



Community Experience Distilled

Less Web Development Essentials

Use CSS preprocessing to streamline the development and maintenance of your web applications

Foreword by Alexis Sellier, creator of Less

Bass Jobsen

[PACKT] open source*
PUBLISHING community experience distilled

Less Web Development Essentials

Use CSS preprocessing to streamline the development and maintenance of your web applications

Bass Jobsen

[PACKT] open source 
PUBLISHING community experience distilled

BIRMINGHAM - MUMBAI

Less Web Development Essentials

Copyright © 2014 Packt Publishing

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, without the prior written permission of the publisher, except in the case of brief quotations embedded in critical articles or reviews.

Every effort has been made in the preparation of this book to ensure the accuracy of the information presented. However, the information contained in this book is sold without warranty, either express or implied. Neither the author nor Packt Publishing, and its dealers and distributors will be held liable for any damages caused or alleged to be caused directly or indirectly by this book.

Packt Publishing has endeavored to provide trademark information about all of the companies and products mentioned in this book by the appropriate use of capitals. However, Packt Publishing cannot guarantee the accuracy of this information.

First published: April 2014

Production Reference: 1170414

Published by Packt Publishing Ltd.
Livery Place
35 Livery Street
Birmingham B3 2PB, UK.

ISBN 978-1-78398-146-5

www.packtpub.com

Cover Image by Faiz J. Fattohi (faizfattohi@gmail.com)

Credits

Author

Bass Jobsen

Project Coordinator

Sageer Parkar

Reviewers

Marcus Bointon

Simone Deponti

Austin Pickett

Proofreaders

Maria Gould

Paul Hindle

Commissioning Editor

Ashwin Nair

Indexer

Tejal Soni

Acquisition Editor

Richard Harvey

Graphics

Ronak Dhruv

Content Development Editor

Sruthi Kutty

Production Coordinator

Arvindkumar Gupta

Technical Editors

Kapil Hemnani

Faisal Siddiqui

Cover Work

Arvindkumar Gupta

Copy Editor

Karuna Narayanan

Foreword

Before you dive into this book, let me give you a little bit of background. In the summer of 2009, I was writing CSS for a project of mine and had developed a habit of questioning every piece of technology I used. I enjoyed CSS for the most part, but one thing bothered me: I couldn't experiment like I wanted to. I was designing a lot back then, and I strongly believed in designing directly in the browser. This meant being able to change the overall tone and style of the page quickly to try different ideas. With the usual way of organizing CSS, this can be difficult. You have to keep classes small and "composable", shifting the burden to the HTML. CSS is great when you need to translate an existing, final design to the Web. However, that's not how things work very often. More and more designers are jumping straight into CSS, closing the gap between design and implementation, and they need a tool that they can use all the way through, from ideation to finished product.

I started thinking of workarounds such as separating colors from other properties so that all classes that were of the same color would be defined together. However, I wanted colors to depend on other colors; I wanted to describe the theme as "relationships" between colors, not static values. I wanted to turn a knob and have the page change from one look to another. This was plainly impossible with the CSS of 2009. I looked for solutions in the form of preprocessors and found a few, but most of them were doing too much; they were fixing things that weren't broken, such as the core syntax of the language that I happened to like.

So, I decided to put something together that would do what I wanted; the first version of Less was born. It was quickly apparent that I wasn't the only one looking for something like this. The idea was simple, but it was a step in the right direction.

Five years later, looking back at this is interesting. If I had run into these problems with the experience I have today, would I have followed the same path? I think my intuition was correct, but never could I have predicted how difficult it is to get something like this right. It's one thing to design something for yourself; it's a completely different thing when it has to work for everyone. This has made me appreciate the quality of the work that went into the CSS specification all the more, as well as the working group's cautiousness in moving forward.

It's important to remember that Less is an extension of CSS, and much of the power of Less comes from its support for plain CSS. It's easy to forget when you have access to all the extra capabilities. However, those who know when and how to use both technologies will enjoy the greatest flexibility and control over their creations.

Alexis Sellier

@cloudhead

Creator of Less

About the Author

Bass Jobsen has been programming for the Web since 1995, from C to PHP, always looking for the most accessible interfaces. He has a special interest in the process between a designer and programmer. He believes that interfaces should work independent of a device or browser. For these reasons, working with grids and meta languages in designs makes him happy. He always looks forward to new opportunities in the Semantic and Responsive Web.

He uses Less in his daily job for web design tasks and WordPress theme development as well as other Twitter Bootstrap apps.

He is always happy to help you. He can be reached at <http://stackoverflow.com/users/1596547/bass-jobsen>.

Currently, he writes a blog (<http://bassjobsen.weblogs.fm/>), programs LBS for mobile devices (<http://www.gizzing.nl>), makes cool websites (such as <http://www.streetart.nl/>), and counsels Jamedo Websites (<http://www.jamedowebbsite.nl/>) in setting up the technical environment and requirements for their business.

You can also check out his Bootstrap WordPress Starters Theme (JBST) and other projects at GitHub at <https://github.com/bassjobsen>.

"I choose a lazy person to do a hard job. Because a lazy person will find an easy way to do it."

-Bill Gates

Acknowledgments

This book is for Colinda, Kiki, Dries, Wolf, and Leny.

Recently, I reviewed *Getting Started with Zurb Foundation 4* by Andrew D. Patterson and *Learning Zurb Foundation* by Kevin Horek. After finishing this book, I will start writing *Less Web Development Cookbook* for Packt Publishing.

Although I have written many blogs and technical project requirements in the past years, this is the first book I have written to be published. Writing this book wasn't possible without the support of my family, Caroliene, and the people of Vivent. Richard Harvey was a patient and excellent motivator and critical reader. Sruthi Kutty helped me dot the i's and cross the t's. Finally, I will thank the reviewers of this book, Simone Deponti, Austin Pickett, and Marcus Bointon, for their critical and valuable suggestions, which make this book even better.

About the Reviewers

Marcus Bointon has been a Less committer for the last couple of years, having developed a taste for Less during the early versions of Twitter Bootstrap. He has a Bachelor's degree in Computer Science from the University of London and a Master's degree from Loughborough University of Technology. He's been involved in computing since 1981 and developing for the Web since 1993. He has extensive experience in many development languages (mainly PHP), Linux and OpenBSD server admin, MySQL database design and admin, e-mail infrastructure, network design, and much more. He is the maintainer of the very popular PHPMailer e-mail sending library.

Marcus is the co-founder and technical director of Synchronmedia Limited, a UK-based company behind the smartmessages.net e-mail marketing service, and a UK partner for the 1CRM open source CRM system.

He lives with his wife and two kids in the French Alps, where he can indulge his passion for skiing and mountain biking.

Simone Deponti is a web developer from Milan, Italy. He has eight years of experience in the field, primarily in CMSes, and has contributed to some open source projects, most notably the Plone CMS. He is also the author of a small debugging tool for Less and FireLess, and he wrote the initial debugging support in the Less compiler.

He works for Abstract, a web technology agency based in Italy and Germany, as a developer and project manager. You can find him at events around the world, focusing on Python, JavaScript, and CMSes.

Austin Pickett is a freelance web developer based out of Boston, MA. He has been interested in programming since he was a child and is never seen without a computer nearby. As a self-taught designer and developer, he has worked with several of his own clients to turn their websites or applications into a reality.

Austin has his own freelance career in which he works closely with clients to create their applications. He has worked with a wide array of clients from The National Academy of Best-selling Authors and vacation property owners to web design firms.

Thanks go out to my father, Shawn Pickett, for without him I would have never been interested in computers, and to my best friend and rival, Talasan Nicholson, for without him I would have no local competition or a buddy to ping at 2 AM.

www.PacktPub.com

Support files, eBooks, discount offers and more

You might want to visit www.PacktPub.com for support files and downloads related to your book.

Did you know that Packt offers eBook versions of every book published, with PDF and ePub files available? You can upgrade to the eBook version at www.PacktPub.com and as a print book customer, you are entitled to a discount on the eBook copy. Get in touch with us at service@packtpub.com for more details.

At www.PacktPub.com, you can also read a collection of free technical articles, sign up for a range of free newsletters and receive exclusive discounts and offers on Packt books and eBooks.



<http://PacktLib.PacktPub.com>

Do you need instant solutions to your IT questions? PacktLib is Packt's online digital book library. Here, you can access, read and search across Packt's entire library of books.

Why Subscribe?

- Fully searchable across every book published by Packt
- Copy and paste, print and bookmark content
- On demand and accessible via web browser

Free Access for Packt account holders

If you have an account with Packt at www.PacktPub.com, you can use this to access PacktLib today and view nine entirely free books. Simply use your login credentials for immediate access.

Table of Contents

Preface	1
<hr/>	
Chapter 1: Improving Web Development with Less	9
<hr/>	
Using CSS3 for styling your HTML	10
Using CSS Selectors to style your HTML	10
Specificity, Inheritance, and Cascade in CSS	11
How CSS specificity works	12
Building your layouts with flexible boxes	13
Compiling Less	14
Getting started with Less	15
Using the watch function for automatic reloading	17
Debugging your code	17
Example code used in this book	19
Your first layout in Less	20
Vendor-specific rules	21
Build rounded corners with border-radius	21
Preventing cross-browser issues with CSS resets	24
Creating background gradients	25
CSS transitions, transformations, and animations	27
Box-sizing	31
Server-side compiling	34
Compressing and minimizing your CSS	35
Graphical user interfaces	36
Summary	37
<hr/>	
Chapter 2: Using Variables and Mixins	39
<hr/>	
Comments	39
Nested comments	40
Special comments	40

Variables	41
Organizing your files	42
Naming your variables	43
Using a variable	44
Organizing variables	45
The last declaration wins	46
Variable declaration is not static	48
Lazy loading	48
Escaping values	49
Mixins	50
Basic mixins	51
Parametric mixins	52
Default values	52
Naming and calling	53
Multiple parameters	54
More complex mixins for linear gradient backgrounds	55
Special variables – @arguments and @rest	58
Return values	60
Changing the behavior of a mixin	61
Switches	61
Argument matching	61
Guarded mixins	62
Using guards and argument matching to construct loops	64
The !important keyword	65
Summary	66
Chapter 3: Nested Rules, Operations, and Built-in Functions	67
The navigation structure	67
Nested rules	68
Mixins and classes	70
Variables	73
Classes and namespaces	73
Operating on numbers, colors, and variables	76
The & symbol	77
Property merging	81
Built-in functions	81
JavaScript	82
List functions	82
Using color functions	85
The darken() and lighten() functions	86
Color manipulation	87
Color operations	87

Color blending with Less	88
Type functions	89
The box-shadow mixin	90
Summary	91
Chapter 4: Avoid Reinventing the Wheel	93
<hr/>	
Revisiting background gradients	93
Unused code	94
Chrome's developer tools	94
Firebug CSS usage add-on	96
Testing your code	96
Understanding TDD	96
All about style guides	97
Building a style guide with StyleDocco	97
Testing your code with tdcss.js	99
Prebuilt mixins	100
Using single-line declarations for vendor-specific rules with Less Elements	101
Less Hat – a comprehensive library of mixins	104
Using the 3L library of prebuilt mixins	105
SEO and HTML debugging	106
ClearLess – another library of prebuilt mixins	107
Using Preboot's prebuilt mixins for your project	109
Integrating other techniques into your projects using Less	110
Using iconic fonts	111
Retina.js	116
Summary	117
Chapter 5: Integrate Less in Your Own Projects	119
<hr/>	
Importing CSS into Less	120
Using the @import rule	120
Migrating your project	122
Organizing your files	123
Converting CSS code to Less code	123
Media queries and responsive design	125
Making your layout fluid	125
Testing your layouts on a mobile phone	128
Coding first for mobile	128
Using grids in your designs and work flow	129
The role of CSS float in grids	129
Making your grid responsive	130
The role of the clearfix	132
Using a more semantic strategy	132

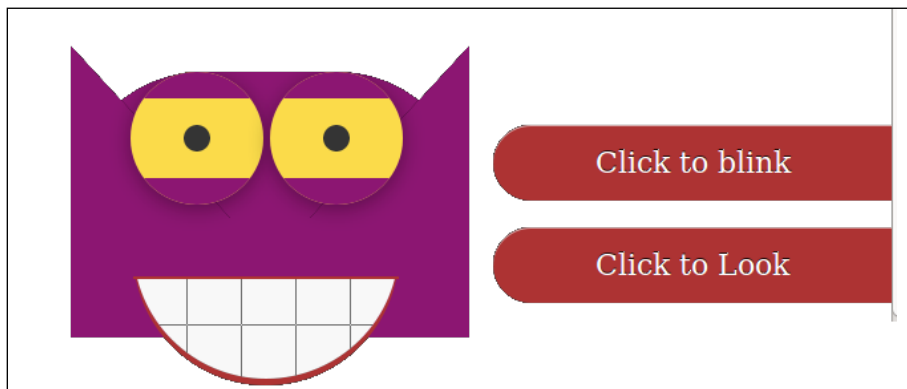
Building your layouts with grid classes	133
Building nested grids	135
Alternative grids	136
Building your project with a responsive grid	137
Using Preboot's grid system	137
Using the grid mixins to build a semantic layout	141
Extending your grids	144
Adding grid classes for the small grid	145
Applying the small grid on your semantic code	148
Summary	149
Chapter 6: Bootstrap 3, WordPress, and Other Applications	151
Bootstrap 3	151
Working with Bootstrap's Less files	153
Building a Bootstrap project with Grunt	153
Compiling your Less files	155
Dive into Bootstrap's Less files	155
Creating a custom button with Less	156
Customizing Bootstrap's navbar with Less	158
Bootstrap classes and mixins	161
Theming Bootstrap with Less	162
The a11y theme for Bootstrap	163
Color schemes with 1pxdeep	163
Using Bootstrap's customizer to build your own version	164
Semantic UI – another Less framework	164
Automatic prefixing of vendor-specific rules	165
Other frameworks to build your grid with Less	166
Using the Golden Grid System to build your grids	166
The Semantic Grid System	167
WordPress and Less	167
Using the Roots theme with Less	168
JBST with a built-in Less compiler	168
The Semantic UI WordPress theme	170
WordPress plugins and Less	170
Theme WooCommerce with Less	171
The WP Less to CSS plugin	171
Alternative compilers for compiling your Less code	171
The Less.php compiler	171
The .less compiler for .NET apps	172
List of tools to develop Less	172
Summary	173
Index	175

Preface

After the introduction of HTML 4.01 in 1999, the Web changed fast. Many new devices such as tablets and mobile phones saw the light of day. Mobile Internet became faster, cheaper, and more stable. The W3C started the HTML5 working group in 2007. In December 2012, W3C designated HTML5 as a candidate recommendation. HTML5 works with CSS3. Today, all major browsers (Chrome, Safari, Firefox, Opera, IE) offer HTML5 support.

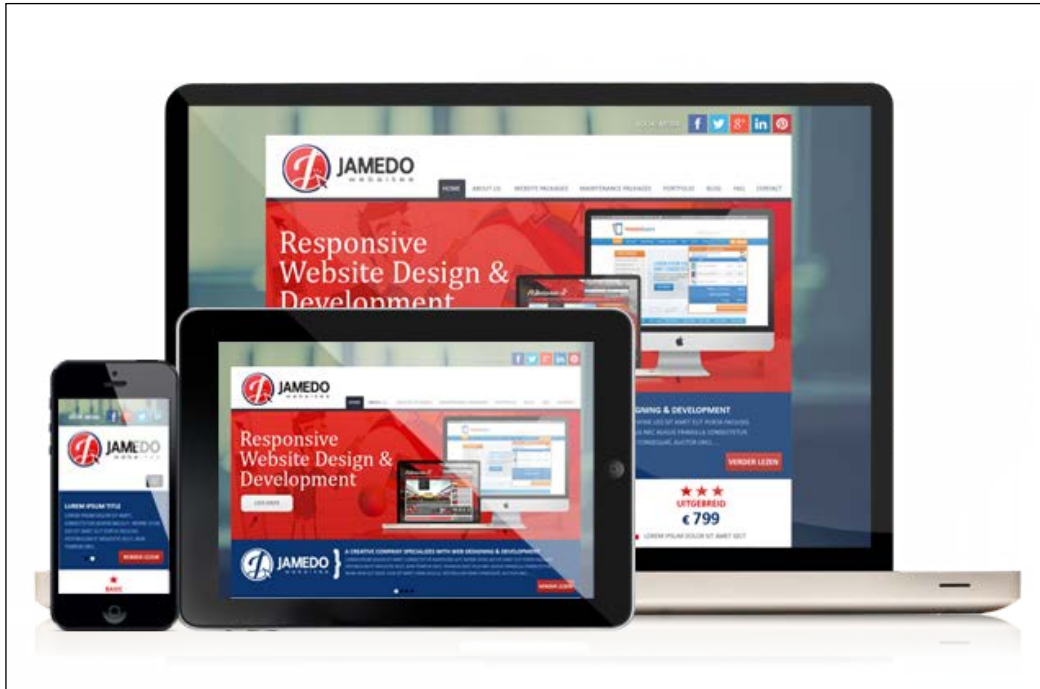
The impact of CSS3 has been huge. Nowadays, CSS3 is not only used to style your HTML documents, but CSS3 also plays an important role in the responsibility of your designs. Last but not least, CSS3 extends CSS with features such as animations and transitions.

We don't need external flash components for complex animation. Take a look at <http://www.hongkiat.com/blog/css3-animation-transition-demos/> or look at the funny owl in the following screenshot:



The owl in the preceding screenshot has been built with HTML5 and CSS3 alone. The live version can wink and look by pressing the buttons.

Responsive designs allow you to build one version of your website with only one code base which functions well and looks good on different devices such as mobile phones, tablets, and desktops. There won't be any technical reason to build different mobile and desktop versions, as shown in the following screenshot:



With all this new stuff, the work of the CSS (or web) developer becomes more complex. A web developer needs to know about complex CSS3, the difference between browsers and devices, animations, and other style effects. Writing correct and functional CSS code will be the first thing; keeping this code readable, maintainable, and working on all major browsers will be the second thing. CSS files grow and become untidy in the development and maintenance processes. CSS doesn't have the ability to modify the existing values or reuse common styles. Also, doing math or defining variables is not possible in CSS. This is where Less comes into the frame.

Less (Leaner CSS) is a dynamic stylesheet language designed by Alexis Sellier. Started in 2010, it is now maintained and extended by the Less core team. Less helps you make your CSS code maintainable, reusable, and prevent code duplications.

In this book, you will learn how to write, compile, and understand Less. We will help you do faster and more cost-effective web development. You will get practical tips to integrate Less in your current and new projects. After reading this book, you will write clear and readable CSS3 with Less. Instead of spending your time on debugging your complex CSS code for a specific device or browser, you can pay more attention to your real design tasks.

Your clients will be happy with your advanced and stable designs. This will reduce the development and maintenance time and hence the cost of designing.

Less extends CSS with functions and variables. In a semantic sense, valid CSS is also valid Less. The initial versions of Less were written in Ruby; now, Less is written in JavaScript.

Less is called a CSS precompiler. This means that the end product will be used for production. The end product in this case will be valid, compact, and readable CSS code. Besides, the precompiling Less code can also compile in real time. Less offers server-side and client-side options to do this. Real-time client-side compilation via LESS.js in a modern web browser makes testing easy. Server-side compilations offer opportunities to build applications with Less as well as create dynamic CSS.

Also, others know the power of Less. Projects such as Twitter's Bootstrap and Roots, a WordPress starter theme, both rely on Less. These projects build clear and extendable frameworks with Less. You can't ignore this proof. Stop writing cumbersome CSS with bugs and browser defects and learn about Less by reading this book.

Less is open source and licensed under the Apache license. At the time of writing this book, the latest version is 1.7. The source code of Less will be maintained on GitHub. Everybody will be allowed to contribute to it. You can use Less free of charge.

What this book covers

Chapter 1, Improving Web Development with Less, shows how CSS3 brought advanced functions such as gradients, transitions, and animations to web designers. It also explains how, on the other hand, CSS code became more complex and difficult to maintain. Less helps you make your CSS maintainable, reusable, and prevent code duplications.

Chapter 2, Using Variables and Mixins, explains why variables allow you to specify widely-used values in a single place and then reuse them throughout the style sheet, thus making global changes as easy as changing one line of code. Mixins allow you to embed all the properties of a class into another class by simply including the class name as one of its properties. The chapter also explains what parametric mixins are and how to use them.

Chapter 3, Nested Rules, Operations, and Built-in Functions, explains the use of nested rules for making inheritance clear and for making shorter style sheets. The chapter also explains how to create complex relationships between properties and how to use the built-in functions of Less.

Chapter 4, Avoid Reinventing the Wheel, teaches you how Less code and mixins can become complex because they handle different browsers and devices. The chapter also explains prebuilt mixins and other sources that help you (re)use them.

Chapter 5, Integrate Less in Your Own Projects, teaches you how to organize your files for new projects or get the projects you maintain ready for using Less.

Chapter 6, Bootstrap 3, WordPress, and Other Applications, explains what Bootstrap is and shows the strength of using Less with Bootstrap. The chapter also teaches you how to build web applications with Less or integrate it in your WordPress themes.

What you need for this book

To understand and get the full benefit of the contents of this book, we expect you to have built a website with CSS previously. A basic understanding of CSS will be required. Understanding CSS selectors and CSS precedence will help you get the most out of this book. We will introduce these CSS aspects briefly in the first chapter as well. Understanding the basics of using functions and parameters in functional languages such as JavaScript will be valuable, but it is not required. Don't panic if you know nothing about functions and parameters. This book contains clear examples. Even without any (functional) programming knowledge you can learn how to use Less, and this book will help you do this. The most important skill will be the willingness to learn.

All chapters of this book contain examples and example code. Running and testing these examples will help you develop your Less skills. You will need a modern web browser such as Google Chrome or Mozilla Firefox to run these examples. Use any preferred text or CSS editor to write your Less code.