

# Introduction to $\text{\LaTeX}$

## Lecture 4: Use Graphics and Tables in $\text{\LaTeX}$

Liu Yihao

SJTU-UMJI Technology Department

March 6, 2018

# Include graphs

It's very useful to include graphs in  $\text{\LaTeX}$ , especially in report and paper writing. Here is a common template of including a single floating graph.

## Command

```
\usepackage{graphicx}
\begin{figure}[position]
  \centering
  \includegraphics[options]{file}
  \caption{caption}
  \label{label}
\end{figure}
```

- **file** - the filename or relative path of the graph you want to insert, usually placed in the same directory as the tex file
- **position** - we usually use **htbp** here, which will be introduced later in this chapter
- **options** - the width, height and other options about the graph
- **caption** - the caption displayed above/under the graph
- **label** - used for references in a document (will be introduced later)

Usually you need to optimize the size and some other properties of the graph, most of them can be set in `options`. (Only some useful options are listed here)

- `height` - use any  $\text{\LaTeX}$  measuring unit.
- `width` - use any  $\text{\LaTeX}$  measuring unit.
- `scale` - scale the graph to this proportion
- `angle` - rotate the graph in anti-clockwise by this angle

## Example

```

1  \usepackage{graphicx}
2  \begin{figure}[htbp]
3      \centering
4      \includegraphics[width=0.5
   ↪ \linewidth]{sample.jpg}
5      \caption{Marshmallow}
6      \label{fig-sample}
7  \end{figure}

```



Figure: Marshmallow

# Include multiple graphs

**Subfigure** package can be used to include a series of graphs.

## Example

```

1  \begin{figure}[htbp]
2      \centering
3      \subfigure[1]{
4          \includegraphics[width=0.4
5              ↪ \linewidth]{sample-1.jpg}
6          \label{fig-sample-1}
7      }
8      \subfigure[2]{
9          \includegraphics[height=0.4
10             ↪ \linewidth]{sample-2.jpg}
11         \label{fig-sample-2}
12     }
13     % more subfigures
14     \caption{Marshmallows}
15     \label{fig-entire}
16 \end{figure}

```



(a) 1

(b) 2



(c) 3



(d) 4

**Figure:** Marshmallows

# Draw tables

Table is another common element in  $\text{\LaTeX}$ , for example, there is a simple table like this:

## Example

```

1 \begin{tabular}{|l|c|r|}
2   \hline
3   Title 1 & Title 2 & Title 3 \\
4   \hline
5   1 & 2 & 3 \\
6   \hline
7 \end{tabular}

```

Title 1	Title 2	Title 3
1	2	3

## Command

```
\begin{tabular}{format}
```

...

```
\end{tabular}
```

`format` can be set as follow

- `|` - represents a vertical separate line between two columns
- `l` - align left in this column
- `c` - align center in this column
- `r` - align right in this column

## Example

`|||`

Title 1	Title 2	Title 3
1	2	3

`||c|cc|`

Title 1	Title 2	Title 3
1	2	3

How to arrange cells in `tabular` environment is very similar to the `align` environment. However, we usually need horizontal lines in tables.

## Command

```
\hline = \cline{1-max_col}
\cline{start-end}
```

## Example

```
1 \begin{tabular}{c|l|c|r}
2   \hline
3   \hline
4   & Title 1 & Title 2 &
   \hookrightarrow Title 3 \\
5   \cline{2-4}
6   Table & 1 & 2 & 3 \\
7   \cline{2-4}
8   & 4 & 5 & 6 \\
9   \hline
10  \hline
11 \end{tabular}
```

	Title 1	Title 2	Title 3
Table	1	2	3
	4	5	6

Here we draw a table with a multirow, but it only works with multirows of odd row number. A more convenient method of drawing multirows will be introduced.

# Multicolumn and Multirow

## Command

`\multicolumn{ncols}{format}{text}`

- `ncols` - the number of columns to be merged
- `format` - the format of the merged column, excluding the left — (eg. `c|`)
- `text` - the text in the merged column

`\usepackage{multirow}`

`\multirow{nrows}{width}[fixup]{text}`

- `nrows` - the number of rows to be merged
- `width` - the width of the merged rows (use `*` for auto)
- `fixup` - the vertical position of the text (optional, default in the center)
- `text` - the text in the merged row



## Example

```

1  \begin{tabular}{|c|c|c|c|c|}
2      \hline
3      \multirow{4}{*}{Table} & Title 1 & Title 2 & Title 3 & Title 4
      ↪  \\
4      \cline{2-5}
5      & \multicolumn{2}{c|}{Text 1} & & 
      ↪  \multicolumn{2}{c|}{\multirow{3}{*}{Text 3}} \\
6      \cline{2-3}
7      & \multicolumn{2}{c|}{Text 2} & \multicolumn{2}{c|}{} \\
8      \cline{2-3}
9      & Text 4 & Text 5 & \multicolumn{2}{c|}{} \\
10     \hline
11     \end{tabular}

```

Table	Title 1	Title 2	Title 3	Title 4
	Text 1		Text 3	
	Text 2			
	Text 4	Text 5		

# Easy ways to create a table

With `multirow` and `multicolumn`, we can almost draw tables of any style, but this coding process can never be as easy as the graphic one, like making tables in Word or Excel. Is there any ways to convert graphic tables into  $\text{\LaTeX}$  codes directly?

- Use  $\text{\LaTeX}$ Table Generator: <http://www.tablesgenerator.com/>
- Excel2latex:  
<https://ctan.org/tex-archive/support/excel2latex/>

# Table environment

A `table` environment is used to arrange the place of a tabular, similar to the `figure` environment

## Command

```
\begin{table(*)}[position]
    \centering
    \begin{tabular}{format}
        ...
    \end{tabular}
    \caption{caption}
    \label{label}
\end{table(*)}
```

The `position`, `caption`, `label` are same as those in `figure` environment.

# About htbp

The htbp order is an official order of displaying graphs and tables.

- **h** - insert to the current place
- **t** - insert to the top of the page
- **b** - insert to the bottom of the page
- **p** - insert to a new page, which is common in dealing with big graphs and tables.

$\text{\LaTeX}$  compiler will try these methods from left to right as you defined. Usually, we use htbp so that it will try to put the graph or table in the current place. If fails, then it will try the top, the bottom, and the next page until success.

We usually want to place the graphs or tables just below or above the content where we mention them, but even when we type `[h]` in position, you can not ensure that it will appear at the ideal position, and there are several methods to make up for this.

- Change `[h]` to `![h]`
- Change `[h]` to `[H]`
- Use `\newpage` to move the following content to the next page

However, the first two methods still cannot ensure the position of the table or graph, so what is the best way to solve this?

# Reference

To solve the problem mentioned in the last page, we can use reference. You may remember the `\label` command used in equations, graphs and tables, they are used for reference in other parts of the document.

## Command

`\ref{label}`

## Example

1	Figure <code>\ref{fig-sample-1}</code> \\	Figure 2(a)
2	Figure <code>\ref{fig-sample}</code>	Figure 1

Once the position of these figures are changed, or some more figures are added between them, the number of them will change, but there label won't. So  $\text{\LaTeX}$  will automatically generate the correct number for them and you don't need to modify them again and again.

# The array environment

When you use `tabular` in maths environment, the text format in the `tabular` won't be italic. However, there is a replacement of `tabular`, which is `array` environment.

## Command

```
\begin{array}{format}
...
\end{array}
```

The properties and usages of these two environment are exactly the same.

Note that there is also a package called `array`, which is an enhancement of both `tabular` and `array`, you may use `texdoc array` to learn about it.