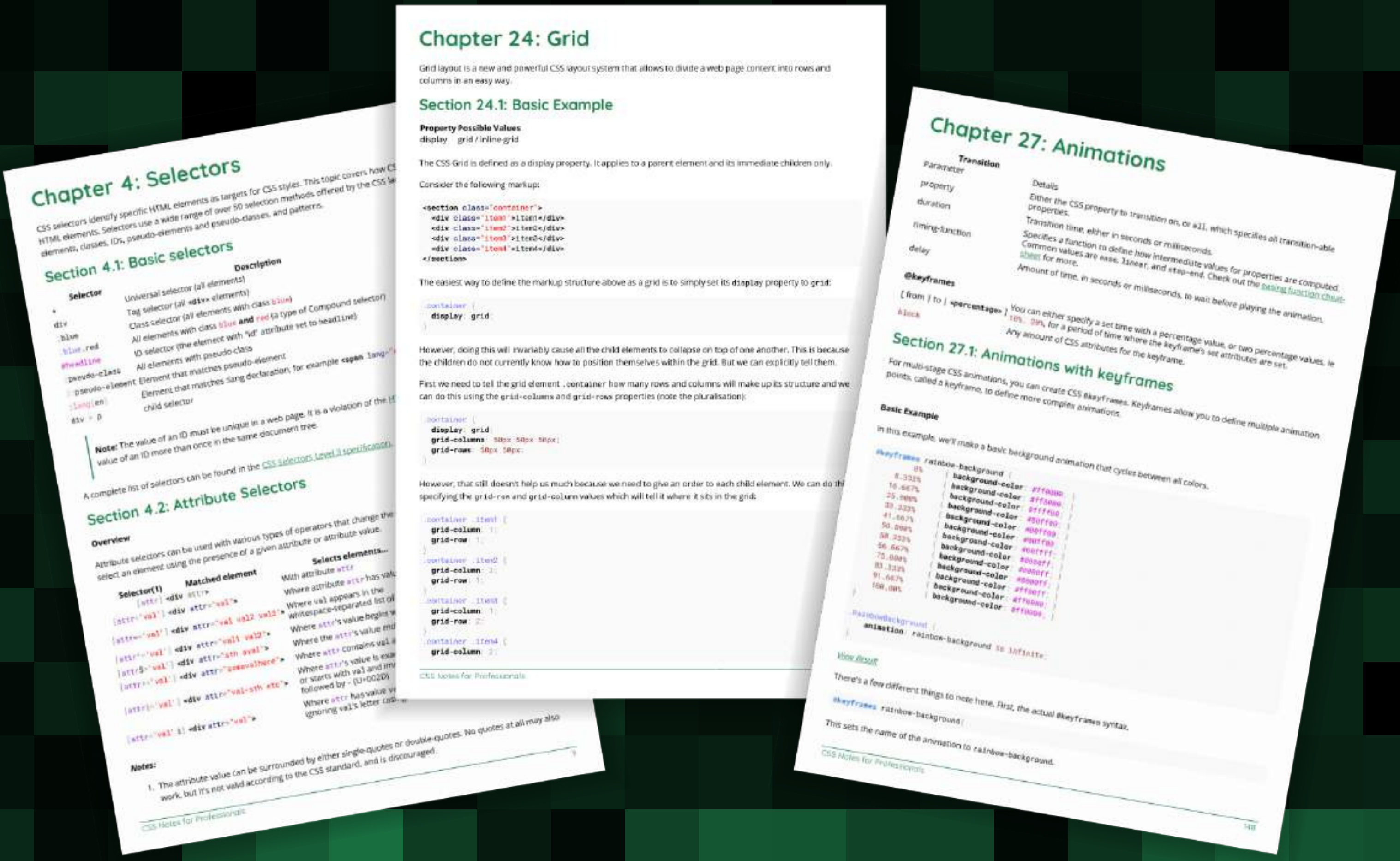
CSS

Notes for Professionals



Chapter 4: Selectors

CSS selectors identify specific HTML elements as targets for CSS styles. This topic covers how CS HTML elements. Selectors use a wide range of over 50 selection methods offered by the CSS la elements, classes, IDs, pseudo-elements and pseudo-classes, and patterns.

Section 4.1: Basic selectors

Selector	Description
`*`	Universal selector (all elements)
`div`	Tag selector (all `<div>` elements)
`.blue`	Class selector (all elements with class `blue`)
`.blue.red`	All elements with class `blue` **and** `red` (a type of Compound selector)
`#headline`	ID selector (the element with "id" attribute set to `headline`)
`:pseudo-class`	All elements with pseudo-class
`::pseudo-element`	Element that matches pseudo-element
`:lang(en)`	Element that matches :lang declaration, for example `<span lang="`
`div > p`	child selector

> **Note:** The value of an ID must be unique in a web page. It is a violation of the H value of an ID more than once in the same document tree.

A complete list of selectors can be found in the CSS Selectors Level 3 specification.

Section 4.2: Attribute Selectors

Overview

Attribute selectors can be used with various types of operators that change the select an element using the presence of a given attribute or attribute value.

Selector(1)	Matched element	Selects elements...
`[attr]`	`<div attr>`	With attribute `attr`
`[attr='val']`	`<div attr="val">`	Where attribute `attr` has valu
`[attr~='val']`	`<div attr="val val2 val3">`	Where `val` appears in the whitespace-separated list of
`[attr^='val']`	`<div attr="val1 val2">`	Where `attr`'s value begins w
`[attr$='val']`	`<div attr="sth aval">`	Where the `attr`'s value end
`[attr*='val']`	`<div attr="somevalhere">`	Where `attr` contains `val` a
`[attr|='val']`	`<div attr="val-sth etc">`	Where `attr`'s value is exa or starts with `val` and imr followed by - (U+002D)
`[attr='val' i]`	`<div attr="val">`	Where `attr` has value w ignoring `val`'s letter cas

Notes:

1. The attribute value can be surrounded by either single-quotes or double-quotes. No quotes at all may also work, but it's not valid according to the CSS standard, and is discouraged.

CSS Notes for Professionals 9

Chapter 24: Grid

Grid layout is a new and powerful CSS layout system that allows to divide a web page content into rows and columns in an easy way.

Section 24.1: Basic Example

Property	Possible Values
display	grid / inline-grid

The CSS Grid is defined as a display property. It applies to a parent element and its immediate children only.

Consider the following markup:

```
<section class="container">
  <div class="item1">item1</div>
  <div class="item2">item2</div>
  <div class="item3">item3</div>
  <div class="item4">item4</div>
</section>
```

The easiest way to define the markup structure above as a grid is to simply set its `display` property to `grid`:

```
.container {
  display: grid;
}
```

However, doing this will invariably cause all the child elements to collapse on top of one another. This is because the children do not currently know how to position themselves within the grid. But we can explicitly tell them.

First we need to tell the grid element .container how many rows and columns will make up its structure and we can do this using the `grid-columns` and `grid-rows` properties (note the pluralisation):

```
.container {
  display: grid;
  grid-columns: 50px 50px 50px;
  grid-rows: 50px 50px;
}
```

However, that still doesn't help us much because we need to give an order to each child element. We can do thi specifying the `grid-row` and `grid-column` values which will tell it where it sits in the grid:

```
.container .item1 {
  grid-column: 1;
  grid-row: 1;
}
.container .item2 {
  grid-column: 2;
  grid-row: 1;
}
.container .item3 {
  grid-column: 1;
  grid-row: 2;
}
.container .item4 {
  grid-column: 2;
```

CSS Notes for Professionals

Chapter 27: Animations

Transition

Parameter	Details
property	Either the CSS property to transition on, or `all`, which specifies all transition-able properties.
duration	Transition time, either in seconds or milliseconds.
timing-function	Specifies a function to define how intermediate values for properties are computed. Common values are `ease`, `linear`, and `step-end`. Check out the easing function cheat-sheet for more.
delay	Amount of time, in seconds or milliseconds, to wait before playing the animation.

@keyframes

[from | to | `<percentage>`]	You can either specify a set time with a percentage value, or two percentage values, ie `10%, 20%`, for a period of time where the keyframe's set attributes are set.
`block`	Any amount of CSS attributes for the keyframe.

Section 27.1: Animations with keyframes

For multi-stage CSS animations, you can create CSS `@keyframes`. Keyframes allow you to define multiple animation points, called a keyframe, to define more complex animations.

Basic Example

In this example, we'll make a basic background animation that cycles between all colors.

```
@keyframes rainbow-background {
      0%     { background-color: #ff0000; }
      8.333% { background-color: #ff8000; }
     16.667% { background-color: #ffff00; }
     25.000% { background-color: #80ff00; }
     33.333% { background-color: #00ff00; }
     41.667% { background-color: #00ff80; }
     50.000% { background-color: #00ffff; }
     58.333% { background-color: #0080ff; }
     66.667% { background-color: #0000ff; }
     75.000% { background-color: #8000ff; }
     83.333% { background-color: #ff00ff; }
     91.667% { background-color: #ff0080; }
    100.00%  { background-color: #ff0000; }
}

.RainbowBackground {
    animation: rainbow-background 5s infinite;
}
```

View Result

There's a few different things to note here. First, the actual `@keyframes` syntax.

```
@keyframes rainbow-background{
```

This sets the name of the animation to `rainbow-background`.

CSS Notes for Professionals 148

200+ pages

of professional hints and tricks

GoalKicker.com
Free Programming Books

Disclaimer
This is an unofficial free book created for educational purposes and is not affiliated with official CSS group(s) or company(s).
All trademarks and registered trademarks are the property of their respective owners

Contents

About ... 1
Chapter 1: Getting started with CSS ... 2
Section 1.1: External Stylesheet ... 2
Section 1.2: Internal Styles ... 3
Section 1.3: CSS @import rule (one of CSS at-rule) ... 4
Section 1.4: Inline Styles ... 4
Section 1.5: Changing CSS with JavaScript ... 4
Section 1.6: Styling Lists with CSS ... 5
Chapter 2: Structure and Formatting of a CSS Rule ... 7
Section 2.1: Property Lists ... 7
Section 2.2: Multiple Selectors ... 7
Section 2.3: Rules, Selectors, and Declaration Blocks ... 7
Chapter 3: Comments ... 8
Section 3.1: Single Line ... 8
Section 3.2: Multiple Line ... 8
Chapter 4: Selectors ... 9
Section 4.1: Basic selectors ... 9
Section 4.2: Attribute Selectors ... 9
Section 4.3: Combinators ... 12
Section 4.4: Pseudo-classes ... 13
Section 4.5: Child Pseudo Class ... 15
Section 4.6: Class Name Selectors ... 16
Section 4.7: Select element using its ID without the high specificity of the ID selector ... 17
Section 4.8: The :last-of-type selector ... 17
Section 4.9: CSS3 :in-range selector example ... 17
Section 4.10: A. The :not pseudo-class example & B. :focus-within CSS pseudo-class ... 18
Section 4.11: Global boolean with checkbox:checked and ~ (general sibling combinator) ... 19
Section 4.12: ID selectors ... 20
Section 4.13: How to style a Range input ... 21
Section 4.14: The :only-child pseudo-class selector example ... 21
Chapter 5: Backgrounds ... 22
Section 5.1: Background Color ... 22
Section 5.2: Background Gradients ... 24
Section 5.3: Background Image ... 25
Section 5.4: Background Shorthand ... 26
Section 5.5: Background Size ... 27
Section 5.6: Background Position ... 31
Section 5.7: The background-origin property ... 32
Section 5.8: Multiple Background Image ... 34
Section 5.9: Background Attachment ... 35
Section 5.10: Background Clip ... 36
Section 5.11: Background Repeat ... 37
Section 5.12: background-blend-mode Property ... 37
Section 5.13: Background Color with Opacity ... 38
Chapter 6: Centering ... 39
Section 6.1: Using Flexbox ... 39
Section 6.2: Using CSS transform ... 40

Section 6.3: Using margin: 0 auto; ... 41
Section 6.4: Using text-align ... 42
Section 6.5: Using position: absolute ... 42
Section 6.6: Using calc() ... 43
Section 6.7: Using line-height ... 43
Section 6.8: Vertical align anything with 3 lines of code ... 44
Section 6.9: Centering in relation to another item ... 44
Section 6.10: Ghost element technique (Michał Czernow's hack) ... 45
Section 6.11: Centering vertically and horizontally without worrying about height or width ... 46
Section 6.12: Vertically align an image inside div ... 47
Section 6.13: Centering with fixed size ... 47
Section 6.14: Vertically align dynamic height elements ... 49
Section 6.15: Horizontal and Vertical centering using table layout ... 49
Chapter 7: The Box Model ... 51
Section 7.1: What is the Box Model? ... 51
Section 7.2: box-sizing ... 52
Chapter 8: Margins ... 55
Section 8.1: Margin Collapsing ... 55
Section 8.2: Apply Margin on a Given Side ... 57
Section 8.3: Margin property simplification ... 58
Section 8.4: Horizontally center elements on a page using margin ... 58
Section 8.5: Example 1: ... 59
Section 8.6: Negative margins ... 59
Chapter 9: Padding ... 61
Section 9.1: Padding Shorthand ... 61
Section 9.2: Padding on a given side ... 62
Chapter 10: Border ... 63
Section 10.1: border-radius ... 63
Section 10.2: border-style ... 64
Section 10.3: Multiple Borders ... 65
Section 10.4: border (shorthands) ... 66
Section 10.5: border-collapse ... 66
Section 10.6: border-image ... 67
Section 10.7: Creating a multi-colored border using border-image ... 67
Section 10.8: border-[left|right|top|bottom] ... 68
Chapter 11: Outlines ... 69
Section 11.1: Overview ... 69
Section 11.2: outline-style ... 69
Chapter 12: Overflow ... 71
Section 12.1: overflow-wrap ... 71
Section 12.2: overflow-x and overflow-y ... 72
Section 12.3: overflow: scroll ... 73
Section 12.4: overflow: visible ... 73
Section 12.5: Block Formatting Context Created with Overflow ... 74
Chapter 13: Media Queries ... 76
Section 13.1: Terminology and Structure ... 76
Section 13.2: Basic Example ... 77
Section 13.3: mediatype ... 77
Section 13.4: Media Queries for Retina and Non Retina Screens ... 78

Section 13.5: Width vs Viewport ... 79
Section 13.6: Using Media Queries to Target Different Screen Sizes ... 79
Section 13.7: Use on link tag ... 80
Section 13.8: Media queries and IE8 ... 80

Chapter 14: Floats ... 81

Section 14.1: Float an Image Within Text ... 81
Section 14.2: clear property ... 82
Section 14.3: Clearfix ... 83
Section 14.4: In-line DIV using float ... 84
Section 14.5: Use of overflow property to clear floats ... 86
Section 14.6: Simple Two Fixed-Width Column Layout ... 86
Section 14.7: Simple Three Fixed-Width Column Layout ... 87
Section 14.8: Two-Column Lazy/Greedy Layout ... 88

Chapter 15: Typography ... 89

Section 15.1: The Font Shorthand ... 89
Section 15.2: Quotes ... 90
Section 15.3: Font Size ... 90
Section 15.4: Text Direction ... 90
Section 15.5: Font Stacks ... 91
Section 15.6: Text Overflow ... 91
Section 15.7: Text Shadow ... 91
Section 15.8: Text Transform ... 92
Section 15.9: Letter Spacing ... 92
Section 15.10: Text Indent ... 93
Section 15.11: Text Decoration ... 93
Section 15.12: Word Spacing ... 94
Section 15.13: Font Variant ... 94

Chapter 16: Flexible Box Layout (Flexbox) ... 96

Section 16.1: Dynamic Vertical and Horizontal Centering (align-items, justify-content) ... 96
Section 16.2: Sticky Variable-Height Footer ... 102
Section 16.3: Optimally fit elements to their container ... 103
Section 16.4: Holy Grail Layout using Flexbox ... 104
Section 16.5: Perfectly aligned buttons inside cards with flexbox ... 105
Section 16.6: Same height on nested containers ... 107

Chapter 17: Cascading and Specificity ... 109

Section 17.1: Calculating Selector Specificity ... 109
Section 17.2: The !important declaration ... 111
Section 17.3: Cascading ... 112
Section 17.4: More complex specificity example ... 113

Chapter 18: Colors ... 115

Section 18.1: currentColor ... 115
Section 18.2: Color Keywords ... 116
Section 18.3: Hexadecimal Value ... 122
Section 18.4: rgb() Notation ... 122
Section 18.5: rgba() Notation ... 123
Section 18.6: hsl() Notation ... 123
Section 18.7: hsla() Notation ... 124

Chapter 19: Opacity ... 126

Section 19.1: Opacity Property ... 126
Section 19.2: IE Compatibility for `opacity` ... 126

Chapter 20: Length Units ... 127
Section 20.1: Creating scalable elements using rems and ems ... 127
Section 20.2: Font size with rem ... 128
Section 20.3: vmin and vmax ... 129
Section 20.4: vh and vw ... 129
Section 20.5: using percent % ... 129
Chapter 21: Pseudo-Elements ... 131
Section 21.1: Pseudo-Elements ... 131
Section 21.2: Pseudo-Elements in Lists ... 131
Chapter 22: Positioning ... 133
Section 22.1: Overlapping Elements with z-index ... 133
Section 22.2: Absolute Position ... 134
Section 22.3: Fixed position ... 135
Section 22.4: Relative Position ... 135
Section 22.5: Static positioning ... 135
Chapter 23: Layout Control ... 137
Section 23.1: The display property ... 137
Section 23.2: To get old table structure using div ... 139
Chapter 24: Grid ... 141
Section 24.1: Basic Example ... 141
Chapter 25: Tables ... 143
Section 25.1: table-layout ... 143
Section 25.2: empty-cells ... 143
Section 25.3: border-collapse ... 143
Section 25.4: border-spacing ... 144
Section 25.5: caption-side ... 144
Chapter 26: Transitions ... 145
Section 26.1: Transition shorthand ... 145
Section 26.2: cubic-bezier ... 145
Section 26.3: Transition (longhand) ... 147
Chapter 27: Animations ... 148
Section 27.1: Animations with keyframes ... 148
Section 27.2: Animations with the transition property ... 149
Section 27.3: Syntax Examples ... 150
Section 27.4: Increasing Animation Performance Using the `will-change` Attribute ... 151
Chapter 28: 2D Transforms ... 152
Section 28.1: Rotate ... 152
Section 28.2: Scale ... 153
Section 28.3: Skew ... 153
Section 28.4: Multiple transforms ... 153
Section 28.5: Translate ... 154
Section 28.6: Transform Origin ... 155
Chapter 29: 3D Transforms ... 156
Section 29.1: Compass pointer or needle shape using 3D transforms ... 156
Section 29.2: 3D text effect with shadow ... 157
Section 29.3: backface-visibility ... 158
Section 29.4: 3D cube ... 159
Chapter 30: Filter Property ... 161
Section 30.1: Blur ... 161

Click here to download full PDF material