

putpdf pagebreak — Add breaks to a PDF file

[Description](#) [Quick start](#) [Syntax](#) [Options](#)
[Remarks and examples](#) [Also see](#)

Description

`putpdf pagebreak` adds a page break to the document, placing subsequent content on the next page of the document.

`putpdf sectionbreak` adds a new section to the active document that starts on the next page. It lets you vary the page size, orientation, margins, and other properties of the pages within a single document. This formatting of sections is most useful when you want to mix portrait and landscape layouts.

Quick start

Add a page break to the document

```
putpdf pagebreak
```

Begin a new section with a landscape layout in the document

```
putpdf sectionbreak, landscape
```

As above, and specify 1 inch margins on the left and right side of the page

```
putpdf sectionbreak, landscape margin(left,1) margin(right,1)
```

Syntax

Add page break to document

```
putpdf pagebreak
```

Add section break to document

```
putpdf sectionbreak [ , section_options ]
```

<i>section_options</i>	Description
<code>pagesize(<i>psize</i>)</code>	set page size of section
<code>landscape</code>	set section orientation to landscape
<code>font(<i>fspec</i>)</code>	set font, font size, and font color
<code>halign(<i>hvalue</i>)</code>	set horizontal alignment of section
<code>margin(<i>type</i>, # [<i>unit</i>])</code>	set page margins of section
<code>bgcolor(<i>color</i>)</code>	set background color

Options

`pagesize(psize)` sets the page size of the section. *psize* may be `letter`, `legal`, `A3`, `A4`, `A5`, `B4`, or `B5`. The default is `pagesize(letter)`.

`landscape` changes the section orientation from portrait (the default) to landscape.

`font(fontname [, size [, color]])` sets the font, font size, and font color for the document.

fontname may be any supported font installed on the user's computer. Base 14 fonts, Type 1 fonts, and TrueType fonts with an extension of `.ttf` and `.ttc` are supported. TrueType fonts that cannot be embedded may not be used. If *fontname* includes spaces, then it must be enclosed in double quotes. The default font is Helvetica.

size is a numeric value that represents font size measured in points. The default is 11.

color sets the text color. *color* may be one of the colors listed in [Colors of \[RPT\] Appendix for putpdf](#); a valid RGB value in the form `### ## #`, for example, `171 248 103`; or a valid RRGGBB hex value in the form `#####`, for example, `ABF867`.

The font size and font color may be specified individually without specifying *fontname*. Use `font("", size)` to specify font size only. Use `font("", "", color)` to specify font color only. For both cases, the default font will be used.

`halign(hvalue)` sets the horizontal alignment of the paragraphs, images, and tables within the section. *hvalue* may be `left`, `right`, or `center`. The default is `halign(left)`.

`margin(type, # [unit])` sets the page margins of the document. This option may be specified multiple times in a single command to account for different margin settings.

type identifies the location of the margin inside the document. *type* may be `top`, `left`, `bottom`, or `right`.

unit may be `in` (inch), `pt` (point), `cm` (centimeter), or `twip` (twentieth of a point). An inch is equivalent to 72 points, 2.54 centimeters, or 1440 twips. The default is `in`.

`bgcolor(color)` sets the background color for the document. *color* may be one of the colors listed in [Colors of \[RPT\] Appendix for putpdf](#); a valid RGB value in the form `### ## #`, for example, `171 248 103`; or a valid RRGGBB hex value in the form `#####`, for example, `ABF867`.

Remarks and examples

stata.com

The `putpdf pagebreak` and `putpdf sectionbreak` commands are useful for organizing your PDF document. Whether you wish to insert page breaks for each new section in your file or you want to format different segments of your document differently, you can do this with these two commands.

To begin all subsequently added content on the next page in the active document, use the `putpdf pagebreak` command. This may be useful, for instance, when you do not want tables or paragraphs split across pages.

You can vary the page layout for sections of your document by using `putpdf sectionbreak`. When inserting a section break, the content that follows will be placed on the following page and formatted according to the options you specify with `putpdf sectionbreak`. You can modify the page size, orientation, font, and other features for each section. This is particularly useful if you have content in your document that varies in dimensions because you can specify a different layout for each element. For example, you may want a portrait layout for a long estimation table and a landscape layout for a wide image.

▷ Example 1: Add section break to PDF file

Suppose we have created a PDF file with the default portrait layout and added some content.

```
putpdf begin
putpdf paragraph
...
```

We now wish to add a wide image that does not fit properly in the confines of the current layout. We can begin a new section with a `landscape` layout and specify quarter-inch margins on the left and right side of the page.

```
putpdf sectionbreak, landscape margin(left,0.25) margin(right,0.25)
```

We can now export the image created in [example 2](#) of [\[RPT\] putpdf paragraph](#) with a much larger width.

```
putpdf paragraph
putpdf image scatter.png, width(11) height(7)
```

To continue adding content on the following page, retaining the `landscape` layout and small margins, we type

```
putpdf pagebreak
```

Inserting a page break simply places the following content on the next page. Any formatting options specified with `putpdf sectionbreak` will still be applied to the following pages.

To conclude this section and resume our document with a portrait layout and the default margin size, we type

```
putpdf sectionbreak
```

◀

Also see

- [\[RPT\] putpdf intro](#) — Introduction to generating PDF files
- [\[RPT\] putpdf begin](#) — Create a PDF file
- [\[RPT\] putpdf paragraph](#) — Add text or images to a PDF file
- [\[RPT\] putpdf table](#) — Add tables to a PDF file
- [\[RPT\] Appendix for putpdf](#) — Appendix for putpdf entries