

Documentation `tikz-tabular`

Version 0.1 – May 18, 2025

English version

Stéphane Pasquet

Sommaire

1	Introduction	2
1.1	Purpose of this package	2
1.2	Python	2
2	Installation	2
3	Usage	3
3.1	A Basic Example	3
3.2	The Corresponding \LaTeX Code	3
3.3	Syntax	4
3.3.1	The <code>\globalpath</code> Variable	4
3.3.2	The <code>tikztabular</code> Environment	4
3.3.3	The Table Title (Optional)	4
3.3.4	The Table Column Headers (Optional)	4
3.3.5	The Column Footers (Optional)	5
3.3.6	The First Columns	5
3.3.7	The Last Columns	5
4	Options	6
5	F.A.Q.	7
5.1	Why define the environment in a scope?	7
5.2	Why are options preceded by a chevron?	7
5.3	Why declare the <code>\globalpath</code> variable?	7

1 Introduction

1.1 Purpose of this package

This package allows creating a table in graphical mode (based on TiKZ) using a simplified syntax.

1.2 Python

It is necessary to have Python installed.

If it is not already done, simply go to the page:

<https://www.python.org/downloads/>

and download the latest version.

Windows Users: Ensure that the path to `python.exe` is included in the Windows system PATH (if you are using this operating system).

environment variables > system variables > path to `python.exe`

The path is of the form:

`C:\Users\<you>\AppData\Local\Programs\Python\Python3xx`

2 Installation

If you perform a manual installation of this package, you must respect the \TeX directory structure. If, for example, your manually installed packages are in:

`c:\texmf\`

you will need to ensure you have:

`c:\texmf\doc\tikz-tabular\`

`c:\texmf\latex\tikz-tabular\`

Important Note: After placing the files in the correct folders, it is necessary to change a line in the `tikz-tabular.sty` file:

```
22 \immediate\write18{python "c:/texmf/tex/latex/tikz-tabular
/tikztabular.py" num-\thetikz@tabular-dir-\globalpath}%
```

This line must contain the exact path to the Python file in the same directory.

Windows Users and MikTeX: Refresh the database once the package is installed. In the MikTeX Console,

Tasks > Refresh file name database

3 Usage

3.1 A Basic Example

My Table				
Header 1	Header 2	Header 3	Header 4	
Cell 1.1	Cell 1.2	Cell 1.3	Cell 1.4	Cell 1.5
Cell 2.1	Cell 2.2	Cell 2.3	Cell 2.4	Cell 2.5
Foot 1	Foot 2	Foot 3	Foot 4	

3.2 The Corresponding L^AT_EX Code

Minimal Code

```
\documentclass{article}
\usepackage{tikz-tabular}
\newcommand{\globalpath}{"c:/texmf/doc/tikz-
tabular/"}

\setlength{\parindent}{0pt}

\begin{document}

\begin{tikzpicture}
\begin{tikztabular}
<columns=5
[tabulartitle]
\fontfamily{lmss}\selectfont\Large\textbf{My Table}
[endtabulartitle]

[tabularhead]
\textbf{Header 1} & \textbf{Header 2} & \textbf{Header
3} & \textbf{Header 4} &
[endtabularhead]

Cell 1.1 & Cell 1.2 & Cell 1.3 & Cell 1.4 & Cell 1.5
Cell 2.1 & Cell 2.2 & Cell 2.3 & Cell 2.4 & Cell 2.5

[tabularfoot]
\textbf{Foot 1} & \textbf{Foot 2} & \textbf{Foot 3} &
\textbf{Foot 4} &
\end{tikztabular}
\end{tikzpicture}

\end{document}
```

3.3 Syntax

3.3.1 The \globalpath Variable

It is necessary to declare the path of the current file.
Without this, the auxiliary files built for the tables will not be created.

Minimal Code

```
\newcommand{\globalpath}{"c:/texmf/doc/tikz-tabular/"}
```

3.3.2 The tikztabular Environment

Minimal Code

```
\begin{tikzpicture}  
\begin{tikztabular}  
...  
\end{tikztabular}  
\end{tikzpicture}
```

It must be within a tikzpicture environment. Indeed, it is built within a scope environment.

3.3.3 The Table Title (Optional)

Minimal Code

```
[tabulartitle]  
My Title  
[endtabulartitle]
```

It allows declaring that the table has a title. It is displayed across the entire width of the table.

3.3.4 The Table Column Headers (Optional)

Minimal Code

```
[tabularhead]  
Headers separated by "&"  
[endtabularhead]
```

This allows displaying the column headers differently from other cells.

3.3.5 The Column Footers (Optional)

Minimal Code

```
[tabularfoot]  
Footers separated by "&"
```

This allows displaying the column footers differently from other cells.
Since the footers are supposed to be at the end of the table, there is no `[endtabularfoot]` command.

3.3.6 The First Columns

My Table

Cell 1.1

Cell 1.2

Cell 1.3

Cell 1.4

Cell 1.5

Cell 2.1

Cell 2.2

Cell 2.3

Cell 2.4

Cell 2.5

Minimal Code

```
\begin{tikztabular}  
<columns=5  
<firstcolumncolor=orange!75!black  
[tabulartitle]  
My Table  
[endtabulartitle]  
Cell 1.1 & Cell 1.2 & Cell 1.3 & Cell 1.4 & Cell 1.5  
Cell 2.1 & Cell 2.2 & Cell 2.3 & Cell 2.4 & Cell 2.5  
\end{tikztabular}
```

3.3.7 The Last Columns

Cell 1.1

Cell 1.2

Cell 1.3

Cell 1.4

Cell 1.5

Cell 2.1

Cell 2.2

Cell 2.3

Cell 2.4

Cell 2.5

Minimal Code

```
\begin{tikztabular}  
<columns=5  
<lastcolumncolor=orange!75!black  
Cell 1.1 & Cell 1.2 & Cell 1.3 & Cell 1.4 & Cell 1.5  
Cell 2.1 & Cell 2.2 & Cell 2.3 & Cell 2.4 & Cell 2.5  
\end{tikztabular}
```

4 Options

Options should be placed in the `tikztabular` environment *before* the content of the cells.

They must be placed *necessarily* in the form:

`<option = value` (with or without space but with the chevron)

The Options		
Option Name	Designation	Example
<code>columns</code>	number of columns	<code>columns = 5</code>
<code>corners</code>	size of rounded corners	<code>corners = 1mm</code>
<code>colorcell</code>	color of cells	<code>colorcell = red</code>
<code>tabularwidth</code>	width of the table	<code>tabularwidth = 20cm</code>
<code>cellheight</code>	height of cells	<code>cellheight = 2em</code>
<code>marginh</code>	horizontal margin between two columns	<code>marginh = 1em</code>
<code>marginv</code>	vertical margin between two lines	<code>marginv = 1em</code>
<code>marginvtitle</code>	vertical margin between the title and the next line	<code>marginvtitle = 1em</code>
<code>tabulartitlecolor</code>	background color of the title	<code>tabulartitlecolor = black</code>
<code>texttitlecolor</code>	color of the title	<code>texttitlecolor = white</code>
<code>firstcolumncolor</code>	color of the first column	<code>firstcolumncolor = black</code>
<code>lastcolumncolor</code>	color of the last column	<code>lastcolumncolor = black</code>
<code>tabularheadcolor</code>	background color of headers	<code>tabularheadcolor = orange</code>
<code>textheadcolor</code>	color of header text	<code>textheadcolor = white</code>
<code>tabularfootcolor</code>	background color of footers	<code>tabularfootcolor = white</code>
<code>textfootcolor</code>	color of footer text	<code>textfootcolor = white</code>
<code>textcolor</code>	color of cell text	<code>textcolor = black</code>

5 F.A.Q.

5.1 Why define the environment in a scope?

I wanted to be able to act on the table, for example by adding arrows.
Each cell has a name in the form:

```
(cell-<row>-<column>)
```

Thus, to join two cells with an arrow, you can write:

```
\draw[->,>=latex] (cell-1-1) to[bend left=10] (cell-3-4);
```

5.2 Why are options preceded by a chevron?

Once the processing of options by Python is complete, the program deletes all lines containing the option names.

Imagine that the options are written without any special symbol (like the chevron).
If a cell in the table contains a word like "columns" (which is an option), the line containing the word is deleted...

I could have adopted another logic and put, for example, `[start]` to indicate that the table begins, but this seemed more restrictive.

5.3 Why declare the `\globalpath` variable?

\TeX cannot transmit (to my knowledge) the path of the compiled file.
However, this path is necessary for the Python program.
Therefore, this variable must be defined manually.