

CSS によるプリントデザイン入門

アンテナハウス株式会社

2008 年 12 月発行



はじめに

CSS (Cascading Style Sheets) は 1996 年にレベル 1 が勧告され、Web ページのレイアウトに広く使われています。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います。

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、ページレイアウト機能はあまり強力ではありません。

これに対して、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増えていくと予想されます。このチュートリアルがそのために役立つことを祈念します。

ご注意

本チュートリアルで説明している CSS の機能は、次の 3 種類に分けることができます。

1. CSS2.1 までに規定されている機能
2. CSS3 で規定されている機能 (CSS3 マークを付加)
3. Antenna House Formatter V5.0 の独自拡張機能 (AH 独自拡張 マークを付加)

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。Antenna House Formatter V5.0 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。(以降、Antenna House Formatter を略して AH Formatter と表記します。)

目次

はじめに

Chapter 1. ページ設定	9
1.1 @page ルール	9
1.2 ページサイズ : size プロパティ <small>CSS3</small>	9
1.3 マージン : margin プロパティ	10
1.4 マージンボックス <small>CSS3</small>	11
1.5 柱とノンブル	11
1.5.1 柱の設定 : string-set プロパティと string() 関数 <small>CSS3</small>	11
1.5.2 ノンブル (ページ番号) : counter(page) <small>CSS3</small>	12
1.5.3 総ページ数 : counter(pages) <small>CSS3</small>	12
1.6 マスターページ (名前付きページ) : page プロパティ <small>CSS3</small>	12
1.7 左・右・先頭ページ	13
1.8 改ページの制御	15
1.8.1 改ページ : page-break-before/page-break-after プロパティ	15
1.8.2 改ページの禁止 : page-break-inside プロパティ	15
1.8.3 左右どちらかのページから始める	15
Chapter 2. 文字設定	16
2.1 書体指定	16
2.1.1 フォント : font プロパティ	16
2.1.2 フォントサイズ : font-size プロパティ	16
2.1.3 フォントの種類 : font-family プロパティ	16
2.1.4 フォントの太さ : font-weight プロパティ	17
2.2 イタリック体・斜体 : font-style プロパティ	17
2.3 スモールキャピタル : font-variant プロパティ	18
2.4 行の高さ : line-height プロパティ	18
2.5 縦方向の揃え位置 : vertical-align プロパティ	18
2.6 下線・上線・打ち消し線	19
2.6.1 下線・上線・打ち消し線 : -ah-text-decoration-line プロパティ	19
2.6.2 線の種類 : -ah-text-decoration-style プロパティ <small>CSS3</small>	19
2.6.3 線の色 : -ah-text-decoration-color プロパティ <small>CSS3</small>	20
2.6.4 線の幅 : -ah-text-line-width プロパティ <small>AH 独自拡張</small>	20
Chapter 3. 段落設定	21
3.1 行揃え	21
3.1.1 行揃え : text-align プロパティ	21
3.1.2 行揃え (均等配置+最終行設定) : -ah-text-align-last プロパティ <small>CSS3</small>	22
3.2 段組	22

3.2.1 段数 : -ah-column-count プロパティ CSS3	22
3.2.2 段間 : -ah-column-gap プロパティ CSS3	23
3.2.3 段間罫 : -ah-column-rule プロパティ CSS3	23
3.2.4 段間罫の種類 : -ah-column-rule-style プロパティ CSS3	23
3.2.5 段間罫の幅 : -ah-column-rule-width プロパティ CSS3	23
3.2.6 段間罫の色 : -ah-column-rule-color プロパティ CSS3	24
3.2.7 行送りの調整 : -ah-line-stacking-strategy プロパティ AH 独自拡張	24
3.3 リーダ : leader() 関数 CSS3	24
3.4 ハイフネーション	25
3.4.1 ハイフネーション : -ah-hyphens プロパティ CSS3	25
3.4.2 ハイフン分割の前の最小文字数 : -ah-hyphens-before プロパティ CSS3	25
3.4.3 ハイフン分割の後の最小文字数 : -ah-hyphens-after プロパティ CSS3	26
3.5 縦書き : -ah-writing-mode: tb-rl CSS3	26
3.6 オブジェクトの装飾	27
3.6.1 ボーダーのスタイル : border-style プロパティ	27
3.6.2 ボーダーの太さ : border-width プロパティ	28
3.6.3 ボーダーの色 : border-color プロパティ	28
3.6.4 角丸 : -ah-border-radius プロパティ AH 独自拡張	28
3.6.5 影付きボックス : box-shadow プロパティ CSS3	29
Chapter 4. 画像配置	30
4.1 インライン画像	30
4.2 表示形式 : display プロパティ	30
4.3 フロート配置 : float プロパティ	31
4.4 フロート配置 : -ah-float プロパティ CSS3	32
4.4.1 ページフロート : -ah-float: top page / -ah-float: bottom page プロパティ CSS3	32
4.4.2 コラムフロート : -ah-float: top / -ah-float: bottom プロパティ CSS3	32
4.5 回り込みの解除 : clear プロパティ	33
Chapter 5. 表組	34
5.1 表の作成方法	34
5.2 表の列幅指定の決定 : table-layout プロパティ	35
5.3 表の中央揃え	35
5.4 表のキャプションの位置 : caption-side プロパティ	35
Chapter 6. 色指定	37
6.1 文字色 : color プロパティ	37
6.2 背景色 : background-color プロパティ	37
Chapter 7. カウンタ	38
7.1 章や節に番号を付ける	38
7.2 文字挿入 : content プロパティ	39
7.3 カウンタの加算 : counter-increment プロパティ	39

7.4 カウンタのリセット : counter-reset プロパティ	40
7.5 ページカウンタ	40
7.6 文字列変数 : string-set プロパティ CSS3	40
Chapter 8. クロスリファレンス (相互参照)	41
8.1 カウンタの参照 : target-counter() 関数 CSS3	41
8.2 テキスト内容の参照 : target-text() 関数 CSS3	41
8.3 目次作成	42
Chapter 9. 脚注	43
9.1 脚注の指定 : -ah-float: footnote CSS3	43
9.2 脚注のスタイル : @footnote ルール CSS3	43
9.3 脚注番号 : ::footnote-call / ::footnote-marker 疑似要素 CSS3	43
Chapter 10. PDF ブックマーク	44
10.1 ブックマークレベル : -ah-bookmark-level プロパティ CSS3	44
10.2 ブックマークラベル : -ah-bookmark-label プロパティ CSS3	44
10.3 ブックマークの状態 : -ah-bookmark-state プロパティ CSS3	44
Chapter 11. MathML と SVG グラフィック	45
11.1 MathML	45
11.2 SVG グラフィック	46
Chapter 12. 日本語文字組版	47
12.1 行頭行末約物半角固定・段落先頭 1 字下げ	47
12.2 約物の詰め処理 : -ah-punctuation-trim プロパティ CSS3	47
12.3 行調整のための約物の詰め : -ah-text-justify-trim プロパティ AH 独自拡張	48
12.4 和欧文間の空き : -ah-text-autospace プロパティ AH 独自拡張	49
12.5 和欧文間の空き量 : -ah-text-autospace-width プロパティ AH 独自拡張	49
12.6 約物と全角文字間の詰め : -ah-punctuation-spacing プロパティ AH 独自拡張	50
12.7 突き出し・ぶら下げ : -ah-hanging-punctuation プロパティ AH 独自拡張	50
12.8 行頭約物半角固定・行末約物全角 / 半角・段落先頭 1 字下げ	51
12.9 行頭約物半角固定・行末句点全角・段落先頭 1 字下げ	51
12.10 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)	52
12.11 縦中横	53
12.12 ルビ・圏点	53
12.13 日本語フォント	55
Chapter 13. 印刷設定	56
13.1 トンボ (クロップマーク)	56
13.1.1 トンボの表示 : -ah-marks プロパティ CSS3	56
13.1.2 トンボの線の色 : -ah-printer-marks-line-color プロパティ AH 独自拡張	57
13.1.3 トンボの線の長さ : -ah-printer-marks-line-length プロパティ AH 独自拡張	57

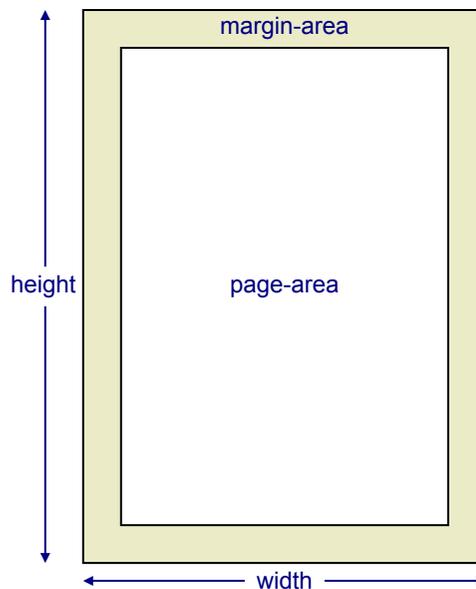
13.1.4 トンボの線の幅 : -ah-printer-marks-line-width プロパティ <small>AH 独自拡張</small>	57
13.2 断ち落とし (塗り足し)	57
13.2.1 断ち落とし (塗り足し) の幅 : -ah-page-bleed プロパティ <small>CSS3</small>	57
13.2.2 出力媒体の端から判型までの距離 : -ah-crop-offset プロパティ <small>AH 独自拡張</small>	58
Chapter 14. 参照仕様	59

Chapter 1. ページ設定

1.1 @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

1.2 ページサイズ : size プロパティ CSS3

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、B5、B4、JIS-B5、JIS-B4、letter、legal、ledger 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) */
}
```

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ヨコ (297mm×210mm) */
}
```

1.3 マージン : margin プロパティ

●初期値 : 各プロパティを参照 ●適用対象 : すべての要素 ●値の継承 : しない

@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;
}
```

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。

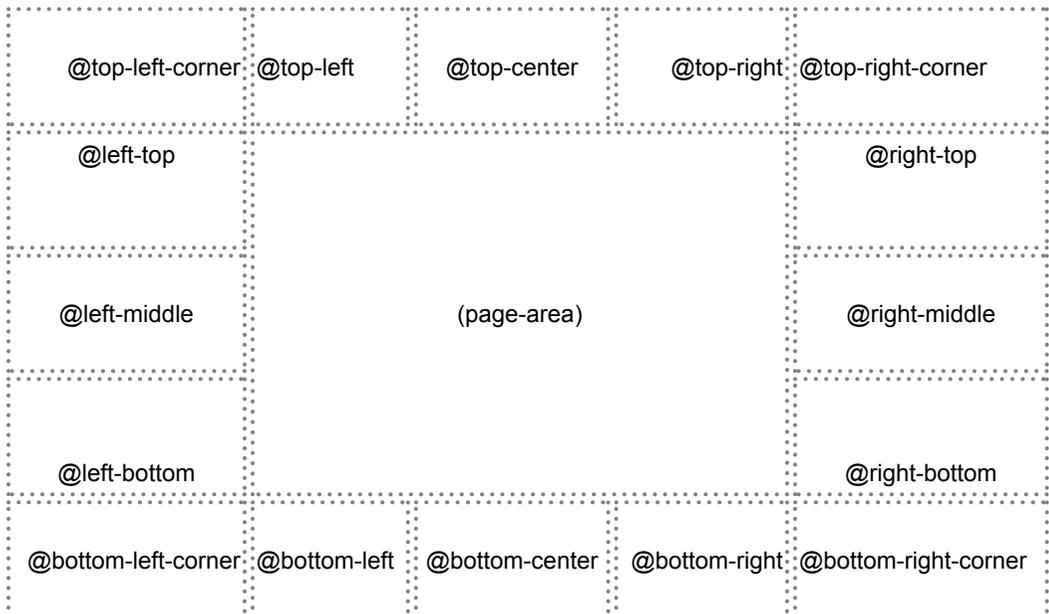
¹⁾ size プロパティは、ISO 標準の名前を使いますが、ISO の B シリーズと JIS で規定している B シリーズでの寸法は異なるため、アンテナハウスの独自拡張として、JIS-B5、JIS-B4などを追加しています。

1.4 マージンボックス CSS3

ページヘッダやページフッタはマージンボックスと呼ばれるページの周辺の領域に割り当てます。

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。

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



各マージンボックスの位置

1.5 柱とノンブル

1.5.1 柱の設定 : `string-set` プロパティと `string()` 関数 CSS3

ページヘッダに本文中の見出しからとった文字列を表示することができます。

```
@page {
  @top-left {
    content: string(Chapter);
  }
}

h1 { string-set: Chapter self; }
```

1.5.2 ノンブル（ページ番号）: counter(page) [CSS3](#)

ノンブル（ページ番号）の出力には、ページカウンタ `counter(page)` を使います²⁾。

```
@page {
  @top-right {
    content: "Page " counter(page);
  }
}
```

1.5.3 総ページ数: counter(pages) [CSS3](#)

「Page 12 of 60」のように、現在のページ番号とともに総ページ数を出力できます。

```
@page {
  @top-right {
    content: "Page " counter(page) " of " counter(pages);
  }
}
```

1.6 マスターページ（名前付きページ）: page プロパティ [CSS3](#)

名前付きの@page ルールを複数用意して、文書内で page プロパティにより切り替えて使うことができます。

マスターページとして使うことができます。

```
@page Landscape { /* 名前付きページ Landscape 定義 */
  size: A4 landscape;
}
@page Appendix { /* 名前付きページ Appendix 定義 */
  @top-center: "Appendix";
}
table.WideTable {
  page: Landscape; /* 横に大きな表を Landscape ページに */
}
div.Appendix {
  page: Appendix; /* 付録を Appendix ページに */
}
```

²⁾ counter() は CSS2.1 で規定されていますが、CSS3 ではページコンテキストにページ番号を表す専用のカウンタを導入することが議論されています。

1.7 左・右・先頭ページ

左ページと右ページで、左右の余白やページヘッダ/フッタの表示内容を変えることができます。また、先頭ページだけ特別にすることもできます。

柱とノンブルの配置位置を左ページは左側、右ページは右側、先頭ページは非表示にすることができます。

マスターページを併せて使うと、マスターページごとに左右ページ、先頭ページのスタイルを指定できます。

```
@page Chapter:left { /* 左ページ設定 */
  margin-left: 23mm;
  margin-right: 27mm;

  @top-left { /* 左ページの柱に本のタイトル */
    content: string(Title);
  }

  @bottom-left { /* ノンブル (ページ番号) */
    content: counter(page);
  }
}

@page Chapter:right { /* 右ページ設定 */
  margin-left: 27mm;
  margin-right: 23mm;

  @top-right { /* 右ページの柱に節のタイトル */
    content: string(Section);
  }

  @bottom-right { /* ノンブル (ページ番号) */
    content: counter(page);
  }
}

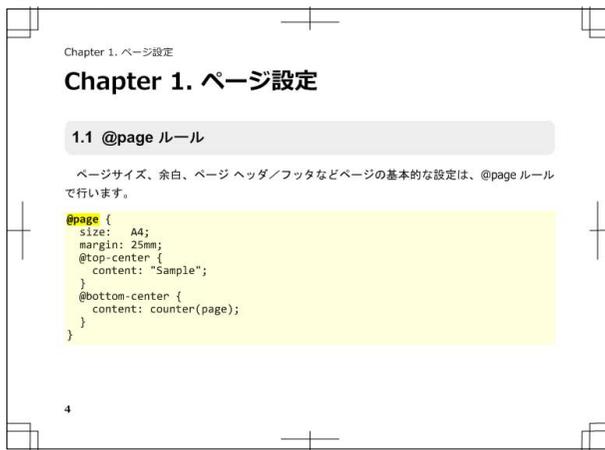
@page Chapter:first { /* Chapterの先頭ページ設定 */
  /* ページヘッダ非表示 */
  @top-right { content: normal }
  @top-left { content: normal }
}
```

string(Title) や string(Chapter) などに対応して [7.6 文字列変数 : string-set プロパティ](#) による設定が必要です。

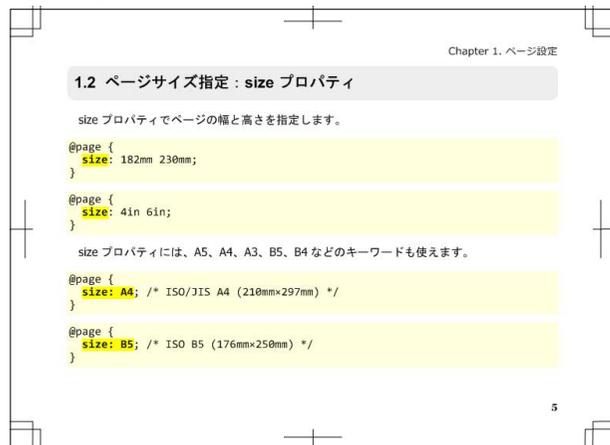
先頭ページ



左ページ



右ページ



1.8 改ページの制御

1.8.1 改ページ : page-break-before/page-break-after プロパティ

●初期値 : auto ●適用対象 : ブロック要素 ●値の継承 : しない

値を always にすると、指定した要素の前か、後ろで必ず改ページされるようになります。

```
/* 大見出し (h1) の前で強制改ページ */
h1 {
  page-break-before: always;
}
```

```
/* このブロックのあとで必ず改ページ */
div.Ending {
  page-break-after: always;
}
```

値を avoid にすると指定した要素での改ページを禁止できる。

```
/* 見出し (h1~h6) の直後で改ページしてはいけない決まり */
h1, h2, h3, h4, h5, h6 {
  page-break-after: avoid;
}
```

1.8.2 改ページの禁止 : page-break-inside プロパティ

●初期値 : auto ●適用対象 : ブロック要素 ●値の継承 : する

値を avoid にすると指定した要素での改ページを禁止できます。

```
/* このブロックの中では改ページ禁止 */
div.NoBreak {
  page-break-inside: avoid;
}
```

1.8.3 左右どちらかのページから始める

章のはじまりのページを常に右側または左側のページにするといった指定ができます。必要に応じて空白ページが挿入されます。

```
/* h2見出しが常に右ページに来るように改ページ */
h2 { page-break-before: right; }
```

Chapter 2. 文字設定

2.1 書体指定

2.1.1 フォント : font プロパティ

●初期値 : 各プロパティの値 ●適用対象 : すべての要素 ●値の継承 : する

font プロパティはフォントに関する、font-style、font-variant、font-weight、font-size、line-height、font-family の6個のプロパティを一つでまとめて指定するためのショートハンドプロパティです。

font プロパティを指定すると各プロパティの値は、初期値に一旦リセットされます。そして明示的に指定したプロパティについてその値に設定されます。font プロパティに指定する順番は次の通りです。

1. font-style、font-variant、font-weight の3つは省略可能で、この3つの中での順序は問いません。
2. font-size は省略できません。
3. line-height は省略可能です。font-size と line-height の値は / で区切ります。
4. font-family は省略できません。

```
/* 文書タイトル */
.CoverPage h1 {
  font: bold 30pt メイリオ, sans-serif;
  text-align: center;
}
```

2.1.2 フォントサイズ : font-size プロパティ

●初期値 : medium ●適用対象 : すべての要素 ●値の継承 : する
フォントのサイズを指定します。

```
/* 本文 */
body {
  font-size: 12pt;
  line-height: 1.5;
  font-family: Tahoma, "MS Gothic", sans-serif;
}
```

2.1.3 フォントの種類 : font-family プロパティ

●初期値 : ユーザーエージェントに依存 ●適用対象 : すべての要素 ●値の継承 : する

フォントの種類を指定します。フォントの種類は、フォント名かキーワードで指定します。

- sans-serif : Arial などのひげ飾りなしフォント、日本語ではゴシック系のフォントです。
- serif : Times などのひげ飾り付きのフォント、日本語では明朝系のフォントです。
- monospace : 等幅フォント、文字幅が等しいフォントです。
- fantasy : 装飾のあるフォントです。
- cursive : 筆記体フォントです。

「Times New Roman」のようにフォント名にスペースが含まれる場合は、ダブルクォーテーションで囲んで指定します。

カンマで区切って複数のフォントを指定できます。ユーザー環境で利用できるフォントが記述順に選ばれます³⁾。

```
font-family: Helvetica, "MS ゴシック", san-serif;
```

なお、指定したフォントがシステムに存在しない場合のために、generic font family (serif、sans-serif などのキーワード) を最後に指定するとよいでしょう。

2.1.4 フォントの太さ : font-weight プロパティ

●初期値 : normal ●適用対象 : すべての要素 ●値の継承 : する
フォントの太さを指定します。

- normal : 標準 (数値指定では 400)。
- bold : 太字 (数値指定では 700)。
- lighter : 1 段階 (数値指定では 100) 細くする。
- bolder : 1 段階 (数値指定では 100) 太くする。
- 数値指定 : 100、200、300、400、500、600、700、800、900 のいずれか。100 が最も細く、900 が最も太い。

2.2 イタリック体・斜体 : font-style プロパティ

●初期値 : normal ●適用対象 : すべての要素 ●値の継承 : する
フォントをイタリック体や斜体に指定します。

- normal : 標準。
- italic : イタリック体。
- oblique : 斜体。

³⁾ AH Formatter V5.0 は文字列のスク립ト種類毎に区切って、スク립ト単位で最初に指定されているフォントから割り当てていきます。例えば、日本語フォントの多くはラテンスク립トをサポートしますので、font-family: "MS ゴシック", Helvetica という並びの場合、ラテンスク립トに "MS ゴシック" が割り当てられる結果となります。

イタリック体は、斜めに傾けて筆記体調にデザインされたものですが、斜体は単純に元のフォントを傾けたものです。italic 指定したフォントにイタリック体がない場合は、そのフォントを斜めに傾けて表示します。

AH Formatter V5.0 では、oblique を指定しても italic と同じように処理します。

2.3 スモールキャピタル : font-variant プロパティ

- 初期値 : normal
 - 適用対象 : すべての要素
 - 値の継承 : する
- フォントをスモールキャピタルにするときに指定します。
日本語フォントでは、指定は有効になりません。

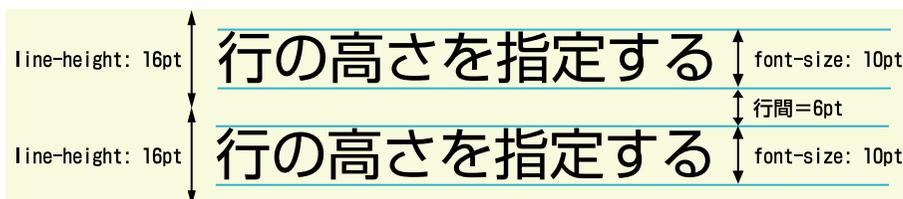
2.4 行の高さ : line-height プロパティ

- 初期値 : normal
- 適用対象 : すべての要素
- 値の継承 : する

行の高さを指定します。normal は標準の行送りで、AH Formatter V5.0 は、1.2 に初期設定されています。初期設定は変更できます。

数値だけを指定すると、その数値にフォントのサイズを掛けた値が行の高さとなります。1.5 と指定すると、150%や 1.5em と指定したのと同じになります。

行の高さと行間



2.5 縦方向の揃え位置 : vertical-align プロパティ

- 初期値 : baseline
 - 適用対象 : インラインとテーブルセル要素
 - 値の継承 : しない
- 縦方向の揃え位置や上付き文字・下付き文字を指定できます。

- baseline : 親要素のベースラインに揃える。
- sub : 下付き文字。
- super : 上付き文字。
- top : 上端に揃える。
- text-top : テキストの上端に揃える。
- middle : 中央に揃える。

- bottom : 下端に揃える。
- text-bottom : テキストの下端に揃える。
- パーセント指定 : line-height プロパティの値に対する割合をパーセントで指定する。
- 数値指定 : 数値に単位を付けて指定する。

2.6 下線・上線・打ち消し線

2.6.1 下線・上線・打ち消し線 : -ah-text-decoration-line プロパティ

- 初期値 : none
- 適用対象 : すべての要素
- 値の継承 : しない

テキストに下線・上線・打ち消し線を指定します。

同時に複数の指定をするときは、スペースで区切って、任意の順序で値を指定します。

- none : テキストに線は付かない。
- underline : テキストに下線が付く。
- overline : テキストに上線が付く。
- line-through : テキストに打ち消し線が付く。

```
/* 章見出し */
.Chapter h2 {
  -ah-text-decoration-line: underline overline;
}
```

2.6.2 線の種類 : -ah-text-decoration-style プロパティ [CSS3](#)

- 初期値 : solid
- 適用対象 : すべての要素
- 値の継承 : しない

この拡張プロパティを使うと、線の種類を指定できます。

- solid : 実線。————
- double : 二重線。————
- dotted : 点線。…………
- dashed : 破線。-----
- dot-dash : 一点鎖線。-.-.-.-
- dot-dot-dash : 二点鎖線。-.-.-.-
- wave : 波線。~~~~~

```
span.solid {
  -ah-text-decoration-style: solid;          /* 線の種類を実線にする */
}
span.wave {
  -ah-text-decoration-style: wave;          /* 線の種類を波線にする */
}
```

```
solid: <span class="solid">実線</span>  
wave: <span class="wave">波線</span>
```

2.6.3 線の色: `-ah-text-decoration-color` プロパティ CSS3

●初期値: 文字の色 ●適用対象: すべての要素 ●値の継承: しない
この拡張プロパティを使うと、線の色を指定できます。

```
span {  
  -ah-text-decoration-color: cmyk(0,1,1,0); /* 線の色を赤にする */  
}
```

2.6.4 線の幅: `-ah-text-line-width` プロパティ AH 独自拡張

●初期値: auto ●適用対象: すべての要素 ●値の継承: しない
この拡張プロパティを使うと、線の幅を指定できます。

Chapter 3. 段落設定

3.1 行揃え

3.1.1 行揃え : text-align プロパティ

●初期値 : 文字の表示方向に依存 ●適用対象 : ブロック要素 ●値の継承 : する
ブロックレベル要素内のテキストを左揃え、右揃え、中央揃え、均等配置にします。

left : 左揃え

Antenna House is very please to announce the release of the Preview version of Formatter V5.0, now with support for page layouts specified using either CSS or XSL-FO for formatting documents for PDF and print.

right : 右揃え

Antenna House is very please to announce the release of the Preview version of Formatter V5.0, now with support for page layouts specified using either CSS or XSL-FO for formatting documents for PDF and print.

center : 中央揃え

Antenna House is very please to announce the release of the Preview version of Formatter V5.0, now with support for page layouts specified using either CSS or XSL-FO for formatting documents for PDF and print.

justify : 均等配置

Antenna House is very please to announce the release of the Preview version of Formatter V5.0, now with support for page layouts specified using either CSS or XSL-FO for formatting documents for PDF and print.

text-align は、CSS3 で start、end の値が拡張されています。-ah-text-align: start;、-ah-text-align: end; と指定すると縦書きで上揃え、下揃えを指定できます。CSS3

end : 下揃え	親譲りの無鉄砲で小供の時から損ばかりしている。小学校に居る時分学校の二階から飛び降りて一週間ほど腰を抜かした事がある。なぜそんな無闇をしたと聞く人があるかも知れぬ。
start : 上揃え	親譲りの無鉄砲で小供の時から損ばかりしている。小学校に居る時分学校の二階から飛び降りて一週間ほど腰を抜かした事がある。なぜそんな無闇をしたと聞く人があるかも知れぬ。

3.1.2 行揃え（均等配置＋最終行設定）：-ah-text-align-last プロパティ CSS3

●初期値：start ●適用対象：すべての要素 ●値の継承：する

すべての要素内のテキストを均等配置し、さらに最終行を上揃え（縦書き）、下揃え（縦書き）、左揃え、右揃え、中央揃え、両端揃えにします。

<p>段深い理由でもない。 人があるかも知れぬ。別 なぜそんな無闇をしたと聞 ど腰を抜かした事がある。 階から飛び降りて一週間ほ 小学校に居る時分学校の二 時から損ばかりしている。 親譲りの無鉄砲で小供の eol:均等配置（下揃え）</p>	<p>段深い理由でもない。 人があるかも知れぬ。別 なぜそんな無闇をしたと聞 ど腰を抜かした事がある。 階から飛び降りて一週間ほ 小学校に居る時分学校の二 時から損ばかりしている。 親譲りの無鉄砲で小供の start:均等配置（上揃え）</p>
----------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------

left：均等配置（左揃え）

Antenna House is very please to announce the release of the Preview version of Formatter V5.0, now with support for page layouts specified using either CSS or XSL-FO for formatting documents for PDF and print.

right：均等配置（右揃え）

Antenna House is very please to announce the release of the Preview version of Formatter V5.0, now with support for page layouts specified using either CSS or XSL-FO for formatting documents for PDF and print.

center：均等配置（中央揃え）

Antenna House is very please to announce the release of the Preview version of Formatter V5.0, now with support for page layouts specified using either CSS or XSL-FO for formatting documents for PDF and print.

justify：均等配置（両端揃え）

Antenna House is very please to announce the release of the Preview version of Formatter V5.0, now with support for page layouts specified using either CSS or XSL-FO for formatting documents for PDF and print.

3.2 段組

3.2.1 段数：-ah-column-count プロパティ CSS3

●初期値：auto ●適用対象：ブロック要素 ●値の継承：しない

ブロックの段数を指定します。

ここは段数 (column-count) を 2 と指定した段組のブロックです。段間 (column-gap)、段間罫 (column-rule) も指定しています。

```
div.MultiCol {
  -ah-column-count: 2;
  -ah-column-gap: 5mm;
  -ah-column-rule: dotted green 1mm;
}
```

段組の指定には、段数 (column-count) を指定するかわりに、段幅 (column-width) を指定

して、全体の幅と段幅により自動的に段数が決まるようにする方法もあります。

段間罫を指定する column-rule はショートハンドプロパティであり、次のように個別に指定することもできます。

```
-ah-column-rule-style: dotted;
-ah-column-rule-color: green;
-ah-column-rule-width: 1mm;
```

3.2.2 段間 : -ah-column-gap プロパティ [CSS3](#)

●初期値 : normal ●適用対象 : マルチコラム要素 ●値の継承 : しない
段間の幅を指定します。

3.2.3 段間罫 : -ah-column-rule プロパティ [CSS3](#)

●初期値 : 各プロパティを参照 ●適用対象 : マルチコラム要素 ●値の継承 : しない
段間罫の幅、線種、色をまとめて指定します。

3.2.4 段間罫の種類 : -ah-column-rule-style プロパティ [CSS3](#)

●初期値 : none ●適用対象 : マルチコラム要素 ●値の継承 : しない
段間罫の線種を指定します。
次の線種が使えます。

- solid : 実線。———
- double : 二重線。—————
- dotted : 点線。……………
- dashed : 破線。-----
- dot-dash : 一点鎖線。-.-.-.-.-
- dot-dot-dash : 二点鎖線。-.-.-.-.-
- wave : 波線。~~~~~

3.2.5 段間罫の幅 : -ah-column-rule-width プロパティ [CSS3](#)

●初期値 : medium ●適用対象 : マルチコラム要素 ●値の継承 : しない
段間罫の幅を指定します。

3.4 ハイフネーション

3.4.1 ハイフネーション：-ah-hyphens プロパティ CSS3

●初期値：manual ●適用対象：すべての要素 ●値の継承：する

AH Formatter V5.0 は、40 言語以上のハイフネーション処理を行うことができます。このとき、辞書を用意する必要はありません。辞書は内蔵されています。内蔵辞書と異なるハイフネーションをさせたい単語を例外辞書に登録しておくこともできます。

- none：ハイフネーション処理とソフトハイフン⁴⁾の両方を無効にします。
- manual：ハイフネーション処理を無効に、ソフトハイフンを有効にします。
- auto：ハイフネーション処理とソフトハイフンを有効にします。

ハイフネーション処理を有効にするには、hyphens プロパティを auto にします。なお、ハイフネーション処理は言語の指定によって言語ごとのハイフネーションを行いますので、当該要素に xml:lang で言語コードを指定してください。

```
.Hyphenated {
  -ah-hyphens: auto;
}
<div class="Hyphenated" xml:lang="en">
  <p>Rainbow PDF Software Products are ...
```

<p>Rainbow PDF Software Products are developed by Antenna House. Antenna House, Inc. is a premier software company founded in 1984 in Tokyo, Japan. Our mission is to create and sell products to make data useful. For over 20 years we have been reliably retrieving information and delivering it to our cus-</p>	<ul style="list-style-type: none"> • tomers in formats and languages our customers can use. • Millions of copies of our software have saved time for individual desktop users, small businesses, and have cemented our leadership role in high end documentation. • •
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

3.4.2 ハイフン分割の前の最小文字数：-ah-hyphenate-before プロパティ CSS3

●初期値：auto ●適用対象：すべての要素 ●値の継承：する

ハイフン分割の前の最小文字数（標準は2）を指定するには、-ah-hyphenate-before プロパティを指定します。

たとえば、hyphen という 6 文字の単語は、hy-phen とハイフネーションされる可能性があります。-ah-hyphenate-before: 3; と指定すると、ハイフン分割の前の文字数が 3 文字以上の単語でないとハイフネーションされないため、hyphen がハイフネーションされることはありません。

4) ソフトハイフンは、U+00AD の文字で、明示的に単語中に挿入してハイフネーションを促すことができます。

3.4.3 ハイフン分割の後の最小文字数 : `-ah-hyphenate-after` プロパティ CSS3

●初期値 : auto ●適用対象 : すべての要素 ●値の継承 : する

ハイフン分割の後の最小文字数（標準は2）を指定するには、`-ah-hyphenate-after` プロパティを指定します。

たとえば、`hyphen` という6文字の単語は、`hy-phen` とハイフネーションされる可能性があります。`-ah-hyphenate-after: 5;` と指定すると、ハイフン分割の後の文字数が5文字以上の単語でない限りハイフネーションされないので、`hyphen` がハイフネーションされることはありません。

3.5 縦書き : `-ah-writing-mode: tb-rl` CSS3

●初期値 : `lr-tb` ●適用対象 : すべての要素 ●値の継承 : する

文書全体を縦書きにするには、ルート要素に `-ah-writing-mode: tb-rl` を指定します。

`tb-rl` は、文字の進行方向が上から下（`top-to-bottom`）、行とブロックの進行方向が右から左（`right-to-left`）を意味しています。横書きの指定は、`-ah-writing-mode: lr-tb`（`left-to-right`、`top-to-bottom`）となります。また、アラビア語やヘブライ語など右から左に書く言語の場合は、`-ah-writing-mode: rl-tb`（`right-to-left`、`top-to-bottom`）となります。

- `lr-tb` : 文字の進行方向が左から右、行とブロックの進行方向が上から下となります。通常の横書きです。
- `rl-tb` : 文字の進行方向が右から左、行とブロックの進行方向が上から下となります。アラビア語などで使います。
- `tb-rl` : 文字の進行方向が上から下、行とブロックの進行方向が右から左となります。日本語などの縦書きで使います。

```
body {  
  -ah-writing-mode: tb-rl; /* 縦書き */  
}
```

ブロックを縦書きにするには、ブロック要素に `-ah-writing-mode` プロパティを指定します。

```
div.VerticalTextBlock {  
  -ah-writing-mode: tb-rl; /* 縦書き */  
  height: 16em; /* 1行の文字数 */  
  padding: 3pt; border: ridge green;  
}
```

日本語は伝統的に縦書きで組まれます。書籍や雑誌など出版物の多くは、今も縦書きが主流です。もちろん、AH Formatter V5.0 は縦書きにも対応しています。このように、部分的にブロックを縦書きにすることも、文書全体を縦書きにすることもできます。

縦書きの指定は **ah-writing-mode: tb-rl** です。tb は、文字の進行方向が上から下 (top-to-bottom)、行とブロックの進行方向が右から左 (right-to-left) を意味しています。

横書きの指定は **ah-writing-mode: lr-tb** (left-to-right、top-to-bottom) です。アラビア語やヘブライ語など右から左に書く言語の横書きの場合は **ah-writing-mode: rl-tb** (right-to-left、top-to-bottom) です。縦書きの中に「08年12月8日」のように部分的に数字などを横書きにすることを「縦中横」といいます。

3.6 オブジェクトの装飾

3.6.1 ボーダーのスタイル : border-style プロパティ

- 初期値 : 各プロパティの値
 - 適用対象 : すべての要素
 - 値の継承 : しない
- ボーダーのスタイルを指定するには、border-style プロパティを使います。
- 値を1つ指定した場合 : 指定した値が [上下左右] のボーダースタイルになります。
- 値を2つ指定した場合 : 記述した順に [上下] [左右] のボーダースタイルになります。
- 値を3つ指定した場合 : 記述した順に [上] [左右] [下] のボーダースタイルになります。
- 値を4つ指定した場合 : 記述した順に [上] [右] [下] [左] のボーダースタイルになります。

次のスタイルが使えます。

none : ボーダーがありません。(太さ = 0)

hidden : ボーダー非表示です。

solid : 実線。

double : 二重線。

dotted : 点線。

dashed : 破線。

dot-dash : 一点鎖線。

dot-dot-dash : 二点鎖線。

wave : 波線。

groove : 立体的にへこんだように見える線。

ridge : 立体的に隆起したように見える線。

inset : ボーダーの範囲が立体的にへこんだように見える線。

outset : ボーダーの範囲が立体的に隆起したように見える線。

3.6.2 ボーダーの太さ : border-width プロパティ

●初期値 : 各プロパティの値 ●適用対象 : すべての要素 ●値の継承 : しない
 ボーダーの太さを指定するには、border-width プロパティを使います。

3.6.3 ボーダーの色 : border-color プロパティ

●初期値 : 各プロパティの値 ●適用対象 : すべての要素 ●値の継承 : しない
 ボーダーの色を指定するには、border-color プロパティを使います。

3.6.4 角丸 : -ah-border-radius プロパティ AH 独自拡張

●初期値 : 0 ●適用対象 : すべての要素 ●値の継承 : しない
 ボーダーの角を丸くするには、-ah-border-radius プロパティで半径を指定します。

```
<p style="-ah-border-radius: 18pt; /* 角の丸め半径 */
border: solid green;
padding: 6pt">Border-radius...
```

Border-radius (丸め罫) を指定しています。

四隅の丸め半径を個別に指定することもできます。

```
<p style="-ah-border-top-left-radius: 2mm; /* 左上 */
-ah-border-top-right-radius: 5mm; /* 右上 */
-ah-border-bottom-right-radius: 2cm 1cm; /* 右下 (水平方向、垂直方向) */
-ah-border-bottom-left-radius: 4cm 2cm; /* 左下 (水平方向、垂直方向) */
border: thin solid;
background-color: lime;
padding: 5mm">...
```

4つの角の丸め半径を個別に指定したり、角の丸めを楕円にすることもできます。

3.6.5 影付きボックス : -ah-box-shadow プロパティ CSS3

●初期値 : none ●適用対象 : すべての要素 ●値の継承 : しない

ボックスに影を付けるには、-ah-box-shadow プロパティに水平方向と垂直方向の影の長さや影の色を指定します。

```
<p style="-ah-box-shadow: 8pt 6pt silver;  
border: solid 1pt black; padding: 6pt">...
```

Box-shadow (影付きボックス) を指定しています。

```
<p style="-ah-box-shadow: -8pt -6pt orange, 8pt 6pt blue;  
-ah-border-radius: 10pt; padding: 6pt">...
```

複数の影を指定することもできます。-ah-border-radius と組み合わせ可能です。

Chapter 4. 画像配置

4.1 インライン画像

文章中にインライン画像を配置するには、インライン要素の要素や<object>要素を使います。

 アンテナハウスは、これまで XML 専用のページ組版言語である XSL-FO 仕様を実装した XSL Formatter で世界的な評価を得てきました。XSL Formatter は、発売以来、日本国内は勿論、海外でも世界 43 カ国（2008 年 6 月末：国内含まず）での導入実績があります。

 アンテナハウスは、XML/HTML を CSS3 でレイアウト指定して、PDF 等のページ媒体に出力することについて理解を広めるため、 AH Formatter V5.0 の正式発売に先行してプレビュー版を幅広く配布いたします。

```
<p>アンテナハウスは、これまでXML専用のページ組版言語であるXSL-FO仕様を実装したXSL Formatterで世界的な評価を得てきました。…
```

4.2 表示形式 : display プロパティ

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。

インライン要素の要素に display: block; と指定すると、ブロック要素として表示されます。なお、display プロパティは要素の表示形式を変更するだけで、要素のもつ役割は変更しません。

 アンテナハウスは、これまで XML 専用のページ組版言語である XSL-FO 仕様を実装した XSL Formatter で世界的な評価を得てきました。XSL Formatter は、発売以来、日本国内は勿論、海外でも世界 43 カ国（2008 年 6 月末：国内含まず）での導入実績があります。

 アンテナハウスは、XML/HTML を CSS3 でレイアウト指定して、PDF 等のページ媒体に出力することについて理解を広めるため、

 AH Formatter V5.0 の正式発売に先行してプレビュー版を幅広く配布いたします。

```
<p>は、これまでXML専用のページ組版言語であるXSL-FO仕様を実装したXSL Formatterで世界的な評価を得てきました。…
```

4.3 フロート配置 : float プロパティ

●初期値 : none ●適用対象 : すべての要素 ●値の継承 : しない

画像をフロートさせて左寄せまたは右寄せにするには、float プロパティを使います。

float プロパティを適用した画像の次の要素は、反対側に回り込みます。

画像の左寄せ float: left



アンテナハウスは、これまで XML 専用のページ組版言語である XSL-FO 仕様を実装した XSL Formatter で世界的な評価を得てきました。XSL Formatter は、発売以来、日本国内は勿論、海外でも世界 43 カ国（2008 年 6 月末：国内含まず）での導入実績があります。



アンテナハウスは、XML/HTML を CSS3 でレイアウト指定して、PDF 等のページ媒体に出力することについて理解を広めるため、AH Formatter V5.0 の正式発売に先行してプレビュー版を幅広く配布いたします。

```
<p style="text-indent: 0">
アンテナハウスは、これまでXML専用のページ組版言語であるXSL-FO仕様を
...
```

画像の右寄せ float: right

アンテナハウスは、これまで XML 専用のページ組版言語である XSL-FO 仕様を実装した XSL Formatter で世界的な評価を得てきました。XSL Formatter は、発売以来、日本国内は勿論、海外でも世界 43 カ国（2008 年 6 月末：国内含まず）での導入実績があります。

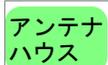
アンテナハウスは、XML/HTML を CSS3 でレイアウト指定して、PDF 等のページ媒体に出力することについて理解を広めるため、AH Formatter V5.0 の正式発売に先行してプレビュー版を幅広く配布いたします。



```
<p style="text-indent: 0">
アンテナハウスは、これまでXML専用のページ組版言語であるXSL-FO仕様を
...
```

画像だけでなく、文字をフロートさせることもできます。

文字の左寄せ float: left



アンテナハウスは、これまで XML 専用のページ組版言語である XSL-FO 仕様を実装した XSL Formatter で世界的な評価を得てきました。XSL Formatter は、発売以来、日本国内は勿論、海外でも世界 43 カ国（2008 年 6 月末：国内含まず）での導入実績があります。

ここは、ページの先頭です。**-ah-float: top page** を指定します。

2番目のページトップフロートです。

文字の右寄せ float: right

アンテナハウスは、これまで XML 専用のページ組版言語である XSL-FO 仕様を実装した XSL Formatter で世界的な評価を得てきました。XSL Formatter は、発売以来、日本国内は勿論、海外でも世界 43 カ国（2008 年 6 月末：国内含まず）での導入実績があります。

アンテナ
ハウス

4.4 フロート配置：-ah-float プロパティ CSS3

●初期値：none ●適用対象：すべての要素 ●値の継承：しない

4.4.1 ページフロート：-ah-float: top page / -ah-float: bottom page CSS3

従来の HTML と CSS のレイアウトで、画像などを左か右に寄せて配置して本文テキストを回り込ませる左右の float はよく使われていますが、ページフロートはそれをページの縦方向に拡張したものです。

-ah-float: top page; でページの先頭にブロックを配置できます。-ah-float: bottom page; でページの最後にブロックを配置できます。

CSS は、すでに Web の世界では確実に普及しており、CSS2 (Cascading Style Sheets Level2) の標準を実装するブラウザも増えています。しかし、ブラウザのページレイアウト、文字配置機能はレベルが低く高品質なページ媒体には向きません。

```
<div style="-ah-float: top page; border: ridge maroon; padding: 3pt;">
<p>ここは、ページの先頭です。<b>-ah-float: top page</b>を指定します。</p>
</div>
<div style="-ah-float: bottom page; border: ridge green; padding: 3pt;">
<p>ここは、ページの最後です。<b>-ah-float: bottom page</b>を指定します。</p>
</div>
<p>CSSは、すでにWebの世界では確実に普及しており、CSS2 (Cascading Style Sheets Level2)
の標準を実装するブラウザも増えています。しかし、ブラウザのページレイアウト、文字配置機能は…
```

4.4.2 コラムフロート：-ah-float: top / -ah-float: bottom CSS3

-ah-float: top; でコラムの先頭にブロックを配置できます。-ah-float: bottom; でコラムの最後にブロックを配置できます。

ここは、ページの最後です。**-ah-float: bottom page** を指定します。

2番目のページボトムフロートです。**-ah-float: bottom page** を指定します。

ここは、コラムの先頭です。

-ah-float: top を指定します。

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec fringilla. Donec luctus ante at dolor scelerisque tempor. Phasellus convallis, nisl sed sollicitudin lacinia, in pulvinar nibh eros non sem.

Donec sit amet velit. Phasellus consectetur. Pellentesque ut magna. Quisque enim turpis, fringilla id, malesuada ut, molestie sed, tellus.

● Cras commodo, pede id dapibus lacinia, nulla ante gravida libero, ac cursus neque diam at massa. Suspendisse et est id eros gravida commodo. Aenean imperdiet tristique urna.

ここは、コラムの最後です。

-ah-float: bottom を指定します。

```
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. ...</p>
<div style="-ah-float: top; border: ridge orange; padding: 3pt;">
<p>ここは、コラムの先頭です。<br /><b>float: top</b>を指定します。</p>
</div>
<p>Donec sit amet velit. Phasellus consectetur. Pellentesque ut magna. ...</p>
<div style="-ah-float: bottom; border: ridge lime; padding: 3pt">
<p>ここは、コラムの最後です。<br /><b>float: bottom</b>を指定します。</p>
</div>
<p>Cras commodo, pede id dapibus lacinia, nulla ante gravida libero, ...</p>
```

4.5 回り込みの解除 : clear プロパティ

●初期値 : none ●適用対象 : ブロック要素 ●値の継承 : しない

右寄せまたは左寄せを指定したブロックへの回り込みを解除するには、clear プロパティを使います。

- none : 回り込みを解除しません。
- left : 左寄せの要素への回り込みを解除します。
- right : 右寄せの要素への回り込みを解除します。
- both : 左寄せおよび右寄せ要素への回り込みを解除します。

Chapter 5. 表組

5.1 表の作成方法

CSSの表モデルは、HTMLの表モデルとほぼ対応しています。

CSS プロパティ	HTML の要素
table	TABLE
inline-table	TABLE
table-row	TR
table-row-group	TBODY
table-header-group	THEAD
table-fooler-group	TFOOT
table-column	COL
table-column-group	COLGROUP
table-cell	TD, TH
table-caption	CAPTION

display プロパティを使って XML 要素を CSS の表にマップすることで、XML を表として表示することができます。

```
<DATA>
  <STACK>
    <ROW><CELL>一行一列目</CELL><CELL>一行二列目</CELL></ROW>
    <ROW><CELL>二行一列目</CELL></ROW>
    <ROW><CELL>三行目一列目</CELL></ROW>
  </STACK>
</DATA>
```

```
DATA {
  display: block;
  margin: 20%;
}
STACK {
  display: block;
  position: fixed;
```

```

top: 100mm;
left: 50mm;
}
ROW {
  display: table-row
}
CELL {
  display: table-cell;
  padding: 10pt;
  font-weight: bolder;
  border: solid thin;
}

```

5.2 表の列幅の決定方法 : table-layout プロパティ

●初期値 : auto ●適用対象 : テーブル要素 ●値の継承 : しない
 表の表示方法を指定するには、table-layout プロパティを使います。

- auto : 表の列幅を自動レイアウトにします。
- fixed : 表の列幅を固定レイアウトにします。

fixed を指定し、一部の列に幅を指定した場合、幅が指定されていない列には残りの幅が均等に割る当てられます。

```

table {
  table-layout: fixed;
  ...
}

```

5.3 表の中央揃え

表を中央揃えにするには、margin-left: auto; margin-right: auto とします。

```

table {
  margin-left: auto;
  margin-right: auto;
  ...
}

```

5.4 表のキャプションの位置 : caption-side プロパティ

●初期値 : top ●適用対象 : キャプション要素 ●値の継承 : する
 表のキャプションの位置を指定するには、caption-side プロパティを使います。

- top : キャプションを表の上に表示します。

- bottom : キャプションを表の下に表示します。

キャプションを表の上に配置した例

製品紹介

オプション名	製品内容
SVG オプション	SVG を出力するためのオプションです。
バーコードオプション	バーコードフォントとバーコードを印刷するためのいくつかのスタイルシートを提供します。

```
table {
  ...
  caption-side: top;
  ...
}
```

キャプションを表の下に配置した例

製品紹介

オプション名	製品内容
SVG オプション	SVG を出力するためのオプションです。
バーコードオプション	バーコードフォントとバーコードを印刷するためのいくつかのスタイルシートを提供します。

製品紹介

```
table {
  ...
  caption-side: bottom;
  ...
}
```

Chapter 6. 色指定

6.1 文字色 : color プロパティ

●初期値 : ユーザーエージェントに依存 ●適用対象 : すべての要素 ●値の継承 : する
文字色やボーダー色の前景色を指定するには、color プロパティを使います。
通常の色指定は、RGB モードで行いますが、以下の 5 つの方法で指定することができます。

- #RGB : R・G・B それぞれ 16 進数 1 桁で指定します。(#5F0 = #55FF00)
- #RRGGBB : R・G・B それぞれ 16 進数 2 桁で指定します。
- rgb(255,0,0) : 左から R・G・B それぞれ 0~255 の整数値で指定します。
- rgb(100%,0%,0%) : 左から R・G・B それぞれ 0%~100%で指定します。
- black など : 定義済みの名前 で指定します。

```
em { color: #F00; } /* #RGB */
em { color: #FF0000; } /* #RRGGBB */
em { color: rgb(255, 0, 0); } /* 0~255の整数値 */
em { color: rgb(100%, 0%, 0%); } /* 0%~100% */
em { color: red; } /* 定義済みの名前 */
```

定義済みの色名

aqua	#00FFFF	水色	black	#000000	黒	blue	#0000FF	青
fuchsia	#FF00FF	ピンク	gray	#808080	灰色	green	#008000	緑
lime	#00FF00	ライム	maroon	#800000	栗色	navy	#000080	紺
olive	#808000	オリーブ	purple	#800080	紫	red	#FF0000	赤色
silver	#C0C0C0	薄い灰色	teal	#008080	青緑	orange	#FFA500	オレンジ
white	#FFFFFF	白	yellow	#FFFF00	黄色			

6.2 背景色 : background-color プロパティ

●初期値 : transparent ●適用対象 : すべての要素 ●値の継承 : しない
背景色を指定するには、background-color プロパティを使います。
適用される範囲は、要素の内容領域、パディング、ボーダーの背景です。マージンは、常に透明で背景は適用されません。
初期値の transparent (透明) は、親要素などの背景が透けて見える状態です。

Chapter 7. カウンタ

7.1 章や節に番号を付ける

章番号や節番号のように要素に連番を付けるには、counter-increment プロパティ、counter-reset プロパティ、content プロパティを使います。

counter-increment プロパティと counter-reset プロパティでカウンタ名とその値を設定します。content プロパティにカウンタ名を指定すると、要素の直前または直後にカウンタの値が挿入されます。counter-increment プロパティを適用した要素が現れるたびにカウンタの値が増え、counter-reset プロパティを適用した要素が現れるたびにカウンタの値がリセットされます。

```
body {
  counter-reset: ChapterNo;          /* 章番号 (ChapterNo) をリセットします */
}
h1:before {
  counter-increment: ChapterNo;      /* 章番号に1を足します */
  /* 章見出し (h1) の前に「第〇章」を挿入します */
  content: "第" counter(ChapterNo) "章 ";
}
h1 {
  /* Chapterにh1:beforeとh1自身の内容をセットします */
  string-set: Chapter before self;
  counter-reset: SectionNo;         /* 節番号をリセットします */
}
h2:before {
  counter-increment: SectionNo;      /* 節番号に1を足します */
  /* 節見出し (h2) の前に「〇.〇」を挿入します */
  content: counter(ChapterNo) ". " counter(SectionNo) " ";
}
h2 {
  /* Sectionにh2:beforeとh2自身の内容をセットします */
  string-set: Section before self;
}
@page :left {
  @top-left {
    content: string(Chapter);        /* 左ページの柱に章のタイトルを挿入します */
  }
}
@page :right {
  @top-right {
    content: string(Section);       /* 右ページの柱に節のタイトルを挿入します */
  }
}
```

7.2 文字挿入 : content プロパティ

●初期値 : normal ●適用対象 : CSS2.1 の場合 :before 疑似要素と :after 疑似要素 (CSS3 ではあらゆる要素に適用) ●値の継承 : しない

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。
- 文字列 : 挿入する文字列を「"」または「'」で囲んで指定します。
- url() : 画像ファイルの URL を指定します。content: url(image.png); などと指定することで、要素の内容を画像とすることができます。
- attr() : 指定した属性の値が content プロパティの値になります
- counter() : カウンタの値を追加します。
- open-quote : quotes プロパティで指定した開始引用符を挿入します。
- close-quote : quotes プロパティで指定した終了引用符を挿入します。
- no-open-quote : 引用符は表示されませんが、quotes プロパティで指定されている引用符の入れ子の深さが一階層深くなります。
- no-close-quote : 引用符は表示されませんが、quotes プロパティで指定されている引用符の入れ子の深さが一階層浅くなります。

```
.Chapter h2:before {
  content: "Chapter " counter(ChapterNo) ". ";
}
```

7.3 カウンタの加算 : counter-increment プロパティ

●初期値 : none ●適用対象 : すべての要素 ●値の継承 : しない

指定したカウンタの値を増やすには、counter-increment プロパティを使います。

- none : カウントしません。
- カウンタ名 : 指定したカウンタの値を 1 増やします。
- カウンタ名+(半角スペース)+整数 : カウンタの値を指定した数分増やします。

```
.Chapter h2 {
  ...
  counter-increment: ChapterNo;
  ...
}
```

7.4 カウンタのリセット : counter-reset プロパティ

●初期値 : none ●適用対象 : すべての要素 ●値の継承 : しない
指定したカウンタの値をリセットするには、counter-reset プロパティを使います。

- none : カウントをリセットしません。
- カウンタ名 : 指定したカウンタの値を 0 にします。
- カウンタ名+(半角スペース)+整数 : カウンタの値を指定した数にします。

```
.Chapter h2 {
  ...
  counter-reset: SectionNo;
  ...
}
```

7.5 ページカウンタ CSS3

現在のページと総ページ数を求めるには、counter() 関数を使います。

現在のページ : counter(page)

総ページ数 : counter(pages)

このページのページ数 = 40

このドキュメントの総ページ数 = 60

```
<p>このページのページ数 = <span style="content: counter(page)"></span></p>
<p>このドキュメントの総ページ数 = <span style="content: counter(pages)"></span></p>
```

7.6 文字列変数 : string-set プロパティ CSS3

●初期値 : none ●適用対象 : すべての要素 ●値の継承 : しない

文字列を名前付きの変数とするには、string-set プロパティを使います。

string-set: に続く文字列が変数名となり、それ以降の文字列が変数に格納する内容になります。

string-set で定義した文字列は、柱などで content: string(Chapter) のように参照されます。

```
h1 {
  /* Chapter に h1:before と h1 自身の内容をセットします */
  string-set: Chapter before self;
}
```

Chapter 8. クロスリファレンス（相互参照）

8.1 カウンタの参照 : target-counter() 関数 CSS3

「[Chapter 7. カウンタ（page 38）](#)を参照」のように、参照先の章番号やページ番号を自動的に付加するには、target-counter() 関数を使います。

target-counter() 関数を使った相互参照の方法は、次のようになります。

1. CSS ファイルの相互参照用クラスに参照元の前または後に挿入する文字列を指定します。
2. target-counter() 関数を使うと、target-counter() 関数内に指定した変数のカウンタを contents プロパティに挿入できます。単なる文字列を挿入する場合は、その文字列を「"」で囲みます。また、target-counter() 関数との間は、半角スペースで区切ります。
3. 文章中の参照元には、アンカーとなる<a>タグを挿入し、そのクラスに相互参照クラス名を指定します。さらに href 属性に参照先の ID 値を指定します。

```
.ChapterRef::before {
  content: "Chapter " target-counter(attr(href, url), ChapterNo) ". ";
}
.PageRef::after {
  content: "(page " target-counter(attr(href, url), page) ")";
}
...
<a class="ChapterRef PageRef" href="#Counters">カウンタ</a>を参照
```

8.2 テキスト内容の参照 : target-text() 関数 CSS3

参照先のテキストを参照元に表示するには、target-text() 関数を使います。

例 : 「[Chapter 8. クロスリファレンス（相互参照）](#)」

target-text() 関数を使った相互参照の方法は、次のようになります。

1. CSS ファイルの相互参照用クラスに参照元に挿入する文字列を指定します。
2. target-text() 関数を使うと、target-text() 関数内に指定した変数のカウンタを contents プロパティに挿入できます。
3. 文章中の参照元には、アンカーとなる<a>タグを挿入し、そのクラスに相互参照クラス名を指定します。さらに href 属性に参照先の ID 値を指定します。
4. アンカータグ内の文字列が、参照先の文字列と差し替わります。

```
.TitleRef {
  content: target-text(attr(href, url), before)
         target-text(attr(href, url), content);
}
```

```
...
例: 「<a class="TitleRef" href="#CrossRef">この章</a>
<p>「この章」という文字が、参照先のタイトルに差し替わります。</p>
```

8.3 目次作成

目次は、`target-counter()` 関数による章番号とページ番号の参照を使って作ることができます。

目次の例

Chapter 1. ページ設定	9
Chapter 2. 文字設定	16
Chapter 3. 段落設定	21
Chapter 4. 画像配置	30
Chapter 5. 表組	34
Chapter 6. 色指定	37
Chapter 7. カウンタ	38
Chapter 8. クロスリファレンス（相互参照）	41
Chapter 9. 脚注	43
Chapter 10. PDF ブックマーク	44

```
.TOC a::before { /* 目次項目に章番号を付ける */
  content: "Chapter " target-counter(attr(href, url), ChapterNo) ". ";
}
.TOC a::after { /* 目次項目にページ番号を付ける */
  content: leader(dotted) " " target-counter(attr(href, url), page);
}
...
<div class="TOC">
  <ul>
    <li><a href="#PageSetup">ページ設定</a></li>
    <li><a href="#PageHeaderFooter">柱とノンブル</a></li>
    <li><a href="#Counters">カウンタ</a></li>
  </ul>
</div>
```

Chapter 9. 脚注

9.1 脚注の指定 : -ah-float: footnote CSS3

-ah-float プロパティの値に footnote を指定するとその要素の内容が脚注になります⁵⁾。

```
<style>
.Footnote { -ah-float: footnote; }
</style>
```

```
<p>-ah-floatプロパティの値にfootnoteを指定するとその要素の内容が脚注
<span class="Footnote">脚注とは、ページの下の方に置かれる注です。</span>
になります。...</p>
```

9.2 脚注のスタイル : @footnote ルール CSS3

脚注領域の上に罫線を引くなど脚注領域の設定をするために@page ルールの中で@footnote ルールを使います。

```
@page {
  @footnote {
    -ah-float: bottom page; /* 脚注領域は、ページの下部へのフロートとして配置される */
    border-top: thin solid black; /* 脚注領域の上の罫線を指定 */
    border-length: 30%; /* 罫線の長さを指定 (ページエリア幅の30%) */
    padding-top: 0.5em;
  }
}
```

9.3 脚注番号 : ::footnote-call / ::footnote-marker 疑似要素 CSS3

脚注番号を設定するには、疑似要素の ::footnote-call と ::footnote-marker を使います。

```
::footnote-call { /* 脚注呼び出し */
  content: "(" counter(footnote) ")"; font-size: 6pt; vertical-align: super;
}
::footnote-marker { /* 脚注の番号 */
  content: "(" counter(footnote) ")"; font-size: 6pt; vertical-align: super;
}
```

⁵⁾ 脚注の配置には、[4.4.1 ページフロートの](#) -ah-float: page bottom; が使われています。脚注領域の設定は、@page ルールの中で@footnote ルールを使います。脚注の番号の形式を設定するには、疑似要素の ::footnote-call と ::footnote-marker を使います。

Chapter 10. PDF ブックマーク

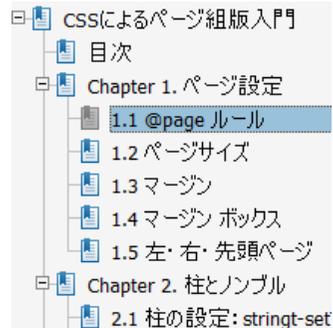
PDFのブックマーク（しおり）を生成することができます。

10.1 ブックマークレベル：-ah-bookmark-level プロパティ CSS3

●初期値：none ●適用対象：すべての要素 ●値の継承：しない

ブックマークのレベルを指定するには、bookmark-level プロパティを使います。

HTMLの<h1>~<h6>要素の見出しレベルに対応します。



10.2 ブックマークラベル：-ah-bookmark-label プロパティ CSS3

●初期値：要素の内容 ●適用対象：すべての要素 ●値の継承：しない

ブックマークのラベルを指定するには、bookmark-label プロパティを使います。

ラベルを指定しない場合、要素の内容がブックマークのラベルになります。

10.3 ブックマークの状態：-ah-bookmark-state プロパティ CSS3

●初期値：open ●適用対象：ブロック要素 ●値の継承：しない

ブックマークを開いた状態にするか閉じた状態にするかを指定するには、bookmark-state プロパティを使います。

-ah-bookmark-state: closed; とすると、指定のブックマークは閉じた状態になります。

```
/* Bookmarks */
h1 { -ah-bookmark-level: 1; }
h2 { -ah-bookmark-level: 2; -ah-bookmark-state: closed; }
h3 { -ah-bookmark-level: 3; -ah-bookmark-state: closed; }
h4 { -ah-bookmark-level: 4; -ah-bookmark-state: closed; }
h5,h6 { -ah-bookmark-level: none; }
```

Chapter 11. MathML と SVG グラフィック

11.1 MathML

AH Formatter V5.0 は、W3C による Mathematical Markup Language (MathML) 2.0 Second Edition を独自開発したエンジンにより描画することができます。そのため、PDF 中へのイメージを高い解像度で描画することが可能です。

AH Formatter V5.0 Lite 版では、MathML を独自描画するためには、別途、「AH Formatter MathML オプション」をご購入していただく必要があります。

MathML を使った組版例

2 次方程式 $ax^2 + bx + c = 0$ の解は、 $D = b^2 - 4ac$ とおくと、

$$D \geq 0 \text{ のとき, } x = \frac{-b \pm \sqrt{D}}{2a}$$

$D < 0$ のとき、実数の解なし。

<p>2次方程式

```
<m:math xmlns:m="http://www.w3.org/1998/Math/MathML" xml:lang="en">
<m:mstyle displaystyle="true" scriptminsize="1pt" scriptsizemultiplier="0.6">
<m:mrow>
<m:msup>
<m:mrow>
<m:mi>a</m:mi>
<m:mo>&#x2062;</m:mo>
<m:mi>x</m:mi>
</m:mrow>
<m:mn>2</m:mn>
</m:msup>
...
の解は,
...
```

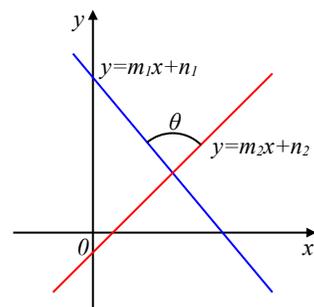
MathML を使った組版例

2 つの直線の方程式を $y = m_1x + n_1$, $y = m_2x + n_2$

とすると、この 2 直線のなす角 θ は、

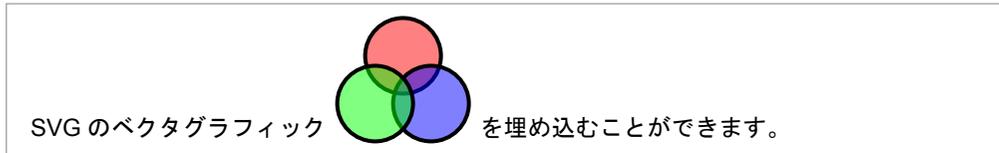
$$\tan \theta = \pm \frac{m_2 - m_1}{1 + m_2m_1} \quad (0^\circ \leq \theta < 180^\circ)$$

となる。



11.2 SVG グラフィック

AH Formatter V5.0 は、W3C による Scalable Vector Graphics (SVG) 1.1 を独自開発したエンジンにより描画します。そのため、PDF 中へのイメージを高い解像度で描画することが可能です。



SVGのベクタグラフィック

```
<s:svg xmlns:s="http://www.w3.org/2000/svg" width="70" height="65"
  viewBox="0 0 70 65">
  <s:g fill-opacity=".5" stroke="black" stroke-width="2">
    <s:circle cx="35" cy="20" r="19" fill="red"/>...
  </s:g>
</s:svg>
```

を埋め込むことができます。</p>

SVG の例



Chapter 12. 日本語文字組版

12.1 行頭行末約物半角固定・段落先頭1字下げ

日本語文書を組版する場合の基本的なスタイル設定は、行頭約物半角固定・行末約物半角固定・段落先頭1字下げとなっています。

- 行頭約物：半角固定とします。
- 行末約物：半角と固定とします。
- 段落先頭：1字下げとします。

```
body { /* 日本語組版のための設定 */
  -ah-punctuation-trim: start end adjacent;
  -ah-text-justify-trim: punctuation;
  -ah-text-autospace: ideograph-numeric ideograph-alpha;
  -ah-text-autospace-width: 25%;
  -ah-punctuation-spacing: 50%;
  -ah-hanging-punctuation: none;
}
p { /* 段落 */
  text-align: justify; /* 行末を揃える */
  text-indent: 1em; /* 段落は全角1字下げ */
  margin: 0; /* 段落間は空けない */
}
```

夏目 漱石（なつめ そうせき、慶応三年一月五日（一八六七年二月九日）―大正五年（一九一六年）一月九日）は、日本の小説家、評論家、英文学者。本名、金之助。『吾輩は猫である』『こころ』などの作品で広く知られる、森鷗外と並ぶ明治・大正時代の文豪である。江戸の牛込馬場下横町（現在の東京都新宿区喜久井町）出身。俳号は愚陀仏。大学時代に正岡子規と出会ひ、俳句を学ぶ。：（中略）：当初は余裕派と呼ばれた。

「修善寺の大患」後は、『行人』『こころ』『硝子戸の中』などを執筆。「則天去私」（そくてんきよし）の境地に達したといわれる。晩年は胃潰瘍に悩まされ、『明暗』が絶筆となった。

フリー百科事典『ウィキペディア』より引用

12.2 約物の詰め処理：-ah-punctuation-trim プロパティ [CSS3](#)

●初期値：none ●適用対象：ブロック要素 ●値の継承：する
行頭、行末の全角約物を半角扱いにするかどうかを指定します。

- none：全角約物の詰めを処理しません。
- start：行頭での全角約物（開き括弧など）を半角に扱います。

- end : text-align="right"、または text-align="justify"（または text-align-last="justify"）のとき、全角約物（閉じ括弧など）が行末に位置した場合、その文字は強制的に半角に扱います。
- allow-end : text-align="right"、または text-align="justify"（または text-align-last="justify"）のとき、全角約物（閉じ括弧など）が行末に位置した場合、行に収まるときは全角のままとし、収まらないときは半角に扱います。
- end-except-fullstop : end と同じですが、次の 2 文字を除外します。
U+3002 = 「。」
U+FF0E = 「.」
- adjacent : 日本語での全角約物と全角文字の間の詰めを処理します。処理されるのは以下の間隔です。全角句読点は全角閉じ括弧と同様に扱われます。
全角閉じ括弧と全角開き括弧の間
全角閉じ括弧と全角閉じ括弧の間
全角閉じ括弧と全角中点類の間
全角開き括弧と全角開き括弧の間
全角中点類と全角開き括弧の間
- all : 全角の括弧類、中点類、句読点類をすべて詰めて半角扱いにします。

「《約物 [やくもの]》、つまり『括弧』・『句読点』の類（たぐい）です。」

このように、約物（句読点や括弧類）が連続する場合や行頭や行末に来たとき、全角文字の約物を半角幅に詰めて、見栄えをよくします。

「《約物 [やくもの] 》、つまり『括弧』・『句読点』の類（たぐい）です。」

約物の詰めを無効にするとこのようになります。(punctuation-trim: none; を指定)

```
/* 行頭、行末、約物どうし隣接のとき、全角約物を半角幅にします */
-ah-punctuation-trim: start end adjacent;
```

12.3 行調整のための約物の詰め : -ah-text-justify-trim プロパティ AH 独自拡張

●初期値 : none ●適用対象 : すべての要素 ●値の継承 : する

日本語での追い込みの処理方法を指定します。指定された詰めを行って、行に入るだけ文字を追いつめます。

- none : 追いつめの処理をしません。
- punctuation : 全角の括弧類、中点類、句読点類を詰めの対象として追いつめ処理をします。
- punctuation-except-fullstop : punctuation と同じですが、次の 2 文字を除外します。
U+3002 = 「。」
U+FF0E = 「.」
- kana : ひらがな、カタカナを多少詰めて追いつめ処理をします。

この指定により、前節で説明した約物を半角幅に詰める場合に該当しない場合でも、行末をびったり揃える行の調整のために、全角の開き括弧類の前、全角閉じ括弧類や句読点の後を、必要に応じて自動的に詰めます。

```
/* 行の調整のために約物の詰めを行います */
-ah-text-justify-trim: punctuation;
```

12.4 和欧文間の空き：-ah-text-autospace プロパティ AH 独自拡張

●初期値：none ●適用対象：ブロック要素 ●値の継承：する
日本語で、和欧文間に空きを入れるかどうかを指定します。

- none：空きを入れません。
- ideograph-numeric：仮名漢字－欧文数字間に空きを入れます。ここで、欧文数字は、Unicode で Nd、NI、No に分類される文字です。
- ideograph-alpha：仮名漢字－欧文アルファベット間に空きを入れます。ここで、欧文アルファベットは、Unicode で Lu、LI、Lt、Lm、Lo に分類される文字です。
- ideograph-parenthesis：仮名漢字－欧文括弧間に空きを入れます。ただし、仮名漢字－欧文閉じ括弧間や、欧文開き括弧－仮名漢字間には空きを入れません。
- ideograph-punctuation：仮名漢字－欧文約物間に空きを入れます。ここでの欧文約物は、Unicode で Ps、Pe、Po、Pi、Pf に分類される文字で、括弧と引用符を除く文字です。ピリオド - 仮名漢字間には空きを入れますが、仮名漢字 - ピリオド間には空きを入れません。カンマも同様です。
- auto：システムの既定値に依存します。設定によって、"none" または "ideograph-numeric ideograph-alpha" とみなされます。

「日本語にも global にも 100%を目指す AH Formatter V5.0 です」のように、日本語の文章の中に欧文や算用数字が混在するとき、間に少し空きを入れて読みやすくします。

「日本語にも global にも 100%を目指す AH Formatter V5.0 です」←こちらは比較のために、和欧文間の空きを無効にした例です (text-autospace: none; を指定)

```
/* 漢字や仮名と数字の間、漢字や仮名と欧文の間に空きを入れます */
-ah-text-autospace: ideograph-numeric ideograph-alpha;
```

12.5 和欧文間の空き量：-ah-text-autospace-width プロパティ AH 独自拡張

●初期値：25% ●適用対象：ブロック要素 ●値の継承：する
日本語で、和欧文間の空き量を指定します。

```
/* 和欧文間の空き量を指定します */
-ah-text-autospace-width: 25%; /* 初期値 */
```

12.6 約物と全角文字間の詰め：-ah-punctuation-spacing プロパティ AH 独自拡張

●初期値：50% ●適用対象：ブロック要素 ●値の継承：する
日本語で、全角約物と全角文字の間の詰めの間隔を指定します。

```
/* 全角約物と全角文字の間の詰めの間隔を指定します */
-ah-punctuation-spacing: 50%; /* 初期値 */
```

12.7 突き出し・ぶら下げ：-ah-hanging-punctuation プロパティ AH 独自拡張

●初期値：none ●適用対象：ブロック要素 ●値の継承：する
行頭での突き出し、行末でのぶら下げを行うかどうかを指定します。

- none：行頭での突き出し、行末でのぶら下げを処理しません。
- start：行頭での突き出しを処理します。このとき、突き出し対象文字が行頭に位置した場合、その文字は強制的に突き出されます。
- first：段落の先頭行でのみ、startと同じ処理をします。
- force-end：行末でのぶら下げを処理します。text-align="right"、または text-align="justify"（または text-align-last="justify"）のとき、ぶら下げ対象文字が行末に位置した場合、その文字は強制的にぶら下げられます。text-align がそれ以外のときは、なりゆきでぶら下げが起こります。
- allow-end：行末でのぶら下げを処理します。text-align="right"、または text-align="justify"（または text-align-last="justify"）のとき、ぶら下げ対象文字が行末に位置した場合、行に収まるときはぶら下げず、収まらないときはぶら下げられます。text-align がそれ以外のときは、なりゆきでぶら下げが起こります。
- last：段落末でのみ、force-endと同じぶら下げを処理します。

対象となる約物は以下のとおりです。

- force-end、allow-end
 - 日本語または簡体字中国語の場合
 - U+3001 = 「、」
 - U+3002 = 「。」
 - U+FF0C = 「,」
 - U+FF0E = 「.」
 - 繁体字中国語の場合
 - U+FE50
 - U+FE51
 - U+FE52
 - U+FF64
 - その他の場合

last と同じ

- last
引用符類、閉じ括弧類、ピリオド、カンマ、ハイフン
- start、first
引用符類、開き括弧類、Bullet 類

```
/* 行頭での突き出し、行末でのぶら下げを行うかどうかを指定します。 */
-ah-hanging-punctuation: none; /* 初期値 */
```

12.8 行頭約物半角固定・行末約物全角／半角・段落先頭 1 字下げ

行頭約物半角固定・行末約物全角または半角・段落先頭 1 字下げとします。

- 行頭約物：半角固定とします。
- 行末約物：全角または半角とします。
- 段落先頭：1 字下げとします。

```
body { /* 日本語組版のための設定 */
-ah-punctuation-trim: start allow-end adjacent;
-ah-text-justify-trim: punctuation;
-ah-text-autospace: ideograph-numeric ideograph-alpha;
-ah-text-autospace-width: 25%;
-ah-punctuation-spacing: 50%;
-ah-hanging-punctuation: none;
}
p { /* 段落 */
text-align: justify; /* 行末を揃える */
text-indent: 1em; /* 段落は全角 1 字下げ */
margin: 0; /* 段落間は空けない */
}
```

夏目 漱石（なつめ そうせき、慶応三年一月五日（一八六七年二月九日）―大正五年（一九一六年）二月九日）は、日本の小説家、評論家、英文学者。本名、金之助。『吾輩は猫である』『こころ』などの作品で広く知られる、森鷗外と並ぶ明治・大正時代の文豪である。江戸の牛込馬場下横町（現在の東京都新宿区喜久井町）出身。俳号は愚陀仏。大学時代に正岡子規と出会ひ、俳句を学ぶ。…（中略）…当初は余裕派と呼ばれた。

「修善寺の大患」後は、『行人』『こころ』『硝子戸の中』などを執筆。「則天去私」（そくてんきよし）の境地に達したといわれる。晩年は胃潰瘍に悩まされ、『明暗』が絶筆となった。

フリー百科事典『ウィキペディア』より引用

12.9 行頭約物半角固定・行末句点全角・段落先頭 1 字下げ

行頭約物半角固定・行末約物は句点のみ全角・段落先頭 1 字下げとします。

- 行頭約物：半角固定とします。
- 行末約物：句点のみ全角、その他は半角固定とします。
- 段落先頭：1字下げとします。

```
body { /* 日本語組版のための設定 */
  -ah-punctuation-trim: start end-except-fullstop adjacent;
  -ah-text-justify-trim: punctuation;
  -ah-text-autospace: ideograph-numeric ideograph-alpha;
  -ah-text-autospace-width: 25%;
  -ah-punctuation-spacing: 50%;
  -ah-hanging-punctuation: none;
}
p { /* 段落 */
  text-align: justify; /* 行末を揃える */
  text-indent: 1em; /* 段落は全角1字下げ */
  margin: 0; /* 段落間は空けない */
}
```

夏目 漱石（なつめ そうせき、慶応三年一月五日（一八六七年二月九日）―大正五年（一九一六年）一月九日）は、日本の小説家、評論家、英文学者。本名、金之助。『吾輩は猫である』『こころ』などの作品で広く知られる、森鷗外と並ぶ明治・大正時代の文豪である。江戸の牛込馬場下横町（現在の東京都新宿区喜久井町）出身。俳号は愚陀仏。大学時代に正岡子規と出会い、俳句を学ぶ。…（中略）…当初は余裕派と呼ばれた。

「修善寺の大患」後は、『行人』『こころ』『硝子戸の中』などを執筆。「則天去私」（そくてんきよし）の境地に達したといわれる。晩年は胃潰瘍に悩まされ、『明暗』が絶筆となった。

フリー百科事典『ウィキペディア』より引用

12.10 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）

行頭約物半角固定・行末約物半角固定・段落先頭の約物は0.5字下げとします。

- 行頭数約物：半角固定とします。
- 行末約物：半角固定とします。
- 段落先頭：1字下げとしますが、約物は0.5字下げとします。

```
body { /* 日本語組版のための設定 */
  -ah-punctuation-trim: start end adjacent;
  -ah-text-justify-trim: punctuation;
  -ah-text-autospace: ideograph-numeric ideograph-alpha;
  -ah-text-autospace-width: 25%;
  -ah-punctuation-spacing: 50%;
  -ah-hanging-punctuation: first;
}
p { /* 段落 */
  text-align: justify; /* 行末を揃える */
  text-indent: 1.5em; /* 段落は全角1.5字下げ */
}
```

```
margin: 0; /* 段落間は空けない */
}
```

夏目 漱石（なつめ そうせき、慶応三年一月五日（一八六七年二月九日）―大正五年（一九一六年）二月九日）は、日本の小説家、評論家、英文学者。本名、金之助。『吾輩は猫である』『こころ』などの作品で広く知られる、森鷗外と並ぶ明治・大正時代の文豪である。江戸の牛込馬場下横町（現在の東京都新宿区喜久井町）出身。俳号は愚陀仏。大学時代に正岡子規と出会い、俳句を学ぶ。：（中略）：当初は余裕派と呼ばれた。

「修善寺の大患」後は、『行人』『こころ』『硝子戸の中』などを執筆。「則天去私」（そくてんきよし）の境地に達したといわれる。晩年は胃潰瘍に悩まされ、『明暗』が絶筆となった。

フリー百科事典『ウィキペディア』より引用

12.11 縦中横

縦書きの中に縦中横（縦書きの行の中に数字などを横書きにする）を指定します。

```
/* 縦中横 */
span.TateChuYoko {
  display: inline-block; /* 行の中に小さなブロックを作る */
  writing-mode: lr-tb; /* この小さなブロックを横書きにする */
  text-align: center; /* 中央揃えに */
  text-indent: 0; /* 字下げ無し（段落pのtext-indentが継承されないように） */
  line-height: 1; /* 行間の空きが縦中横の上下につかないように */
}

<div class="VerticalTextBlock">
  ...
  <p>縦書きの中に「<span class="TateChuYoko">'08</span>年
  <span class="TateChuYoko">12</span>月<span class="TateChuYoko">8</span>日」
  のように部分的に数字などを横書きにすることを「縦中横」といいます。</p>
</div>
```

縦書きの中に「'08年12月8日」のように部分的に数字などを横書きにすることを「縦中横」といいます。

12.12 ルビ・圏点

XHTML では<ruby>要素を使うと、文字にルビ（フリガナ）をふることができます。

<rb>タグで親文字（ルビをふる対象となる文字）を指定します。

<rt>タグでルビ文字（ルビの内容）を指定します。

これを CSS を使って実現する例を次に示しています。縦書きでも横書きでも使えるように-ah-margin-before [AH独自拡張](#) を使っています。

```
/* ルビ指定 */
ruby {
  display: inline-table;
  vertical-align: text-top;
  -ah-margin-before: -0.5em;
}
ruby > * {
  display: table-row-group;
}
ruby > rt {
  display: table-header-group;
  font-size: 50%;
}
ruby rt[rbspan] {
  -ah-table-column-span: attr(rbspan);
}
ruby rp {
  display: none;
}

<p><ruby><rb>吾輩</rb><rp> (わがはい</rp><rt>わがはい</rt><rp> )</rp></ruby>は猫である。</p>
<p>名前はまだ無い。どこで生れたかけんとうと見当<rb>見当</rb><rp> (けんとう</rp><rt>けんとう</rt>
<rp> )</rp></ruby>がつかぬ。</p>
```

横書き

わがはい
吾輩は猫である。
名前はまだ無い。どこで生れたかけんとうと見当けんとうがつかぬ。

縦書き
わがはい
吾輩は猫である。
名前はまだ無い。どこ
で生れたかけんとうと見当けんとう
がつかぬ。

圏点は、ルビ文字に中点や読点などを使います。<ruby>要素の設定を圏点用に変更してください。

```
/* 圏点 */
ruby.kenten {
  -ah-margin-before: -0.7em;
}
ruby.kenten > rt {
  font-size: 70%;
}

<p>横書きでは、<ruby class="kenten"><rb>圏</rb><rp> (けん</rp><rt>・</rt><rp> )</rp>
```

```
</ruby><ruby class="kenten"><rb>点</rb><rp> ( </rp><rt>・ </rt><rp> ) </rp></ruby>は、  
ルビ文字に中点を使います。</p>
```

横書きでは、[・] 圈点は、ルビ文字に中点を使います。

12.13 日本語フォント

日本語フォント、たとえば株式会社モリサワのオープンタイプフォントの指定は、以下のように
なります。

- リュウミン R font-family: "A-OTF リュウミン Pro R-KL", serif;
- 中ゴシック BBB font-family: "A-OTF 中ゴシック BBB Pro Medium", sans-serif;
- 新ゴ B font-family: "A-OTF 新ゴ Pro B", sans-serif;
- 見出ゴ MB31 font-family: "A-OTF 見出ゴ MB31 Pro MB31", sans-serif;
- 見出ミン MA31 font-family: "A-OTF 見出ミン MA31 Pro MA31", serif;

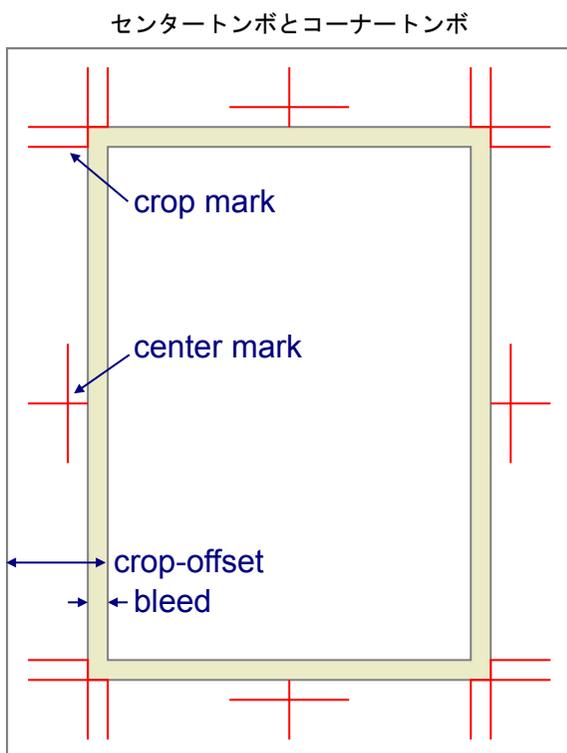
Chapter 13. 印刷設定

13.1 トンボ（クロップマーク）

13.1.1 トンボの表示：-ah-marks プロパティ CSS3

●初期値：none ●適用対象：@page で使用 ●値の継承：しない
印刷時にトンボを印刷するかどうかを指定します。

- crop：コーナートンボ（crop marks）を出力します。
- cross：センタートンボ（cross marks、registration marks）を出力します。



判型に断裁するための位置や多色刷りの見当合せのため、版下の天地・左右の中央と四隅などに付ける目印です。天地・左右の中央に付けるものをセンタートンボ、版面の四隅に付けるものをコーナートンボ（角トンボ）と呼びます。

トンボの形状は指定できません。

AH Formatter V5.0 Lite 版では、この拡張プロパティは利用できません。

```
@page {
  size: B5; /* ISO B5 (176mm 250mm) */
  margin: 28mm 21.325mm;
  -ah-marks: crop cross; /* トンボの表示 */
  -ah-crop-offset: 14mm; /* 出力媒体の物理的な端から判型までの距離 */
  -ah-printer-marks-line-color: auto; /* トンボの線の色 */
  -ah-printer-marks-line-length: 10mm; /* トンボの線の長さ */
  -ah-printer-marks-line-width: 0.12mm; /* トンボの線の幅 */
  -ah-page-bleed: 3mm; /* 断ち落としの長さ */
}
```

13.1.2 トンボの線の色 : -ah-printer-marks-line-color プロパティ AH 独自拡張

●初期値 : auto ●適用対象 : @page で使用 ●値の継承 : しない

トンボの線の色を指定します。

値が auto のときは、レジストレーションカラーとみなされます。

AH Formatter V5.0 Lite 版では、この拡張プロパティは利用できません。

13.1.3 トンボの線の長さ : -ah-printer-marks-line-length プロパティ AH 独自拡張

●初期値 : auto ●適用対象 : @page で使用 ●値の継承 : しない

トンボの線の長さを指定します。

値が auto のときは、システム既定値に依存します。デフォルトのトンボの長さは 10mm ですが、オプション設定ファイルで調整することができます。

AH Formatter V5.0 Lite 版では、この拡張プロパティは利用できません。

13.1.4 トンボの線の幅 : -ah-printer-marks-line-width プロパティ AH 独自拡張

●初期値 : auto ●適用対象 : @page で使用 ●値の継承 : しない

トンボの線の幅を指定します。

値が auto のときは、システム既定値に依存します。デフォルトのトンボの幅は 0.2pt ですが、オプション設定ファイルで調整することができます。

AH Formatter V5.0 Lite 版では、この拡張プロパティは利用できません。

13.2 断ち落とし (塗り足し)

13.2.1 断ち落とし (塗り足し) の幅 : -ah-page-bleed プロパティ CSS3

●初期値 : 0 ●適用対象 : @page で使用 ●値の継承 : しない

断ち落とし（塗り足し）の幅を指定します。断ち落とし領域は、判型の外側に取られます。マージンボックスのマージンにマイナスの値を指定することで、断ち落とし領域までブロックの範囲を伸ばすことができます。

AH Formatter V5.0 Lite 版では、この拡張プロパティは利用できません。

13.2.2 出力媒体の端から判型までの距離：-ah-crop-offset プロパティ AH 独自拡張

●初期値：0 ●適用対象：@page で使用 ●値の継承：しない

出力媒体の物理的な端から判型までの距離を指定します。

AH Formatter V5.0 Lite 版では、この拡張プロパティは利用できません。

Chapter 14. 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 V5.0 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)**
..... ドラフト仕様 <http://www.w3.org/TR/css3-gcpm/>
..... Editor's Draft <http://dev.w3.org/csswg/css3-gcpm/>
- ✓ **CSS3 Multi-column layout** ドラフト仕様 <http://www.w3.org/TR/css3-multicol/>
- ✓ **CSS3 Backgrounds and Borders** ドラフト仕様 <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** 最終ドラフト <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 V5.0 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/>

CSS によるプリントデザイン入門

2008 年 12 月 8 日 初版発行

2009 年 3 月 13 日 第 2 版発行

著者 アンテナハウス株式会社

発行所 アンテナハウス株式会社

〒103-0004 東京都中央区東日本橋 2 丁目 1 番 6 号 東日本橋藤和ビル 5 階

電話 03-5829-9021 / FAX 03-5829-9023

Copyright © アンテナハウス株式会社 2008

Web ページ <http://www.antenna.co.jp/>

本冊子の内容は XHTML 1.1 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 V5.0 で PDF 化しています。