If you want to emphasize a longer passage, and you wish to have it appear as a separate paragraph, you should use one of the standard environments of LaTeX. These are used as follows:

```
\begin{environment-name}
      some text goes here
\end{environment-name}
```

For example, to set a whole paragraph in emphasized text, use:

```
\begin{em}
      some text goes here
\end{em}
```

Notice that the backslash is not used in front of the command `em` when the command is used as the argument in the `\begin` command.

For Centered Text

```
\begin{center}
Text on line 1 \\
Text on line 2 \\
.
.
.
\end{center}
```

The `center` environment allows you to create a paragraph consisting of lines that are centred within the left and right margins on the current page. Each line must be terminated with the string `\\`, which is the short form of the command `\newline`, and gives a line which is not justified to the right.

Environments for Listing Text

There are three environments for listing text.

For numbered lists

```
\begin{enumerate}
\item First item
\item Second item
.
.
.
\end{enumerate}
```

The `enumerate` environment produces a numbered list. Enumerations can be nested within one another, up to four levels deep. They can also be nested within other paragraph-making environments.

Each item of an enumerated list begins with an `\item` command. There must be at least one `\item` command within the environment.

Labelled lists

```
\begin{description}
\item [label] First item
\item [label] Second item
.
.
.
\end{description}
```

The `description` environment is used to make labelled lists. The `label` is bold face and flushed right.

Bulleted lists

```
\begin{itemize}
\item First item
\item Second item
.
.
.
\end{itemize}
```

The `itemize` environment produces a "bulleted" list. Itemizations can be nested within one another, up to four levels deep. They can also be nested within other paragraph-making environments

Bold, italics and underlining

```
Some of the \textbf{greatest}
discoveries in \underline{science}
were made by \textbf{\textit{accident}}.
```

Some of the **greatest** discoveries in science were made by ***accident***.

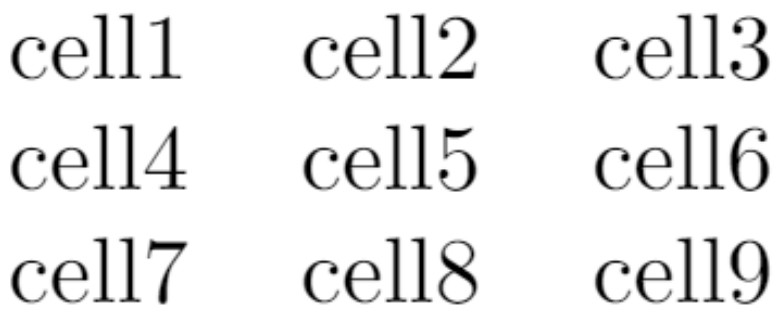
Tables

Introduction

This article explains how to use LaTeX to create and customize tables: changing size/spacing, combining cells, applying colour to rows or cells, and so forth.

We can start with one of the simplest examples of a table:

```
\begin{center}
\begin{tabular}{ c c c }
 cell1 & cell2 & cell3 \\
 cell4 & cell5 & cell6 \\
 cell7 & cell8 & cell9
\end{tabular}
\end{center}
```



The `tabular` environment is the default LaTeX method to create tables. You must specify a parameter to this environment; here we use `{c c c}` which tells LaTeX that there are three columns and the text inside each one of them must be **centred**.

Creating a simple table in LaTeX

The `tabular` environment provides additional flexibility; for example, you can put separator lines in between each column:

```
\begin{center}
\begin{tabular}{ |c|c|c| }
 \hline
 cell1 & cell2 & cell3 \\
 cell4 & cell5 & cell6 \\
 cell7 & cell8 & cell9 \\
 \hline
\end{tabular}
\end{center}
```

Open this example in Overleaf.

cell1	cell2	cell3
cell4	cell5	cell6
cell7	cell8	cell9

Here is a description of the constructions used in the example above:

`{ |c|c|c| }`

> This declares that three columns, separated by a vertical line, are going to be used in the table. Each `c` means that the contents of the column will be centred. You can also use `r` to align the text to the right and `l` for left alignment.

`\hline`

> This will insert a horizontal line on top of the table and at the bottom too. There is no restriction on the number of times you can use `\hline`.

`cell1 & cell2 & cell3 \\`

> Each `&` is a cell separator and the double-backslash `\\` sets the end of this row.

Below you can see a second example which uses various vertical and horizontal lines (often called "rules"):

```
\begin{center}
\begin{tabular}{||c c c c||}
 \hline
 Col1 & Col2 & Col2 & Col3 \\ [0.5ex]
 \hline\hline
 1 & 6 & 87837 & 787 \\
 \hline
 2 & 7 & 78 & 5415 \\
 \hline
 3 & 545 & 778 & 7507 \\
 \hline
 4 & 545 & 18744 & 7560 \\
 \hline
 5 & 88 & 788 & 6344 \\ [1ex]
 \hline
\end{tabular}
\end{center}
```

Open this example in Overleaf.

Col1	Col2	Col2	Col3
1	6	87837	787
2	7	78	5415
3	545	778	7507
4	545	18744	7560
5	88	788	6344

Tables with fixed length

When formatting a table you might require a fixed length either for each column or for the entire table. The following example adds the `array` package to document preamble:

```
\usepackage{array}
```

and uses it to set fixed column widths:

```
\documentclass{article}
\usepackage{array}
\begin{document}
\begin{center}
\begin{tabular}{ | m{5em} | m{1cm}| m{1cm} | }
  \hline
  cell1 dummy text dummy text dummy text& cell2 & cell3 \\
  \hline
  cell1 dummy text dummy text dummy text & cell5 & cell6 \\
  \hline
  cell7 & cell8 & cell9 \\
  \hline
\end{tabular}
\end{center}
\end{document}
```

Open this example in Overleaf.

cell1 dummy text dummy text dummy text	cell2	cell3
cell1 dummy text dummy text dummy text	cell5	cell6
cell7	cell8	cell9

Combining rows and columns

Rows and columns can be merged to create larger table cells. The following example uses the `\multicolumn` command to merge several columns:

```
\documentclass{article}
\usepackage{multirow}
\begin{document}
\begin{tabular}{ |p{3cm}||p{3cm}|p{3cm}|p{3cm}|  }
 \hline
 \multicolumn{4}{|c|}{Country List} \\
 \hline
 Country Name or Area Name& ISO ALPHA 2 Code &ISO ALPHA 3
Code&ISO numeric Code\\
 \hline
 Afghanistan   & AF    &AFG&   004\\
 Aland Islands&   AX  & ALA   &248\\
 Albania &AL & ALB&  008\\
 Algeria    &DZ & DZA&  012\\
 American Samoa&   AS  & ASM&016\\
 Andorra& AD  & AND   &020\\
 Angola& AO  & AGO&024\\
```

```
 \hline
\end{tabular}
\end{document}
```

Open this example in Overleaf.

Country List			
Country Name or Area Name	ISO ALPHA 2 Code	ISO ALPHA 3 Code	ISO numeric Code
Afghanistan	AF	AFG	004
Aland Islands	AX	ALA	248
Albania	AL	ALB	008
Algeria	DZ	DZA	012
American Samoa	AS	ASM	016
Andorra	AD	AND	020
Angola	AO	AGO	024

Let's review each component of the command `\multicolumn{4}{|c|}{Country List} \\`:

`{4}`

The number of columns to be combined: 4 in this case.

`{|c|}`

Delimiters and alignment of the resulting cell: in this case the text will be centred and a vertical line will be drawn at each side of the cell.

`{Country List}`

Text to be displayed inside the cell.

To combine rows you need to add the `multirow` package to your document preamble:

```
\usepackage{multirow}
```

You can then use the `\multirow` command in your document:

```
\documentclass{article}
\usepackage{multirow}
\begin{document}
\begin{center}
\begin{tabular}{ |c|c|c|c| }
\hline
col1 & col2 & col3 \\
\hline
\multirow{3}{4em}{Multiple row} & cell2 & cell3 \\
& cell5 & cell6 \\
& cell8 & cell9 \\
\hline
\end{tabular}
\end{center}
\end{document}
```

Open this example in Overleaf.

col1	col2	col3
Multiple row	cell2	cell3
	cell5	cell6
	cell8	cell9

The `multirow` command takes three parameters; our example uses the following settings:

1. the number of rows to be combined: `3`
2. the width of the column: `4em`
3. the content of the cell: `Multiple row`

Positioning tables

Positioning a table is easy if they're inside a float `table` environment.

```
\documentclass{article}
\begin{document}
Below is a table positioned exactly here:
\begin{table}[h!]
\centering
 \begin{tabular}{||c c c c||}
 \hline
 Col1 & Col2 & Col2 & Col3 \\ [0.5ex]
 \hline\hline
 1 & 6 & 87837 & 787 \\
 2 & 7 & 78 & 5415 \\
 3 & 545 & 778 & 7507 \\
 4 & 545 & 18744 & 7560 \\
 5 & 88 & 788 & 6344 \\ [1ex]
 \hline
 \end{tabular}
\end{table}
\end{document}
```

Open this example in Overleaf.

Below is a table positioned exactly here:

Col1	Col2	Col2	Col3
1	6	87837	787
2	7	78	5415
3	545	778	7507
4	545	18744	7560
5	88	788	6344

The parameter `h!` passed to the table environment declaration establishes that this table must be placed *here*, and override LaTeX defaults. The positioning parameters that can be passed-in include:

`h`

Will place the table *here* approximately.

`t`

Position the table at the *top* of the page.

`b`

Position the table at the *bottom* of the page.

`p`

Put the table in a special page, for tables only.

`!`

Override internal LaTeX parameters.

`H`

Place the table at this precise location, pretty much like h!.

For further examples on table positioning see the Positioning images and tables article.

In this example there are a few more commands:

`\centering`

Centres the table relative to the float container element.

`\[1ex]`

This adds extra space to the cell.

Captions, labels and references

Tables can be captioned, labelled and referenced by means of the `table` environment.

```
\documentclass{article}
\begin{document}
Table \ref{table:1} is an example of a referenced \LaTeX{}
element.

\begin{table}[h!]
\centering
\begin{tabular}{||c c c c||}
 \hline
 Col1 & Col2 & Col2 & Col3 \\ [0.5ex]
 \hline\hline
 1 & 6 & 87837 & 787 \\
 2 & 7 & 78 & 5415 \\
 3 & 545 & 778 & 7507 \\
 4 & 545 & 18744 & 7560 \\
 5 & 88 & 788 & 6344 \\ [1ex]
 \hline
\end{tabular}
\caption{Table to test captions and labels.}
\label{table:1}
\end{table}
\end{document}
```

Open this example in Overleaf.

Table 1 is an example of a referenced LaTeX element.

Col1	Col2	Col2	Col3
1	6	87837	787
2	7	78	5415
3	545	778	7507
4	545	18744	7560
5	88	788	6344

Table 1: Table to test captions and labels.

There are three important commands in the example:

`\caption{Table to test captions and labels}`

> As you may expect this command sets the caption for the table. If you create a list of tables this caption will be used there. You can place it above or below the table.

\label{table:1}

If you need to reference the table within your document, set a label with this command. The label will number the table and, when combined with the `\ref` command, will allow you to reference it.

\ref{table:1}

This code will be substituted by the number corresponding to the referenced table.

List of tables

To create a list of tables use the `\listoftables` command. The caption of each table will be used to generate this list. For languages supported by the `babel` package, the title "List of tables" will be translated accordingly. See the article about International language support for more info.

```
\documentclass{article}
\begin{document}
\listoftables
\vspace{5pt}
The table \ref{table:1} is an example of referenced \LaTeX{}
elements.

\begin{table}[h!]
\centering
\begin{tabular}{||c c c c||}
 \hline
 Col1 & Col2 & Col2 & Col3 \\ [0.5ex]
 \hline\hline
 1 & 6 & 87837 & 787 \\
 2 & 7 & 78 & 5415 \\
 3 & 545 & 778 & 7507 \\
 4 & 545 & 18744 & 7560 \\
 5 & 88 & 788 & 6344 \\ [1ex]
 \hline
\end{tabular}
\caption{This is the caption for the first table.}
\label{table:1}
\end{table}

\begin{table}[h!]
\centering
\begin{tabular}{||c c c c||}
 \hline
 Col1 & Col2 & Col2 & Col3 \\ [0.5ex]
 \hline\hline
  4 & 545 & 18744 & 7560 \\
 5 & 88 & 788 & 6344 \\ [1ex]
 \hline
\end{tabular}
```

```
\caption{This is the caption for the second table.}
\label{table:2}
\end{table}
\end{document}
```

Open this example in Overleaf.

List of Tables

The table 1 is an example of referenced LaTeX elements.

Col1	Col2	Col2	Col3
1	6	87837	787
2	7	78	5415
3	545	778	7507
4	545	18744	7560
5	88	788	6344

Table 1: This is the caption for the first table.

Col1	Col2	Col2	Col3
4	545	18744	7560
5	88	788	6344

Table 2: This is the caption for the second table.

Changing the appearance of a table

Several table elements can be modified to suit the needs of your document. Below you will learn how to modify the line thickness, the line colour and the background colour of the cells in your table.

Line width and cell padding

The legibility of a table can sometimes be improved by incrementing the column spacing and row stretch.

```
\documentclass{article}
\setlength{\arrayrulewidth}{0.5mm}
\setlength{\tabcolsep}{18pt}
\renewcommand{\arraystretch}{1.5}
\begin{document}
\begin{tabular}{ |p{3cm}|p{3cm}|p{3cm}|   }
```

```
\hline
\multicolumn{3}{|c|}{Country List} \\
\hline
Country Name or Area Name& ISO ALPHA 2 Code &ISO ALPHA 3 \\
\hline
Afghanistan & AF &AFG \\
Aland Islands & AX   & ALA \\
Albania &AL & ALB \\
Algeria    &DZ & DZA \\
American Samoa & AS & ASM \\
Andorra & AD & AND   \\
Angola & AO & AGO \\
\hline
\end{tabular}
\end{document}
```

Open this example in Overleaf.

Country List		
Country Name or Area Name	ISO ALPHA 2 Code	ISO ALPHA 3
Afghanistan	AF	AFG
Aland Islands	AX	ALA
Albania	AL	ALB
Algeria	DZ	DZA
American Samoa	AS	ASM
Andorra	AD	AND
Angola	AO	AGO

A description of the commands is provided below:

`\setlength{\arrayrulewidth}{0.5mm}`

This sets the thickness of the borders of the table. In the example is 0.5mm but you can use other units—see the article Lengths in LaTeX for a complete list.

`\setlength{\tabcolsep}{18pt}`

The space between the text and the left/right border of its containing cell is set to 18pt with this command. Again, you may use other units if needed.

`\renewcommand{\arraystretch}{1.5}`

The height of each row is set to 1.5 relative to its default height.

Colour alternating rows

You can apply alternating colours to the rows of your table by using the `xcolor` package with the `table` option, as demonstrated in the following example:

```
\documentclass{article}
\usepackage[table]{xcolor}
\setlength{\arrayrulewidth}{0.5mm}
\setlength{\tabcolsep}{18pt}
\renewcommand{\arraystretch}{2.5}
\begin{document}
{\rowcolors{3}{green!80!yellow!50}{green!70!yellow!40}
\begin{tabular}{ |p{3cm}|p{3cm}|p{3cm}|  }
\hline
\multicolumn{3}{|c|}{Country List} \\
\hline
Country Name or Area Name& ISO ALPHA 2 Code &ISO ALPHA 3 \\
\hline
Afghanistan & AF &AFG \\
Aland Islands & AX   & ALA \\
Albania &AL & ALB \\
Algeria    &DZ & DZA \\
American Samoa & AS & ASM \\
Andorra & AD & AND   \\
Angola & AO & AGO \\
\hline
\end{tabular}
\end{document}
```

Country List		
Country Name or Area Name	ISO ALPHA 2 Code	ISO ALPHA 3
Afghanistan	AF	AFG
Aland Islands	AX	ALA
Albania	AL	ALB
Algeria	DZ	DZA
American Samoa	AS	ASM
Andorra	AD	AND
Angola	AO	AGO

Notice the braces right before the command

`\rowcolors{3}{green!80!yellow!50}{green!70!yellow!40}`
and after the `tabular` environment. The command `\rowcolors` takes three parameters each passed inside braces:

- the row to start,
- the colour for odd rows and
- the colour for even rows.

See the `xcolor` package documentation (at the further reading section) for a list of available colours and how to create your own. In the example the colours green and yellow are mixed in different proportions.

For the command to work make sure to add

`\usepackage[table]{xcolor}`
to the preamble of your LaTeX file.

Colouring a table (cells, rows, columns and lines)

All elements in a table can be customized to use a specific colour. Again, this functionality is provided by **xcolor** so you must add

`\usepackage[table]{xcolor}`

to the preamble. Below you can see an example.

```
\documentclass{article}
\usepackage[table]{xcolor}
\setlength{\arrayrulewidth}{1mm}
\setlength{\tabcolsep}{18pt}
\renewcommand{\arraystretch}{2.5}
\newcolumntype{s}{>{\columncolor[HTML]{AAACED}} p{3cm}}
\arrayrulecolor[HTML]{DB5800}
\begin{document}
\begin{tabular}{ |s|p{3cm}|p{3cm}| }
\hline
\rowcolor{lightgray} \multicolumn{3}{|c|}{Country List} \\
\hline
Country Name or Area Name& ISO ALPHA 2 Code &ISO ALPHA 3 \\
\hline
Afghanistan & AF &AFG \\
\rowcolor{gray}
Aland Islands & AX & ALA \\
Albania   &AL & ALB \\
Algeria  &DZ & DZA \\
American Samoa & AS & ASM \\
Andorra & AD & \cellcolor[HTML]{AA0044} AND    \\
Angola & AO & AGO \\
\hline
\end{tabular}
\end{document}
```

Open this example in Overleaf.

Country List		
Country Name or Area Name	ISO ALPHA 2 Code	ISO ALPHA 3
Afghanistan	AF	AFG
Aland Islands	AX	ALA
Albania	AL	ALB
Algeria	DZ	DZA
American Samoa	AS	ASM
Andorra	AD	AND
Angola	AO	AGO

Below is a description about how to change the colour of each element in the table:

- **Colour of the lines**. The command `\arrayrulecolor` is used for this. In the example an HTML format is used, but other formats are available too, see the xcolor documentation for a complete list (link provided at the further reading section).

- **Background colour of a cell**. Use the command `\cellcolor`. You can either enter the name directly inside the braces (red, gray, green and so on) or pass a format parameter inside brackets (HTML in the example) and then set the desired colour inside the braces using the established format.

- **Background colour of a row**. In this case `\rowcolor` will accomplish that. The same observations about colour selection mentioned in the two previous commands are valid for this one.

- **Background colour of a column**. This one is a bit tricky, but the easiest way is to define a new column type. The command
- `\newcolumntype{s}{>{\columncolor[HTML]{AAACED}} p{3cm}}`

  define a column type called `s` whose alignment is `p`, the column width is `3cm` and the colour is set with HTML format to `AAACED`. This new column type is used in the `tabular` environment.

Reference guide

A brief description of parameters in the `tabular` environment.

Tables can be created using `tabular` environment.

```
\begin{tabular}[pos]{cols}
 table content
\end{tabular}
```

where options can be:

- **`pos`** : Vertical position. It can assume the following values:

`t`	the line at the top is aligned with the text baseline
`b`	the line at the bottom is aligned with the text baseline
`c` or none	the table is centred to the text baseline

- **`cols`** : Defines the alignment and the borders of each column. It can have the following values:

`l`	left-justified column
`c`	centred column
`r`	right-justified column
`p{'width'}`	paragraph column with text vertically aligned at the top
`m{'width'}`	paragraph column with text vertically aligned in the middle (requires `array` package)
`b{'width'}`	paragraph column with text vertically aligned at the bottom (requires `array` package)
`|`	vertical line
`||`	double vertical line
`*{num}{form}`	the format form is repeated *num* times; for example `*{3}{|l|}` is equal to `|l|l|l|`

To separate between cells and introducing new lines use the following commands:

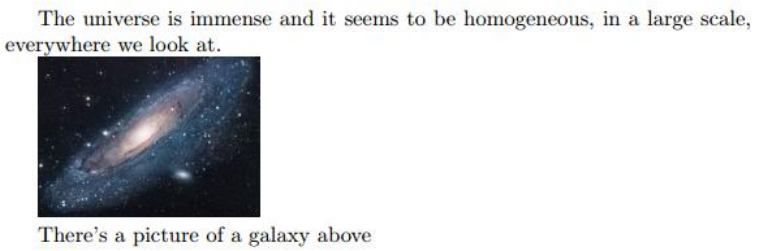
`&`	column separator
`\\`	start new row (additional space may be specified after `\\` using square brackets, such as `\\[6pt]`)
`\hline`	horizontal line between rows
`\newline`	start a new line within a cell (in a paragraph column)
`\cline{i-j}`	partial horizontal line beginning in column *i* and ending in column *j*

Inserting Images

Introduction

Below is an example on how to import a picture.

```
\documentclass{article}
\usepackage{graphicx}
\graphicspath{ {./images/} }

\begin{document}
The universe is immense and it seems to be homogeneous,
in a large scale, everywhere we look at.

\includegraphics{universe}

There's a picture of a galaxy above
\end{document}
```



Latex can not manage images by itself, so we need to use the `graphicx` package. To use it, we include the following line in the preamble: `\usepackage{graphicx}`.

The command `\graphicspath{ {./images/} }` tells LaTeX that the images are kept in a folder named `images` under the directory of the main document.

The `\includegraphics{universe}` command is the one that actually included the image in the document. Here `universe` is the name of the file containing the image without the extension, then `universe.PNG` becomes `universe`. The file name of the image should not contain white spaces nor multiple dots.

Note: The file extension is allowed to be included, but it's a good idea to omit it. If the file extension is omitted it will prompt LaTeX to search for all the supported formats.

Changing the image size and rotating the picture

If we want to further specify how LaTeX should include our image in the document (length, height, etc), we can pass those settings in the following format:

```
\begin{document}

Overleaf is a great professional tool to edit online
documents,
share and backup your \LaTeX{} projects. Also offers a
rather large help documentation.

\includegraphics[scale=1.5]{overleaf-logo}
```

Overleaf is a great professional tool to edit online documents, share and backup your LaTeX projects. Also offers a rather large help documentation.

The command `\includegraphics[scale=1.5]{overleaf-logo}` will include the image `overleaf-logo` in the document, the extra parameter `scale=1.5` will do exactly that, scale the image 1.5 of its real size.

You can also scale the image to a some specific **width and height**.

```
\begin{document}

Overleaf is a great professional tool to edit online
documents,
share and backup your \LaTeX{} projects. Also offers a
```

```
rather large help documentation.

\includegraphics[width=5cm, height=4cm]{overleaf-logo}
```

Overleaf is a great professional tool to edit online documents, share and backup your LaTeX projects. Also offers a rather large help documentation.

As you probably have guessed, the parameters inside the brackets `[width=3cm, height=4cm]` define the width and the height of the picture. You can use different units for these parameters. If only the *width* parameter is passed, the height will be scaled to keep the aspect ratio.

The length units can also be relative to some elements in document. If you want, for instance, make a picture the same width as the text:

```
\begin{document}

The universe is immense and it seems to be homogeneous,
in a large scale, everywhere we look at.

\includegraphics[width=\textwidth]{universe}
```

The universe is immense and it seems to be homogeneous, in a large scale, everywhere we look at.

Instead of `\textwidth` you can use any other default LaTeX length: `\columnsep`, `\linewidth`, `\textheight`, `\paperheight`, etc. See the reference guide for a further description of these units.

There is another common option when including a picture within your document, to rotate it. This can easily accomplished in LaTeX:

```
\begin{document}

Overleaf is a great professional tool to edit online,
share and backup your \LaTeX{} projects. Also offers a
rather large base of help documentation.

\includegraphics[scale=1.2, angle=45]{overleaf-logo}
```

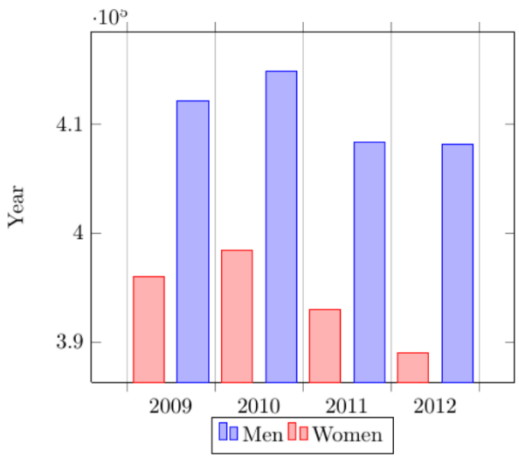
The parameter `angle=45` rotates the picture 45 degrees counter-clockwise. To rotate the picture clockwise use a negative number.

Positioning

In the previous section was explained how to include images in your document, but the combination of text and images may not look as we expected. To change this we need to introduce a new *environment*.

```
In the next example the figure will be positioned
right below this sentence.

\begin{figure}[h]
\includegraphics[width=8cm]{Plot}
\end{figure}
```

In the next example the figure will be positioned right below this sentence.



The `figure` environment is used to display pictures as floating elements within the document. This means you include the picture inside the `figure` environment and you don't have to worry about it's placement, LaTeX will position it in a such way that it fits the flow of the document.

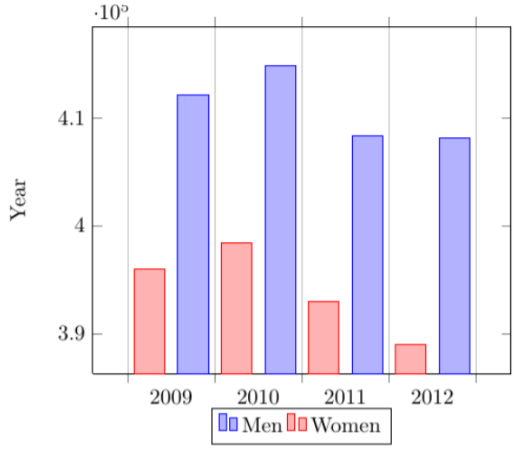
Anyway, sometimes we need to have more control on the way the figures are displayed. An additional parameter can be passed to determine the figure positioning. In the example, `begin{figure}[h]`, the parameter inside the brackets set the position of the figure to ***here***. Below a table to list the possible positioning values.

Parameter	Position
`h`	Place the float *here*, i.e., *approximately* at the same point it occurs in the source text (however, not *exactly* at the spot)
`t`	Position at the *top* of the page.
`b`	Position at the *bottom* of the page.
`p`	Put on a special *page* for floats only.
`!`	Override internal parameters LaTeX uses for determining "good" float positions.
`H`	Places the float at precisely the location in the LaTeX code. Requires the `float` package, though may cause problems occasionally. This is somewhat equivalent to `h!`.

In the next example you can see a picture at the `t`op of the document, despite being declared below the text.

```
In this picture you can see a bar graph that shows
the results of a survey which involved some important
```

```
data studied as time passed.

\begin{figure}[t]
\includegraphics[width=8cm]{Plot}
\centering
\end{figure}
```



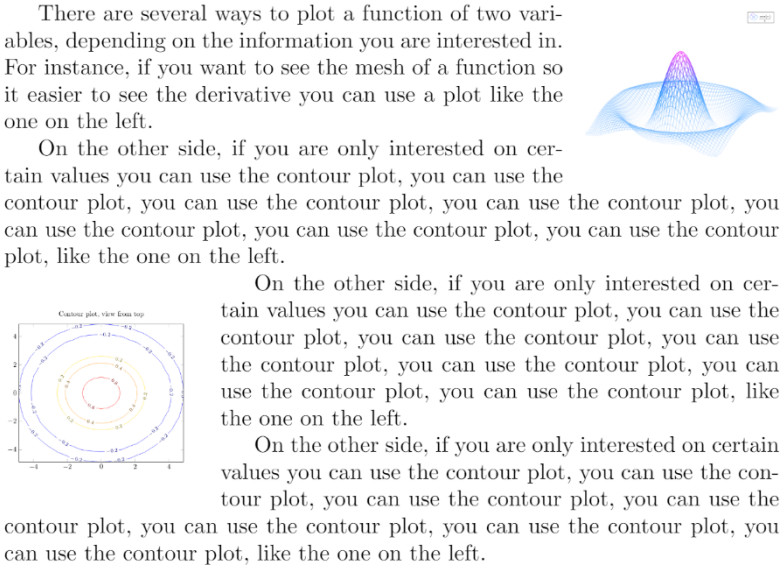
In this picture you can see a bar graph that shows the results of a survey which involved some tricky data studied as time passed.

The additional command `\centering` will centre the picture. The default alignment is *left*.

It's also possible to **wrap** the text around a figure. When the document contains small pictures this makes it look better.

```
\begin{wrapfigure}{r}{0.25\textwidth} %this figure will be at
the right
    \centering
    \includegraphics[width=0.25\textwidth]{mesh}
\end{wrapfigure}

There are several ways to plot a function of two variables,
depending on the information you are interested in. For
instance, if you want to see the mesh of a function so it
easier to see the derivative you can use a plot like the
one on the left.


\begin{wrapfigure}{l}{0.25\textwidth}
    \centering
    \includegraphics[width=0.25\textwidth]{contour}
\end{wrapfigure}

On the other side, if you are only interested on
certain values you can use the contour plot, you
can use the contour plot, you can use the contour
plot, you can use the contour plot, you can use
the contour plot, you can use the contour plot,
```

```
you can use the contour plot, like the one on the left.

On the other side, if you are only interested on
certain values you can use the contour plot, you
can use the contour plot, you can use the contour
plot, you can use the contour plot, you can use the
contour plot, you can use the contour plot,
you can use the contour plot,
like the one on the left.
```



There are several ways to plot a function of two variables, depending on the information you are interested in. For instance, if you want to see the mesh of a function so it easier to see the derivative you can use a plot like the one on the left.

On the other side, if you are only interested on certain values you can use the contour plot, you can use the contour plot, you can use the contour plot, you can use the contour plot, you can use the contour plot, you can use the contour plot, you can use the contour plot, like the one on the left.

On the other side, if you are only interested on certain values you can use the contour plot, you can use the contour plot, you can use the contour plot, you can use the contour plot, you can use the contour plot, you can use the contour plot, you can use the contour plot, like the one on the left.

On the other side, if you are only interested on certain values you can use the contour plot, you can use the contour plot, you can use the contour plot, you can use the contour plot, you can use the contour plot, you can use the contour plot, you can use the contour plot, like the one on the left.

For the commands in the example to work, you have to import the package **wrapfig**. Add to the preamble the line `\usepackage{wrapfig}`.

Now you can define the `wrapfigure` environment by means of the commands `\begin{wrapfigure}{l}{0.25\textwidth} \end{wrapfigure}`. Notice that the environment has two additional parameters enclosed in braces. Below the code is explained with more detail:

`{l}`

> This defines the alignment of the figure. Set **l** for left and **r** for right. Furthermore, if you are using a book or any similar format, use instead **o** for the outer edge and **i** for the inner edge of the page.

`{0.25\textwidth}`

> This is the width of figure box. It's not the width of the image itself, that must be set in the `\includegraphics` command. Notice that the length is relative to the text width, but normal units can also be used (cm, in, mm, etc). See the reference guide for a list of units.

`\centering`

> This was already explained, but in this example the image will be centred by using its container as reference, instead of the whole text.

For a more complete article about image positioning see Positioning images and tables

Open an images example in Overleaf

Captioning, labelling and referencing

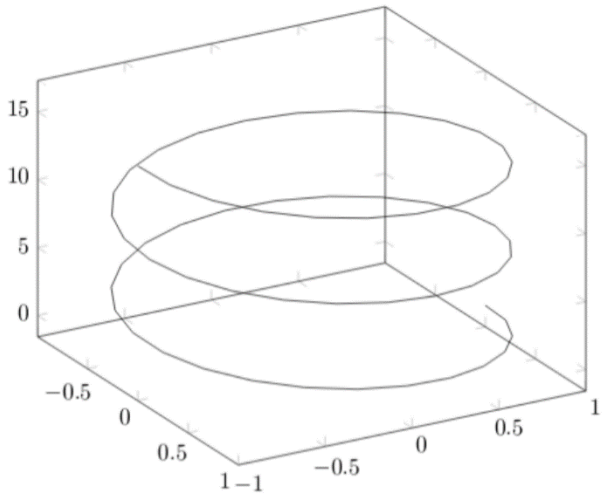
Captioning images to add a brief description and labelling them for further reference are two important tools when working on a lengthy text.

Captions

Let's start with a caption example:

```
\begin{figure}[h]
\caption{Example of a parametric plot ($\sin (x), \cos(x), x$)}
\centering
\includegraphics[width=0.5\textwidth]{spiral}
\end{figure}
```

Figure 1: Example of a parametric plot $(\sin(x), \cos(x), x)$



It's really easy, just add the `\caption{Some caption}` and inside the braces write the text to be shown. The placement of the caption depends on where you place the command; if it's above the `\includegraphics` then the caption will be on top of it, if it's below then the caption will also be set below the figure.

Captions can also be placed right after the figures. The `sidecap` package uses similar code to the one in the previous example to accomplish this.

```
\documentclass{article}
\usepackage[rightcaption]{sidecap}
```

```
\usepackage{graphicx} %package to manage images
\graphicspath{ {images/} }

\begin{SCfigure}[0.5][h]
\caption{Using again the picture of the universe.
This caption will be on the right}
\includegraphics[width=0.6\textwidth]{universe}
\end{SCfigure}
```

Figure 2: Using again the picture of the universe. This caption will be on the right

There are two new commands

`\usepackage[rightcaption]{sidecap}`

> As you may expect this line will import a package named `sidecap`, but there is an additional parameter: `rightcaption`. This parameter establishes the placement of the caption at the right of the picture, you can also use `leftcaption`. In book-like documents `outercaption` and `innercaption` are also available. The names of these are self-descriptive.

`\begin{SCfigure}[0.5][h] \end{SCfigure}`

> Defines an environment similar to `figure`. The first parameter is the width of the caption relative to the size of the image, as declared in `\includegraphics`. The second parameter `h` works exactly as in the `figure` environment. See the placement section for more information.

You can do a more advanced management of the caption formatting. Check the further reading section for references.

Labels and cross-references

Figures, just as many other elements in a LaTeX document (equations, tables, plots, etc) can be referenced within the text. This is very easy, just add a `\label` to the `figure` or `SCfigure` environment, then later use that label to refer the picture.

```
\begin{figure}[h]
    \centering
    \includegraphics[width=0.25\textwidth]{mesh}
    \caption{a nice plot}
    \label{fig:mesh1}
```

```
\end{figure}

As you can see in the figure \ref{fig:mesh1}, the
function grows near 0. Also, in the page \pageref{fig:mesh1}
is the same example.
```

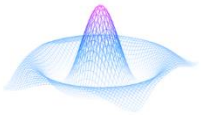


Figure 3: a nice plot

As you can see in the figure 3, the function grows near 0. Also, in the page 7 is the same example.

The `\caption` is mandatory to reference a figure.

Another great characteristic in a LaTeX document is the ability to automatically generate a *list of figures*. This is straightforward.

```
\listoffigures
```

List of Figures

This command only works on captioned figures, since it uses the caption in the table. The example above lists the images in this article.

Important Note: When using cross-references your LaTeX project must be compiled twice, otherwise the references, the page references and the table of figures won't work—Overleaf takes care of that for you.

Positioning images and tables

LaTeX is an editing tool that takes care of the format so you only have to worry about the contents of your document; nevertheless, better control of floating elements is sometimes necessary. This article explains how to position images and tables in a LaTeX document.

Contents

Introduction

The default alignment for images and tables is set to *left*

```
Lorem ipsum dolor sit amet, consectetuer adipiscing elit.
Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra
sollicitudin.

\includegraphics[width=0.5\textwidth]{overleaf-logo}

Praesent imperdiet mi nec
ante. Donec ullamcorper, felis non sodales commodo, lectus
velit
ultrices augue, a dignissim nibh lectus placerat pede.
 Vivamus nunc nunc, molestie ut, ultricies
vel, semper in, velit. Ut porttitor.
```

This is a simple example, for a description of this and other ways to include images in your LaTeX file see the article Inserting Images.

Open an example in Overleaf

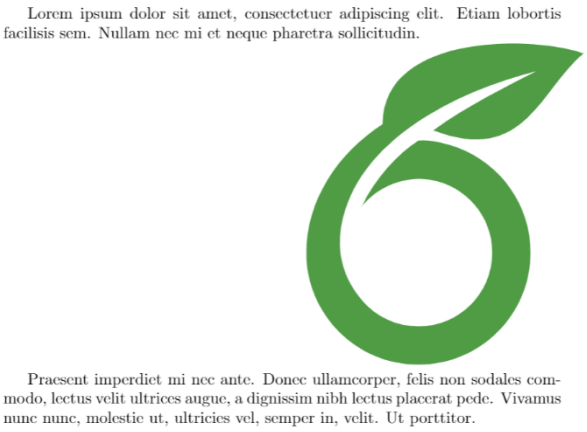
Positioning images

Basic positioning

To change the default alignment of a image from *left* or *right* an easy manner is to add

`\usepackage[export]{adjustbox}`
to the preamble of your file and then use an additional option in your image importing statement

```
Lorem ipsum dolor sit amet, consectetuer adipiscing elit.
Etiam lobortis facilisis sem. Nullam nec mi et neque
pharetra sollicitudin.

\includegraphics[width=0.5\textwidth, right]{overleaf-logo}

Praesent imperdiet mi necante. Donec ullamcorper, felis
non sodales commodo, lectus velit ultrices augue,
a dignissim nibh lectus placerat pede. Vivamus nunc nunc,
molestie ut, ultriciesvel, semper in, velit. Ut porttitor.
```



The package **adjustbox** enables an additional option in the `\includegraphics` command, in the example the picture is aligned to `right`. The available values are: *left*, *right*, *center*, *outer* and *inner*, the last two are intended for two-sided documents.

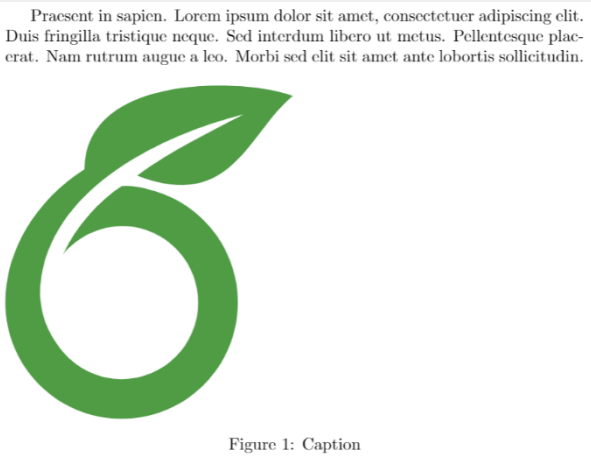
Open an example in Overleaf

The figure environment

The figure environment (see Inserting Images) is intended to provide automatic positioning.

```
Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer
adipiscing
```

```
elit. Duis fringilla tristique neque. Sed interdum libero ut
metus.
Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit
sit amet
ante lobortis sollicitudin.

\begin{figure}[h]
\includegraphics[width=0.5\textwidth, inner]{overleaf-logo}
\caption{Caption}
\label{fig:figure2}
\end{figure}
```



This environment uses a positioning parameter passed inside brackets, it can take the next values:

Parameter	Position
`h`	Place the float *here*, i.e., *approximately* at the same point it occurs in the source text (however, not *exactly* at the spot)
`t`	Position at the *top* of the page.
`b`	Position at the *bottom* of the page.
`p`	Put on a special *page* for floats only.
`!`	Override internal parameters LaTeX uses for determining "good" float positions.
`H`	Places the float at precisely the location in the LaTeX code. Requires the `float` package. This is somewhat equivalent to `h!`.

You can put more than one value in the parameter, for instance, if you write `[ht]` LaTeX will try to position the figure **here**, but if it's not possible (the space may be insufficient) then the figure will appear at the **top** of the page. It is

recommended to use more than one positioning parameter to prevent unexpected results.

Open an example in Overleaf

Multiple images in one figure

It is possible to insert several images in one figure, each one with its own reference and label

```
Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer
adipiscing elit. Duis fringilla tristique neque...

\begin{figure}[h]

\begin{subfigure}{0.5\textwidth}
\includegraphics[width=0.9\linewidth, height=6cm]{overleaf-
logo}
\caption{Caption1}
\label{fig:subim1}
\end{subfigure}
\begin{subfigure}{0.5\textwidth}
\includegraphics[width=0.9\linewidth, height=6cm]{mesh}
\caption{Caption 2}
\label{fig:subim2}
\end{subfigure}

\caption{Caption for this figure with two images}
\label{fig:image2}
\end{figure}

Praesent blandit blandit mauris. Praesent lectus tellus,
aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia
lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan
semper.
```

Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. In the document you can enter a reference to image 2a, it's actually easy you to do it.

(a) Caption1 (b) Caption 2

Figure 2: Caption for this figure with two images

Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

First, you must import the package **subcaption** by adding to the preamble

```
\usepackage{subcaption}
```

then you can use the environment `\subfigure` that takes one parameter, the width of the figure. This environment must be used inside a `figure` environment, captions and labels can be set to each *subfigure*.

Open an example in Overleaf

Wrapping text around a figure

The package **wrapfig** provides a useful feature, text can be floated around the images.

```
Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer
adipiscing elit. Duis fringilla tristique neque. Sed interdum
libero ut metus. Pellentesque placerat.

\begin{wrapfigure}{l}{0.25\textwidth}
\includegraphics[width=0.9\linewidth]{overleaf-logo}
\caption{Caption1}
\label{fig:wrapfig}
\end{wrapfigure}

Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer
adipiscing elit. Duis fringilla tristique neque. Sed interdum
```

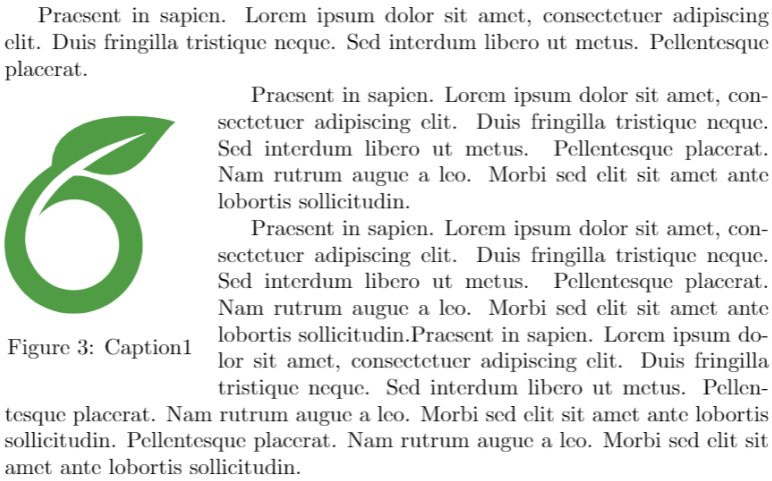


Figure 3: Caption1

First import the package *wrapfig* by adding

```
\usepackage{wrapfig}
```
to the preamble.

After that you can use the environment `wrapfig`, it takes two parameters that are passed inside braces: the alignement that can be *l*, *r*, *c*, *i* or *o*; this letters stand for left, right, centre, inner and outer (the last two intended for two-sided documents). The second parameter is the width of the figure, in the example is 0.25 the width of the text. See the reference guide for a list of possible length units.

Open an example in Overleaf

Positioning tables

Options for table positioning are similar to those available for figures.

Basic positioning

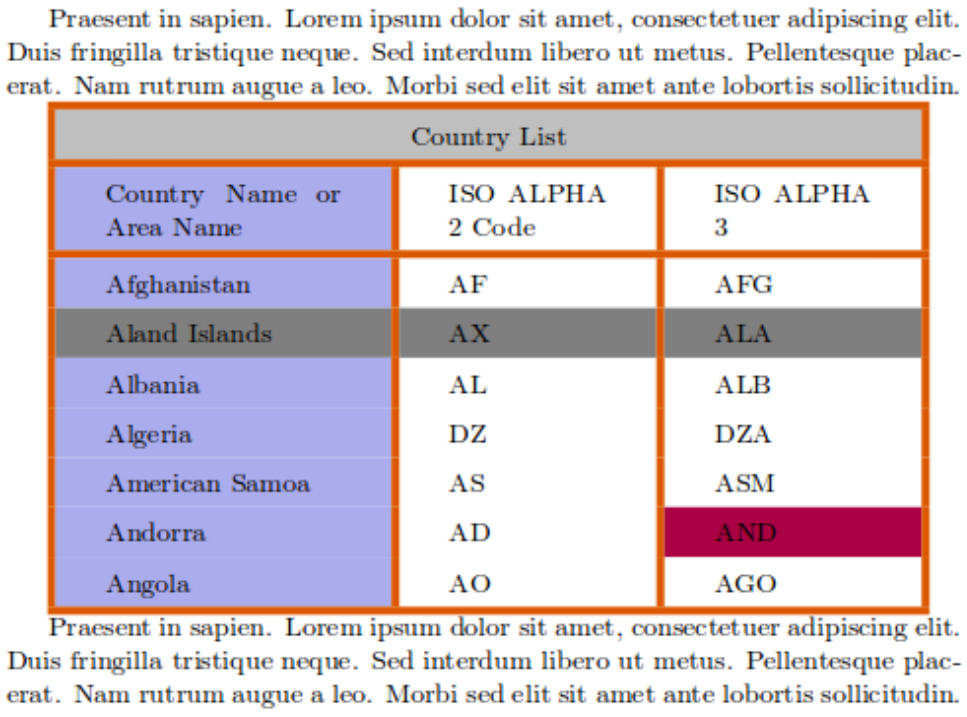
Default position of the tabular environment is *centre*.

```
Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer
adipiscing elit. Duis fringilla tristique neque.
Sed interdum libero ut metus. Pellentesque placerat. Nam
rutrum augue a leo. Morbi sed elit sit amet
ante lobortis sollicitudin.

\arrayrulecolor[HTML]{DB5800}
\begin{tabular}{ |s|p{2cm}|p{2cm}|  }
\hline
\rowcolor{lightgray} \multicolumn{3}{|c|}{Country List} \\
\hline
Country Name or Area Name& ISO ALPHA 2 Code &ISO ALPHA 3 \\
\hline
```

```
Afghanistan & AF &AFG \\
\rowcolor{gray}
Aland Islands & AX  & ALA \\
Albania    &AL & ALB \\
Algeria   &DZ & DZA \\
American Samoa & AS & ASM \\
Andorra & AD & \cellcolor[HTML]{AA0044} AND \\
Angola & AO & AGO \\
\hline
\end{tabular}

Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer
adipiscing
elit. Duis fringilla tristique neque. Sed interdum libero ut
metus.
Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit
sit
amet ante lobortis sollicitudin.
```

Open this code fragment in Overleaf

The following graphic shows the result of the code fragment above:



Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin.

Country List		
Country Name or Area Name	ISO ALPHA 2 Code	ISO ALPHA 3
Afghanistan	AF	AFG
Aland Islands	AX	ALA
Albania	AL	ALB
Algeria	DZ	DZA
American Samoa	AS	ASM
Andorra	AD	AND
Angola	AO	AGO

Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin.

You can also open a complete project example project in Overleaf.

To learn about how to create tables see the Tables article.

The table environment

The table environment is intended to automatically position tables so they fit nicely in the flow of your document.

```
Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer
adipiscing elit. Duis fringilla tristique neque. Sed interdum
libero ut metus. Pellentesque placerat. Nam rutrum augue a
leo.
Morbi sed elit sit amet ante lobortis sollicitudin.

\begin{table}[ht]
\arrayrulecolor[HTML]{DB5800}
\centering
\begin{tabular}{ |s|p{2cm}|p{2cm}|  }
\hline
\rowcolor{lightgray} \multicolumn{3}{|c|}{Country List} \\
\hline
Country Name or Area Name& ISO ALPHA 2 Code &ISO ALPHA 3 \\
\hline
Afghanistan & AF &AFG \\
\rowcolor{gray}
Aland Islands & AX  & ALA \\
Albania    &AL & ALB \\
Algeria   &DZ & DZA \\
American Samoa & AS & ASM \\
Andorra & AD & \cellcolor[HTML]{AA0044} AND \\
Angola & AO & AGO \\
\hline
\end{tabular}
\caption{Table inside a floating element}
\label{table:ta}
\end{table}

Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer
adipiscing elit. Duis fringilla tristique neque. Sed interdum
libero ut metus. Pellentesque placerat. Nam rutrum augue a
leo.
Morbi sed elit sit amet ante lobortis sollicitudin.
```

Open this code fragment in Overleaf

The following graphic shows the output produced by the Overleaf link:

Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin.

Country List		
Country Name or Area Name	ISO ALPHA 2 Code	ISO ALPHA 3
Afghanistan	AF	AFG
Aland Islands	AX	ALA
Albania	AL	ALB
Algeria	DZ	DZA
American Samoa	AS	ASM
Andorra	AD	AND
Angola	AO	AGO

Table 1: Table inside a floating element

Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin.

A position parameter, inside brackets, can be passed to the **table** environment. This parameter can take the next values:

Parameter	Position
`h`	Place the float *here*, i.e., *approximately* at the same point it occurs in the source text (however, not *exactly* at the spot)
`t`	Position at the *top* of the page.
`b`	Position at the *bottom* of the page.
`p`	Put on a special *page* for floats only.
`!`	Override internal parameters LaTeX uses for determining "good" float positions.
`H`	Places the float at precisely the location in the LaTeX code. Requires the `float` package. This is somewhat equivalent to `h!`.

You can set more than one value in the parameter, for instance, if you write `[ht]` LaTeX will try to position the table **here**, but if it's not possible (the space may be insufficient) then the table will appear at the **top** of the page. It is

recommended to use more than one positioning parameter to prevent unexpected results.

Notice also the command `\centering`. This changes the alignment of the table within its container to *centre* instead of the default *left*.

Open an example in Overleaf

Wrapping text around a table

If your table don't take all available space and you want to put text next or before it, is possible with the package `wrapfig`.

First, import the package

```
\usepackage{wrapfig}
```

then you can use the environment `wraptable` which takes two parameters: The first one is the alignment that can be `l`, `r`, `c`, `i` or `o` for left, right, centre, inner and outer respectively. The second one is the width of the table container, keep in mind that this latter parameter must be the same as the width of the table, otherwise things may not be properly aligned.

```
Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer
adipiscing elit. Duis fringilla tristique neque. Sed interdum
libero ut metus. Pellentesque placerat. Nam rutrum augue a
leo.
Morbi sed elit sit amet ante lobortis sollicitudin.

\begin{wraptable}{r}{8cm}
\arrayrulecolor[HTML]{DB5800}
\centering
\begin{tabular}{ |s|p{2cm}|  }
\hline
\rowcolor{lightgray} \multicolumn{2}{|c|}{Country List} \\
\hline
Country Name or Area Name& ISO ALPHA 2 Code \\
\hline
Afghanistan & AF \\
\rowcolor{gray}
Aland Islands & AX \\
Albania    &AL  \\
Algeria   &DZ \\
American Samoa & AS \\
Andorra & \cellcolor[HTML]{AA0044} AD   \\
Angola & AO \\
\hline
\end{tabular}
\caption{Table inside a wraptable}
\label{table:ta2}
\end{wraptable}
```

```
Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer
adipiscing elit. Duis fringilla tristique neque. Sed interdum
libero ut metus. Pellentesque placerat. Nam rutrum augue a
leo.
Morbi sed elit sit amet ante lobortis sollicitudin...
```

Open this `wrapfig` code fragment in Overleaf

The following graphic shows the output produced by the Overleaf link:

Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin.

Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin... Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin...

Country List	
Country Name or Area Name	ISO ALPHA 2 Code
Afghanistan	AF
Aland Islands	AX
Albania	AL
Algeria	DZ
American Samoa	AS
Andorra	AD
Angola	AO

Table 1: Table inside a wraptable

Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin... Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin...

Reference guide

LaTeX units and lengths

Abbreviation	Definition
`pt`	A point, is the default length unit. About 0.3515mm
`mm`	a millimetre

`cm`	a centimetre
`in`	an inch
`ex`	the height of an **x** in the current font
`em`	the width of an **m** in the current font
`\columnsep`	distance between columns
`\columnwidth`	width of the column
`\linewidth`	width of the line in the current environment
`\paperwidth`	width of the page
`\paperheight`	height of the page
`\textwidth`	width of the text
`\textheight`	height of the text
`\unitlength`	units of length in the picture environment.

Typesetting quotations

Introduction

When it comes to quotations and quotation marks, each language has its own symbols and rules. For this reason, several LaTeX packages have been created to assist in typesetting quotations in-line, in display mode or at the beginning of each chapter. It's important to remark that even if you are typing English quotes, different quotation marks used in **English (UK)** and **English (US)**. Plenty of different quotation marks can be typeset with LaTeX, and there are options for almost every language (see the reference guide).

We will look at several packages suited to typesetting different types of quotation.

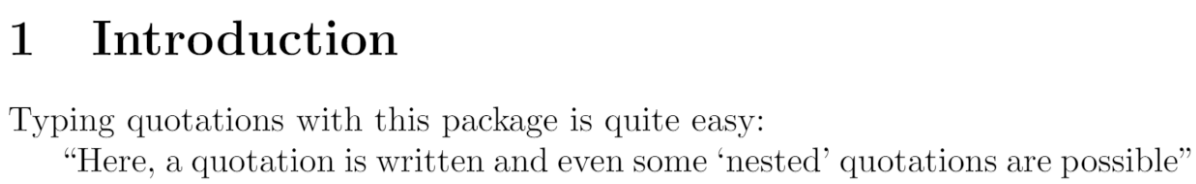
`dirtytalk` package

`dirtytalk` is a small LaTeX package with only one available command: `\say`, as shown in the next example:

```
\documentclass{article}
\usepackage{dirtytalk}
\begin{document}
\section{Introduction}

Typing quotations with this package is quite easy:

\say{Here, a quotation is written and even some \say{nested}
quotations
are possible}
\end{document}
```

Open this example in Overleaf.

This example produces the following output:



The `dirtytalk` package can be loaded by putting the following line in your document preamble:

```
\usepackage{dirtytalk}
```

`dirtytalk` supports one nested quotation and has options to redefine the characters used for the quotes. For example, in a document written in French the following code could be used:

```
\usepackage[french]{babel}
\usepackage[T1]{fontenc}
\usepackage[
    left = \flqq{},%
    right = \frqq{},%
    leftsub = \flq{},%
    rightsub = \frq{} %
]{dirtytalk}
```

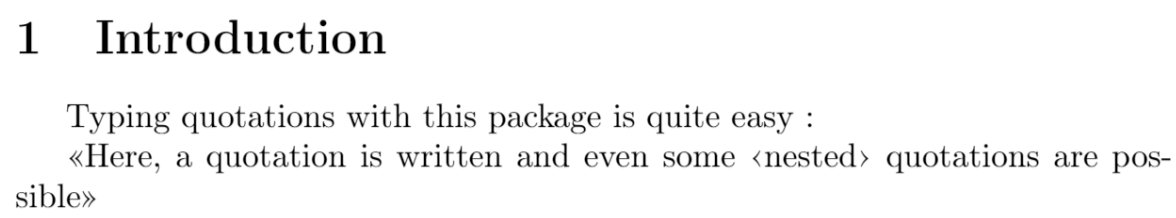
The first two commands define the primary left and right quotation marks, the second pair of commands define the secondary set of quotation marks. Here is an example using the code above:

```
\documentclass{article}
\usepackage[french]{babel}
\usepackage[T1]{fontenc}

\usepackage[
    left = \flqq{},%
```

```
    right = \frqq{},%
    leftsub = \flq{},%
    rightsub = \frq{} %
]{dirtytalk}

\begin{document}
\section{Introduction}

Typing quotations with this package is quite easy:

\say{Here, a quotation is written and even some \say{nested}
quotations are possible}
\end{document}
```

Open this example in Overleaf.

This example produces the following output:



This package is suitable for most situations: it's very simple, since only one command is needed, and it supports nesting quotations to one degree. If a more complex quotation mark structure is required, the options listed in the following sections may be more effective.

`csquotes` package

The `csquotes` package provides advanced facilities for in-line and display quotations. It supports a wide range of commands, environments and user-definable quotes. Quotes can be automatically adjusted to the current language by means of the `babel` or `polyglossia` packages. This package is suitable for documents with complex quotation requirements, therefore it has a vast variety of commands to insert in-line quotes, quotes with sources, block-quotes with the support of changing language.

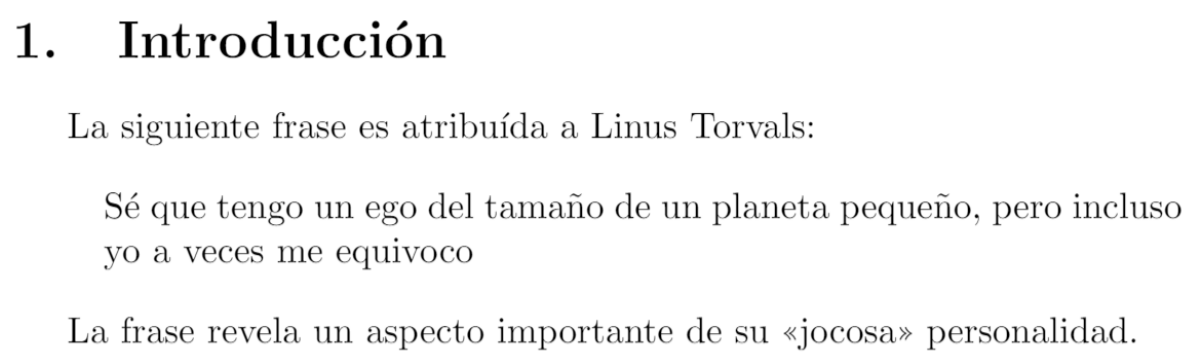
The following example uses the `csquotes` package, in conjunction with `babel`, within a document written in Spanish. It automatically loads the correct quotation characters "«" and "»"— known as guillemets (or "comillas angulares", in Spanish).

```
\documentclass{article}
\usepackage[spanish]{babel}
\usepackage{csquotes}

\begin{document}
\section{Introducción}
```

```
La siguiente frase es atribuída a Linus Torvals:

\begin{displayquote}
Sé que tengo un ego del tamaño de un planeta pequeño, pero
incluso yo a veces me equivoco
\end{displayquote}

La frase revela un aspecto importante de su \textquote{jocosa}
personalidad.
\end{document}
```

Open this example in Overleaf.

This example produces the following output:



In the example the environment `displayquote` prints a display quotation and the command `\textquote` and in-line quotation.

`epigraph` package

Some authors like to write quotations at the beginning of a chapter: those quotations are known as epigraphs. The `epigraph` package provides a vast set of options to typeset epigraphs and epigraphs lists. To use the package, add the following line to your document preamble:

```
\usepackage{epigraph}
```

Here is an example showing an epigraph quotation typed using the command `\epigraph{}{}`, whose first parameter is the quotation itself and the second parameter is the quotations source (author, book, etc.):

```
\documentclass{book}
\usepackage{blindtext} %This package generates automatic text
\usepackage{epigraph}

\title{Epigraph example}
\author{Overleaf}
\date{August 2021}

\begin{document}
\frontmatter
\mainmatter
```

```
\chapter{Something}
\epigraph{All human things are subject to decay, and when fate summons, Monarchs must obey}{\textit{Mac Flecknoe \\ John Dryden}}
\blindtext
\end{document}
```

Open this example in Overleaf

This example produces the following output:

The `epigraph` package can handle several quotations by means of a special environment and also has many customization options.

`quotchap` package

The package `quotchap` redefines the commands `chapter`, and its starred version, to reformat them. You can change the colour of the chapter number with this package.

It also provides a special environment to typeset quotations and the corresponding authors.

To use this package include the following line in your document preamble:

```
\usepackage{quotchap}
```

Quotes are typed inside the environment `savequote`. In the example below, the parameter inside brackets, `[45mm]`, sets the width of the quotation area. After each quote the command `\qauthor{}` is used to typeset and format the author's name.

```
\documentclass{book}
\usepackage{blindtext}
\usepackage{quotchap}

\begin{document}
\begin{savequote}[45mm]
---When shall we three meet again
in thunder, lightning, or in rain?
---When the hurlyburly's done,
when the battle's lost and won.
\qauthor{Shakespeare, Macbeth}
Cookies! Give me some cookies!
\qauthor{Cookie Monster}
\end{savequote}

\chapter{Classic Sesame Street}
\blindtext
\end{document}
```

Open this example in Overleaf.

This example produces the following output:

—When shall we three meet again
in thunder, lightning, or in rain?
—When the hurlyburly's done,
when the battle's lost and won.

Shakespeare, Macbeth

Cookies! Give me some cookies!

Cookie Monster

1

Classic Sesame Street

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Using colours in LaTeX

-

Introduction

This article explains how to use colour in your LaTeX document via the `color` or `xcolor` packages. Note that user-level documentation of the `color` package is contained in The LaTeX standard graphics bundle.

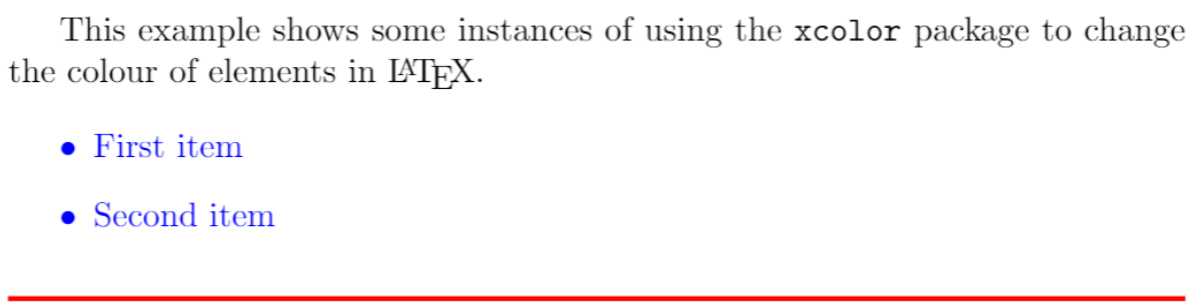
Both packages provide a common set of commands for colour manipulation, but `xcolor` is more flexible and supports a larger number of colour models, so is the recommended approach.

We'll start with the following example:

```
\documentclass{article}
\usepackage{xcolor}
\begin{document}
This example shows some instances of using the \texttt{xcolor}
package
to change the colour of elements in \LaTeX.

\begin{itemize}
\color{blue}
\item First item
\item Second item
\end{itemize}

\noindent
{\color{red} \rule{\linewidth}{0.5mm}}
\end{document}
```

Open this `xcolor` example in Overleaf

This example produces the following output:



In this example, the package `xcolor` is imported with

```
\usepackage{xcolor}
```

then the command `\color{blue}` sets the `blue` colour for the current block of text. In this case for the `itemize` environment.

The code to typeset the horizontal line created by `\rule{\linewidth}{0.5mm}` is contained in a group, delimited by { and }, in order to keep the effects of the `\color{red}` local to that group.

Named colours provided by the `xcolor` package

As noted in the xcolor package documentation, the following named colours are always available without needing to load any package options:

Loading and using named colours in the `color` package

You can also use the `color` package and load named colours via its `usenames` and `dvipsnames` package options:

```
\usepackage[usenames,dvipsnames]{color}
```

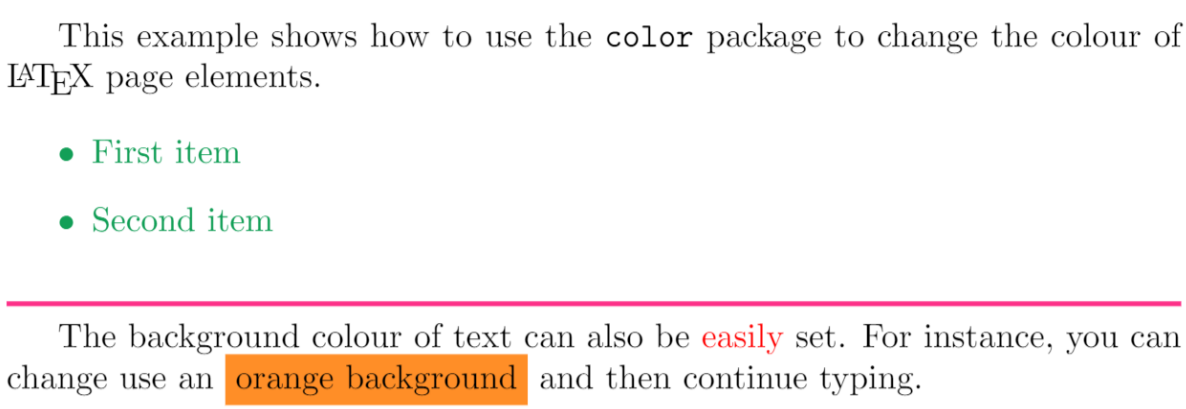
The following code uses the `color` package to apply the same named colours used in the previous `xcolor` package example.

```
\documentclass{article}
\usepackage[usenames,dvipsnames]{color} %using the color
package, not xcolor
\begin{document}
This example shows how to use the \texttt{\bfseries color}
package
```

```
to change the colour of \LaTeX{} page elements.

\begin{itemize}
\color{ForestGreen}
\item First item
\item Second item
\end{itemize}

\noindent
{\color{RubineRed} \rule{\linewidth}{0.5mm}}

The background colour of text can also be
\textcolor{red}{easily} set. For
instance, you can change use an \colorbox{BurntOrange}{orange
background} and then continue typing.
\end{document}
```

Open this **color** example in Overleaf

This example produces the same output as the previous `xcolor` version:



Setting the page background colour

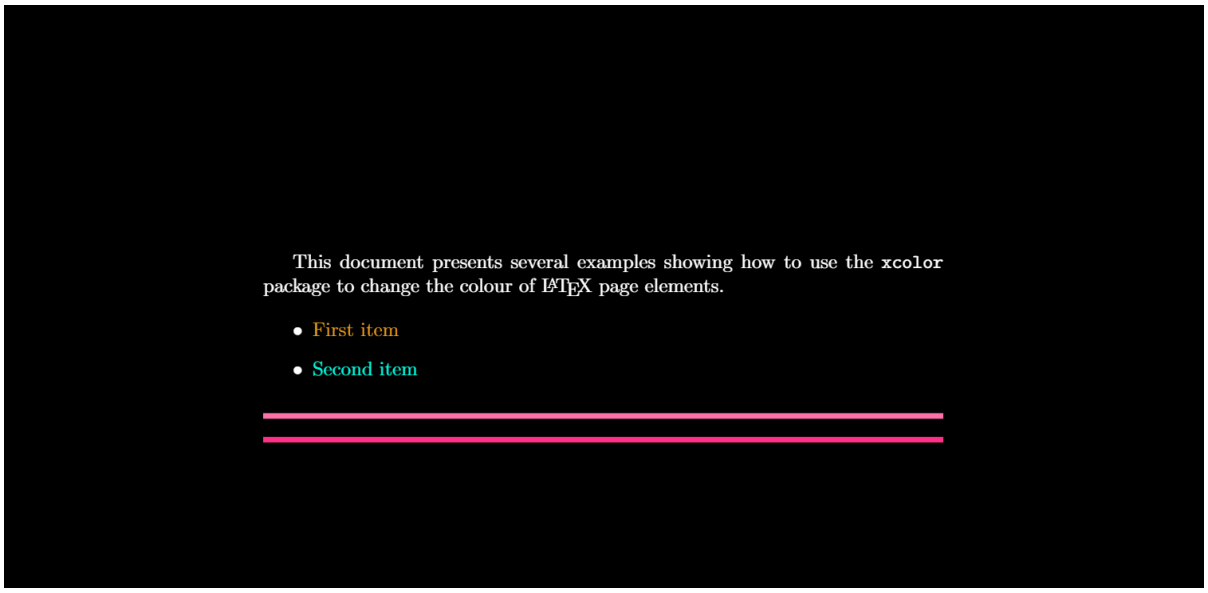
The background colour of the entire page can be easily changed with `\pagecolor`. The following code demonstrates this, using the text of an earlier example::

```
\documentclass{article}
\usepackage[dvipsnames]{xcolor}
\colorlet{LightRubineRed}{RubineRed!70}
\colorlet{Mycolor1}{green!10!orange}
\definecolor{Mycolor2}{HTML}{00F9DE}
\begin{document}
\pagecolor{black}
\color{white}% set the default colour to white
This document presents several examples showing how to use the
\texttt{xcolor} package
to change the colour of \LaTeX{} page elements.

\begin{itemize}
```

```
\item \textcolor{Mycolor1}{First item}
\item \textcolor{Mycolor2}{Second item}
\end{itemize}

\noindent
{\color{LightRubineRed} \rule{\linewidth}{1mm}}

\noindent
{\color{RubineRed} \rule{\linewidth}{1mm}}
\end{document}
```

Open this example in Overleaf

This example produces the following output:



The command `\pagecolor{black}` set the page colour to `black`. This is a switch command, meaning it will take effect in the entire document unless another switch command is used to revert it. `\nopagecolor` will change the background back to normal.

Multiple columns

Introduction

Two-column documents can be easily created by passing the parameter `\twocolumn` to the document class statement. If you need more flexibility in the column layout, or to create a document with multiple columns, the package `multicol` provides a set of commands for that. This article explains how use the `multicol` package, starting with this basic example:

```
\documentclass{article}
\usepackage{blindtext}
\usepackage{multicol}
\title{Multicols Demo}
\author{Overleaf}
\date{April 2021}

\begin{document}
\maketitle

\begin{multicols}{3}
[
\section{First Section}
All human things are subject to decay. And when fate summons,
Monarchs must obey.
]
\blindtext\blindtext
\end{multicols}

\end{document}
```

Open this `multicols` example in Overleaf

Multicols Demo

Overleaf

April 2021

1 First Section

All human things are subject to decay. And when fate summons, Monarchs must obey.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

To import the package, the line

```
\usepackage{multicol}
```

is added to the preamble. Once the package is imported, the environment `multicols` can be used. The environment takes two parameters:

- Number of columns. This parameter must be passed inside braces, and its value is 3 in the example.
- "Header text", which is inserted in between square brackets. This is optional and will be displayed on top of the multicolumn text. Any LaTeX command can be used here, except for floating elements such as figures and tables. In the example, the section title and a small paragraph are set here.

The text enclosed inside the tags `\begin{multicols}` and `\end{multicols}` is printed in multicolumn format.

Column separation

The column separation is determined by `\columnsep`. See the example below:

```
\documentclass{article}
\usepackage{blindtext}
\usepackage{multicol}
\setlength{\columnsep}{1cm}
\title{Second multicols Demo}
\author{Overleaf}
\date{April 2021}

\begin{document}
\maketitle

\begin{multicols}{2}
[
\section{First Section}
All human things are subject to decay. And when fate summons,
Monarchs must obey.
]
\blindtext\blindtext
\end{multicols}

\end{document}
```

Open this `multicols` example in Overleaf

Second multicols Demo

Overleaf

April 2021

1 First Section

All human things are subject to decay. And when fate summons, Monarchs must obey.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Here, the command `\setlength{\columnsep}{1cm}` sets the column separation to 1cm. See Lengths in LaTeX for a list of available units.

Unbalanced columns

In the default `multicols` environment the columns are balanced so each one contains the same amount of text. This default format can be changed by the stared environment `multicols*`:

```
\documentclass{article}
\usepackage{blindtext}
\usepackage{multicol}
\setlength{\columnsep}{1cm}
\title{Second multicols Demo}
\author{Overleaf}
\date{April 2021}
\begin{document}
\maketitle
\begin{multicols*}{3}
[
\section{First Section}
All human things are subject to decay. And when fate summons,
Monarchs must obey.
]
\blindtext\blindtext
\end{multicols*}

\end{document}
```

Open this `multicols` example in Overleaf

Second multicols Demo

Overleaf

April 2021

1 First Section

All human things are subject to decay. And when fate summons, Monarchs must obey.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

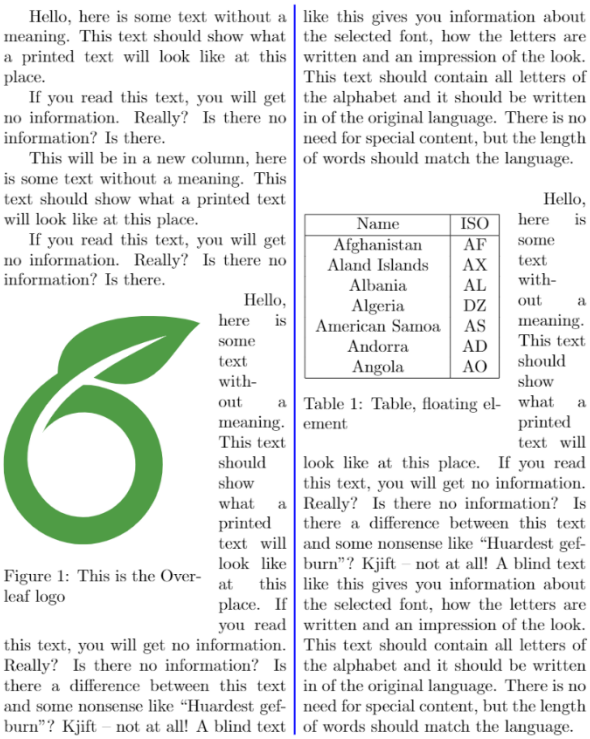
If you open this example on Overleaf you'll see that the text is printed in a column till the end of the page is reached, then the in continues in the next column, and so on.

Inserting floating elements

Floating elements (tables and figures) can be inserted in a multicolumn document with `wrapfig` and `wraptable`.

```
\begin{multicols}{2}
[
\section{First Section}
All human things are subject to decay. And when fate summons,
Monarchs must obey.
]

Hello, here is some text without a meaning.  This text should
show what
a printed text will look like at this place.
If you read this text, you will get no information.  Really?
Is there
no information?  Is there.

\vfill
```

```
\begin{wrapfigure}{l}{0.7\linewidth}
\includegraphics[width=\linewidth]{overleaf-logo}
\caption{This is the Overleaf logo}
\end{wrapfigure}

A blind text like this gives you information about the
selected font, how
the letters are written and an impression of the look.  This
text should
contain all...

\begin{wraptable}{l}{0.7\linewidth}
\centering
\begin{tabular}{|c|c|}
\hline
Name & ISO \\
\hline
Afghanistan & AF \\
Aland Islands & AX \\
Albania    &AL  \\
Algeria   &DZ \\
American Samoa & AS \\
Andorra & AD   \\
Angola & AO \\
\hline
\end{tabular}
\caption{Table, floating element}
\label{table:ta}
\end{wraptable}

\end{multicols}

\end{document}
```



Hello, here is some text without a meaning. This text should show what a printed text will look like at this place.

If you read this text, you will get no information. Really? Is there no information? Is there.

This will be in a new column, here is some text without a meaning. This text should show what a printed text will look like at this place.

If you read this text, you will get no information. Really? Is there no information? Is there.

Figure 1: This is the Overleaf logo

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Name	ISO
Afghanistan	AF
Aland Islands	AX
Albania	AL
Algeria	DZ
American Samoa	AS
Andorra	AD
Angola	AO

Table 1: Table, floating element

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Open this `multicols` example in Overleaf

Floats in the **multicol** package are poorly supported in the current version. Elements inserted with the conventional `figure*` and `table*` environments will show up only at the top or bottom of the next page after they are inserted, and will break the layout. The example presented here is a workaround, but you may expect some rough edges. For instance, if the float width is set to `\linewidth` it causes a weird text overlapping. This said, below is a brief description of the commands:

- `\usepackage{wrapfig}`. Put this line in the preamble to import the package **wrapfig**
- The environment `wrapfigure` will insert a figure wrapped in the text. For more information and further examples about this environment see Positioning images and tables.
- The environment `wraptable` is the equivalent to *wrapfigure* but for tables. See Positioning images and tables for more information.

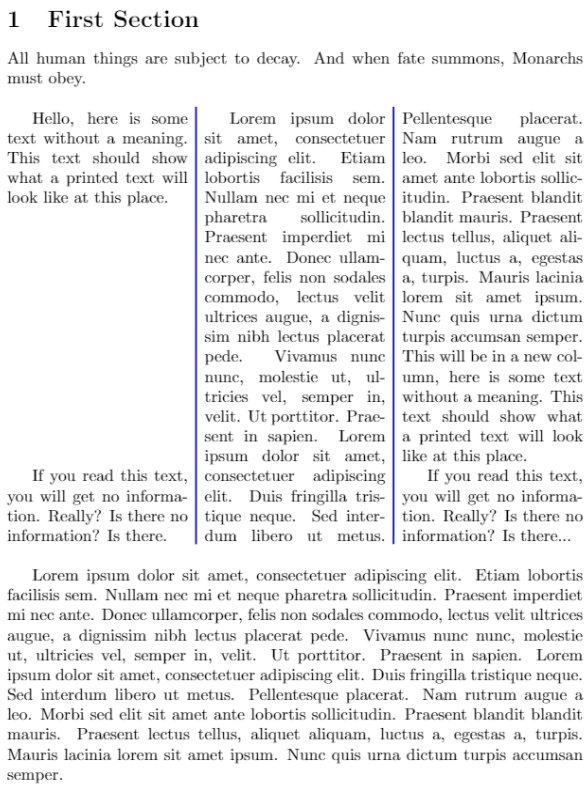
Inserting vertical rulers

A vertical ruler can be inserted as column separator to may improve readability in some documents:

```
\documentclass{article}
\usepackage{blindtext}
\usepackage{multicol}
\usepackage{color}
\setlength{\columnseprule}{1pt}
\def\columnseprulecolor{\color{blue}}
```

```
\begin{document}

\begin{multicols}{3}
[
\section{First Section}
All human things are subject to decay. And when fate summons,
Monarchs must obey.
]
Hello, here is some text without a meaning.  This text should
show what
a printed text will look like at this place.

If you read this text, you will get no information.  Really?
Is there
no information?  Is there.

\columnbreak
\blindtext
This will be in a new column, here is some text without a
meaning.  This text
should show what a printed text will look like at this place.

If you read this text, you will get no information.  Really?
Is there
no information?  Is there...
\end{multicols}

\blindtext

\end{document}
```

Open this `multicols` example in Overleaf



1 First Section

All human things are subject to decay. And when fate summons, Monarchs must obey.

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place.

If you read this text, you will get no information. Really? Is there no information? Is there.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus.

Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper. This will be in a new column, here is some text without a meaning. This text should show what a printed text will look like at this place.

If you read this text, you will get no information. Really? Is there no information? Is there...

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

If you open this example on Overleaf you will see the column separator can be set to a specific colour also. Below a description of each command:

`\usepackage{color}.`

This line is inserted in the preamble to enable the use of several colours within the document.

`\setlength{\columnseprule}{1pt}`

This determines the width of the ruler to be used as column separator, it's set to 0 by default. In the example a column whose width is 1pt is printed.

`\def\columnseprulecolor{\color{blue}}`

The colour of the separator ruler is set to *blue*. See the article about using colours in LaTeX for more information on colour manipulation.

`\columnbreak`

This command inserts a column breakpoint. In this case, the behaviour of the text is different from what you may expect. The column break is inserted, then the paragraphs before the breakpoint are evenly distributed to fill all available space. In the example, the second paragraph is at the bottom of the column and a blank space is inserted in between the second and the first paragraphs.