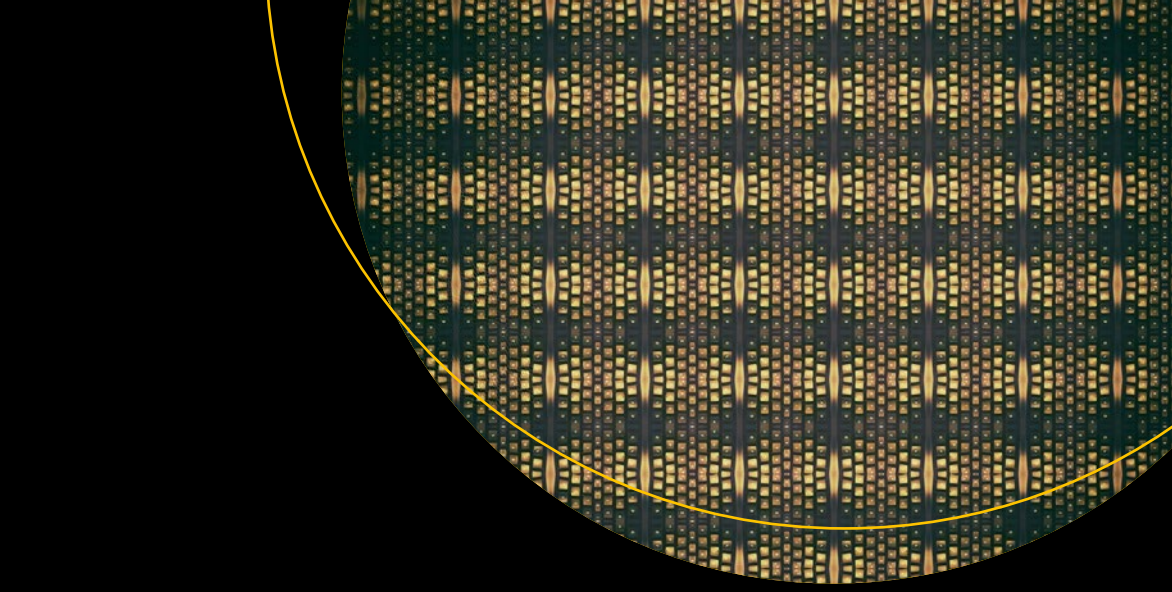

Introducing Bootstrap 4

Create Powerful Web Applications Using Bootstrap 4.5

—

Second Edition

—

Jörg Krause

Apress®

Introducing Bootstrap 4

Create Powerful Web Applications Using Bootstrap 4.5

Second Edition

Jörg Krause

Apress®

Introducing Bootstrap 4: Create Powerful Web Applications Using Bootstrap 4.5

Jörg Krause
Berlin, Germany

ISBN-13 (pbk): 978-1-4842-6202-3 ISBN-13 (electronic): 978-1-4842-6203-0
https://doi.org/10.1007/978-1-4842-6203-0

Copyright © 2020 by Jörg Krause

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

Trademarked names, logos, and images may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, logo, or image we use the names, logos, and images only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Managing Director, Apress Media LLC: Welmoed Spahr
Acquisitions Editor: Spandana Chatterjee
Development Editor: Rita Fernando
Coordinating Editor: Shrikant Vishwakarma

Cover designed by eStudioCalamar

Cover image designed by Pexels

Distributed to the book trade worldwide by Springer Science+Business Media New York, 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail orders-ny@springer-sbm.com, or visit www.springeronline.com. Apress Media, LLC is a California LLC and the sole member (owner) is Springer Science + Business Media Finance Inc (SSBM Finance Inc). SSBM Finance Inc is a **Delaware** corporation.

For information on translations, please e-mail booktranslations@springernature.com; for reprint, paperback, or audio rights, please e-mail bookpermissions@springernature.com.

Apress titles may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Print and eBook Bulk Sales web page at http://www.apress.com/bulk-sales.

Any source code or other supplementary material referenced by the author in this book is available to readers on GitHub via the book's product page, located at www.apress.com/978-1-4842-6202-3. For more detailed information, please visit http://www.apress.com/source-code.

Printed on acid-free paper

Table of Contents

About the Author xiii

About the Technical Reviewer xv

Chapter 1: Introduction to Bootstrap 1

Where to Obtain Bootstrap 1

Content Delivery Network 2

Repository for Local Installation 3

Installation Using Npm 4

Structure of the CSS Files 5

Page-building 6

Browser Support 8

ARIA 9

HTML5: The Role-Attribute 9

Optimization 11

What's New in Bootstrap 4? 15

Global Changes 15

Grid System 15

Tables 15

Forms 16

Buttons 16

Pull-Down Menus 16

Panels....16
Other Updates....16
Summary....17

Chapter2: It Begins with CSS....19

Basics....20
Syntax....21
Selector....22
Elements (Tags)....22
IDs....22
Classes....23
Attributes....24
Logical Selection....24
More Selectors....26
Units....30
Absolute Units....30
Relative Units....32
The Box Model....34
Blocks of the Box....36
Box Model Extensions....37
The Flexbox Model....38
Terminology....38
The Grid Model....45
Terminology....46
Grid Container Properties....46
Grid Item Properties....54
Compliance with Media....55
Syntax....56
Parameters....59

The Viewport 66
Setting Viewport 66
Viewport Settings 68
Summary 69

Chapter 3: Structure of the Page 71

Basis for the Grid 71
The Overall Structure of a Page 71
Settings of the Viewing Area 73
The Grid System 74
Container 74
The Grid in Detail 75
Fallbacks and Utilities 88
Equal Distribution 88
Row Columns 89
Alignments 89
Auxiliary Classes 92
The Flex Classes 92
Within the Flexbox 93
Summary 93

Chapter 4: Typography 95

Headings 96
Display Headings 97
Text and Text Elements 98
Basic Font 98
Text Elements 98
Alignment and Wrapping 103
Text Transformations 104

Lists 105
Tables 108
Styles for Tables 109
Auxiliary Classes 122
Spacing Classes 122
Sizing Classes 124
Shadows 126
Stretched Link 126
Simple Classes 127
Summary 127

Chapter 5: Forms 129

Structure of a Form 129
Simple Form Elements 129
Single-Line Forms 131
Form Elements with Blocks 135
Sizes 137
Dealing with Checkboxes and Radio Buttons 138
Additional Buttons 139
Horizontal Forms 144
Input Elements 146
Input Element Syntax 147
Select Elements 151
Static Texts in Forms 152
Behavior of the Form Elements 154
Validation Information in Forms 156
Form Elements in the Grid 164
Adaptation of the Field Height 165
Help Text in Forms 169

Buttons....170
Semantic Buttons....171
Size and Appearance....172
States....175
Summary....177

Chapter 6: Images, Effects, and Icons....179

Icons....179
Alternatives....180
Use of Symbols....182
Use of Buttons....183
Messages....184
Common Symbols....184
Responsive Images....186
Embedded Sources....187
Colors and Backgrounds....188
Text Color....188
Background Color....190
Alignment of Elements in Flow....191
Aligning Text....192
Wrapping and Overflow....192
Transformation and Font Weight....193
Break: Clearfix....193
Show and Hide Content....194
Printer Support....195
Summary....196

Chapter 7: Components 197

Requirements 197

Drop-Down Menus 198

Alignment of the Menu 200

Decorative Elements 201

Toolbars 205

Vertical Alignment 208

General Options 209

Menu Button 210

Simple Button with Menu 210

Split-Button Menu 212

Sizes of Menu Buttons 213

Special Menu Options 215

Navigation 215

Tabs 216

Navigation Buttons (Pills) 217

Universal Settings 218

Expansion Through Pop-Up Menus 220

Navigation Bar 222

Form Elements 226

Buttons, Hyperlinks, and Text 226

Breadcrumb Navigation 229

Page Scrolling 230

Size 234

Identification Labels (Badges) 234

Big Screen (Jumbotron) 236

Page Headers 237

Messages 238
Progress Bar 241
Media 248
Orientation 249
Media Lists 249
Common Lists 250
Badges in Lists 250
Links in Lists 251
Buttons in Lists 252
Cards 257
Headings 258
Footers 260
Cards with Tables 260
Cards with Lists 262
Card with Pictures 263
Cards in the Grid 264
Summary 271

Chapter 8: Active Components 273

Setup and Activation 273
The Programming Interface 274
Conflict Prevention 276
Events 276
Transitions 277
Modal Dialogs 277
Features 278
Dialog Sizes 280
Dialog with Grid 282

General Information About Behavior 288
Options 289
Pull-Down Menu (Drop-Down) 291
Common Information About Behavior 291
Options 292
Scroll Bar Supervisor (ScrollSpy) 295
Features 295
Options 297
Reversible Tabs (Tab) 298
Features 298
Options 300
Tooltip 301
Features 301
Multiline Links 303
General Tips 303
Options 304
Content Overlay (Popover) 306
Features 306
Options 309
Message (Alert) 312
Features 312
Options 313
Action Buttons (Button) 313
Features 313
Options 315
Content Insertion (Collapse) 315
Features 315
Options 320

Image Roundabout (Carousel) 321
Features 322
Options 324
Summary 326

Appendix: A Gentle Introduction to Sass 327

How It Works 327
First Steps 328
Using Sass in Applications 329
Interactive Sass 337
Variables 337
Nesting Styles 340
Operators 341
Interpolation 342
Functions 343
Modularization 344
Partials 345
Mixins 345
Summary 349

Index 351

About the Author

Jörg Krause has been working with software and software technology since the early 1980s, beginning with a Sinclair and taking his first steps as a programmer in BASIC and assembly language. He studied Information Technology at Humboldt University, Berlin, but left early, in the 1990s, to start his own company. He has worked with Internet technology and software development since the early days when CompuServe and FidoNet dominated. In 1998, he worked on one of the first commercial e-commerce solutions, and wrote his first book in Germany. Due to its wide success, he started working as a freelance consultant and author in order to share his experience and knowledge with others. He has written several books with Apress, Hanser, Addison Wesley, and other major publishers along with several self-published books—a total of over sixty titles. He also publishes articles in magazines and speaks at major conferences in Germany. Currently, Jörg works as an independent consultant, software developer, and author in Berlin. The main focus is web development (nowadays called full-stack), cloud native architectures, and web security. In his occasional spare time, Jörg enjoys reading thrillers and science fiction novels, and playing a round of golf.

About the Technical Reviewer

Kirti Mahadane is a freelance web professional based in India. Since 2015, she has been active in front-end development technologies including Bootstrap. She is particular in writing clean code.

CHAPTER 1

Introduction to Bootstrap

Bootstrap 4.5 is the latest version of the Bootstrap framework, formerly known as Twitter Bootstrap. It was built by Twitter for the mobile-first apps. It's a complete and easy-to-use system of styles that helps deal with the daily tasks of a web developer. Compared with other CSS frameworks, the biggest advantage of Bootstrap is the huge selection of additional templates, themes, and boilerplate code. It makes it very easy and fast to create a sophisticated web site without a design agency or without having any professional design skills. There are also endless additions and extensions available to fill the rare gaps. Moreover, Bootstrap can handle desktop web sites and mobile device pages equally as well, so it's really a one-stop solution.

Where to Obtain Bootstrap

Generally, Bootstrap 4 is distributed using the official web site, `https://getbootstrap.com` and npm (node package manager). Moreover, you also can create your own distribution using the source code. Bootstrap utilizes the raw files of the cascading style sheets language Sass, a pre-compiler that translates into CSS (unlike the previous version, Bootstrap 3, where the primary language was LESS). All common packaging and building tools such as WebPack, SystemJS, or Browserify can be used without restrictions to add Bootstrap directly or from the sources to existing projects.

© Jörg Krause 2020
J. Krause, *Introducing Bootstrap 4*, https://doi.org/10.1007/978-1-4842-6203-0_1

Content Delivery Network

Bootstrap is also available via Bootstrap's own content delivery network (CDN). A CDN enables a web site to frequently retrieve used public files from globally distributed servers. For example, when a user from the United States invokes your host in Germany, the CDN will ensure that the Bootstrap files are retrieved from a server in the United States. This relieves your server, the lines of the provider, and the Internet in general, resulting in a faster download for the user. Basically, this is a win–win situation. If you program on the intranet, a CDN is not to your advantage. If you expect only local users in Germany, there is no advantage to a CDN.

The Bootstrap files are included when using the CDN as follows:

```
<link rel="stylesheet"
      href="https://stackpath.bootstrapcdn.com/
      bootstrap/4.5.0/css/bootstrap.min.css"
      integrity="sha384-9aIt2nRpC12Uk9gS9baDl411NQApFmC26E
      wAOH8WgZl5MYYxFfc+NcPb1dKGj7Sk"
      crossorigin="anonymous">
<script
      src="https://stackpath.bootstrapcdn.com/
      bootstrap/4.5.0/js/bootstrap.min.js"
      integrity="sha384-OgVRvuATP1z7JjHLkuOU7Xw704+h835Lr+
      6QL9UvYjZE3Ipu6Tp75j7Bh/kROJKI"
      crossorigin="anonymous">
</script>


<!—JavaScript support -->

<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"
```

```
15          integrity="sha384-DfXdz2htPHOlsSSs5nCTpuj/zy4C+OGp
            amoFVy38MVBnE+IbbVYUew+OrCXaRkfj"
16          crossorigin="anonymous">
17  </script>
18  <script src="https://cdn.jsdelivr.net/npm/popper.
    js@1.16.0/dist/umd/popper.min.js"
19          integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9I
            OYy5n3zV9zzTtmI3UksdQRVvoxMfooAo"
20          crossorigin="anonymous">
21  </script>
```

The JavaScript libraries are jQuery for Browser abstraction and DOM access, Popper for notifications, and Bootstrap.js for all the interactive Bootstrap components. If you don't use the components you can remove all three links. If you don't use notification, simply remove Popper. The CSS parts are entirely autonomous and don't need any JavaScript.

Bootstrap 4.5.x This book was reviewed with version 4.5.0. If you use a newer version, you will need to replace all the version numbers of the type 4.5.0 accordingly.

Repository for Local Installation

To obtain a local copy, you would usually use the node package manager, npm.[1] Npm is both the repository for NodeJs and the tool to access the same. If you develop in Ruby on Rails, you should look for the gem "Bootstrap for Sass."[2] Other environments may come with their own package repositories, but because Bootstrap is entirely for frontend

[1]Npm, https://www.npmjs.com.

[2]Gem for Ruby Developers, https://rubygems.org/gems/bootstrap.

development, a package manager from the JavaScript world is the best bet. Here, npm is the most important one. However, the code world is changing constantly and other providers arise all the time, so keep an eye out for new sites for download or public repositories.

Installation Using Npm

Bootstrap can be installed as follows using the npm command line tool:

```
1   $ npm install bootstrap@latest
```

The files are now in the folder *node_modules* of your local project. This gets referenced by whatever development tool you use. Either you add the paths to *index.html* or another start file manually, or let a packer do this. (Packers are beyond the scope of this book, so you need to do your own research.) The example files provided in the Github repository for this book don't use any such dependencies, and use the manual integration instead. This simplifies the setup for beginners.

Bootstrap's file structure in *node_modules/bootstrap* contains three major parts:

- *scss*: Path to Bootstrap's Sass source files
- *js*: Path to Bootstrap's JavaScript source files
- *dist*: Path to Bootstrap's minimized CSS and JS that has been precompiled

Bootstrap was developed in version 4 with Sass.[3] Sass is available on all platforms. Bootstrap uses *Autoprefixer* to work with the vendor prefixes in CSS. If you want to compile Bootstrap by using the Sass source and not use the supplied Gruntfile, you must use an *Autoprefixer* integrated into your own build process. If you use the precompiled Bootstrap files or

[3]The Sass Language, http://sass-lang.com.

the Gruntfile, the prefixes are already integrated. The procedure applies to Gulp and other taskRunners equivalent if they are to be used. The appendix reviews this in more depth.

Bootstrap can be downloaded in two types of packages: the compiled version and additionally minimized variants.

Structure of the CSS Files

The complete structure of an environment that uses Bootstrap is as follows:

```
bootstrap/
├── css/
│   ├── bootstrap.css
│   ├── bootstrap.css.map
│   ├── bootstrap.min.css
│   ├── bootstrap.min.css.map
│   ├── bootstrap-grid.css
│   ├── bootstrap-grid.css.map
│   ├── bootstrap-grid.min.css
│   ├── bootstrap-grid.min.css.map
│   ├── bootstrap-reboot.css
│   ├── bootstrap-reboot.css.map
│   ├── bootstrap-reboot.min.css
│   └── bootstrap-reboot.min.css.map
├── js/
│   ├── bootstrap.js
│   ├── bootstrap.min.js
│   ├── bootstrap.js.map
│   ├── bootstrap.min.js.map
│   ├── bootstrap.bundle.js
│   ├── bootstrap.bundle.js.map
│   ├── bootstrap.bundle.min.js
│   └── bootstrap.bundle.min.js.map
```

Precompiled files are the easiest way to use Bootstrap. The *min* versions are additionally minimized (compressed). The *map* files support the developer tools in the browser by mapping the sources to the distributed codes. They are not supposed to be deployed to production environments.

Warning If you have a Minimizer (sometimes called Minifier or Uglifier) in your project, you pass the nonminimized files (the ones without ".min." infix). Some minimizers destroy the code if it has already been compressed by another minimizer.

In addition to CSS and JavaScript, your app may require fonts that provide the icons. Font symbols are a particularly compact and simple method to incorporate monochrome icons in web sites. Bootstrap 4 does not bundle anything here, and you need to add a font package by yourself. One well- known package is FontAwesome (more on this in Chapter 6, where various icon options are presented). For now, just note that an additional installation step required.

Page-building

Once everything is ready, you can create the first page. This page should provide the basic layout of the entire application. A first version might look like this:

```
1   <!doctype html>
2   <html lang="en">
3     <head>
4       <!-- Required meta tags -->
5       <meta charset="utf-8">
```

```
6       <meta name="viewport" content="width=device-width,
        initial-scale=1, shrink-to-fit=no">
7
8       <!-- Bootstrap CSS -->
9   <link rel="stylesheet" href="https://stackpath.
    bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.css"
    integrity="sha384-9aIt2nRpC12Uk9gS9baDl411NQApFmC26EwAOH8W
    gZl5MYYxFfc+NcPb1dKGj7Sk" crossorigin="anonymous">
10
11
12      <title>Hello, world!</title>
13    </head>
14    <body>
15      <h1>Hello, world!</h1>
16
17      <!-- Optional JavaScript -->
18      <!-- jQuery first, then Popper.js, then Bootstrap JS -->
19  <script src="https://code.jquery.com/jquery-3.5.1.slim.
    min.js" integrity="sha384-DfXdz2htPHOlsSSs5nCTpuj/
    zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj"
    crossorigin="anonymous"></script>
20  <script src="https://cdn.jsdelivr.net/npm/popper.
    js@1.16.0/dist/umd/popper.min.js" integrity="sha384-Q6E9R
    HvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3zV9zzTtmI3UksdQRVvoxMfoo
    Ao" crossorigin="anonymous"></script>
21
22    </body>
23  </html>
```

The first two metatags from line 5 must be at the beginning of the block.

Warning If you load this page locally via *file:///*, consider that some parts may not work as expected. Always try to carry out all stages of development with a local web server (IIS express with Visual Studio, IIS, NodeJs, or a local Apache or nginx are perfect for this).

Note that Bootstrap itself must be loaded after jQuery.

Tip Download only from a CDN when your web site will be public. For an intranet, a local copy is more suitable.

Browser Support

Even with Bootstrap 4, browser support remains an issue. It has been taken a good deal of effort to reach as many browsers and platforms as it does. Currently it looks like this:

Table 1-1. *Current Browser Support*

	Chrome > 45	Firefox > 38	Edge > 12	Opera > 30	Safari > 9
Android	OK	OK	OK	Not possible	N/A
iOS	OK	N/A	OK	Not possible	OK
MacOS X	OK	OK	OK	OK	OK
Windows	OK	OK	OK	OK	Not possible

Chromium and Chrome for Linux and Firefox for Linux should work, but are not officially supported. The broadest coverage with browsers you currently have is with MacOS and Windows. Internet Explorer is no longer an option in any version, but officially there is support for version 10/11. However, this support will be dropped with Bootstrap 5.

ARIA

Support for barrier-free applications (Accessible Rich Internet Applications Suite = ARIA)[4] is actually an HTML topic. Some examples in the text are already geared and carry these attributes. Here is an overview of how this works. Bootstrap supports ARIA implicitly and comprehensively.

ARIA Bootstrap's code makes use of several of these ARIA tags and attributes. Even if you don't have any intention of using it, it makes sense to understand something about Bootstrap's internal settings.

HTML5: The Role-Attribute

The `role`-attribute is set in the relevant HTML tags. It improves semantic markup and thus helps screen readers and other devices to present the correct output. It's part of the effort to help advanced screen readers to render properly and make use of distinct local resources. There is a complete description available online.[5]

For example, imagine a device that has a hardware "search" button. On your page are two forms. One is for searching your site, one for sending a contact form. With `<form role="search">` the device would be able to link the right form to the hardware button, making your page appear a lot more professionally designed.

---

[4]Accessible Rich Internet Applications (WAI-ARIA) 1.1, W3C Recommendation, 14 December 2017, http://www.w3.org/TR/wai-aria.

[5]ARIA, "Accessibility and Mozilla," https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA.

Here is the list of role-attributes:

- `banner`: The element is a banner
- `complementary`: The element adds a section, usually instead of an `<aside>`
- `content`: Regular content
- `info`: Additional information
- `form`: Form
- `main`: Main area
- `navigation`: Navigation area
- `search`: Search

The following `role`-attributes describe the structure of the page:

- `article`: Article (text)
- `columnheader`: Column header
- `definition`: Definition
- `directory`: Directory
- `document`: Document
- `group`: Group
- `heading`: Head area
- `img`: Images
- `list`: List
- `listitem`: List item
- `math`: Mathematical formula
- `note`: Note, or supplement

- `presentation`: Presentation, display support
- `region`: Area
- `row`: Row
- `rowheader`: Head of a row (left turn)
- `separator`: Separation line in menus or lists
- `toolbar`: Toolbar

An application example looks like this:

```
1   <hr role="separator" />
```

In tags like `<nav>` or `<aside>`, the `role`-attribute is redundant. The following is therefore not necessary:

```
1   <nav role="navigation">
2   <aside role="complementary">
```

ARIA Is Not an Issue? In addition to the barrier-free access, ARIA facilitates semantic attributes that take care and help the maintenance of your pages source code. It is always easier to work with `role="banner"` compared to work interspersed with dozens of `<div>` elements.

Optimization

Correctly created Bootstrap pages can be considerably larger than classic HTML pages. The stability of the styles has its price. You should therefore carefully consider how elements are created. A typical example are long lists with many options. This is where the additions come into the list of

elements, buttons, or menus you want to use. This is especially the case when the code is generated on the server. Here's an example:

```
<ul class="list-group">
  <li class="list-group-item">First Element
  <div class="btn-group">
    <button type="button"
            class="btn btn-default dropdown-toggle"
            data-toggle="dropdown" aria-haspopup="true"
            aria-expanded="false">
      Action <span class="caret"></span>
    </button>
    <ul class="dropdown-menu">
      <li><a href="#">Delete</a></li>
      <li><a href="#">Move</a></li>
      <li><a href="#">Rename</a></li>
      <li role="separator" class="divider"></li>
      <li><a href="#">Download</a></li>
    </ul>
  </div>
  </li>
  <li>... Other elements</li>
  <li>... Other elements</li>
</ul>
```

This list requires about 530 characters in UTF-16 that are more than 1 KB (in UTF-8 only the characters outside of ASCII have several characters). When you view 40 items on the page, which is not usually a problem because of the drop-downs, there are 40 KB HTML and a payload of roughly 2 KB (40 times the text for each entry with 50 bytes). Here it is worthwhile to use JavaScript.

The following code defines a template with the code of one element, and the JavaScript then adds it to the running time of each list element. For control, HTML5 attributes are used:

```
1  <ul class="list-group" data-list-target>
2    <li class="list-group-item">First Element</li>
3    <li>...other elements</li>
4    <li>...other elements</li>
5  </ul>
6  <div class="btn-group" data-list-template>
7    <button type="button"
8            class="btn btn-default dropdown-toggle"
9            data-toggle="dropdown" aria-haspopup="true"
10           aria-expanded="false">
11     Action <span class="caret"></span>
12   </button>
13     <ul class="dropdown-menu">
14       <li><a href="#">Delete</a></li>
15       <li><a href="#">Move</a></li>
16       <li><a href="#">Rename</a></li>
17       <li role="separator" class="divider"></li>
18       <li><a href="#">Download</a></li>
19     </ul>
20 </div>
```

This is then read by JavaScript using `data-list-template` attributes to address this portion in the code (line 6), and then cloned at the achievable place through `data-list-target` attributes. The code block uses jQuery here:

```
// Execution if the document was loaded
$(function(){
  // Load Template, clone and hide
  var template = $('[data-list-template]').hide().clone();
  // Search list items, Attach copied template, Show
  $('[data-list-target] li').append($(template).show());
});
```

The script will only cost about 250 bytes (130 characters without the comments). Instead of a maximum of 40 KB, this solution requires less than 2.5 KB: if you want it to look more striking, it's 6% of the original size or a decrease of 94%. In addition, the JavaScript code can be outsourced and cached in the browser.

Interactive Surfaces The script can be further refined so that the switching times only appear when the mouse pointer hovers over the entry. The `append` is only performed when a `mouseenter` emerges and `mouseleave` all buttons are removed. Ensure that events always in some `$(document).on('click')` pattern, so it works well with the dynamically appended elements. Tip: RTFM jQuery!

Why does this matter for Bootstrap? The smart structure of connected classes and the consequent usage of HTML may lead to many more elements and complex trees compared with pages written entirely manually. It clearly has advantages, and it will speed up your development time. But it will definitely come with a price. And if you want to refine your project, such techniques can be helpful. This is only one example of many such ways to improve performance.

What's New in Bootstrap 4?

This section gives an overview of the changes for those readers switching from Bootstrap 3.

Global Changes

The unit system has been changed from pixels (px) to *rem* (CSS) or *em* (Media Queries). The global font, which serves as the starting point, has been increased from 14 pixels to 16 pixels. This is mainly a reference to higher-resolution screens of mobile devices.

Grid System

So far there have been four raster layers: *xs*, *sm*, *md*, and *lg*. In the future, there will be another level: *xl*. This is designed to support extremely large displays (3,000 x 2,000 pixels and more).

Tables

Tables come with new options:

- *.table-inverse* inverts the table
- *.thead-default* and *.thead-inverse* format the header area
- *.table-sm* creates smaller tables
- All context classes now have the prefix *.table-*
- *.responsive-table* can now be directly assigned to the table element, the container is no longer needed
- *.table-reflow* supports reflow tables

Forms

Instead of the special class *.form-horizontal*, *.row* can now be used. All *.control-*- and *.input-*-classes were unified to *.form-control-*. This means that the special classes *.has-feedback* and *.help-block* are obsolete.

Buttons

The classes *.btn-default* and *.btn-info* have been omitted in favor of *.btn-secondary*. Another new feature is *.btn-xx-outline*—a bezel button. The class *.btn-xs* is removed.

Pull-Down Menus

Instead of `<ul><li>`-structures, these can now be formatted using `<div><a>`-blocks directly. These are the classes *.dropdown-item* and *.dropdown-menu*. The class *.dropdown-header* can now be applied directly to `<h1>` and the like. Intervals are created with `</div class="dropdown-divider">` rather than the previous method, which used `<li>`-tags.

Panels

The display forms Wells, Panels, and Thumbnails have been deleted and replaced by a new component with the name Card.

Other Updates

Progress bar can now be created with the HTML element `<progress>`. Quotations need the class *.blockquote*. Due to lack of browser support, this has some limitations.