


見本

ページ組版のための

CSS 指南

2010年7月 V1.0 版

 有限会社イー・エイド



アンテナハウス株式会社

はじめに

1996年にレベル1が勧告されたCSS（Cascading Style Sheets）は、Webページのレイアウトに広く使われ、最近ではWebページのスタイルを豊富にするため富に注目され、多くのWebページで採用されています。現在、CSS2.1（Cascading Style Sheets Level 2 Revision 1）が勧告候補として提唱されており、WebブラウザでもCSS2.1仕様サポートが進んでおり、Webページの表現を豊かなものへと大きく発展してきております。

最近のCSSは、「Webデザイン」のためばかりではなく、幅広い印刷用途、PDFなどページ媒体のレイアウトにも使うことのできるスタイルシート仕様です。CSS2.1では、ページ媒体への出力については簡単な機能のみ規定しておりますが、W3Cで策定作業中のCSSレベル3では、ページサイズを始めとして高度なページレイアウト機能、段組、縦書き、さまざまな言語の文字レイアウトなど、プロの印刷品質の組版もできる機能が多く定義されています。

CSSを使ってXMLやHTMLをページ媒体向けにデザインすることは、現在、一般的ではありませんが、CSS3の仕様策定が進展するにつれて益々増えていくと予想されます。この指南書がそのために役立つことを祈念します。

AH Formatter V5.x を基準に

本書はCSSによる書籍組版について解説します。CSSによるページ媒体向けのフォーマッタは、日本語対応が完成しているものは現在のところ（株）アンテナハウスのAH Formatter V5.xですので、これをベースにして解説しております。Web画面表示用のバリエーションに富むスタイルシートの使い方は、市販の教科書が沢山ありますので、敢えて言及いたしません。

CSSという規定は、制定しているW3Cにおいても目下仕様の検討が進められている最中でして、新たな機能が追加される可能性があります。また、（株）アンテナハウスのAH Formatterは、特に日本語組版に必要な新しい機能を追加しております。そのため、次々と改善される機能に追従していかなければなりません。本書は、平成22年（2010年）4月にリリースされたVersion 5.1 MR2に基づいて解説しております。

ご注意

本チュートリアルで説明しているCSSの機能には、Antenna House Formatter（以降、Antenna House Formatterを略してAH Formatterと表記します。）の独自拡張機能があります。（AH独自拡張マークを付加）

AH Formatterの独自拡張機能につきましては、-ah-プレフィックスを付けないと正しく動作しません。

CSS3は開発中の仕様のため、CSS3で規定されている仕様は、将来、変更になることがあります。そこで、このようなプロパティには、**-ah-**プレフィックスを付けてお使いになることを強くお勧めします。本チュートリアルでも**(-ah-)**プレフィックスを付けて表示しています。なお、CSS3の仕様の中で、変更になる可能性が小さいと判断したものは、**(-ah-)**プレフィックスを付けていないものもあります。

謝辞

この解説書のベースは、アンテナハウス（株）の「CSS プリントデザイン入門」（以下単に「入門」とします）を用いております。「入門」においては、CSSの基礎的な機能や、プリントデザインに特化された機能を端的に解説しています。「CSS 指南」では、これらにW3Cで定める規格や、アンテナハウス（株）のAH Formatterのオンラインマニュアルで解説している独自の追加機能、そしてアンテナハウス（株）のWebページに紹介されている事例などを取り込み、さらに編者独自の分かりやすい例題を付け加えました。

このような流用をアンテナハウス株式会社の小林徳滋社長から快くご承諾下さいましたこと、心より感謝致します。

また、内容について詳細な校正と適切なアドバイスを下さいました、同社技術担当の村上真雄取締役には深謝いたします。ありがとうございました。

また、フォントのバリエーションを示すために株式会社ニイスのフォント製品を使わせて頂きました。同社の伊藤晃社長にはサンプル用として快くご提供下さいましたこと、心より感謝致します。

目次

はじめに

Chapter 1. HTML について	1
1.1 HTML で使われる要素と属性	1
1.1.1 HTML に含まれる要素と属性	1
1.1.2 body 要素に含まれる要素と属性	1
1.1.3 body 要素に含まれる inline 要素と属性	2
1.1.4 form 要素と frame 要素	3
1.2 CSS の記述方法	4
1.2.1 HTML の head 部に style 要素として記述する	4
1.2.2 スタイルシートを別ファイルにもち、head 部に@import として記述する	4
1.2.3 CSS の記述形式	4
1.2.4 セレクタのバリエーション	5
1.2.5 属性（プロパティ）に対する規格の読み方	5
1.2.6 使用例とコード表示	6
Chapter 2. セレクタ、@ルール、関数など	7
2.1 CSS セレクタ	7
2.1.1 タイプセレクタ——要素 { …… }	7
2.1.2 セレクタのグループ化——セレクタ, セレクタ { …… }	7
2.1.3 全ての要素を指定する——* { …… }	7
2.1.4 子要素または子孫要素を分別する——親要素名 子要素名 { …… }	8
2.1.5 子供要素を分別する——親要素名 > 子要素名 { …… }	8
2.1.6 隣接要素を分別する——要素名 1 + 要素名 2 { …… }	8
2.1.7 属性により要素を分別する——要素名[属性名="値"] { …… }	8
2.1.8 属性による要素分別の拡張——要素名[属性名~="値"] { …… }	8
2.1.9 クラスで分別する省略形——要素名.クラス名 { …… }	9
2.1.10 ID で分別する省略形——要素名#ID 名 { …… }	9
2.2 疑似クラスと疑似要素	9
2.2.1 root 指定要素——要素:root { …… }	9
2.2.2 最初の子供の要素分別——要素名:first-child { …… }	9
2.2.3 最後の子要素——要素名:last-child { …… }	10
2.2.4 最初の子要素——要素名:first-of-type { …… }	10
2.2.5 最後の子要素——要素名:last-of-type { …… }	10
2.2.6 nth-child クラス——要素:nth-child() { …… }	10
2.2.7 nth-last-child クラス——要素:nth-last-child() { …… }	11
2.2.8 同じタイプの n 番目の要素——要素:nth-of-type { …… }	11
2.2.9 同じタイプの最後の n 番目の要素——要素:nth-last-of-type { …… }	11
2.2.10 一人っ子要素——要素:only-child { …… }	11
2.2.11 単独の同じタイプ名の要素——要素:only-of-type { …… }	11
2.2.12 Empty 指定——:empty { …… }	12
2.2.13 リンク疑似クラス分別——要素:link { …… }	12
2.2.14 lang 指定要素——要素:lang { …… }	12

2.2.15 FirstLetter 要素——要素::first-letter { …… }	12
2.2.16 以外の要素——要素:not { …… }	12
2.2.17 before 疑似要素——要素名::before { …… }	13
2.2.18 after 疑似要素——要素名::after { …… }	13
2.3 脚注・側注およびリスト番号	13
2.3.1 ——::footnote-call	13
2.3.2 ——::footnote-marker	13
2.3.5 marker 疑似要素分別	14
2.4 @ルール	14
2.4.1 CSS ファイル取り込み——@import ルール	14
2.4.2 対象媒体指定——@media	15
2.4.3 文字コード指定——@charset	15
2.4.4 ネームスペース指定——@namespace	15
2.4.5 フォント指定——@font-face	15
2.4.6 ページ指定——@page	16
2.4.7 名前付き page 指定——@page Name { …… }	16
2.4.8 空白ページ指定——@page :blank { …… }	16
2.4.9 最初のページ指定——@page :first { …… }	16
2.4.10 左のページ指定——@page :left { …… }	16
2.4.11 右のページ指定——@page :right { …… }	17
2.4.12 マージンボックス——@top-left-corner, etc.	17
2.4.13 脚注——@footnote	18
2.4.14 傍注——@sidenote	18
2.5 関数	18
2.5.1 ——attr(ident, type, value)	18
2.5.2 ——calc(expr)	18
2.5.3 ——cmyk(c, m, y, k)	19
2.5.4 ——counter(ident, list-style-type)	19
2.5.5 ——counters(ident, string, list-style-type)	19
2.5.6 リーダー——leader()	19
2.5.7 ——rect(top, right, bottom, left)	19
2.5.8 ——inset-rect(top, right, bottom, left)	19
2.5.9 ——string(ident)	20
2.5.10 ——target-counter(attr, ident, list-style-type)	20
2.5.11 ——target-counters(attr, ident, string, list-style-type)	20
2.5.12 ——target-text(attr, content)	20
2.5.13 ——url(string)	20
2.5.14 ——-ah-attr-from(from, ident, type, value)	20
2.5.15 ——-ah-rgb-icc(r, g, b, name, num, num)	20
2.6 CSS キーワード	21
2.6.1 ——default	21
2.6.2 ——!important	21
2.6.3 ——inherit	21
2.6.4 ——initial	21
2.6.5 ——page	21

2.6.6	—pages	21
Chapter 3. ボックスによるレイアウト 22		
3.1	ボックスの表示と印刷	22
3.2	ボックスの配置	22
3.3	ボックスの大きさを決める特性	22
3.4	ボックス生成の制御	23
3.4.1	block レベル要素と block ボックス	23
3.4.2	表示形式—display プロパティ	23
3.5	位置決め仕組み	24
3.5.1	位置決めの選択—position プロパティ	24
3.6	コンテンツ	25
3.6.1	コンテンツの幅—width	25
3.6.2	コンテンツの最小幅—min-width	25
3.6.3	コンテンツの最大幅—max-width	25
3.6.4	コンテンツの高さ—height	26
3.6.5	コンテンツの最小高さ—min-height	26
3.6.6	コンテンツの最高高さ—max-height	26
3.7	パディング	27
3.7.1	パディングの幅—padding-top, etc.	27
3.7.2	パディングの幅—(-ah-)padding-before, etc.	27
3.7.3	パディングの幅簡略記法—padding	27
3.8	ボーダー	28
3.8.1	ボーダーのスタイル—border-top-style, etc.	28
3.8.2	ボーダーのスタイル簡略記法—border-style	28
3.8.3	ボーダーの太さ—border-top-width, etc.	29
3.8.4	border-width 簡略記法—border-width	29
3.8.5	ボーダーの色—border-top-color, etc.	30
3.8.6	ボーダーの色簡略記法—border-color	30
3.8.7	四辺の border の簡略記法—border-top, etc.	30
3.8.8	border 簡略記法—border	31
3.8.9	—(-ah-)border-before-color, etc.	31
3.8.10	—(-ah-)border-before-style, etc.	31
3.8.11	—(-ah-)border-before-width, etc.	31
3.8.12	—(-ah-)border-break	32
3.8.13	—(-ah-)border-length	32
3.9	角丸	33
3.9.1	角丸簡略記法—(-ah-)border-radius	33
3.9.2	四隅の角丸—(-ah-)border-top-left-radius, etc.	33
3.10	影付け	34
3.10.1	影付きボックス—(-ah-)box-shadow	34
3.10.2	—(-ah-)box-sizing	35
3.11	斜線罫	35
3.11.1	斜線罫の色指定—ah-diagonal-border-color <small>AH 独自拡張</small>	35
3.11.2	斜線罫のスタイル指定—ah-diagonal-border-style <small>AH 独自拡張</small>	35

3.11.3 斜線罫の幅指定—— <code>ah-diagonal-border-width</code> <small>AH 独自拡張</small>	36
3.11.4 逆斜線罫の色指定—— <code>ah-reverse-diagonal-border-color</code> <small>AH 独自拡張</small>	36
3.11.5 逆斜線罫のスタイル指定—— <code>ah-reverse-diagonal-border-style</code> <small>AH 独自拡張</small>	36
3.11.6 逆斜線罫の幅指定—— <code>ah-reverse-diagonal-border-width</code> <small>AH 独自拡張</small>	36
3.12 一般的なブロックレベルのマージン特性	37
3.12.1 マージンの幅—— <code>margin-top</code> , etc.	37
3.12.2 マージンの幅簡略記法—— <code>margin</code>	37
3.12.3 拡張マージン幅—— <code>ah-margin-before</code> , etc.	38
3.12.4 マージンの取捨—— <code>(-ah-)margin-break</code>	38
3.13 インデント・ボーダー・パディング	39
Chapter 4. ページ設定	41
4.1 <code>@page</code> ルール	41
4.2 ページに対する用紙サイズ設定	42
4.2.1 ページサイズ—— <code>size</code>	42
4.3 ページに対する <code>margin</code> 設定	43
4.3.1 マージン指定簡略記法—— <code>margin</code>	43
4.3.2 マージンボックス	44
4.4 <code>content</code> を挿入又は置き換える	44
4.4.1 文字挿入—— <code>content</code>	44
4.5 柱とノンブル	45
4.5.1 柱の設定—— <code>(-ah-)string-set</code> プロパティと <code>string()</code> 関数	45
4.5.2 —— <code>(-ah-)string-set</code>	45
4.5.3 <code>string</code> の使い方	46
4.5.4 連載要素	47
4.6 ページ番号	48
4.6.1 ノンブル (ページ番号) —— <code>counter(page)</code>	48
4.6.2 総ページ数—— <code>counter(pages)</code>	49
4.7 マスターページ	49
4.7.1 名前付きページ—— <code>page</code> プロパティ	49
4.7 左・右・先頭ページ—— <code>:left</code> 、 <code>:right</code> 、 <code>:first</code>	49
4.8 縦書き横書きの設定	52
4.8.1 <code>(-ah-)writing-mode</code>	52
4.8.2 ブロックの縦書き—— <code>(-ah-)writing-mode: tb-rl</code>	52
4.9 改ページの制御	53
4.9.1 前方改ページ—— <code>page-break-before</code>	53
4.9.2 後方改ページ—— <code>page-break-after</code>	54
4.9.3 改ページの禁止—— <code>page-break-inside</code>	54
4.10 段ブレイクの拡張	55
4.10.1 前方改段—— <code>(-ah-)break-before</code>	55
4.10.2 後方改段—— <code>(-ah-)break-after</code>	55
4.10.3 改段の禁止—— <code>(-ah-)break-inside</code>	56
4.11 XSL-FO における "keep-" 特性との違い	56
Chapter 5. 文字設定	57

5.1 書体指定	57
5.1.1 フォント——font	57
5.1.2 フォントの種類——font-family	58
5.1.3 フォントサイズ——font-size	60
5.1.4 長さの単位	62
5.1.5 フォントの太さ——font-weight	63
5.1.6 イタリック体・斜体——font-style	64
5.1.7 フォント幅指定——(-ah-)font-stretch	65
5.1.8 文字サイズ調整——(-ah-)font-size-adjust	66
5.1.9 スモールキャピタル——font-variant	67
5.1.10 ——-ah-font-selection-strategy <small>AH 独自拡張</small>	67
5.2 行の高さ——line-height	67
5.3 行ブロック進行方法	68
5.3.1 —— -ah-line-stacking-strategy <small>AH 独自拡張</small>	68
5.3.2 —— -ah-line-height-shift-adjustment <small>AH 独自拡張</small>	70
5.3.3 —— -ah-linefeed-treatment <small>AH 独自拡張</small>	70
5.4 ベースライン揃え	72
5.4.1 縦方向の揃え——vertical-align	72
5.4.2 揃え位置——(-ah-)alignment-adjust	73
5.4.3 ベースライン揃え——(-ah-)alignment-baseline	74
5.4.4 主要ベースライン——(-ah-)dominant-baseline	75
5.4.5 縦揃え——-ah-display-align <small>AH 独自拡張</small>	76
5.4.6 ベースライン移動——(-ah-)baseline-shift	76
5.4.7 相対揃え——-ah-relative-align <small>AH 独自拡張</small>	78
5.5 テキスト修飾	78
5.5.1 テキスト修飾簡略記法——(-ah-)text-decoration	78
5.5.2 下線・上線・打ち消し線の選択——(-ah-)text-decoration-line	79
5.5.3 線の色指定——(-ah-)text-decoration-color	80
5.5.4 線の種類——(-ah-)text-decoration-style	80
5.5.5 線の幅——-ah-text-line-width <small>AH 独自拡張</small>	81
5.5.6 下線位置指定——(-ah-)text-underline-position	81
5.5.7 縦組み側線制御——-ah-vertical-underline-side <small>AH 独自拡張</small>	82
Chapter 6. 段落設定	84
6.1 行揃え	84
6.1.1 行頭字下げ——text-indent	84
6.1.2 行揃え——text-align	84
6.1.3 先頭行行揃え——-ah-text-align-first <small>AH 独自拡張</small>	85
6.1.4 行揃え (均等配置+最終行設定) —— (-ah-)text-align-last	86
6.1.5 最終行行末字上げ——-ah-last-line-end-indent <small>AH 独自拡張</small>	87
6.2 Word 及び letter スペース	87
6.2.1 語間スペース——word-spacing	87
6.2.2 文字間スペース——letter-spacing	88
6.3 段組	88
6.3.1 段数——(-ah-)column-count	88

6.3.2 段間—— (-ah-)column-gap	89
6.3.3 段間罫—— (-ah-)column-rule	90
6.3.4 段間罫の種類—— (-ah-)column-rule-style	90
6.3.5 段間罫の幅—— (-ah-)column-rule-width	90
6.3.6 段間罫の色—— (-ah-)column-rule-color	90
6.3.7 段長さ揃え—— (-ah-)column-fill	91
6.3.8 段抜き指定—— (-ah-)column-span	91
6.3.9 段幅設定—— (-ah-)column-width	91
6.3.10 段数設定—— (-ah-)columns	92
6.3.11 段間罫揃え指定—— -ah-column-rule-align AH 独自拡張	92
6.3.12 段間罫長さ指定—— -ah-column-rule-length AH 独自拡張	93
6.3.13 段間罫表示制御—— -ah-column-rule-display AH 独自拡張	93
6.4 ハイフネーション	94
6.4.1 国名指定—— -ah-country AH 独自拡張	94
6.4.2 言語指定—— -ah-language AH 独自拡張	94
6.4.3 スクリプト指定—— -ah-script AH 独自拡張	95
6.4.4 ハイフネーション—— (-ah-)hyphens	95
6.4.5 ソフトハイフネーション—— -ah-soft-hyphen-treatment AH 独自拡張	96
6.4.6 分割の前の最小文字数—— (-ah-)hyphens-before	96
6.4.7 分割の後の最小文字数—— (-ah-)hyphens-after	96
6.4.8 —— (-ah-)hyphenate-character	97
6.4.9 —— (-ah-)hyphenate-lines	97
6.4.10 分離制御—— -ah-hyphenation-keep AH 独自拡張	97
6.4.11 最小文字数—— -ah-hyphenation-minimum-character-count AH 独自拡張	98
6.4.12 可能範囲—— -ah-hyphenation-zone AH 独自拡張	98
6.4.13 単語の分割—— (-ah-)word-break	98
6.4.14 —— (-ah-)word-wrap	99
6.4.15 widow 制御—— -ah-avoid-widow-words AH 独自拡張	99
Chapter 7. 日本語文字組版	101
7.1 行頭行末約物半角固定・段落先頭 1 字下げ	101
7.1.1 約物の詰め処理—— (-ah-)punctuation-trim	101
7.1.2 行調整のための約物の詰め—— (-ah-)text-justify-trim	103
7.1.3 カーニングモード—— -ah-kerning-mode AH 独自拡張	103
7.1.4 和欧文間の空き—— -ah-text-autospace AH 独自拡張	104
7.1.5 和欧文間の空き量—— -ah-text-autospace-width AH 独自拡張	105
7.1.6 約物と全角文字間の詰め—— -ah-punctuation-spacing AH 独自拡張	105
7.1.7 突き出し・ぶら下げ—— (-ah-)hanging-punctuation	105
7.2 行頭末約物処理の色々	107
7.2.1 行頭約物半角固定・行末約物全角／半角・段落先頭 1 字下げ	107
7.2.2 行頭約物半角固定・行末句点全角・段落先頭 1 字下げ	107
7.2.3 行頭行末約物半角固定・段落先頭 1 字下げ（起こし食い込み）	108
7.3 日本語の行分割制御	109
7.3.1 行分割制御—— -ah-line-break AH 独自拡張	109
7.3.2 追加行頭禁則文字—— -ah-append-non-starter-characters AH 独自拡張	111

7.3.3 追加行末禁則文字—— <code>-ah-append-non-end-of-line-characters</code> <small>AH 独自拡張</small>	111
7.3.4 除外行頭禁則文字—— <code>-ah-except-non-starter-characters</code> <small>AH 独自拡張</small>	112
7.3.5 除外行末禁則文字—— <code>-ah-except-non-end-of-line-characters</code> <small>AH 独自拡張</small>	112
7.4 変更マーク： <code>change-bar</code>	113
7.4.1 変更マークのペア認識—— <code>(-ah-)change-bar-class</code>	113
7.4.2 変更マークの色指定—— <code>(-ah-)change-bar-color</code>	113
7.4.3 変更バーの距離制御—— <code>(-ah-)change-bar-offset</code>	113
7.4.4 変更バー位置制御—— <code>(-ah-)change-bar-placement</code>	113
7.4.5 変更バーのスタイル指定—— <code>(-ah-)change-bar-style</code>	114
7.4.6 変更バーの太さ制御—— <code>(-ah-)change-bar-width</code>	114
7.5 レイアウト関連特性	114
7.5.1 オーバーフロー制御—— <code>overflow</code>	114
7.5.2 <code>condense</code> の方法選択—— <code>-ah-overflow-condense</code> <small>AH 独自拡張</small>	115
7.5.3 <code>font-size</code> の下限値—— <code>-ah-overflow-condense-limit-font-size</code> <small>AH 独自拡張</small>	116
7.5.4 <code>font-stretch</code> の下限—— <code>-ah-overflow-condense-limit-font-stretch</code> <small>AH 独自拡張</small>	116
7.5.5 オーバーフロー代替—— <code>-ah-overflow-replace</code> <small>AH 独自拡張</small>	116
7.6 縦中横	119
7.7 ルビ・圏点	119
7.8 日本語フォント	121
Chapter 8. 画像配置	122
8.1 画像配置制御	122
8.2 画像の大きさ制御	123
8.2.1 画像拡大・縮縦横比制御	123
8.2.2 画像の高さ指定—— <code>-ah-content-height</code> <small>AH 独自拡張</small>	123
8.2.3 画像の幅指定—— <code>-ah-content-width</code> <small>AH 独自拡張</small>	124
8.2.4 画像拡大・縮尺高さスケール—— <code>-ah-allowed-height-scale</code> <small>AH 独自拡張</small>	124
8.2.6 インライン画像	124
Chapter 9. フロート	126
9.1 フロート位置制御	126
9.1.1 フロート配置—— <code>float</code>	126
9.1.2 フロート配置—— <code>(-ah-)float</code>	127
9.1.3 ページフロート—— <code>(-ah-)float</code>	128
9.1.4 コラムフロート—— <code>(-ah-)float</code>	129
9.2 回り込みの制御	129
9.2.1 回り込みの解除—— <code>clear</code>	129
9.2.2 テキストの回り込み制御—— <code>-ah-intrusion-displace</code> <small>AH 独自拡張</small>	130
Chapter 10. 表組	133
10.1 表組の基礎	133
10.1.1 表の CSS デフォルト指定	133
10.1.2 デフォルトのテーブルに味付けをする	134
10.1.3 色々な指定を験す	135
10.2 表の作成方法	136
10.3 ボックス基本特性の表への適用	137

10.4 表のボーダーモデル	138
10.4.1 ボーダーの分離制御—— border-collapse	138
10.4.2 ボーダー間の空き—— border-spacing	138
10.5 ボーダーの優先度指定	140
10.5.1 ボーダーの優先度	140
10.6 表のサイズ等の指定	140
10.6.1 表の列幅指定の決定—— table-layout	140
10.6.2 表のカラム幅設定—— width	141
10.6.3 表のロー高さ設定—— height	142
10.6.4 表の中央揃え	142
10.6.5 表のキャプションの位置—— caption-side	142
10.6.6 カラムスパン—— (-ah-)table-column-span	143
10.6.7 ロースパン—— (-ah-)table-row-span	143
10.7 セル内の文字の配置	145
10.7.1 ——text-align	145
10.7.2 ——text-align="string"	146
10.7.3 ——text-align="string"の拡張	146
10.7.4 テーブル内での縦方向揃え	147
10.7.5 ——vertical-align	147
10.8 段・ページ跨り制御	148
10.8.1 フッターの省略—— -ah-table-omit-footer-at-break AH 独自拡張	148
10.8.2 ヘッダーの省略—— -ah-table-omit-header-at-break AH 独自拡張	149
10.8.3 セル内容の繰り返し—— -ah-repeat-cell-content-at-break AH 独自拡張	149
10.8.4 表脚注の繰り返し—— -ah-repeat-footnote-in-table-footer AH 独自拡張	149
10.8.5 表頭注の繰り返し—— -ah-repeat-footnote-in-table-header AH 独自拡張	150
10.9 テーブルの作例	151
10.9.1 斜線を用いたテーブルの例	151
10.9.2 テーブルヘッダに変化を付ける	151
10.9.3 テーブルの外枠を外す例	152
10.9.4 header を縦組みにする例	153
10.9.5 header を横向きに回転する例	154
10.9.6 テーブル全体を回転させる	155
10.9.7 縦組みの表	156
Chapter 11. リスト	157
11.1 リストプロパティ	157
11.1.1 リスト指定の簡略記法—— list-style	157
11.1.2 リストマークの指定—— list-style-type	157
11.1.3 リストマークにイメージ指定—— list-style-image	161
11.1.4 マーカー位置の指定—— list-type-position	161
11.1.5 ::marker 疑似要素—— !::marker	162
11.1.6 HTML4.0 用のスタイルシート設定	164
11.1.7 リストに対する html.css の設定	165
Chapter 12. 色指定	167
12.0.1 文字色—— color	167

Chapter 13. 背景色・画像指定	168
13.0.1 背景色—— background-color	168
13.0.2 背景画像—— background-image	168
13.0.3 背景画像の繰り返し—— background-repeat	169
13.0.4 背景画連結—— background-attachment	170
13.0.4 背景画付属——background-attachment	170
13.0.5 背景画位置指定—— background-position	170
13.0.6 背景簡略記法——background	171
13.0.7 背景画像幅・高さ—— -ah-background-content-width <small>AH 独自拡張</small>	171
13.0.8 高さ基準尺——-ah-allowed-height-scale <small>AH 独自拡張</small>	171
13.0.9 幅基準尺——-ah-allowed-width-scale <small>AH 独自拡張</small>	171
13.0.10 背景画像サイズ——background-size	172
13.0.11 背景画像の種類—— -ah-background-content-type <small>AH 独自拡張</small>	172
13.0.12 背景画像の倍率——-ah-background-scaling <small>AH 独自拡張</small>	173
13.0.13 背景画像解像度—— (-ah-)background-image-resolution	173
13.0.14 背景画像切り抜き—— (-ah-)background-clip	174
13.0.15 背景画像基準—— (-ah-)background-origin	174
Chapter 14. カウンタ	176
14.1 自動カウンタ及びナンバリング	176
14.1.1 カウンタのスタイル	176
14.1.2 'display:none'の要素内のカウンタ	176
14.1.3 カウンタ名の留保	176
14.2 章や節に番号を付ける	176
14.3 カウンタの加算	177
14.3.1 カウンタの増分—— counter-increment	177
14.4 カウンタのリセット	178
14.4.1 カウンタのリセット—— counter-reset	178
14.5 ページカウンタ	179
Chapter 15. クロスリファレンス（相互参照）	180
15.1 参照	180
15.1.1 カウンタの参照—— target-counter() 関数	180
15.1.2 テキスト内容の参照—— target-text() 関数	180
15.2 目次作成	181
15.2.1 リーダーを挿入する	181
Chapter 16. 脚注	183
16.1 脚注の指定—— (-ah-)float: footnote	183
16.2 脚注のスタイル—— @footnote ルール	183
16.3 脚注番号—— ::footnote-call/::footnote-marker 疑似要素	183
Chapter 17. PDF ブックマーク	184
17.1 ブックマーク	184
17.1.1 ブックマークレベル—— (-ah-)bookmark-level	184
17.1.2 ブックマークラベル—— (-ah-)bookmark-label	184
17.1.3 ブックマークの状態—— (-ah-)bookmark-state	184

17.2 しおり	185
17.2.1 しおりのタイトル色——-ah-outline-color <small>AH 独自拡張</small>	185
17.2.2 しおりの内部リンク——-ah-outline-internal-destination <small>AH 独自拡張</small>	185
17.2.3 しおりの外部リンク——-ah-outline-external-destination <small>AH 独自拡張</small>	186
17.2.4 しおりタイトルの font-style 指定——-ah-outline-font-style <small>AH 独自拡張</small>	186
17.2.5 しおりの font-weight 指定——-ah-outline-font-weight <small>AH 独自拡張</small>	186
17.2.6 しおりをまとめて出力——-ah-outline-group <small>AH 独自拡張</small>	187
Chapter 18. MathML と SVG グラフィック	188
18.1 MathML	188
18.2 SVG グラフィック	189
Chapter 19. 印刷設定	194
19.1 トンボ (クロップマーク)	194
19.1.1 トンボの表示——(-ah)-marks	194
19.1.2 トンボの線の色——-ah-printer-marks-line-color <small>AH 独自拡張</small>	195
19.1.3 トンボの線の長さ——-ah-printer-marks-line-length <small>AH 独自拡張</small>	195
19.1.4 トンボの線の幅——-ah-printer-marks-line-width <small>AH 独自拡張</small>	196
19.1.5 版面・トンボ間マージン——-ah-printer-marks-zero-margin <small>AH 独自拡張</small>	196
19.2 断ち落とし (塗り足し)	196
19.2.1 断ち落とし (塗り足し) の幅——(-ah)-bleed	196
19.2.2 出力媒体の端から判型までの距離——-ah-crop-offset <small>AH 独自拡張</small>	197
Chapter 20. PDF 出力における注釈	198
20.1 注釈	198
20.1.1 注釈のタイプ——-ah-annotation-type <small>AH 独自拡張</small>	198
20.1.2 注釈のタイトル——-ah-annotation-title <small>AH 独自拡張</small>	198
20.1.3 注釈の内容——-ah-annotation-content <small>AH 独自拡張</small>	199
20.1.4 注釈の背景色指定——-ah-annotation-color <small>AH 独自拡張</small>	199
20.1.5 注釈のフラグ——-ah-annotation-flag <small>AH 独自拡張</small>	199
20.1.6 注釈の最初の状態——-ah-annotation-open <small>AH 独自拡張</small>	200
20.1.7 注釈の水平位置——-ah-annotation-position-horizontal <small>AH 独自拡張</small>	201
20.1.8 注釈の垂直位置——-ah-annotation-position-vertical <small>AH 独自拡張</small>	202
20.1.9 注釈の幅指定——-ah-annotation-width <small>AH 独自拡張</small>	202
20.1.10 注釈の高さ指定——-ah-annotation-height <small>AH 独自拡張</small>	202
20.1.11 注釈の書体指定——-ah-annotation-font-family <small>AH 独自拡張</small>	202
20.1.12 注釈のフォントサイズ指定——-ah-annotation-font-size <small>AH 独自拡張</small>	203
20.1.13 注釈のフォント太さ指定——-ah-annotation-font-weight <small>AH 独自拡張</small>	203
20.1.14 注釈のフォントスタイル指定——-ah-annotation-font-style <small>AH 独自拡張</small>	203
20.1.15 注釈の文字色指定——-ah-annotation-text-color <small>AH 独自拡張</small>	203
20.1.16 注釈の揃え指定——-ah-annotation-text-align <small>AH 独自拡張</small>	204
20.1.17 注釈の添付ファイル指定——-ah-annotation-file-attachment <small>AH 独自拡張</small>	204
Chapter 21. デフォルト設定	205
21.1 デフォルト設定について	205
21.1.1 HTML Default の概要	205
21.1.2 AH Formatter のデフォルト CSS	209

21.1.3 html.css	209
Chapter 22. 参照仕様	215

Chapter 1. HTML について

1.1 HTML で使われる要素と属性

現在普及している HTML の規格は、HTML4.01 と呼ばれるものと、XHTML1.0 でしょう。最近 HTML5 が注目されています。CSS は HTML から独立した規格になっていますが、CSS の設定の仕方も HTML の規格が定める要素や属性に対応して記述する必要があります。特に HTML5 では追加されたり廃止された要素や、使い方が明確に規定される要素や属性などもありますので、状況の変化に敏感に対応していく必要があります。しかしながら HTML5 が確定していないので、ここでは現状の HTML4.01 ないしは XHTML1.0 を意識して話を進めます。

現状の HTML では使用できる要素の親子関係の構造や、要素に使用できるプロパティが決められています。それらの関係を簡単に表にまとめてみました。

1.1.1 HTML に含まれる要素と属性

要素	子要素	属性
html	head, body	dir, lang, version, (xml:ns)
head	meta, title, style	profile, dir, lang
title		dir, lang
meta		name, content, http-equiv, scheme
link		href, type, rel, rev, media, charset, hreflang, target, id, class, style, title, lang, dir
style		type, media, title, lang, dir
body	<i>inline-elements</i> , <i>block</i>	bgcolor, text

1.1.2 body 要素に含まれる要素と属性

次に示す表は、HTML で使われている要素と属性をまとめたものです。要素の主な子要素となるものを列記しました。*inline-elements* は inline 要素の代表的なものを示します。body 要素には多くの子要素が含まれます。

要素	子要素	属性
h1~h6	<i>inline-elements</i>	align, id, class, style, title, lang, dir
p	<i>inline-elements</i>	align, id, class, style, title, lang, dir
div		align, id, class, style, title, lang, dir
base		href, target
address		id, class, style, title, lang, dir
blockquote		cite, id, class, style, title, lang, dir

要素	子要素	属性
hr		id, class, style, title, lang, dir
ul	li	id, class, style, title, lang, dir
ol	li	id, class, style, title, lang, dir
li	<i>inline-elements</i> , <u>block</u>	id, class, style, title, lang, dir
dl	dt, dd	id, class, style, title, lang, dir
dt	<i>inline-elements</i>	id, class, style, title, lang, dir
dd	<i>inline-elements</i> , <u>block</u>	id, class, style, title, lang, dir
table	caption, colgroup, col, thead, tfoot, tbody	border, cellpadding, cellspacing, frame, rules, width, height, id, class, style, title, lang, dir
tr	td, th	align, valign, char, charoff, id, class, style, title, lang, dir
td, th	<i>inline-elements</i> , <u>block</u>	align, valign, colspan, rowspan, width, height, char, charoff, id, class, style, title, lang, dir
caption	<i>inline-elements</i>	id, class, style, title, lang, dir
thead, tfoot, tbody	tr	align, valign, char, charoff, id, class, style, title, lang, dir
colgroup	col	span, width, align, valign, char, chraaoff, id, class, style, title, lang, dir
col		span, width, align, valign, char, chraaoff, id, class, style, title, lang, dir

1.1.3 body 要素に含まれる inline 要素と属性

要素	子要素	属性
<i>inline-elements</i>	<i>a, img, br, b, i, tt, u, strike, big, small, sup, sub, acronym, cite, code, kbd, samp, stron, var, dfn, em</i>	
<i>a</i>		href, name, target, charset, hreflang, rel, rev, type, coords, shape, id, class, style, title, lang, dir, acceskey, tabindex
<i>img</i>		src, alt, width, height, id, class, style, title, lang, dir
<i>br</i>		clear, id, class, style, title
<i>b, i, tt, u</i>		id, class, style, title, lang, dir
<i>big, small</i>		id, class, style, title, lang, dir
<i>sup, sub</i>		id, class, style, title, lang, dir
<i>abbr, acronym</i>		id, class, style, title, lang, dir
<i>cite</i>		id, class, style, title, lang, dir

要素	子要素	属性
<i>code, kbd, samp, var</i>		id, class, style, title, lang, dir
<i>dfn</i>		id, class, style, title, lang, dir
<i>em, strong</i>		id, class, style, title, lang, dir
<i>pre</i>		width, id, class, style, title, lang, dir
<i>q</i>		cite, id, class, style, title, lang, dir
<i>del, ins</i>		cite, datetime, id, class, style, title, lang, dir
<i>bdo</i>		id, class, style, title, lang, dir
<i>span</i>		id, class, style, title, lang, dir
<i>ruby</i>	<i>rb, rt, rp, rbc, rtc</i>	id, class, style, title, dir
<i>rb</i>		id, class, style, title, dir
<i>rt</i>		rtspan, id, class, style, title, dir
<i>rp</i>		id, class, style, title, dir
<i>rbc</i>	<i>rb</i>	id, class, style, title, dir
<i>rtc</i>	<i>rt</i>	id, class, style, title, dir

1.1.4 form 要素と frame 要素

Web 対応として form の機能は益々重要になってきます。しかしながら、ページ組版においては、それらもつダイナミックな機能は表現できません。また、frame の機能は今後使われなくなるもので、特にページ組版では扱えません。

要素	子要素	属性
form	<u>block</u> , <i>input</i> , <i>textarea</i>	action, method, enctype, name, accept, accept-charset, target, id, class, style, title, lang, dir
<i>input</i>		type, name, value, size, maxlength, checked, disabled, readonly, src, alt, align, ismap, usemap, axcept, id, class, style, title, lang, dir
<i>textarea</i>		cols, rows, name, disable, readonly, wrap, id, class, style, title, lang, dir
frameset	frameset, frame, noframe	cols, rows, border, id, class, title, style
frame		src, name, frameborder, marginwidth, marginheight, noresize, id, class, title, style
noframe	<u>inline-elements</u> , <u>block</u>	id, class, style, title, lang, dir

1.2 CSS の記述方法

1.2.1 HTML の head 部に style 要素として記述する

前項の HTML の構造でお分かりのように、head 部分に style 要素が含まれています。head 部に書かれた style はその HTML データのみに機能する固有のもので、他への流用はそっくりコピーして移植することになります。

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
  "http://www.3c.org/TR/html4/loose.dtd">
<html lang="ja">
  <head>
    <meta http-equiv="content-type" content="text/html; charset=UTF-8" />
    <meta http-equiv="content-style-type" content="text/css" />
    <title>CSSの手始め</title>
    <style>
      h1 { color: red; }
    </style>
  </head>
```

1.2.2 スタイルシートを別ファイルにもつ

スタイルシートを別ファイルにもち、HTML と連携を取るようにすると、そのスタイルシートは汎用化され他の HTML でも利用できます。

```
@charset "Shift-JIS";
h1 { color: red; }
```

上記のようなスタイルシート (mostsimple.css) を別ファイルとして作ったとします。これを使用する HTML の head 部分に以下のような記述をすれば、スタイルシートとの連携がとれます。

```
<head>
  @import url("mostsimple.css") print; />
  <title>CSSの手始め</title>
</head>
```

または、以下のような記述をすれば、スタイルシートとの連携がとれます。

```
<head>
  <link rel="stylesheet" type="text/css" media="print" href="mostsimple.css"/>
  <title>CSSの手始め</title>
</head>
```

1.2.3 CSS の記述形式

CSS の記述形式は、ブレース "{" と "}" とで囲まれる組版指示を指定する "プロパティ" 及びその値の "セット" と、そのプロパティが適用される要素を示す "セレクタ" からなります。

```
h1 { font-size: 24pt; }
```

ここで、"h1"が"セクタ"です。"{"以下の指示を h1 に適用させることを示しています。続く "{"と"}"で囲まれた部分が組版指示です。コロン":"の前の"font-size"がプロパティで、後の"24pt"がその値を示します。

指示するプロパティが複数ある時は、セミコロン";"で区切って、プロパティとプロパティの値との組を羅列することができます。それらを一括して "{"と"}"で括ります。

例に示す"_"はスペースを示します。このようにスペースで空きを付けた方が見やすくなります。空きの量は幾つでも同じです。またタブや改行を使って読みやすくすることもできます。

1.2.4 セクタのバリエーション

HTML の要素タグは、前章に示すようにそれほど多くはありません。表題は h1~h6 で示され、段落は p で表されます。テーブルやリスト、引用やプログラムを示す要素などがありますが、それだけでは色々な状況に対応した記述を行うには少なすぎます。

そこで例えば div 要素を使って表題や段落をグループ化したり、その div のプロパティを使って、"`<div class="section">`¹⁾ とか、"`<div class="special-report">`"などとすれば、色々なバリエーションが作られます。

これらのバリエーションに対応してセクタにも工夫ができれば、それぞれに適合した組版指示自由に行えます。次章ではセクタの色々なバリエーションを列記しました。記号や書き方に馴染みがないかも知れませんが、セクタの拡張性が豊富であることを見て下さい。

1.2.5 属性（プロパティ）に対する規格の読み方

プロパティの取りうる値などを W3C の規格に合わせて記述しているところがあります。書き方が馴染めないかと思いますが、CSS 規格を間違えなくお伝えするため敢えてそのまま掲載しています。

プロパティの定義は次のような形式で記述します。

Name:	属性（プロパティ）の名前
Value:	可能なプロパティ値及び記述方を示します
Initial:	省略値を示します
Applies to:	プロパティが使用できる要素を示します ²⁾
Percentage:	特に注意を要するときに基準となる値を示します

Value : の項に示す、可能なプロパティ値の記述の仕方に次のような規則があります。

1. auto などのキーワード
2. `<length>`、`<percentage>`などと書かれる基本的なデータ形式
3. `<border-width>`などと書かれる、他の処で定義されているプロパティの値を参照するもの
4. `<border-width>`などと値の表示にクォーテーションを付けずに書かれるもので、近くにその値に対する定義がある場合です。

1) HTML5 では、"section"という要素が追加されます。
2) 特に指定しない場合はほとんどの要素に使えるものです。

Chapter 3. ボックスによるレイアウト

3.1 ボックスの表示と印刷

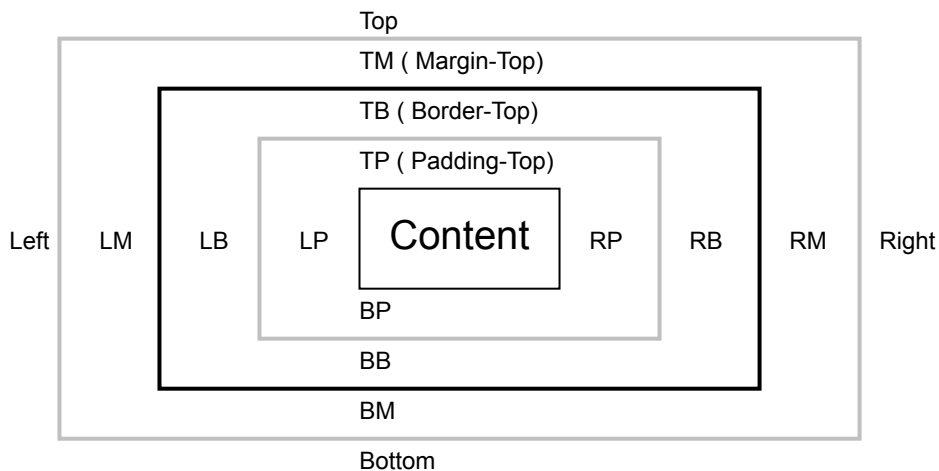
CSS では、テキスト、図、表などの組版対象物はボックスとして配置されます。最下位の文字列を含むボックスから、最上位のルート要素のボックスまで階層構造になります。出力先が画面のときは、ルート要素のボックスをウィンドウに表示します。出力先がページ媒体のときは、ページボックスでページの雛形を設定し、ページ上に他のボックスを配置します。

3.2 ボックスの配置

表以外のボックスは、内側から、コンテンツ、パディング、ボーダー（境界）をもち、最外部にマージンをもちます。それぞれに特有のプロパティによってレイアウト特性を指定します。ページボックス、ルート要素のボックス、ブロックのボックスという階層になっているとき、それぞれのコンテンツ、パディング、ボーダーは、次の図のように配置されます。表については、「[Chapter 10. 表組 \(page 133\)](#)」を参照してください。

3.3 ボックスの大きさを決める特性

ボックスの大きさは、コンテンツの幅と高さとその周りのパディング、ボーダー、マージンの4つの特性値の合計となります。



3.4 ボックス生成の制御

CSS で生成されるボックスのタイプについて説明します。

3.4.1 block レベル要素と block ボックス

block レベル要素とは、可視的なブロック（例えば段落）としてフォーマットされる原始文書の要素です。'display'プロパティのいくつかの値は、'block'、'list-item'、及び'table'などの block レベルの要素を作ります。

block レベル要素は（後に説明する'table'要素を除いて）、block ボックスのみか inline ボックスのみかのいずれかを包含する主要 block ボックスを生成します。主要 block ボックスは子孫のボックスや生成されるコンテンツに対してこれらの包含する block を生成し、位置決めの仕事を含むボックスです。主要 block ボックスは、block 生成文脈に関与します。

block レベル要素は、主要ボックスの外に、'list-item'という付加的要素を生成します。これらの付加的ボックスは、主要ボックスに準じて置かれます。

3.4.2 表示形式——display プロパティ

Name:	display
Value:	inline block list-item <i>run-in</i> inline-block table inline-table table-row-group table-header-group table-footer-group table-row table-column-group table-column table-cell table-caption none
Initial:	inline
Applies to:	all elements

block

block ボックスを作ります。

inline-block

この値は、差し替え要素と同じように単独の inline ボックスとして流し込まれる block ボックスを生成します。inline-block の中は block ボックスとしてフォーマットされ、要素自身は inline 差し替え要素としてフォーマットされます。

inline

一つ以上の inline ボックスを生成します。

list-item

要素（例えば HTML の li）に、主要 block ボックスと list-item inline ボックスを生成させます。

none

要素は何も作りません。

run-in

block または inline ボックスのどちらかを生成します。

table、etc.

table の要素を生成します。

要素の表示形式を指定するには、`display` プロパティを使います。

3.5 位置決めの仕組み

CSS2.1 では、ボックスは 3 つの位置決めスキームによって配置されます。

1. Normal. CSS2.1 における normal flow は、block ボックス、inline ボックスの block フォーマットと相対的位置決め、及び run-in ボックスの位置決めを含みます。
2. Float. float モデルでは、ボックスは、最初は normal flow で配置され、flow によって左又は右へシフトされます。
3. Absolute positioning. ボックスは normal flow から離れて block に関する位置に割り当てられます。

3.5.1 位置決めの選択——position プロパティ

Name:	position
Value:	static relative absolute fixed
Initial:	static
Applies to:	all element

static

ボックスは normal ボックスで、normal flow によって配置されます。'top'、'right'、'bottom'、及び'left'は使えません。

relative

ボックスは normal flow によって計算されます。したがって、ボックスは normal position に相対的な位置となります。

absolute


配置は'top'、'right'、'bottom'及び'left'によります。

fixed

ボックスは absolute モデルで計算され、screen モードなどではスクロールされても動きません。print モードではボックスは各ページに反映され、ページボックスに固定されているように振る舞います。

"position: relative;"を指定したインライン画像

一つ前の行。

文字列の中に  がある場合、どのように表現されるだろうか？

```
<p class="TextSample">一つ前の行。</p> <p class="TextSample" style="margin-top: 5mm;"> 文字列の中に <img src="images/07_eto_tobi05s.gif" style="width: 20mm;
```



```
position: relative; top: -1em;" />   がある
場合、どのように表現されるだろうか? </p>
```

行の高さは、画像が正規の位置にあるとして評価されています。直前の行と重ならないように margin をとる必要があります。

3.6 コンテンツ

3.6.1 コンテンツの幅——width プロパティ

Name:	width
Value:	<length> <percentage> auto inherit
Initial:	auto
Applies to:	all elements but non-replaced inline elements, table row, and rowv group

適用対象で、表の行、および文字列などを含むインライン要素の幅は、自動的に決定されるので適用対象外です。ただし、インライン要素のうちイメージ要素には適用されます。min-width、max-width プロパティも同様です。

コンテンツの幅を指定します。

3.6.2 コンテンツの最小幅——min-width プロパティ

Name:	min-width
Value:	<length> <percentage>
Initial:	0
Applies to:	all elements but non-replaced inline elements, tablerows, and row group

3.6.3 コンテンツの最大幅——max-width プロパティ

Name:	max-width
Value:	<length> <percentage> none
Initial:	none
Applies to:	all elements but non-replaced inline elements, tablerows, and row group

3.6.4 コンテンツの高さ——height プロパティ

Name:	height
Value:	<length> <percentage> inherit
Initial:	auto
Applies to:	all elements but non-replaced inline elements, table columns, and column group

適用対象で、表の列、および文字列などを含むインライン要素の高さは、自動的に決定されますので適用対象外です。ただし、インライン要素のうちイメージには適用されます。

3.6.5 コンテンツの最小高さ——min-height プロパティ

Name:	min-height
Value:	<length> <percentage> inherit
Initial:	auto
Applies to:	all elements but non-replaced inline elements, table columns, and column group

3.6.6 コンテンツの最高高さ——max-height プロパティ

Name:	max-height
Value:	<length> <percentage> auto inherit
Initial:	auto
Applies to:	all elements but non-replaced inline elements, table columns, and column group

コンテンツの大きさ指定例



画像を幅 20mm で左側に置きました。

画像を幅 80mm、高さ 15mm で右側に置きました。



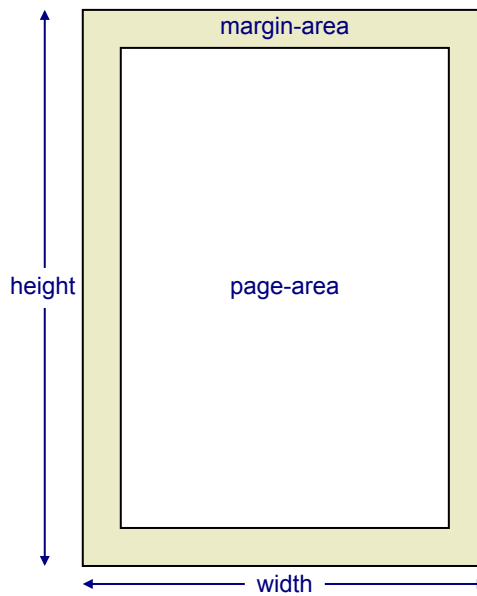
```
<p>
画像を幅20mmで左側に置きました。</p>
<p style="clear: both;">
画像を幅80mm、高さ15mmで右側に置きました。</p>
```

Chapter 4. ページ設定

4.1 @page ルール

ページサイズ、余白、ページヘッダ/ページフッタなどページの基本的な設定は、@page ルールで行います。

```
@page {  
  size: A4;  
  margin: 25mm;  
  @top-center {  
    content: "Sample";  
  }  
  @bottom-center {  
    content: counter(page);  
  }  
}
```



ページエリアとマージンエリア

4.2 ページに対する用紙サイズ設定

4.2.1 ページサイズ : size プロパティ

Name:	size
Value:	<length>{1,2} auto [<page-size> [portrait landscape]]
Initial:	auto
Applies to:	page context

<length>{1,2}

用紙の横サイズと縦サイズを指定します。値が1つだけ指定されて居るときは、縦横同寸と見なします。

<page-size> || [portrait | landscape]

A4、B5などの呼称で指定できます。用紙の縦・横指定を"portrait"、"landscape"で指定します。

size プロパティでページの幅と高さを指定します。

```
@page {
  size: 210mm 297mm; ; /* ISO/JIS A4 */
}
```

```
@page {
  size: 4in 6in;
}
```

size プロパティには、A5、A4、A3、などのページサイズ名も使えます¹⁾。

```
@page {
  size: A4; /* ISO/JIS A4 (210mm×297mm) */
}
```

```
@page {
  size: B5; /* ISO B5 (176mm×250mm) */
}
```

```
@page {
  size: JIS-B5; /* JIS B5 (182mm×257mm) */
}
```

page-size に指定できる呼称は以下の通りです。

page-size の呼称とサイズ

1) size プロパティは、ISO 標準の名前を使いますが、ISO の B シリーズと JIS で規定している B シリーズでの寸法は異なるため、アンテナハウスの独自拡張として、JIS-B5、JIS-B4などを追加しています。

<page-size>	長×短	<page-size>	長×短	<page-size>	長×短
A3 *	297×420mm	Folio	210×330mm	Letter *	8.5×11in
A4 *	210×297mm	Hagaki	100×148mm	Lagal *	8.5×14in
A5 *	148×210mm	ISO-Designated	110×220mm	Ledger *	11×17in
A6	105×148mm	ISO-C3	324×458mm	Statement	5.5×8.5in
B4(ISO-B4) *	250×353mm	ISO-C4	229×324mm	Executive	7.25×10.5in
B5(ISO-B5) *	176×250mm	ISO-C5	162×229mm	Quarto	8.5×10.83in
B6(ISO-B6)	125×176mm	ISO-C6	114×162mm	C	17×22in
JIS-B4	257×364mm			D	22×34in
JIS-B5	182×257mm			E	34×44in
JIS-B6	128×182mm				

注：*印は CSS3 Paged Media 仕様を示します。他は AH 独自拡張です。
 キーワード **landscape** を同時に指定することで、横長の指定ができます。

```
@page {
  size: A4 landscape; /* A4ヨヨ (297mm×210mm) */
}
```

4.3 ページに対する margin 設定

4.3.1 マージン指定簡略記法——margin プロパティ

一般的なブロックレベルの margin 設定で解説した margin プロパティが、@page に適用できます。margin は、margin-top、margin-bottom、margin-left、margin-right のショートハンド（簡略記法）です。

値を 1 つ指定した場合：指定した値が [上下左右] のマージンになります。
 値を 2 つ指定した場合：記述した順に [上下] [左右] のマージンになります。
 値を 3 つ指定した場合：記述した順に [上] [左右] [下] のマージンになります。
 値を 4 つ指定した場合：記述した順に [上] [右] [下] [左] のマージンになります。

@page ルールの margin プロパティでページ余白を指定します。

```
@page {
  margin: 10%; /* 上下左右の余白は、ページ幅の10%ずつ */
}
```

```
@page {
  /* 上下余白を2cm、左右余白を3cm とする */
}
```

```

margin-top: 2cm;
margin-bottom: 2cm;
margin-left: 3cm;
margin-right: 3cm;
}

```

margin-*が指定されていないときの初期値はゼロです。¹⁾。(X)HTML の body 要素に指定された margin-*は、ページエリアの内側に取られます。したがって、(X)HTML で@page と body 要素の両方に margin-*を指定していると余白は合計になります。

4.3.2 マージンボックス

マージンボックスは、ページの周辺のマージン領域に割り当てられます。

マージンボックスは、@top-left、@top-center、@top-right、@bottom-left、@bottom-center、@bottom-right などページ周辺の位置により名前がついています。

ヘッダーやフッターはこれらのマージンボックスに割り当てます。

```

@page {
  @top-right { /* ページヘッダ */
    content: "Sample";
  }
  @bottom-center { /* ページフッタ */
    content: counter(page);
  }
}

```

4.4 文字挿入又は置き換え

4.4.1 文字挿入——content プロパティ

Name:	content
Value:	normal none [<string> <uri> <counter> attr(<identifier>) open-quote close-quote no-open-quote no-close-quote]+
Initial:	normal

要素の直前、または直後に文字列を挿入するには、content プロパティを使います。

```

normal :
    文字を挿入しません。
none :
    文字を挿入しません。

```

1) AH Formatter では、組版オプション設定でデフォルト余白を指定できます。規定値として、10%が設定されています。

Chapter 22. 参照仕様

ここまで、CSS によるページ組版で何ができるかをざっと紹介しました。以下は関係する主な CSS の仕様で、AH Formatter V5.1 が実装を進めているものです。

- ✓ **CSS2.1 (CSS Level 2 Revision 1)** W3C 勧告候補 <http://www.w3.org/TR/CSS21/>
- ✓ **CSS3 Paged Media** 最終ドラフト <http://www.w3.org/TR/css3-page/>
- ✓ **CSS3 GCPM (Generated Content for Paged Media)**
..... Editor's Draft <http://dev.w3.org/csswg/css3-gcpm/>
- ✓ **CSS3 Multi-column layout** W3C 勧告候補 <http://www.w3.org/TR/css3-multicol/>
- ✓ **CSS3 Backgrounds and Borders** 最終ドラフト <http://www.w3.org/TR/css3-background/>
- ✓ **CSS3 Lists** ドラフト仕様 <http://www.w3.org/TR/css3-lists/>
- ✓ **CSS3 Text** ドラフト仕様 <http://www.w3.org/TR/css3-text/>
- Editor's Draft <http://dev.w3.org/csswg/css3-text/>
- ✓ **CSS3 Text Layout** Editor's Draft <http://dev.w3.org/csswg/css3-text-layout/>
- ✓ **CSS3 Selectors** W3C 勧告案 <http://www.w3.org/TR/css3-selectors/>
- ✓ **CSS3 Namespaces** W3C 勧告候補 <http://www.w3.org/TR/css3-namespace/>
- ✓ **HTML 5 – A vocabulary and associated APIs for HTML and XHTML**
..... ドラフト仕様 <http://www.w3.org/TR/html5/>

なお、AH Formatter は、XSL-FO による組版も行えます。

- ✓ **Extensible Stylesheet Language (XSL) Version 1.1** W3C 勧告 <http://www.w3.org/TR/xsl/>

ページ組版のための「CSS 指南」

2010年7月8日 V1.0 版発行

編・著 藤島 雅宏

監 修 アンテナハウス株式会社

発行所 有限会社イー・エイド

〒 103-0004 東京都千代田区西神田 2-2-6-401

電話 03-5212-1891 / FAX 03-5212-1891

<http://www.eaid.jp/>

©有限会社イー・エイド

本冊子の内容は HTML で記述されており、CSS スタイルシートでレイアウト指定をし、AH Formatter V5.1 MR2 で PDF 化しています。