

Web 追加資料

平成 17 年 10 月 17 日

第 I 部 訂正

p.4 I.5 追加: は通常の空白を表します。漢字と同じ幅の空白ではありませんので注意して下さい。

p.22 II.19–20 変更前: font-style はフォントスタイル(立体か, 斜字体か)を指定します。normal, italic, oblique が指定できます。

変更後: font-style はフォントスタイルを指定します。normal(立体), italic(デザインが考慮された斜字体), oblique(単なる斜字体) が指定できます。

p.22 最後に追加

list-style-type 箇条書 (li 要素) のマークを指定します。disc(黒丸), circle(白丸), square(四角), decimal(十進数), lower-roman(英字の小文字), upper-roman(英字の大文字), lower-greek(小文字のギリシャ文字), hiragana(ひらがなのいろは), none(なし)などを指定できます。

p.27 II.2–5 変更前: 実験用サーバを使用する際、自分のノートパソコンの Windows の設定を行う必要があります。ウィルススキャンソフトあるいは Windows XP に付属しているファイアウォールが設定されているとファイル共有ができませんので、ファイアウォールの設定は解除して下さい。なお、実験終了後は必ずもとに戻して下さい。

変更後: 実験用サーバを使用する際、自分のノートパソコンの Windows の設定を行う必要があります。なお、Windows XP のファイアウォールは「例外」で「ファイルやプリンタの共有」がチェックされていれば解除しなくてもファイル共有ができます。もし、チェックできない、あるいは Service Pack 1 以前で例外が指定できない場合はファイアウォールを解除して下さい。なお、ウィルススキャンソフトが動作しているとファイルの共有ができないことがあります。ファイルの共有がうまく動作しない場合は動作を一時停止させてください。なお、セキュリティの設定変更に関しては、実験終了後は必ずもとに戻して下さい。

p.28 図 25 まるごと差し替え

p.29 II.4–5 実験 3 変更前: なお、少なくとも見出しと本文は区別して異なるスタイルを与えて下さい。

変更後: なお、段落は枠で囲み、箇条書は背景色を変更して下さい。

p.31 l.2 変更前: (4) 国際規格に準拠してコンテンツを作る

変更後: (4) 今回実験で行ったように国際規格に準拠してコンテンツを作る

p.31 l.7 変更前: (1) HTML 4.01 Strict

変更後: (1) 今回実験で使用した HTML 4.01 Strict

p.32 l.15 変更前:

```
<table border=1>
```

変更後:

```
<div id="table">  
<table border="1">
```

p.32 l.31 変更前:

```
</table>
```

変更後:

```
</table>  
</div>
```

p.34 l.10 変更前: "現金"

変更後: "cash"

p.34 ll.32-34 変更前:

```
if ($cgi->param("item$i") ne ""){  
    $imax=$i;  
    $item[$i]=Jcode::convert($cgi->param("item$i"), 'euc');
```

変更後:

```
if ($cgi->param("price$i") > 0){  
    $imax++;  
    $item[$imax]=Jcode::convert($cgi->param("item$i"), 'euc');
```

p.35 ll.1-5 変更前:

```
$price[$i]=$cgi->param("price$i");  
$num[$i]=$cgi->param("num$i");  
} else {  
    last;  
}
```

変更後:

```
    $price[$imax]=$cgi->param("price$i");
    $num[$imax]=$cgi->param("num$i");
}
```

p.35 l.19 变更前:

```
<table border=1>
```

变更后:

```
<div id="table">
<table border="1">
```

p.36 l.10 变更前:

```
</table>
```

变更后:

```
</table>
</div>
```

10.2 内容の幅: 'width' プロパティ

width

値:

「長さ」 | 「% 表示」 | auto | inherit

初期値:

auto

適応範囲:

再配置されないインライン要素と table の列と row group を除くすべて

継承:

しない

% 表示:

含まれているブロックの幅を基準とする

メディア:

視覚 (visual)

このプロパティはブロックレベルまたは再配置要素により生成されるボックスの幅を規定します。

このプロパティは再配置されないインラインレベルの要素には適応されません。再配置されないインラインレベルの要素のボックスの幅は、(子要素の相対位置補正の前の) それを含んでいる表示されている内容の幅になります。なおインラインのボックスは行のボックスに流し込まれることに注意してください。行のボックスの幅はそれが含まれているボックスにより与えられます。但し、float が存在する時は縮められるかも知れません。

再配置された要素のボックスの幅は固有のものなので、このプロパティの値が auto 以外ならユーザエージェントにより調整されます。

値は以下のような意味を持ちます

長さ

決まった長さを表す。

% 表示

幅の割合を表す。割合はブロックに含まれる生成された box の幅に対して計算される。

auto

幅は他の属性の値に依存する。以下の章 (10.3) を参照。

width に対する負の値は誤りになります。

例

例えば、次のルールは段落の内容の幅を 100 ピクセルに固定します。

```
p {width: 100px}
```

図 25: 差し替え

第II部

発展学習

1 XHTML 1.1

2005年現在、最新のHTMLの規格はXHTML 1.1です。

XHTML 1.1はXHTML 1.0 Strictをモジュール化したものです。またXHTML 1.0 StrictはHTML 4.01 StrictをXML化したものです。

1.1 XML

XMLはHTMLを作成する際のメタ言語SGMLを制限したものです。従って機能は低くなりますが、その反面、コンピュータプログラムなどの処理が非常にし易くなっています。データベースのデータの記述の他、OpenOffice.orgや新しいMicrosoft Office Wordなど各種ワープロの文書ファイルのフォーマットなどに使用されています。また、JavaやC#言語などでも簡単に読み書きできます。同様に、WebのドキュメントもXML化することでコンピュータによる処理がより簡単になります。

SGMLに対してXMLが異なる主な点は以下の通りです。

1. デフォルトでは文字コードとしてUTF-8またはUTF-16が想定されている
2. 名前空間がサポートされた
3. タグの省略ができなくなった
4. 要素名の大文字と小文字が区別されるようになった
5. オプション(属性)の値は必ず"'"で括らなければならなくなった。
6. SGML由来ではinput要素のcheckedオプションのようにオプション(属性)の値なしでも機能していたが、XMLではchecked="checked"と省略もできなくなった。

HTMLでは終了タグを禁止している要素がありましたが、XML化したXHTMLではそのような要素に対しても終了タグが必要になります。但し、そうすると互換性に問題が出てきますので、通常そのような要素に対しては開始タグの最後に/を付けるというルールがあります。以下は例です。

```
<link rel="stylesheet" type="text/css" href="style.css" />
```

1.2 モジュール化

モジュール化とは様々な機能の拡張を従来のように言語の拡張で行わず、機能拡張モジュールの呼び出しで行うと言うものです。現在利用できるXHTMLのモジュールとしては数式を

表現できる MathML や、ブラウザの解像度に合わせて図形を表示できる SVG(Scalable Vector Graphics) などがあります。例として、以下の数式を MathML で表示したものを掲げます。

$$\sqrt{2} = 1 + \frac{1}{2 + \frac{1}{2 + \frac{1}{2 + \dots}}}$$

```
<?xml version="1.0" encoding="Shift_JIS"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1 plus MathML 2.0//EN"
  "http://www.w3.org/TR/MathML2/dtd/xhtml-math11-f.dtd"
  [

```

このようにかなり煩瑣にはなりますが、数式を表現できます。

1.3 HTML 4.01 Strict から XHTML 1.1 への移行

ここでは、作成したドキュメントを XHTML 1.1 仕様にするための変更点を挙げます。HTML 4.01 と XHTML 1.1 は同じではありませんが、高い互換性を持ちます (完全な互換性はありません)。従って変更範囲は限られています。なお、実験書では HTML 4.01 で許されている XHTML 1.1 型の表記はすでに取り込んであります。

1. 要素名 (CSS を含む) や属性名をすべて小文字にする (実験書は対応済み)
2. 属性値を必ず書き、`”` で囲む (実験書は対応済み)
3. 終了タグを省略しない。終了タグが禁止されている要素に対しては、開始タグの最後に `/` を加える。
4. lang 属性は廃止されたので `xml:lang` を使用する。
5. DOCTYPE と html 開始タグを次のように変更する。

```
<?xml version="1.0" encoding="Shift_JIS" ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD_XHTML_1.1//EN"
    "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ja">
```

6. a 要素と map 要素で name 属性が廃止された。id 属性を使用する。
7. ファイル名の拡張子を .html から .xml にする。

以上で完成です。ブラウザで表示させてみましょう。但し、XHTML をブラウザで見る時、タグの対応がきちんと取れてないとドキュメントの表示の代わりにエラーメッセージが表示されます。

参考文献

- [AM01] Murray Altheim and Shane McCarron. XHTML 1.1 — module-based XHTML. World Wide Web Consortium, Recommendation REC-xhtml11-20010531, May 2001.
- [APMI04] Daniel Austin, Subramanian Peruvemba, Shane McCarron, and Masayasu Ishikawa. Modularization of XHTML — second edition. World Wide Web Consortium, Working Draft WD-xhtml-modularization-20040218, February 2004.
- [CIMP01] David Carlisle, Patrick Ion, Robert Miner, and Nico Poppelier. Mathematical markup language (mathML) version 2.0. World Wide Web Consortium, Recommendation REC-MathML2-20010221, February 2001.
- [Con00] World Wide Web Consortium. Extensible markup language (XML) 1.0 (second edition) – W3C recommendation. Available at <http://www.w3.org/TR/2000/WD-xml-2e-20000814>, 2000.
- [Pem00] Steven Pemberton. XHTML 1.0: The extensible hypertext markup language. World Wide Web Consortium, Recommendation REC-xhtml1-20000126, January 2000.
- [大藤 02] 大藤幹. 詳解 HTML & XHTML & CSS 辞典. 秀和システム, 2002.