

**PHPWord Beta 0.6.2**

**开发者指南**

**目 录**

[首先我们要了解文档最基本的信息和设置： 4](#_Toc361845588)

[计量单位：缇（twips） 4](#_Toc361845589)

[字体设置 4](#_Toc361845590)

[文档属性设置 4](#_Toc361845591)

[新建文档 5](#_Toc361845592)

[添加页面 5](#_Toc361845593)

[页面样式 5](#_Toc361845594)

[页面样式属性 6](#_Toc361845595)

[文本 7](#_Toc361845596)

[添加文本 7](#_Toc361845597)

[添加文本资源 7](#_Toc361845598)

[文本样式 8](#_Toc361845599)

[样式属性列表 9](#_Toc361845600)

[添加换行符 10](#_Toc361845601)

[添加分页符 10](#_Toc361845602)

[列表 10](#_Toc361845603)

[添加列表 10](#_Toc361845604)

[列表样式 11](#_Toc361845605)

[列表样式属性列表 11](#_Toc361845606)

[超链接 11](#_Toc361845607)

[添加超链接 11](#_Toc361845608)

[超链接样式 12](#_Toc361845609)

[图片 13](#_Toc361845610)

[添加图片 13](#_Toc361845611)

[图片样式 13](#_Toc361845612)

[图片样式属性 13](#_Toc361845613)

[添加GD生成图片 14](#_Toc361845614)

[添加水印 14](#_Toc361845615)

[添加对象 15](#_Toc361845616)

[添加标题 15](#_Toc361845617)

[添加目录 16](#_Toc361845618)

[表格 17](#_Toc361845619)

[添加表格 17](#_Toc361845620)

[添加行 17](#_Toc361845621)

[添加单元格 17](#_Toc361845622)

[单元格样式 19](#_Toc361845623)

[表格样式 20](#_Toc361845624)

[页脚 22](#_Toc361845625)

[页眉 23](#_Toc361845626)

[模版 23](#_Toc361845627)

[其他问题修改 25](#_Toc361845628)

[解决文本缩进问题 25](#_Toc361845629)

[表格对齐和表格缩进 27](#_Toc361845630)

[图片缩进和绝对相对悬浮定位 30](#_Toc361845631)

# 首先我们要了解文档最基本的信息和设置：

 因为是国外编辑的类库，存在对中文支持的问题，使用前，我们需要进行一些修正：

1、解决编码问题，PHPword 会对输入的文字进行utf8\_encode编码转化，如果你使用GBK、GB2312或者utf8编码的话就会出现乱码，如果你用utf8编码，就查找类库中所有方法中的 utf8\_encode 转码将其删除，如果你采用GBK或者GB2312编码，使用iconv进行编码转换。

2、解决中文字体支持，在writer/word2007/base.php中 312行添加 $objWriter->writeAttribute('w:eastAsia',$font)

3、启动php zip支持，windows环境下在php配置文件php.ini中，将extension=php\_zip.dll前面的分号“;”去除；(如果没有，请添加extension=php\_zip.dll此行并确保php\_zip.dll文件存在相应的目录)，然后同样在php.ini文件中，将 zlib.output\_compression = Off 改为zlib.output\_compression = On ；

# 计量单位：缇（twips）

首先解释一下PHPWord最基本的计量单位：“缇”（twips），我们常常在文件中看到或使用计量单位“缇”，它是开源办公软件中最基本的计量单位，“缇”是"TWentieth of an Inch Point"的简写，意思 1/20磅，与其他常用剂量单位的换算是1缇=1/1,440英寸，1缇=1/567厘米，1缇=1/15像素

# 字体设置

文档默认字体是Arial，字号10号，我们可以通过以下方法设置默认字体和字号：

注，该库存在中文字体支持问题，解决方法：见文档开头

$PHPWord->setDefaultFontName('Tahoma');

$PHPWord->setDefaultFontSize(12);

# 文档属性设置

我们可以设置下列文档属性

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 描述 |
| Creator | String | 创建者 |
| Company | String | 公司 |
| Title | String | 标题 |
| Description | String | 描述 |
| Category | String | 分类 |
| Last modified by | String | 最后修改者 |
| Created | Datetime | 创建时间 |
| Modified | Datetime | 修改时间 |
| Subject | String | 主题 |
| Keywords | String | 关键词 |

我们可以通过以下方法设置文档属性

$properties = $PHPWord->getProperties();

$properties->setCreator('My name');

$properties->setCompany('My factory');

$properties->setTitle('My title');

$properties->setDescription('My description');

$properties->setCategory('My category');

$properties->setLastModifiedBy('My name');

$properties->setCreated( mktime(0, 0, 0, 3, 12, 2010) );

$properties->setModified( mktime(0, 0, 0, 3, 14, 2010) );

$properties->setSubject('My subject');

$properties->setKeywords('my, key, word');

# 新建文档

## 添加页面

添加默认页面（默认页面方向和页边距）：

$section = $PHPWord->createSection();

## 页面样式

调整页面样式和布局有两种方法：

创建样式数组:

$sectionStyle = array('orientation' => null,

'marginLeft' => 900,

'marginRight' => 900,

'marginTop' => 900,

'marginBottom' => 900);

$section = $PHPWord->createSection($sectionStyle);

直接调用样式属性设置方法进行设置:

$section = $PHPWord->createSection();

$sectionStyle = $section->getSettings();

$sectionStyle->setLandscape();

$sectionStyle->setPortrait();

$sectionStyle->setMarginLeft(900);

$sectionStyle->setMarginRight(900);

$sectionStyle->setMarginTop(900);

$sectionStyle->setMarginBottom(900);

## 页面样式属性

**注意：所有的属性对大小写敏感 !**

|  |  |
| --- | --- |
| 属性 | 描述 |
| orientation | 页面方向：  默认竖向：null  横向：landscape |
| marginTop | 上边距，单位：twips. |
| marginLeft | 左边距，单位：twips. |
| marginRight | 右边距，单位：twips. |
| marginBottom | 下边距，单位：twips.. |
| borderTopSize | 上边框尺寸，单位：twips. |
| borderTopColor | 上边框颜色 |
| borderLeftSize | 左边框尺寸，单位 ：twips. |
| borderLeftColor | 左边框颜色 |
| borderRightSize | 右边框尺寸，单位：twips. |
| borderRightColor | 右边框颜色 |
| borderBottomSize | 底边框尺寸，单位：twips. |
| borderBottomColor | 底边框颜色 |

页面高度和宽度是自动设置的，你可以通过以下两个属性来修改，但不推荐进行修改。

|  |  |
| --- | --- |
| 属性 | 描述 |
| pageSizeW | 页面宽度，单位： twips. |
| pageSizeH | 页面高度，单位：twips. |

# 文本

## 添加文本

向文档添加文本使用方法函数： addText.（注意PHPword 会对输入的文字进行utf8\_encode编码转化，如果你使用GBK、GB2312或者utf8编码的话就会出现乱码，如果你用utf8编码，就查找类库中所有方法中的 utf8\_encode 转码将其删除，如果你采用GBK或者GB2312编码，使用iconv进行编码转换。）

$section->addText( $text, [$fontStyle], [$paragraphStyle] );

|  |  |  |
| --- | --- | --- |
| addText() | | |
| 参数 | 类型 | 描述 |
| $text | String | 文本内容. |
| $fontStyle | String / Array | 字体样式. |
| $paragraphStyle | String / Array | 段落样式 |

## 添加文本资源

文本资源可以包含文本和链接，可以统一赋予段落样式，添加文本资源使用函数方法createTextrun.

|  |  |  |
| --- | --- | --- |
| createTextRun() | | |
| 参数 | 类型 | 描述 |
| $paragraphStyle | String / Array | 文本样式. |

添加文本资源后，就可以添加具有独特样式的文本或链接了。

$textrun = $section->createTextRun();

$textrun->addText('I am bold', array('bold'=>true));

$textrun->addText('I am italic, array('italic'=>true));

$textrun->addText('I am colored, array('color'=>'AACC00'));

当然也可以继承使用段落或文字样式

## 文本样式

设置文本样式有两种方法：

内嵌样式:

$fontStyle = array('color'=>'006699', 'size'=>18, 'bold'=>true);

$section->addText('helloWorld', $fontStyle);

$text = $section->addText('helloWorld');

$style = $text->getStyle();

$style->setColor('006699');

$style->setSize(18);

$style->setBold();

或者定义一个样式定义设置文本样式，定义一种样式后，必须把第二个参数设置为样式名称，使用方法函数addFontStyle:

$PHPWord->addFontStyle( $styleName, $fontStyle);

|  |  |  |
| --- | --- | --- |
| addFontStyle() | | |
| 参数 | 类型 | 描述 |
| $styleName | String | 样式名称 |
| $fontStyle | Array | 样式风格. |

示例:

$fontStyle = array('color'=>'006699', 'size'=>18, 'bold'=>true);

$PHPWord->addFontStyle('myOwnStyle', $fontStyle);

$text = $section->addText('helloWorld', 'myOwnStyle');

添加段落样式，使用方法函数addParagraphStyle:

|  |  |  |
| --- | --- | --- |
| addParagraphStyle() | | |
| 参数 | 类型 | 描述 |
| $styleName | String | 段落样式名称. |
| $paragraphStyle | Array | 段落样式. |

## 样式属性列表

**属性大小写敏感 !**

|  |  |
| --- | --- |
| Font Style | |
| 名称 | 描述 |
| size | 字号. |
| name | 字体 |
| bold | 粗体 |
| italic | 斜体 |
| superScript | 上标 |
| subScript | 下标 |
| underline | 下划线，使用常量: PHPWord\_Style\_Font::UNDERLINE\_... |
| Color | 字体颜色 |
| fgColor | 前景色. 只能使用预定义常量： PHPWord\_Style\_Font::FGCOLOR\_... |

|  |  |
| --- | --- |
| Paragraph Style | |
| 名称 | 描述 |
| align | 水平对齐:   * left * right * center * both / justify |
| spaceBefore | 段前间距，单位： twips. |
| spaceAfter | 段后间距，单位：twips |
| spacing | 行间距，单位： twips. |

## 添加换行符

添加换行符，使用方法函数 addTextBreak:

$section->addTextBreak();

添加多个换行符：

$section->addTextBreak(15);

.

# 添加分页符

添加分页符，使用方法函数：addPageBreak:

$section->addPageBreak();

# 列表

## 添加列表

添加列表使用方法函数： addListItem:

$section->addListItem( $text, [$depth], [$styleText], [$styleList], [$styleParagraph] );

|  |  |  |
| --- | --- | --- |
| addListItem() | | |
| 参数 | 类型 | 描述 |
| $text | String | 文本内容. |
| $depth | Integer | 编号 |
| $styleText | String / Array | 文本样式. |
| $styleList | Array | 列表样式. |
| $styleParagraph | String / Array | 段落样式 |

## 列表样式

示例:

$listStyle = array('listType' => PHPWord\_Style\_ListItem::TYPE\_NUMBER);

$section->addListItem('Listitem 1', 0, null, $listStyle);

## 列表样式属性列表

**属性大小写敏感!**

|  |  |
| --- | --- |
| 名称 | 描述 |
| listType | 列表符号样式.  使用常量 PHPWord\_Style\_ListItem::TYPE\_... |

# 超链接

## 添加超链接

添加超链接，使用方法函数： addLink:

$section->addLink( $linkSrc, [$linkName], [$styleFont], [$styleParagraph]);

|  |  |  |
| --- | --- | --- |
| addListItem() | | |
| 参数 | 类型 | 描述 |
| $linkSrc | String | 链接地址 |
| $linkName | String | 链接名称. |
| $styleFont | String / Array | 文本样式 |
| $styleParagraph | String / Array | 段落样式 |

注意在添加链接地址时最好trim一下前后有空格很可有可能导致文档打不开

## 超链接样式

定义超链接风格的两种方法

内嵌样式:

$linkStyle = array('color'=>'0000FF',

'underline'=>PHPWord\_Style\_Font::UNDERLINE\_SINGLE);

$section->addLink('http://www.google.de', null, $linkStyle);

或者定义一个样式定义设置超链接样式，定义一种样式后，必须把第三个参数设置为样式名称

$linkStyle = array('color'=>'0000FF',

'underline'=>PHPWord\_Style\_Font::UNDERLINE\_SINGLE);

$PHPWord->addLinkStyle('myHyperlinkStyle', $linkStyle);

$section->addLink('http://www.google.de', null, 'myHyperlinkStyle');

|  |  |  |
| --- | --- | --- |
| addLinkStyle() | | |
| 参数 | 类型 | 描述 |
| $styleName | String | 超链接样式名称. |
| $styles | Array | 链接样式.可以使用各种字体样式属性 |

# 图片

## 添加图片

添加图片的函数方法： addImage:

$section->addImage( $src, [$style] );

|  |  |  |
| --- | --- | --- |
| addImage() | | |
| 参数 | 类型 | 描述 |
| $src | String | 图像的服务器路径，支持相对和绝对路径 |
| $style | Array | 图片样式. |

注意在添加图片路径时最好trim一下前后有空格很可有可能导致文档打不开

## 图片样式

添加图片样式只能使用数组方式 :

$imageStyle = array('width'=>350, 'height'=>350, 'align'=>'center');

$section->addImage('EARTH.jpg', $imageStyle);

## 图片样式属性

**大小写敏感!**

|  |  |
| --- | --- |
| 名称 | 描述 |
| width | 图像宽度，单位像素 |
| height | 图像高度，单位像素 |
| align | 图像对齐方式   * left * right * center |

如果没有指定图片高或宽的属性，系统将使用PHP原生函数”getimagesize”来获取相关属性。

PHPWord 支持的图片格式: gif, jpeg, png, bmp, tiff.

## 添加GD生成图片

你也可以添加由GD库生成的图片，使用函数方法：addMemoryImage:

$section->addMemoryImage( $link, [$style] );

|  |  |  |
| --- | --- | --- |
| addMemoryImage() | | |
| 参数 | 类型 | 描述 |
| $link | String | 生成图片的php文件的路径**.**  **注意: 应设置文件的绝对路径（就像你在浏览器中调用php文件），否则会发生错误。** |
| $style | Array | 图像样式. |

示例:

$section->addMemoryImage('http://localhost/image.php');

你GD图片样式的设置和本地图片一样.

PHPWord 支持的 GD 图片类型： png, jpeg, gif.

## 添加水印

添加水印的页面需要一个头部引用，添加水印方法函数：addWatermark

|  |  |  |
| --- | --- | --- |
| addWatermark() | | |
| 参数 | 类型 | 描述 |
| $src | String | 水印图片的文件地址 |
| $style | Array | 水印图片样式 |

水印图片是在页面是绝对定位的，所以水印图片至少需要两个样式属性

|  |  |
| --- | --- |
| 名称 | 描述 |
| marginLeft | 左边距，单位像素 |
| marginTop | 上边距，单位像素 |

注：图片样式并没有提供图像并排，文字环绕等功能，可以通过与表格想结合进行解决。

# 添加对象

我们可以使用方法函数 addObject，添加对象和链接

$section->addObject( $src, [$style] );

|  |  |  |
| --- | --- | --- |
| addObject() | | |
| 参数 | 类型 | 描述 |
| $src | String | 文件的服务器，支持相对和绝对路径. |
| $style | Array | 对象样式. |

对象属性是有一个样式:

**属性区分大小写!**

|  |  |
| --- | --- |
| 名称 | 描述 |
| align | 对齐方式   * left * right * center |

PHPWord 支持的对象类型： XLS, DOC, PPT.

# 添加标题

我们可以使用标题来为结构化文档或为文档建立目录，添加标题使用方法函数addTitleStyle 和 addTitle:

$PHPWord->addTitleStyle( $titleCount, [$fontStyle] );

|  |  |  |
| --- | --- | --- |
| addTitleStyle() | | |
| 参数 | 类型 | 描述 |
| $src | Integer | 标题级别，最多支持9级标题 |
| $fontStyle | Array | 标题字体样式 |

**需要添给标题添加一个样式，否则文档不会将其作为一个真正的标题来处理。**

定义标题样式后，定义标题就很简单了，可以使用函数方法：addTitle;

$section->addTitle( $text, [$depth] );

|  |  |  |
| --- | --- | --- |
| addTitle() | | |
| 参数 | 类型 | 描述 |
| $text | String | 标题文本内容 |
| $depth | Integer | 标题级别编号，通过该参数调用addTtileStyle()设置的标题样式 |

# 添加目录

添加目录使用方法函数： addTOC:

$styleTOC = array('tabLeader'=>PHPWord\_Style\_TOC::TABLEADER\_DOT);

$styleFont = array('spaceAfter'=>60, 'name'=>'Tahoma', 'size'=>12);

$section->addTOC($styleFont, $styleTOC);

|  |  |  |
| --- | --- | --- |
| addTOC() | | |
| 参数 | 类型 | 描述 |
| $styleFont | Array | 目录字体样式 |
| $styleTOC | Array | 目录样式 |

目录样式属性列表:

**样式属性区分大小写 !**

|  |  |
| --- | --- |
| 名称 | 描述 |
| tabLeader | 标题的类型和对应页码.默认使用系统常量 PHPWord\_Style\_TOC::TABLEADER\_... |
| tabPos | 标题与页码的位置，单位： twips. |
| Indent | 标题缩进，单位： twips. |

# 表格

## 添加表格

添加表格使用函数方法：addTable:

$table = $section->addTable( *[$tableStyle]* );

参数 $tableStyle 是可选的. 表格样式这章有关于表格样式的详细说明。**为addTable建立一个本地对象，我们需要使用这个对象来调用相关函数方法。**

### 添加行

$table->addRow( *[$height]* );

行的高度可以通过$height参数来设置，单位：twips.

### 添加单元格

单元格添加前必须先添加行，添加单元格的函数方法为： addCell

$cell = $table->addCell(h, *[$cellStyle]* );

|  |  |  |
| --- | --- | --- |
| addCell() | | |
| 参数 | 类型 | 描述 |
| $width | Integer | 单元格宽度： twips. |
| $cellStyle | Array | 单元格样式 |

**为addcell创建一个本地对象，需要使用该对象来 调用以下函数**

|  |  |
| --- | --- |
| 名称 | 描述 |
| addText | 添加文本 |
| addTextBreak | 添加换行符 |
| addLink | 添加链接 |
| addImage | 添加图片 |
| addMemoryImage | 添加水印 |
| addListItem | 添加列表 |
| addObject | 添加对象 |
| addPreserveText | 添加页码，只对页眉和页脚有效 |

示例1:

$table = $section->addTable();

$table->addRow();

$cell = $table->addCell(2000);

$cell->addText('Cell 1');

$cell = $table->addCell(2000);

$cell->addText('Cell 2');

$cell = $table->addCell(2000);

$cell->addText('Cell 3');

示例2:

$table = $section->addTable();

$table->addRow(400);

$table->addCell(2000)->addText('Cell 1');

$table->addCell(2000)->addText('Cell 2');

$table->addCell(2000)->addText('Cell 3');

$table->addRow(1000);

$table->addCell(2000)->addText('Cell 4');

$table->addCell(2000)->addText('Cell 5');

$table->addCell(2000)->addText('Cell 6');

## 单元格样式

使用addCell的第二个参数来给单元格设置样式

示例:

$cellStyle = array('textDirection'=>PHPWord\_Style\_Cell::TEXT\_DIR\_BTLR, 'bgColor'=>'C0C0C0');

$table = $section->addTable();

$table->addRow(1000);

$table->addCell(2000, $cellStyle)->addText('Cell 1');

$table->addCell(2000, $cellStyle)->addText('Cell 2');

$table->addCell(2000, $cellStyle)->addText('Cell 3');

$table->addRow();

$table->addCell(2000)->addText('Cell 4');

$table->addCell(2000)->addText('Cell 5');

$table->addCell(2000)->addText('Cell 6');

单元格样式属性列表:

**属性大小写敏感 !**

|  |  |
| --- | --- |
| 名称 | 描述 |
| valign | 单元格内容对齐方式： left, right, center |
| textDirection | 文本方向. 使用预定常量 PHPWord\_Style\_Cell:: TEXT\_DIR\_... |
| bgColor | 单元格背景色 |
| borderTopSize | 单元格上边框尺寸，单位 twips. |
| borderTopColor | 单元格上边框 颜色 |
| borderLeftSize | 单元格左边框尺寸，单位twips |
| borderLeftColor | 单元格左边框颜色 |
| borderRightSize | 单元格右边框尺寸，单位twips |
| borderRightColor | 单元格右边框颜色 |
| borderBottomSize | 单元格下边框尺寸 ，单位twips |
| borderBottomColor | 单元格下边框颜色 |
|  |  |

## 表格样式

我们可以设置整个表格的样式，通过创建表格函数addTable的参数$tableStyle，表格具有如下样式属性

**属性名称大小写敏感!**

|  |  |
| --- | --- |
| 名称 | 描述 |
| cellMarginTop | 单元格上边距，单位： twips. |
| cellMarginLeft | 单元格左边距，单位： twips. |
| cellMarginRight | 单元格右边距，单位： twips. |
| cellMarginBottom | 单元格下边距，单位： twips. |

示例:

$tableStyle = array('cellMarginTop'=>80,

'cellMarginLeft'=>80,

'cellMarginRight'=>80,

'cellMarginBottom'=>80);

$table = $section->addTable($tableStyle);

我们可以使用函数方法： addTableStyle，为表格定义一个完整的样式。

$PHPWord->addTableStyle($styleName, $styleTable, [$styleFirstRow] );

|  |  |  |
| --- | --- | --- |
| addTableStyle() | | |
| 参数 | 类型 | 描述 |
| $styleName | String | 表样式名称 |
| $styleTable | Array | 这个表的样式 |
| $styleFirstRow | Array | 表头样式（第一行） |

示例:

$styleTable = array('borderColor'=>'006699',

'borderSize'=>6,

'cellMargin'=>50);

$styleFirstRow = array('bgColor'=>'66BBFF');

$PHPWord->addTableStyle('myTable', $styleTable, $styleFirstRow);

$table = $section->addTable('myTable');

$table->addRow(400);

$table->addCell(2000)->addText('Cell 1');

$table->addCell(2000)->addText('Cell 2');

$table->addCell(2000)->addText('Cell 3');

$table->addRow(1000);

$table->addCell(2000)->addText('Cell 4');

$table->addCell(2000)->addText('Cell 5');

$table->addCell(2000)->addText('Cell 6');

表格样式属性，注意属性名称大小写敏感**!**

|  |  |
| --- | --- |
| 名称 | 描述 |
| cellMarginTop | 单元格上边距，单位：twips. |
| cellMarginLeft | 单元格左边距，单位：twips. |
| cellMarginRight | 单元格右边距，单位：twips. |
| cellMarginBottom | 单元格下边距，单位：twips. |
| cellMargin | 单元格间距，单位：twips. |
| bgColor | 表格背景色 |
| borderTopSize | 表格上边框尺寸，单位：twips. |
| borderTopColor | 表格上边框颜色 |
| borderLeftSize | 表格左边框尺寸，单位：twips. |
| borderLeftColor | 表格左边框颜色 |
| borderRightSize | 表格右边框尺寸，单位：twips. |
| borderRightColor | 表格右边框颜色 |
| borderBottomSize | 表格下边框尺寸，单位：twips.. |
| borderBottomColor | 表格下边框颜色 |
| borderInsideHSize | 表格内水平网格尺寸，单位： twips. |
| borderInsideHColor | 表格内水平网格颜色 |
| borderInsideVSize | 表格内垂直网格尺寸，单位： twips. |
| borderInsideVColor | 表格内垂直网格颜色 |
| borderSize | 表格边框尺寸，单位：twips. |
| borderColor | 表格边框颜色 |

**注意：表格在word布局中的功能可以进行体现，例如进行图片，对象等的布局可以考虑与表格结合进行处理**

# 页脚

添加文档页脚使用函数方法： createFooter:

$footer = $section->createFooter();

**确保在本地对象中保存页脚，并使用下列函数**

|  |  |
| --- | --- |
| 名称 | 描述 |
| addText | 添加文本 |
| addTextBreak | 添加换行符 |
| addImage | 添加图像 |
| addMemoryImage | 添加GD生成图像 |
| addListItem | 添加列表 |
| addPreserveText | 添加页码，只能在页眉或页脚使用 |
| addTable | 添加表格 |
| createTextrun | 添加文本资源 |

向页脚（页眉）添加页码使用函数方法：addPreserveText:

addPreserveText( $text, [$style] );

|  |  |  |
| --- | --- | --- |
| addPreserveText() | | |
| 参数 | 类型 | 描述 |
| $text | String | 页脚（页眉）的文本内容 |
| $style | Array | 文字样式. |

示例:

$footer->addPreserveText('Page {PAGE} of {NUMPAGES}.');

# 页眉

添加页眉，使用函数方法： createHeader:

$header = $section->createHeader();

**确保页眉是建立在一个文档中，页眉和页脚使用相同的属性和函数，详见页脚章节** 。

注意：只用添加了页眉的页面，才能添加和使用图片水印

# 模版

我们可以利用搜索替换功能创建一个docx格式的模版文档，来替换文档中你想替换的文本. 但是要注意，只有文本或链接可以被替换。加载模版文档使用函数方法：loadTemplate function.

|  |  |  |
| --- | --- | --- |
| loadTemplate() | | |
| 参数 | 方法 | 描述 |
| $strFilename | String | 模版文件路径和名称 |

加载完成模版文档后，你可以使用函数方法： setValue 来搜索替换相关内容

|  |  |  |
| --- | --- | --- |
| setValue() | | |
| 参数 | Type | Description |
| $search | Mixed | 搜索的值 |
| $replace | Mixed | 替换的值 |

$template = $PHPWord->loadTemplate('Template.docx');

$template->setValue('Name', 'Somebody someone');

$template->setValue('Street', 'Coming-Undone-Street 32');

被搜索替换的标签格式为： **${YOUR\_SEARCH\_PATTERN}**

不能添加新的PHPWORD元素到加载的模版文档中、

模版使用的几个注意事项：

1、从模板生成word文档，支持在word模板文档里写替换标签，标签格式为${xxx}，不过一定要注意，不要直接在word里编辑这些标签，一定要在文本文档里先写好标签，直接拷贝粘贴上去，千万不要编辑，否则无法替换，原因也很简单，把word文档另存为xml，然后看xml里标签的位置，会发现标签中间被插入了很多没用的xml节点，还有中文字体的节点。。。

2、模版文档要一次性完成在保存，否则会出现替换失败问题。

3、中文乱码问题，这个一定是存在的，如果php环境已经是utf8了，要找到关键地方，转utf8的代码，去掉，否则就是转两次编码了，典型的是phpword\template.php文件，把这行注释掉：$replace = utf8\_encode($replace);

4、linux下报“Could not close zip file.”错误，这个你永远想不到，要将模板文件所在目录权限改为可写，因为要在那个目录下生成临时文件

5、引用模板文件和另存文件路径什么的，最好用绝对路径

# 其他问题修改

PHPWord库相对功能比较简单，部分功能需要进行二次开发才能解决

这里要感谢网友[**yukirin\_fans**](http://my.csdn.net/yukirin_fans)，以下是他发布的开发代码

## 解决文本缩进问题

1、/PHPWord/Style/Paragraph.php  添加以下属性

/\*\*

\* 缩进 indentleft and indentright段落缩进值,单位为twips

\*

\* 缩进indentFirstLine and indentFirstChars 首行缩进twips数

\*

\* @var int

\*/

private $\_indentLeft;

private $\_indentRight;

private $\_indentFirstLine;

private $\_indentFirstLineChars;

    // 获取左缩进值

    public function getIndentLeft()

    {

        return $this->\_indentLeft;

    }

    // 设置左缩进值

    public function setIndentLeft($pValue = null)

    {

        $this->\_indentLeft = $pValue;

        return $this;

    }

    // 获取右缩进值

    public function getIndentRight()

    {

        return $this->\_indentRight;

    }

    // 设置右缩进值

    public function setIndentRight($pValue = null)

    {

        $this->\_indentRight = $pValue;

        return $this;

    }

    // 首行缩进相关方法

    public function setIndentFirstLine($pValue = null)

    {

        $this->\_indentFirstLine = $pValue;

        return $this;

    }

    public function getIndentFirstLine()

    {

        return $this->\_indentFirstLine;

    }

    public function setIndentFirstLineChars($pValue = null)

    {

        $this->\_indentFirstLineChars = $pValue;

        return $this;

    }

    public function getIndentFirstLineChars()

    {

        return $this->\_indentFirstLineChars;

    }

2、/PHPword/Writer/Word2007/Base.php，在\_writeParagraphStyle方法中大约第117行添加

                $indentLeft = $style->getIndentLeft();

$indentRight = $style->getIndentRight();

$indentFirstLine = $style->getIndentFirstLine();

$indentFirstLineChars = $style->getIndentFirstLineChars();

然后同文件大约第156行

if (!is\_null($indentLeft) || !is\_null($indentRight) || !is\_null($indentFirstLine) || !is\_null($indentFirstLineChars))

{

$objWriter->startElement('w:ind');

if (!is\_null($indentLeft))

{

$objWriter->writeAttribute('w:left', $indentLeft);

}

if (!is\_null($indentRight))

{

$objWriter->writeAttribute('w:right', $indentRight);

}

if (!is\_null($indentFirstLine))

{

$objWriter->writeAttribute('w:firstLine', $indentFirstLine);

}

if (!is\_null($indentFirstLineChars))

{

$objWriter->writeAttribute('w:firstLineChars', $indentFirstLineChars);

}

$objWriter->endElement();

}

这样就可以在addtext时在段落样式中用缩进属性了

$section->addText('test', array('bold' => true), array('indentLeft' => 1440));

注意计量单位，首行缩进2个字符，设置indentFirstLineChars的值是 200而不是2

## 表格对齐和表格缩进

/PHPWord/Style/TableFull.php添加相关属性和方法

// 表格居中属性，用法：表格style数组中：'alignMent' => 'center'

private $\_alignMent = null;

// 表格缩进属性

private $\_tableIndent = null;

    public function setTableAlign($pValue = null)

    {

        $this->\_alignMent = $pValue;

    }

    public function getTableAlign()

    {

        return $this->\_alignMent;

    }

    public function getTableIndent()

    {

        return $this->\_tableIndent;

    }

    public function setTableIndent($pValue = null)

    {

        $this->\_tableIndent = $pValue;

        return $this;

    }

构造函数中添加

unset($this->\_firstRow->\_alignMent);

unset($this->\_firstRow->\_tableIndent);

/PHPWord/Writer/Word2007/Styles.php大约144行\_writeFullTableStyle方法内添加

$tableAlign = $style->getTableAlign();

$tableIndent = $style->getTableIndent();

同文件$objWriter->startElement('w:tblPr');下面

if ($tableAlign) {

$objWriter->startElement('w:jc');

$objWriter->writeAttribute('w:val', $tableAlign);

$objWriter->endElement();

}

if (!is\_null($tableIndent))

{

$objWriter->startElement('w:tblInd');

if (!is\_null($tableIndent))

{

$objWriter->writeAttribute('w:w', $tableIndent);

$objWriter->writeAttribute('w:type', 'dxa');

}

$objWriter->endElement(); // w:ind

}

这样就可以设置表格对齐方式和缩进了

$styleTable = array('borderSize'=>6, 'alignMent' => 'right', 'tableInden' => 1440);

$styleFirstRow = array('bgColor'=>'#1F497D');

$PHPWord->addTableStyle('tableStyle', $styleTable, $styleFirstRow);

$table = $section->addTable('tableStyle');

2、单元格合并

/PHPWord/Style/Cell.php添加属性和方法

private $\_rowMerge = null;

private $\_cellMerge = null;

public function getRowMerge()

{

return $this->\_rowMerge;

}

public function setRowMerge($pValue = null)

{

$this->\_rowMerge = $pValue;

return $this;

}

public function getCellMerge()

{

return $this->\_cellMerge;

}

public function setCellValue($pValue = null)

{

$this->\_cellMerge = $pValue;

return $this;

}

 /PHPWord/Writer/Word2007/base.php中\_writeCellStyle方法添加

$rowMerge = $style->getRowMerge();

$cellMerge = $style->getCellMerge();

在同方法中修改$styles(感谢网友提出，还真忘了加上这个)，通过这个才能进入if ($styles)代码块里面:

$styles = (!is\_null($bgColor) || !is\_null($valign) || !is\_null($textDir)

|| $borders || !is\_null($rowMerge) || !is\_null($cellMerge)) ? true : false;

在同方法if ($styles)中添加

if (!is\_null($cellMerge))

{

//$objWriter->startElement('w:gridSpan');

$objWriter->startElement('w:hMerge');

if ((string)$cellMerge !== 'continue')

{

$objWriter->writeAttribute('w:val', $cellMerge);

}

$objWriter->endElement();

}

if (!is\_null($rowMerge))

{

$objWriter->startElement('w:vMerge');

if ((string)$rowMerge !== 'continue')

{

$objWriter->writeAttribute('w:val', $rowMerge);

}

$objWriter->endElement();

}

使用方法：

$table->addRow(400);

$table->addCell(1600, array('cellMerge' => 'restart', 'valign' => "center"))->addText('横向合并');

$table->addCell(1600, array('cellMerge' => 'continue'));

$table->addCell(1600, array('cellMerge' => 'continue'));

$table->addCell(1600, array('cellMerge' => 'continue'));

附调试心得：如果生成的word文件和自己预想的样式不太一致，可以直接用解压缩软件如rar、zip打开word文件解压出来，直接查看里面的相关文件。

## 图片缩进和绝对相对悬浮定位

如果不修改代码，也可以通过表格与图片配合进行功能实现

/Style/image.php添加以下属性和方法

// 图片缩进+绝对定位

private $\_indentLeft = null;

private $\_indentRight = null;

private $\_position = null;

private $\_top = null;

private $\_left = null;

private $\_zIndex = null;

public function getIndentLeft()

    {

        return $this->\_indentLeft;

    }

    public function setIndentLeft($pValue = null)

    {

        $this->\_indentLeft = $pValue;

        return $this;

    }

    public function getIndentRight()

    {

        return $this->\_indentRight;

    }

    public function setIndentRight($pValue = null)

    {

        $this->\_indentRight = $pValue;

        return $this;

    }

    public function getPosition()

    {

        return $this->\_position;

    }

    public function setPosition($pValue = null)

    {

        $this->\_position = $pValue;

        return $this;

    }

    public function getTop()

    {

        return $this->\_top;

    }

    public function getLeft()

    {

        return $this->\_left;

    }

    public function setTop($pValue = null)

    {

        $this->\_top = $pValue;

        return $this;

    }

    public function setLeft($pValue = null)

    {

        $this->\_left = $pValue;

        return $this;

    }

    public function getZIndex()

    {

        return $this->\_zIndex;

    }

    public function setZIndex($pValue = null)

    {

        $this->\_zIndex = $pValue;

        return $this;

    }

base.php方法修改为如下方法

protected function \_writeImage(PHPWord\_Shared\_XMLWriter $objWriter = null, $image, $is\_textrun = false) {

$rId = $image->getRelationId();

$style = $image->getStyle();

$width = $style->getWidth();

$height = $style->getHeight();

$align = $style->getAlign();

$indentLeft = $style->getIndentLeft();

$indentRight = $style->getIndentRight();

$position = $style->getPosition();

$top = $style->getTop();

$left = $style->getLeft();

$zIndex = $style->getZIndex();

if (!$is\_textrun)

{

$objWriter->startElement('w:p');

}

if(!is\_null($align) || !is\_null($indentLeft) || !is\_null($indentRight)) {

$objWriter->startElement('w:pPr');

if (!is\_null($align))

{

$objWriter->startElement('w:jc');

$objWriter->writeAttribute('w:val', $align);

$objWriter->endElement(); // w:jc

}

if (!is\_null($indentLeft) || !is\_null($indentRight))

{

$objWriter->startElement('w:ind');

if (!is\_null($indentLeft))

{

$objWriter->writeAttribute('w:left', $indentLeft);

}

if (!is\_null($indentRight))

{

$objWriter->writeAttribute('w:right', $indentRight);

}

$objWriter->endElement(); // w:ind

}

$objWriter->endElement(); // w:pPr

}

$objWriter->startElement('w:r');

$objWriter->startElement('w:pict');

$objWriter->startElement('v:shape');

$objWriter->writeAttribute('type', '#\_x0000\_t75');

if (!empty($position) && $top !== null && $left !== null && $zIndex !== null)

{

$objWriter->writeAttribute('style',

'width:'.$width.'px;height:'.$height.'px;position:'.$position.';top:'.$top.'px;left:'.$left.'px;z-index:'.$zIndex.';');

}

else

{

$objWriter->writeAttribute('style', 'width:'.$width.'px;height:'.$height.'px');

}

$objWriter->startElement('v:imagedata');

$objWriter->writeAttribute('r:id', 'rId'.$rId);

$objWriter->writeAttribute('o:title', '');

$objWriter->endElement();

$objWriter->endElement(); // v:shape

$objWriter->endElement();// w:pict

$objWriter->endElement(); // w:r

if (!$is\_textrun)

{

$objWriter->endElement(); // w:p

}

}

同文件\_writeTextRun方法中修改

if(count($elements) > 0) {

foreach($elements as $element) {

if($element instanceof PHPWord\_Section\_Text) {

$this->\_writeText($objWriter, $element, true);

} elseif($element instanceof PHPWord\_Section\_Link) {

$this->\_writeLink($objWriter, $element, true);

} elseif ($element instanceof PHPWord\_Section\_Image) {

$this->\_writeImage($objWriter, $element,true);

}

}

}

调用方法

$tmp\_textrun = $section->createTextRun(array('indentLeft' => 2600));

$tmp\_textrun->addText("段落", array('size' => 11), array('indentLeft' => 0));

$tmp\_textrun->addImage('./test.jpg', array('position' => 'absolute', 'top' => 0, 'left' => 25, 'zIndex' => 4));

2、使用PHPWord循环生成word文件样式bug问题。

在项目中需要循环生成不同的word文件，有个很奇怪的问题就是除了第一个生成的以外后面生成的word表格样式会有各种bug，和写完预期的不一样。

这是因为存储表格样式用的是一个静态变量，当重新实例化PHPWord时没有重置此变量引起的。通过以下代码可修复此问题，如果没有循环生成不同样式word需求可不做修改。

/PHPWord/Style.php添加一个静态变量和一个静态方法

private static $\_initElements = array();

    /\*\*

     \* clear the $\_styleElements static property

     \*/

    public static function clearStyles() {

       //self::$\_styleElements = array();

       self::$\_styleElements = self::$\_initElements;

    }

/PHPWord/PHPWord.php构造方法中添加一行

PHPWord\_Style::clearStyles();