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1">
<title>Gulp Demo</title>

<link rel="stylesheet" href="//maxcdn.bootstrapcdn.com/bootstrap/3.3.4/css/bootstrap.min.css">
<link rel="stylesheet" href="//maxcdn.bootstrapcdn.com/bootstrap/3.3.4/css/bootstrap-theme.min.css">
<link rel="stylesheet" href="/css/global.css">

</head>
<body>
  <div class="container">
    <nav class="navbar navbar-inverse">
      . . . [removed for slide]
    </nav>
  </div>

  <div class="container">
    <div class="jumbotron">
      . . . [removed for slide]
    </div>
  </div>

  <script src="//ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"></script>
  <script src="//maxcdn.bootstrapcdn.com/bootstrap/3.3.4/js/bootstrap.min.js"></script>
  <script src="/js/global.js"></script>

</body>
</html>
```

CSS (Sass) Example

Plugin: gulp-sass



Slides and examples
github.com/durkes

global.css + bootstrap_overrides.scss + ... + ... = global.css

```
/* Custom */
@import './_incl/bootstrap_overrides';

/* Alertify */
@import './_incl/alertify.core';
@import './_incl/alertify.bootstrap';

#gulp-logo {
    margin: 0 auto;
}
```

global.css + bootstrap_overrides.scss + ... + ... = global.css

```
.jumbotron {  
  background-color: #22A7F0;  
  
  h1, h2, h3, h4, p {  
    color: #fff;  
  }  
}
```

global.css + bootstrap_overrides.scss + ... + ... = global.css

```
/* Custom */
.jumbotron {
  background-color: #22A7F0; }
.jumbotron h1, .jumbotron h2, .jumbotron h3, .jumbotron h4, .jumbotron p {
  color: #fff; }

/* Alertify */
.alertify,
.alertify-show,
.alertify-log {
  -webkit-transition: all 500ms cubic-bezier(0.175, 0.885, 0.32, 1.275);
  -moz-transition: all 500ms cubic-bezier(0.175, 0.885, 0.32, 1.275);
  -ms-transition: all 500ms cubic-bezier(0.175, 0.885, 0.32, 1.275);
  -o-transition: all 500ms cubic-bezier(0.175, 0.885, 0.32, 1.275);
  transition: all 500ms cubic-bezier(0.175, 0.885, 0.32, 1.275);
  /* easeOutBack */ }

. . . [removed for slide]

#gulp-logo {
  margin: 0 auto; }
```

JavaScript Example

Plugin: browserify



Slides and examples
github.com/durkes

global.js + alertify.js + flood.js = global.js

```
/*import and instantly run alertify with global argument*/
```

```
require('./_incl/alertify')(global);
```

```
/*flood calls alertify.log, alertify.success, and alertify.error at once*/
```

```
var flood = require('./_incl/flood');
```

```
/*flood is scoped to the current module, so let's make it usable in the browser*/
```

```
global.flood = flood;
```

```
/*This is verbose for demo. We could have combined the assignments:
```

```
global.flood = require('./_incl/flood');*/
```

global.js + alertify.js + flood.js = global.js

```
/* alertify @author Fabien Doiron <fabien.doiron@gmail.com> */
module.exports = function (global, undefined) {
    "use strict";

    var document = global.document,
        Alertify;

    Alertify = function () {
        var _alertify = {},
            dialogs    = {},
            isopen     = false,
            keys       = { ENTER: 13, ESC: 27, SPACE: 32 },
            queue      = [],
            $, btnCancel, btnOK, btnReset, btnResetBack, btnFocus, elCallee,
            elCover, elDialog, elLog, form, input, getTransitionEvent;

        . . . [removed for slide] . . .

        if (typeof define === "function") {
            define([], function () { return new Alertify(); });
        } else if (typeof global.alertify === "undefined") {
            global.alertify = new Alertify();
        }
    };
};
```

global.js + alertify.js + flood.js = global.js

```
module.exports = function (message) {  
    alertify.log(message);  
    alertify.success(message);  
    alertify.error(message);  
};
```

global.js + alertify.js + flood.js = global.js

```
(function e(t,n,r){function s(o,u){if(!n[o]){if(!t[o]){var a=typeof
require=="function"&&require;if(!u&&a)return a(o,!0);if(i)return i(o,!0);var f=new Error("Cannot find
module '"+o+"'");throw f.code="MODULE_NOT_FOUND",f}var
l=n[o]={exports:{}};t[o][0].call(l.exports,function(e){var n=t[o][1][e];return
s(n?n:e)},l,l.exports,e,t,n,r)}return n[o].exports}var i=typeof require=="function"&&require;for(var
o=0;o<r.length;o++)s(r[o]);return s})({1:[function(require,module,exports){
/* alertify @author Fabien Doiron <fabien.doiron@gmail.com> */
  module.exports = function (global, undefined) {
    "use strict";
    var document = global.document,
        Alertify;
    . . . [removed for slide] . . .
  };
},{}],2:[function(require,module,exports){
module.exports = function (message) {
  alertify.log(message);
  alertify.success(message);
  alertify.error(message);
};
},{}],3:[function(require,module,exports){
(function (global){
require('./_incl/alertify')(global);
var flood = require('./_incl/flood');
global.flood = flood;
}).call(this,typeof global !== "undefined" ? global : typeof self !== "undefined"
? self : typeof window !== "undefined" ? window : {})),{"./_incl/alertify":1,"./_incl/flood":2}}},{},[3]);
```

Installation



Slides and examples
github.com/durkes

Prerequisite

Install Node.js (and npm) from nodejs.org

From command line

Install gulp (-g: globally)

```
> npm install -g gulp
```

Create a new project

```
> mkdir project
```

```
> cd project
```

```
> mkdir gulp src build
```

```
> cd gulp
```

```
> npm init
```

*you may press enter for all EXCEPT
name your project so it does not default to 'gulp'*

Install gulp packages

```
> npm install --save-dev gulp-file-include
```

...



Setup

Organization Tasks



Slides and examples
github.com/durkes

Directory structure used in this demo

FOLDERS

▼ project

▶ build

▼ gulp

▶ tasks

gulpfile.js

package.json

▼ src

▼ _html

▼ css

▶ _incl

▼ js

▶ _incl

▼ noop

▶ img

index.html

page2.html

page3.html

Run gulp from here

HTML file includes

SASS/CSS includes

JS module exports

No operation – just copy files to dest



A basic gulpfile.js

```
var gulp = require('gulp');
var fileInclude = require('gulp-file-include');
var minifyHTML = require('gulp-minify-html');

gulp.task('html', function () {
  return gulp.src('../src/*.html')
    .pipe(fileInclude({prefix: '@@', basepath: '@file'}))
    .pipe(minifyHTML({empty: true, cdata: true, conditionals: true, spare: true, quotes: false}))
    .pipe(gulp.dest('../build'));
});
```

return is optional, but use it! You'll see why...



A basic gulpfile.js

```
var gulp = require('gulp');
var fileInclude = require('gulp-file-include');
var minifyHTML = require('gulp-minify-html');

gulp.task('html', function () {
  return gulp.src('../src/*.html')
    .pipe(fileInclude({prefix: '@@', basepath: '@file'}))
    .pipe(minifyHTML({empty: true, cdata: true, conditionals: true, spare: true, quotes: false}))
    .pipe(gulp.dest('../build'));
});
```

Task name is html

Run task from command line: **gulp html**



Source glob examples

```
return gulp.src(['../src/*.html'])
```

`../src/*.html`

all html files in the directory

`../src/**/*.html`

all html files in src and sub dirs
sub dirs are created in dest folder

`../src/specific.html`

just run on a specific file

`['../src/specific.html', '../src/html/*']` multiple source array



Task Dependencies



Slides and examples
github.com/durkes

```
var gulp = require('gulp');
var fileInclude = require('gulp-file-include');
var minifyHTML = require('gulp-minify-html');

/*tasks*/
gulp.task('html', function () {
  return gulp.src('../src/*.html')
    .pipe(fileInclude({prefix: '@@', basepath: '@file'}))
    .pipe(gulp.dest('../build'));
});
```



```
gulp.task('minify', ['html'], function () {
  return gulp.src('../build/**/*.html')
    .pipe(minifyHTML({empty: true, cdata: true, conditionals: true, spare: true, quotes: false}))
    .pipe(gulp.dest('../build'));
});
```

Because *minify* depends on *html*, **gulp minify = gulp html minify**



Use 'hints' with dependencies

```
gulp.task('html', function () {  
  ➡ return gulp.src('../src/*.html')  
    .pipe(fileInclude({prefix: '@@', basepath: '@file'}))  
    .pipe(gulp.dest('../build'));  
});
```



```
gulp.task('html', function (cb) {  
  gulp.src('../src/*.html')  
    .pipe(fileInclude({prefix: '@@', basepath: '@file'}))  
    .pipe(gulp.dest('../build'));  
  ➡ cb();  
});
```

A hint is a **returned** stream or promise, or a callback



Task Groups



Slides and examples
github.com/durkes

```
gulp.task('build', ['html', 'css', 'js', 'minify']);  
gulp.task('default', ['build']);
```

> **gulp build** = gulp html css js minify

Run the default task

> **gulp**

In this example

> **gulp** = gulp build = gulp html css js minify



Running tasks (recap)

From command line

> gulp [task, task]

Just html

> gulp html

html and js

> gulp html js

Default task

> gulp



Watch

Build on save



Slides and examples
github.com/durkes

```
gulp.task('build', ['html', 'css', 'js', 'minify']);  
gulp.task('default', ['watch']);
```



```
gulp.task('watch', ['build'], function () {  
    var watcher = gulp.watch('../src/**/*.js', ['build']);  
  
    watcher.on('change', function (event) {  
        console.log('\nFile ' + event.path + ' was ' + event.type + ', running tasks...');  
    });  
});
```



Run **gulp** or **gulp watch**

Gulp will run the 'build' task on file change/add/delete



Browsersync

Local server with auto-reload



Slides and examples
github.com/durkes

```

var gulp = require('gulp');
var browserSync = require('browser-sync').create();

/*groups*/
gulp.task('build', ['clean', 'noop', 'html', 'css', 'js', 'minify']);
gulp.task('default', ['watch']);

/*browsersync + watch*/
gulp.task('browsersync', ['build'], function() {
  browserSync.init({
    server: {
      baseDir: '../build'
    }
  });
});

gulp.task('reload', ['build'], function() {
  browserSync.reload();
});

gulp.task('watch', ['browsersync'], function () {
  var watcher = gulp.watch('../src/**/*.js', ['reload']);

  watcher.on('change', function (event) {
    console.log('\nFile ' + event.path + ' was ' + event.type + ', running tasks...');
  });
});

```

Browsersync will launch
<http://localhost:3000>
 and automatically reload
 the page on re-build





Slides and examples
github.com/durkes

Packages used in this demo

```
npm install --save-dev gulp  
npm install --save-dev del  
npm install --save-dev browser-sync  
npm install --save-dev gulp-file-include  
npm install --save-dev gulp-sass  
npm install --save-dev gulp-minify-html  
npm install --save-dev gulp-minify-css  
npm install --save-dev gulp-uglify  
npm install --save-dev browserify  
npm install --save-dev through2  
npm install --save-dev vinyl-buffer  
npm install --save-dev vinyl-source-stream
```

