

# Sass and Compass IN ACTION

Wynn Netherland  
Nathan Weizenbaum  
Chris Eppstein  
Brandon Mathis



---

# Sass and Compass in Action

---

Wynn Netherland, Nathan Weizenbaum, Chris Eppstein, and Brandon Mathis



# Copyright

---

For online information and ordering of this and other Manning books, please visit [www.manning.com](http://www.manning.com). The publisher offers discounts on this book when ordered in quantity. For more information, please contact

Special Sales Department  
Manning Publications Co.  
20 Baldwin Road  
PO Box 261  
Shelter Island, NY 11964  
Email: [orders@manning.com](mailto:orders@manning.com)

©2013 by Manning Publications Co. All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by means electronic, mechanical, photocopying, or otherwise, without prior written permission of the publisher.

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in the book, and Manning Publications was aware of a trademark claim, the designations have been printed in initial caps or all caps.

♻️ Recognizing the importance of preserving what has been written, it is Manning's policy to have all books we publish printed on acid-free paper, and we exert our best efforts to that end. Recognizing also our responsibility to conserve the resources of our planet, Manning books are printed on paper that is at least 15 percent recycled and processed without the use of elemental chlorine.



Manning Publications Co.  
20 Baldwin Road  
PO Box 261  
Shelter Island, NY 11964

Development editor: Scott Stirling  
Technical proofreader: Matt Martini  
Copyeditor: Andy Carroll  
Proofreader: Katie Tennant  
Typesetter: Dottie Marsico  
Cover designer: Marija Tudor

ISBN 9781617290145

Printed in the United States of America

1 2 3 4 5 6 7 8 9 10 - MAL - 18 17 16 15 14 13

# Dedication

---

*To those who craft the web and delight in the work of their hands*

# Brief Table of Contents

---

[Copyright](#)

[Brief Table of Contents](#)

[Table of Contents](#)

[Preface](#)

[Acknowledgments](#)

[About this Book](#)

[About the Authors](#)

[About the Cover Illustration](#)

[1. Getting acquainted with Sass and Compass](#)

[Chapter 1. Sass and Compass make stylesheets fun again](#)

[Chapter 2. Basic Sass syntax](#)

[2. Using Sass and Compass in practice](#)

[Chapter 3. CSS grids without the math](#)

[Chapter 4. Eliminate the mundane using Compass](#)

[Chapter 5. CSS3 with Compass](#)

[3. Tuning for production](#)

[Chapter 6. Spriting](#)

[Chapter 7. From prototype to production](#)

[Chapter 8. High-performance stylesheets](#)

[4. Advanced Sass and Compass](#)

[Chapter 9. Scripting with Sass](#)

[Chapter 10. Creating and sharing a Compass extension](#)

[Appendix A. Installing Sass and Compass](#)

[Appendix B. Getting started with Compass](#)

[Appendix C. The Sass syntaxes](#)

[Index](#)

[List of Figures](#)

[List of Tables](#)

[List of Listings](#)



# Table of Contents

---

[Copyright](#)

[Brief Table of Contents](#)

[Table of Contents](#)

[Preface](#)

[Acknowledgments](#)

[About this Book](#)

[About the Authors](#)

[About the Cover Illustration](#)

## [1. Getting acquainted with Sass and Compass](#)

[Chapter 1. Sass and Compass make stylesheets fun again](#)

[1.1. Getting started with Sass](#)

[1.1.1. From CSS to Sass](#)

[1.1.2. Think dynamic](#)

[1.1.3. Don't Repeat Yourself](#)

[1.2. Hello Sass: DRYing up your stylesheets](#)

[1.2.1. Reuse property values with variables](#)

[1.2.2. Write long selectors more quickly with nesting](#)

[1.2.3. Reuse chunks of style with mixins](#)

[1.2.4. Avoid property duplication with selector inheritance](#)

[1.3. What is Compass?](#)

[1.3.1. The Compass library](#)

[1.3.2. Simple stylesheet projects](#)

[1.3.3. Community ecosystem](#)

[1.4. Create a Compass project](#)

[1.5. Solve real-world CSS problems with Compass](#)

[1.5.1. Clear the canvas with resets](#)

[1.5.2. Create layouts without a calculator](#)

[1.5.3. Zebra-stripe like a pro with table helpers](#)

[1.5.4. Easy CSS3 without vendor prefixes](#)

[1.6. Summary](#)

### 2.1. Working with variables

#### 2.1.1. Declaring

#### 2.1.2. Referencing

#### 2.1.3. Variable names: dashes or underscores?

### 2.2. Nesting CSS rules

#### 2.2.1. &, the parent selector

#### 2.2.2. Nesting selector groups

#### 2.2.3. Child and sibling combinators: >, +, and ~

#### 2.2.4. Nested properties

### 2.3. Importing Sass files

#### 2.3.1. Using Sass partials

#### 2.3.2. Default variable values

#### 2.3.3. Nested imports

#### 2.3.4. Plain CSS imports

### 2.4. Silent comments

### 2.5. Introducing mixins

#### 2.5.1. When to use mixins

#### 2.5.2. CSS rules in mixins

#### 2.5.3. Passing arguments to a mixin

#### 2.5.4. Default argument values

### 2.6. Trimming CSS with selector inheritance

#### 2.6.1. When to use inheritance

#### 2.6.2. Advanced inheritance

#### 2.6.3. How inheritance works

#### 2.6.4. Best practices when using inheritance

### 2.7. Summary

## 2. Using Sass and Compass in practice

## Chapter 3. CSS grids without the math

### 3.1. What is a grid?

#### 3.1.1. Without CSS grids, or designing without a net



### [3.1.2. What is a grid system or framework and how does it work?](#)

---

### [3.1.3. Grids with Sass and Compass](#)

## [3.2. Getting started with grids](#)

### [3.2.1. Terminology](#)

### [3.2.2. Choosing a grid style, semantic versus pragmatic](#)

### [3.2.3. Fixed versus fluid grids](#)

## [3.3. Using Blueprint](#)

### [3.3.1. Blueprint with plain CSS](#)

### [3.3.2. Blueprint grids with Compass](#)

### [3.3.3. Blueprint in Compass without the classes](#)

## [3.4. Using 960.gs](#)

### [3.4.1. A basic 960 layout](#)

### [3.4.2. Using the 960 Grid System with Compass](#)

## [3.5. Vertical rhythm with Compass](#)

### [3.5.1. Establishing a baseline](#)

### [3.5.2. Leading and trailing whitespace](#)

## [3.6. Summary](#)

# [Chapter 4. Eliminate the mundane using Compass](#)

## [4.1. A better blank slate with targeted resets](#)

### [4.1.1. Global resets](#)

### [4.1.2. Gain more control with targeted resets](#)

## [4.2. Utilities for faster, better-looking typography](#)

### [4.2.1. Anchors away: link helpers](#)

### [4.2.2. Creating versatile lists](#)

### [4.2.3. Taming text with helpers](#)

## [4.3. Layout helpers](#)

### [4.3.1. Sticky footers](#)

### [4.3.2. Stretching elements](#)

## [4.4. Summary](#)

### 5.1. What is CSS3?

5.1.1. New properties: vendor prefixes got you down?

5.1.2. Compass to the rescue

### 5.2. Using CSS3 with Compass

5.2.1. Rounded corners

5.2.2. CSS3 shadows

5.2.3. Gradients

5.2.4. Embedding fonts with @font-face

### 5.3. Support for Internet Explorer with CSS PIE

### 5.4. Summary

## 3. Tuning for production

### Chapter 6. Spriting

6.1. How do CSS sprites work?

6.2. Why is spriting necessary?

6.2.1. The fewer HTTP requests, the better

6.2.2. The soul-crushing tedium of doing it manually

6.2.3. The Compass solution

6.3. Spriting with Compass

6.3.1. Creating a sprite map

6.3.2. Generating spriting CSS

6.4. Configuring Compass sprites

6.4.1. Customizing the sprite map

6.4.2. Customizing the Sprite CSS

6.5. Mastering the magic with sprite helpers

6.5.1. Creating sprite maps

6.5.2. Writing sprite CSS

6.6. Summary

### Chapter 7. From prototype to production

7.1. Abstracting URLs

### [7.1.1. Generating URLs to assets](#)

---

### [7.1.2. Avoiding broken links](#)

### [7.1.3. Avoiding stale images with cache busting](#)

## [7.2. Prototyping with Sass and Compass](#)

### [7.2.1. Simplifying your development environment](#)

### [7.2.2. Designing in the browser](#)

## [7.3. Deploying to production](#)

### [7.3.1. Surprise! It's moving time](#)

### [7.3.2. Compiling for production](#)

### [7.3.3. Generating domain-relative assets](#)

### [7.3.4. Adding copyright notices](#)

### [7.3.5. Deploying CSS is simple](#)

### [7.3.6. Working with source control and the deployment process](#)

### [7.3.7. Working with staging servers](#)

## [7.4. Summary](#)

## [Chapter 8. High-performance stylesheets](#)

### [8.1. Measuring client-side performance](#)

### [8.2. Avoiding HTTP requests with server-side @import](#)

### [8.3. Reducing transfer time with compression](#)

#### [8.3.1. gzip compression](#)

#### [8.3.2. Image compression](#)

### [8.4. Speeding up page loads with asset hosts](#)

#### [8.4.1. Generating URLs with asset hosts](#)

#### [8.4.2. Avoiding mixed content warnings with domain-based assets](#)

### [8.5. Inline data URIs](#)

### [8.6. Selector performance](#)

#### [8.6.1. It all adds up](#)

#### [8.6.2. The danger of over-nesting](#)

## [8.7. Summary](#)

## [4. Advanced Sass and Compass](#)

### 9.1. Using expressions

### 9.2. Understanding data types

#### 9.2.1. Strings and names

#### 9.2.2. Numbers

#### 9.2.3. Colors

#### 9.2.4. Lists

#### 9.2.5. Booleans

### 9.3. Functions

#### 9.3.1. Number functions

#### 9.3.2. Color functions

#### 9.3.3. List functions

#### 9.3.4. Other Sass functions

#### 9.3.5. User-defined functions

### 9.4. Using expressions in selectors and property names

### 9.5. Control directives

#### 9.5.1. Repeating styles for a range of numbers

#### 9.5.2. Repeating styles for a list of values

#### 9.5.3. Conditional styling

### 9.6. Summary

## Chapter 10. Creating and sharing a Compass extension

### 10.1. Sharing and reusing stylesheets

#### 10.1.1. Sass is easier to share than CSS

#### 10.1.2. Share-ready Sass

#### 10.1.3. Sharing Sass isn't enough

#### 10.1.4. Why use a Compass extension?

### 10.2. A simple extension

#### 10.2.1. Installing ad hoc extensions

#### 10.2.2. Testing your extension

### 10.3. Creating an extension demo project

### 10.4. Writing an advanced extension

[10.4.1. Automating the hard parts](#)

---

[10.4.2. Refactoring your extension](#)

[10.5. Creating a template](#)

[10.6. Distributing extensions](#)

[10.6.1. Distributing extensions in an archive](#)

[10.6.2. Distributing an extension as a Ruby gem](#)

[10.6.3. Social coding on GitHub](#)

[10.7. Summary](#)

[Appendix A. Installing Sass and Compass](#)

[A.1. Installation on Windows](#)

[A.1.1. Opening the Windows command prompt](#)

[A.1.2. Installing Ruby on Windows](#)

[A.1.3. Installing Sass and Compass on Windows](#)

[A.2. Installation on Mac OS X](#)

[A.2.1. Opening the Mac OS X Terminal](#)

[A.2.2. Installing Ruby](#)

[A.2.3. Installing Sass and Compass on Mac](#)

[A.3. Installation on Linux](#)

[A.3.1. Opening the Linux Terminal](#)

[A.3.2. Installing Ruby](#)

[A.3.3. Installing Sass and Compass on Linux](#)

[Appendix B. Getting started with Compass](#)

[B.1. Create a new project](#)

[B.1.1. Configuring options during setup](#)

[B.1.2. Adding Compass to a Rails project](#)

[B.2. Installing Compass extensions](#)

[B.2.1. Installing extensions released as Ruby gems](#)

[B.2.2. Install extensions for an existing project](#)

[B.2.3. Installing ad hoc Compass extensions](#)

[B.2.4. Installing an extension's patterns](#)

## [B.3. Configuring your Compass project](#)

---

### [B.3.1. Working with assets](#)

### [B.3.2. Configuring asset locations](#)

## [B.4. The command line](#)

### [B.4.1. Getting help](#)

## [Appendix C. The Sass syntaxes](#)

### [C.1. Indented Sass versus SCSS](#)

#### [C.1.1. Whitespace versus braces and semicolons](#)

#### [C.1.2. The @import directive](#)

#### [C.1.3. Mixins](#)

#### [C.1.4. Comments](#)

#### [C.1.5. Which is better?](#)

## [Index](#)

## [List of Figures](#)

## [List of Tables](#)

## [List of Listings](#)

# Preface

---

Just a few short years ago, the idea of a book on Sass or Compass seemed absurd. As early adopters, we knew we had seen the future of stylesheet authoring, but we struggled to gain much traction outside the Ruby community in which Sass was born. Developers often didn't see the dichotomy of using frameworks to create dynamic web pages while still writing static CSS by hand. Yet others were distrustful of Sass's only syntax at the time, the original indented, whitespace-significant syntax. It felt too rigid, like too much of a departure from CSS.

In 2010, as we worked to evangelize the benefits of Sass to our designer friends across the industry (and making some converts, we should add), Sass and the idea of preprocessed CSS began to get a foothold in development and designer circles. When Sass introduced the SCSS syntax, many of the objections to adopting Sass began to fade away and we saw a real tipping point in projects using Sass for stylesheet authoring.

At the time, many other languages with a similar vision were emerging. Much like Sirius and XM validated the idea of satellite radio, healthy competition helped validate the idea of preprocessed CSS. It was in this environment of initial industry curiosity that Manning approached us to write a book about Sass and Compass. We agreed to write this book because we wanted to share Sass with a broad audience. While it's taken much longer to produce than we wanted due to career moves and major life events for each of us, we're excited to offer this book to the community that has grown up around Sass.

If you're new to Sass, we hope it provides a solid foundation for the language and opens your eyes to new techniques. Even if you've been writing Sass for many years, we're confident you'll deepen your understanding of advanced Sass and Compass features that you'll take back to your own projects.

# Acknowledgments

---

We couldn't write a book about Sass (and Compass by extension) without thanking Hampton Catlin. Sass has made CSS fun again for so many of us. Though the syntax has evolved, Sass has always kept to the spirit of CSS while extending it in powerful ways. Hampton's vision and hard work have made an indelible mark on the project and the community.

Chris Eppstein, we (your coauthors), would like to extend our sincere appreciation to you. Without your tireless efforts extending and maintaining Sass and Compass over the last few years, we can say with certainty that the community would not be what it is today.

We'd like to thank the folks at Manning for sticking with us during a long, long journey to get this book published. Writing a book about fast-moving open source is always difficult, as the landscape is constantly shifting. We're excited to put this book in the hands of designers and developers wanting to level up their front-end tools.

Finally, special thanks to our technical proofreader, Matt Martini, for his careful reading of the final manuscript shortly before it went into production, and to the following reviewers, who read our chapters several times at different stages during development and offered invaluable feedback: Adam Michela, Adam Yonk, Andrea Ferretti, David A. Mosher, David Landau, Ezekiel Templin, Graham Ashton, Jacob Rohde, Jake Stutzman, James Hafner, Jason J. W. Williams, Jeremiah Stover, Jeroen van Dijk, Ken Paulsen, Kerrick Long, Kevin Sylvestre, Kyle Wild, Ron Chloupek, Ryan Kelln, and William Dodson.

WYNN NETHERLAND

I would like to thank my wife Polly for dealing with the stress and deadlines of yet another manuscript. Thanks for loving this crazy person.

I'd also like to thank Jason J. W. Williams, another Manning author, for sharing his polyglot authoring toolchain and countless hours of technical support.



# About this Book

---

So many of us pick up techniques from the community, learning stylesheet hacks and other tricks in short-form blog posts or screencasts. This book aims to present a top-down survey of two tools—Sass and Compass—to expand your CSS toolkit and make you a better stylesheet author. While focusing on practical application, we take a systematic approach to teaching Sass syntax and applying the patterns in the Compass framework. Hopefully, the reader will walk away with a more complete understanding of both Sass and Compass.

---

## Audience

This book is designed for two main audiences. First, we want to reach out to web designers, those who write a lot of CSS but might not have considered ways to automate parts of the stylesheet authoring process. Second, we want to show full stack developers how to treat stylesheets, images, and fonts like any other project asset and how to handle them throughout the lifecycle of a project from development to production.

---

## Roadmap

If you're new to Sass and Compass, you might find yourself jumping to [appendixes A and B](#) as you begin the book. Those appendixes provide setup instructions and other prerequisites you'll need for the book.

[Chapter 1](#) dives right into the powerful features of the Sass language. You'll discover not only exciting features, but hopefully a renewed joy for CSS when the tedium of static stylesheets disappears. We'll also give you a taste of the Compass framework in examples that provide practical application of Sass's features.

[Chapter 2](#) goes deeper into Sass and covers variables, mixins, and other language features that provide the building blocks for the rest of the book.

[Chapter 3](#) jumps right into one of the most common uses for CSS, building grid systems. As you'll see, with Sass, there's far less math involved.

[Chapter 4](#) takes a step back to give a broader view of how the Compass framework can reduce the mundane tasks that come with stylesheet authoring.

In [Chapter 5](#), we take a survey of Compass's CSS3 module and how it provides vendor-independent implementation of the most commonly used aspects of CSS3.

[Chapter 6](#) is a fun experiment with CSS sprites, an advanced technique every designer should know.

[Chapter 7](#) demonstrates how to optimize your stylesheets for both development debugging and production deployment using Compass's compile features. [Chapter 8](#) builds on this theme and shows advanced techniques to compress and minify your stylesheet assets for deployment.

[Chapter 9](#) is aimed at the advanced developer who would like to use Sass's advanced scripting techniques. [Chapter 10](#) expands on this topic and walks you through creating your own Compass plugin.

---

## Code conventions and downloads

Source code in listings or in text appears in a fixed-width font like this to separate it from the ordinary text. Code annotations accompany many of the listings, highlighting important concepts. In some cases, numbered cueballs link to additional explanations that follow the listing.

Source code for the examples in this book can be downloaded from the publisher's website at [www.manning.com/SassandCompassinAction](http://www.manning.com/SassandCompassinAction). Updates to code will be available at <https://github.com/pengwynn/sass-and-compass-in-action>.

---

## Author Online

Purchase of *Sass and Compass in Action* includes free access to a private web forum run by Manning Publications where you can make comments about the book, ask technical questions, and receive help from the authors and from other users. To access the forum and subscribe to it, point your web browser to [www.manning.com/SassandCompassinAction](http://www.manning.com/SassandCompassinAction). This page provides information on how to get on the forum once you're registered, what kind of help is available, and the rules of conduct on the forum.

Manning's commitment to our readers is to provide a venue where a meaningful dialog between individual readers and between readers and the author can take place. It's not a commitment to any specific amount of participation on the part of the authors, whose contribution to the AO remains voluntary (and unpaid). We suggest you try asking the authors some challenging questions lest their interest stray!

The Author Online forum and the archives of previous discussions will be accessible from the publisher's website as long as the book is in print.

# About the Authors

---

WYNN NETHERLAND has been building the web for nearly twenty years. He's authored or contributed several books on topics from web development to open government. When he's not shipping at GitHub, you can find him speaking at industry conferences, hanging out at developer meetups, or picking his guitar on the back porch.

CHRIS EPPSTEIN is an engineering graduate from the California Institute of Technology and has more than ten years of experience building websites and applications for Silicon Valley startups. He has a passion for front-end engineering and is currently working on front-end architecture and developer relations at LinkedIn. An active member of the Ruby open source community, Chris created the Compass Stylesheet Authoring Framework, is a member of the Sass core team, maintains many open source projects, and has contributed to dozens of others.

NATHAN WEIZENBAUM is a graduate of the University of Washington, majoring in Computer Science and Philosophy, and has been the lead developer for Sass since it was first conceived. He's currently software engineer working on Gmail at Google.

BRANDON MATHIS is on the Compass core team and creator of Octopress, a beautifully extensible blogging framework for hackers based on Jekyll. He currently is a designer at MongoHQ.

## About the Cover Illustration

---

The figure on the cover of *Sass and Compass in Action* is captioned “Silanka,” a woman from a Slavic tribe that lived in the Gail River Valley. The river, called Zilja in Slovene, originates in southern Austria and flows through some of the most picturesque landscapes of the Julian Alps. This illustration is taken from a recent reprint of *Balthasar Hacquet’s Images and Descriptions of Southwestern and Eastern Wenda, Illyrians, and Slavs* published by the Ethnographic Museum in Split, Croatia, in 2008. Hacquet (1739–1815) was an Austrian physician and scientist who spent many years studying the botany, geology, and ethnography of many parts of the Austrian Empire, as well as the Veneto, the Julian Alps, and the western Balkans, inhabited in the past by peoples of many different tribes and ethnicities. Hand-drawn illustrations accompany the many scientific papers and books that Hacquet published.

The rich diversity of the drawings in Hacquet’s publications speaks vividly of the uniqueness and individuality of the eastern Alpine and northwestern Balkan regions just 200 years ago. This was a time when the dress codes of two villages separated by a few miles identified people uniquely as belonging to one or the other, and when members of a social class or trade could be easily distinguished by what they were wearing. Dress codes have changed since then and the diversity by region, so rich at the time, has faded away. It is now often hard to tell the inhabitant of one continent from another, and today the residents of the picturesque towns and villages in the Slovenian Alps or Balkan coastal towns are not readily distinguishable from the residents of other parts of Europe.

We at Manning celebrate the inventiveness, the initiative, and the fun of the computer business with book covers based on costumes from two centuries ago brought back to life by illustrations such as this one.

# Part 1. Getting acquainted with Sass and Compass

---

The first part of this book introduces you to Sass and Compass, looking at Sass's core and covering some of the principles behind writing dynamic stylesheets. In [chapter 1](#), we look at what it means to author stylesheets dynamically and discuss development principles to help wield this power wisely. You'll see how Sass simplifies stylesheet authoring by helping you avoid repetition with selector nesting and variables, and how you can intelligently reuse common styles and patterns using `@extend` and mixins. We discuss the Compass framework and how it provides patterns and tools to make styling websites smooth and efficient.

[Chapter 2](#) helps you get acquainted with the Sass syntax and its many powerful features. We discuss how to use variables in Sass and how scoping works. You'll learn how nesting selectors and subproperties can make your stylesheets cleaner and easier to read. You'll see how Sass has improved CSS's `@import` to let you combine many stylesheets into one, allowing you to break up your styles into smaller, more manageable files. We look at using mixins to easily share common styles while avoiding repetition, and how to pass arguments and use variables in mixins, making it easy to customize the styles while preserving patterns. You'll learn to use selector inheritance with `@extend`, another way to reduce repetition, when to use inheritance and when to use mixins, and best practices.

After reading the first two chapters, you should feel comfortable with the Sass syntax and have some great ideas for how to improve your stylesheets. You'll have a good grasp of what it means to think dynamically about stylesheets. In the next part, we move from the principle to the practical, and solve some real-world problems using Sass and Compass.



- [The Curious Enlightenment of Professor Caritat: A Novel of Ideas for free](#)
- [\*\*download online Yin Yoga: Outline of a Quiet Practice \(2nd Edition\)\*\*](#)
- [download When Islam and Democracy Meet: Muslims in Europe and in the United States](#)
- [read Memory \(2nd Edition\)](#)
  
- <http://fitnessfatale.com/freebooks/Wild-Man-Creek--Virgin-River--Book-12-.pdf>
- <http://www.gateaerospaceforum.com/?library/The-Great-Book-of-Chess-Combinations.pdf>
- <http://www.uverp.it/library/Arte-Precolombiana--Art-dossier-Giunti-.pdf>
- <http://www.freightunlocked.co.uk/lib/Deeper-Than-That.pdf>