

# Tools for JavaScript/TypeScript Development

Unleash the Power of JavaScript/TypeScript Tooling

Press Space for next page →



# Table of Contents

- JavaScript Editors
  - JetBrains WebStorm
  - Sublime Text 2/3
  - Atom
  - Visual Code Editor
- Project tools
  - Package Management: NPM/Yarn & Bower
  - Scaffolding: Yeoman
  - Task Automation: Grunt && Gulp && npm

# JavaScript Editors

# JetBrains WebStorm

- JavaScript IDE suitable for both client-side JavaScript and server (Node.js) development
  - Livereload
  - Build
  - Code highlighting/Intellisense for:
    - LESS/SASS/Stylus CSS preprocessors
    - CoffeeScript/TypeScript JavaScript preprocessors
    - AngularJS directives
    - Emmet Coding
  - Has only 30-days-long free trial

# Sublime Text 2/3

- Editor for scripting languages
  - JavaScript, PHP, Python, Ruby, etc...
  - Basic code highlighting with vanilla installation
    - Endless number of configurable plugins
  - Free product
    - Paid only if used commercially
  - With plugins support for:
    - LESS/SASS/Stylus
    - CoffeeScript/TypeScript

# Atom: A hackable text editor

- Cross-platform editing
- Built-in package manager
- Smart autocomplete
- File system browser
- Under the hood
  - Atom is a desktop application built with HTML, JavaScript, CSS, and Node.js integration. It runs on Electron, a framework for building cross platform apps using web technologies.
- Open source
  - support by Github

# Visual Code Editor

- Smart autocomplete with IntelliSense.
  - smart completions based on variable types, function definitions, and imported modules.
- Debugging support
- Git commands built-in
- Extensible and customizable
- great integration of TypeScript and Angular 2
- Open source
  - support by Microsoft

# Project Tools

No matter the Editor

# Project Tools

- NPM, Yarn & Bower
  - Install Node.js packages or client libraries
- Grunt & Gulp
  - Tasks runner
  - Create different tasks for build/development/test cases
- Yeoman
  - Scaffolding of applications
  - One-line-of-code to create a project template with views/routes/modules/etc...

# Package Management

NPM, Yarn & Bower

# Package Management: NPM

- Node.js Package Management (NPM)
- Package manager for Node.js modules

```
npm init #in CMD (Win) or Terminal (MAC/Linux)
```

- Initializes an empty Node.js project with package.json file

```
npm init
//enter package details
name: "NPM demos"
version: 0.0.1
description: "Demos for the NPM package management"
entry point: main.js
test command: test
git repository: http://github.com/user/repository-name
keywords: npm, package management
author: doncho.minkov@telerik.com
license: BSD-2-Clause
```

# Package Management: NPM

## - Installing modules

```
npm install package-name [--save][--save-dev][--save-optional]  
# Installs a package to the Node.js project
```

\*-S\*, \*--save\*: Package will appear in your dependencies in package.json  
\*-D\*, \*--save-dev\*: Package will appear in your devDependencies  
\*-O\*, \*--save-optional\*: Package will appear in your optionalDependencies.

```
npm install express --save-dev
```

## Before running the project

```
npm install ## Installs all missing packages from package.json
```

# Package Management: Bower (Deprecated)

- Bower is a package management tool for installing client-side JavaScript libraries
  - Like jQuery, KendoUI, AngularJS, etc...
  - It is a Node.js package and should be installed first

```
npm install -g bower
bower init # in CMD (Win) or Terminal (Mac/Linux)
```

- Asks for pretty much the same details as  $\$ npm init$
- Creates bower.json file to manage libraries

# Package Management: Bower (Deprecated)

- Searching for libraries

```
bower search kendo
```

```
Search results: -  
kendo-ui git://github.co  
angular-kendo-ui git://g  
kendo-backbone git://git  
knockout-kendo git://git  
kendo-global git://githu
```

- Installing libraries

```
bower install kendo-ui
```

# Grunt & Gulp & NPM

Tasks Runner

# Tasks Runner

- Grunt/Gulp are Node.js task runners
  - They can run different tasks, based on configuration
  - Tasks can be:
    - Concat and minify JavaScript/CSS files
    - Compile SASS/LESS/Stylus
    - Run jshint, csshint
    - Run Unit Tests
    - Deploy to Git, Cloud, etc...
    - And many many more

# Task Runner

- Why use a task runner?
  - Task runners gives us automation, even for different profiles:

DEVELOPMENT	TEST	BUILD
jshint	jshint	jshint
stylus	stylus	stylus
csshint	csshint	csshint
connect	mocha	concat
watch		uglify
	copy	

# Yeoman

Application Scaffolding

# Yeoman

- Yeoman is a Node.js package for application scaffolding
  - Uses bower & NPM to install the js package
  - Has lots of generators for many types of applications:
    - MEAN, AngularJS, Kendo-UI, WebApp, WordPress, Backbone, Express, etc...
    - Each generators install both needed Node.js packages and client-side JavaScript libraries
    - Generated Gruntfile.js for build/test/serve

# Yeoman

Install Yeoman:

```
npm install -g yo
```

Install Yeoman generator:

```
npm install -g generator-jhipster
```

Scaffold Express application:

```
cd path/to/app/directory  
yo jhipster
```

Generates:

