

PHPでのCSVの書き出し方

2019年12月3日 火曜日 10:46

ExcelでBOM付きが無双するPHPでのCSVの書き出し方

2016年4月8日

こんばんは、Webプログラマの篠田です。

4月に入り、暖かくなってきましたね。1年のなかでも活動しやすい季節になりましたね。が睡魔との戦いが最も激しい季節でもありますね。

みなさんも睡魔に負けないようお過ごしください。

さて、前回（私の回の）ご紹介した「[BOMの地雷を踏み抜かない正しい文字コード接し方](#)」でご紹介した「BOM付きUTF-8」が無双するコードを、ご紹介したいとおもいます。

この記事の目次 [\[hide\]](#)

- [1 「BOM付きUTF-8」の無双（？）のコード](#)
- [2 コード解説（1）](#)
- [3 さすが「Microsoft Excel」と思いました！](#)
- [4 「Mac版Microsoft Excel」で表示させる](#)
- [5 コード解説（2）](#)
- [6 まとめ](#)

「BOM付きUTF-8」の無双（？）のコード

では早速、下記のコードで「BOM付きUTF-8」のCSVファイルを作ることができます。

```
1 <?php
2 $view = "";
3
4 $list = array(
5     array(
6         '三石琴乃',
7         '田中敦子',
8         '榊原良子',
```

で

ました

ドとの
もい

す。

```

 9      '池田昌子',
10      ),
11      array(
12          '高橋孝治',
13          '関俊彦',
14          '田の中勇',
15      ),
16  );
17
18  foreach ($list as $value) {
19      //配列の内容を「,」区切りで連結する
20      $view .= implode(",", $value). "\r\n";
21  }
22
23  $view = pack('C*',0xEF,0xBB,0xBF). $view;
24
25  //「$view」を「koe.csv」ファイルに書き出しする
26  file_put_contents("koe.csv", $view);

```

では、ポイントだけ見ていきましょう。

コード解説（１）

ポイントは、「UTF-8」の「BOM」であるコード「0xEF」「0xBB」「0xBF」をカン切りされた文字列の先頭に連結します。

「0xEF」「0xBB」「0xBF」は特殊なバイナリ文字列として扱わないといけないため、通常の文字連結では「BOM」として認識してくれません。

そのため、PHP関数の「pack関数」でバイナリ文字列に変換して連結させます。

「pack関数」の第１引数にある「C*」は「符号なしchar型の繰り返し」という意味になります。

これで書きだされたCSVファイルを「Windows版 Microsoft Excel」で開いてみてください。

綺麗に表示されたのではないですか？

※特殊文字の「高（ハシゴたか）」も綺麗に表示されていますね。

めでたし、めでたし。

さすが「Microsoft Excel」と思いました！

はい、めでたくなかった方がいるのではないのでしょうか。

そう、「Mac版Microsoft Excel」で開いた方は残念な結果になったと思います。

これは完全に盲点だったのですが、どうやらMac版とWindows版では同じMicrosoftでも文字コードの捉え方が違うようです。

さすが「Microsoft Excel」一筋縄ではいかないですねww

解決方法のコードをご紹介します。

「Mac版Microsoft Excel」で表示させる

結論からいうと「BOM付きUTF-8」では無理です！！！！

マ区

め通常

味にな

どさ

t Excel

何冊からいつと「BOM付きUTF-8」では無理です！！！！

「Mac版Microsoft Excel」で表示させるためには「BOM付きUTF-16LE」という文字コードでかつ「TSV形式」でないとMac、Windows両方で表示可能なファイルを作ることができません。

というわけで下記のコードをご紹介します。

```
1  <?php
2  $view = "";
3
4  $list = array(
5      array(
6          '三石琴乃',
7          '田中敦子',
8          '榊原良子',
9          '池田昌子',
10         ),
11     array(
12         '高橋孝治',
13         '関俊彦',
14         '田の中勇',
15     ),
16 );
17
18 foreach ($list as $value) {
19     //配列の内容を「,」区切りで連結する
20     $view .= implode("\t", $value). "\r\n";
21 }
22
23 $view = pack('C*',0xFF,0xFE). mb_convert_encoding($view, 'UTF-16LE', 'UTF-8');
24
25 //「$view」を「koe.csv」ファイルに書き出しする
26 file_put_contents("koe.csv", $view);
```

コード解説（２）

ポイントは2箇所です。

作成したファイルがUTF-8であることが前提ですが、PHP関数「mb_convert_encoding」を使って「UTF-16LE」という文字コードに変換します。もうこの段階で負けていますが。。。気にせず次に行きましょう。

あとは「BOM」を付けてあげるだけなのですがUTF-8と同じBOMでは効果がありません。

文字コードごとに「BOM」が変わります。

今回は「UTF-16」の「LE（リトルエンディアン）」なのでBOMは「0xFF」「0xFE」になります。

※「LE（リトルエンディアン）」については長くなるので割愛

ロード
ができ

せ

にな

...

ちなみに「UTF-16のLE」と「UTF-16LE」といつのかあるのですかこれは分類上別物
り、「UTF-16LE」の方は「BOM」が付けることができません。
ただ、ややこしいのがPHP関数「mb_convert_encoding」で「UTF-16のLE」というの
「UTF-16LE」に該当します。

※PHP関数「mb_convert_encoding」が取り扱える文字コードは[こちら](#)
なので、PHP関数「mb_convert_encoding」で変換した「UTF-16LE」の文字列に「BOM」
をつけることができます。

あとは、CSVである「,」をタブを表す「\t」にして配列を連結させます。

出力ファイルの拡張子は「.csv」のままでOKです

※厳密には「.tsv」ですが、それではExcelが反応してくれないので「.csv」で偽装し
ます。

さあ、これでMacでも綺麗に表示されるはずです！！

まとめ

あまり大きな声では言えませんが、、、「仕様をまとめることはできなかったの
か！？」と声を大にして言いたいというきもちですが、胸に閉まっておきたいと思
います。

検証は、Mac・Windows両方で見ましょう！

※ちなみにOpenOfficeとかで動かないとかで動かない！というクレームはお断りで

切にな

のが

[COM](#)」

しま

思いま

ですw