

# L<sup>A</sup>T<sub>E</sub>X FOR UNDERGRADUATES

## TABULAR ENVIRONMENT (TABLES)

### Lecture Notes

Tom Schenk Jr.

*Version of January 11, 2006*

## 1 Motivation

New users sometime catch on to tables slowly; however, once adapted, making tables is very fast. This lecture describes how to make tables in a L<sup>A</sup>T<sub>E</sub>X document.

## 2 Tabular Environment

The first thing is to point out that the name of the basic table environment is called `tabular`. The `table` environment, which you may have encountered, is similar, but a little more advanced. When using the `tabular` environment, the table is placed at that exact place; that is, if a table is large, it may be cut off by a page break. The `table` environment will “float” so it will appear on another page instead of being interrupted. Nevertheless, the most important thing to learn is the `tabular` environment since `table` builds upon this.<sup>1</sup>

Tables can be broken down into two parts: columns (which run vertical) and rows (which run horizontal). An easy way to remember the difference is columns of a building are large pillars, which stand vertically. A row is similar to an auditorium where seats are aligned side-to-side in relation to the stage.

The first step is to start the tabular environment with `\begin{tabular}`. Immediately following this, the author must determine the number of columns and how the text should be justified with `l`, `c`, or `r` for left, right, and centered, respectively. Let’s look at the following: `\begin{tabular}{l l c}` The text in the first and second column is justified to the left. The text in the third column is centered. As you can see, it is important to know the number of columns before starting a table. Knowing the number of columns is not needed as it can be determined along the way.

The next few lines is the actual content of the cells.

---

<sup>1</sup>The `table` environment is discussed in the advanced portion of the tutorial.

```

\begin{tabular}{l l c}
Cell 1 & Cell 2 & Cell 3 \\
Cell 4 & Cell 5 & Cell 6 \\
Cell 7 & & Cell 8
\end{tabular}

```

The & will start the next cell. Double backslashes will start a new column. Two breaks (e.g. &) will create three cells, one break will create two cells, and so on. The number of cells in a row needs to match the number of columns that you determined previously, if not, L<sup>A</sup>T<sub>E</sub>X will complain bitterly. As you can see, new columns will be created if needed. Below is an example of the above code:

```

Cell 1   Cell 2   Cell 3
Cell 4   Cell 5   Cell 6
Cell 7           Cell 8

```

You may include horizontal and vertical lines. Placing the — (pipe) character between the column justifications will create vertical lines (e.g. `\begin{tabular}{l|l|c}`). Likewise, `\hline` will insert horizontal lines between rows. Below is an example of including horizontal and vertical lines.

```

\begin{tabular}{l|l|c|}
\hline
Cell 1 & Cell 2 & Cell 3 \\
\hline
Cell 4 & Cell 5 & Cell 6 \\
\hline
Cell 7 & & Cell 9
\end{tabular}

```

Cell 1	Cell 2	Cell 3
Cell 4	Cell 5	Cell 6
Cell 7		Cell 9

Partial lines can be made with `\cline{m-n}`, where  $m$  is the beginning column number and  $n$  is the ending column number. For instance, if you wanted a line from column 2 to column 4, the corresponding command would be `\cline{2-4}`.

Likewise, you may span a cell across multiple columns with `\multicolumn{#}{x}{content}`, where  $\#$  is the number of columns,  $x$  is the justification of the text (e.g. c, l, or r), and *content* is the text. Below is an example of a complex code:

```

\begin{tabular}{l|l|}
\hline
\multicolumn{2}{c|}{18 and over} \\
\hline
C. Anderson & 20 pts \\
\hline
D. Smith & 15 pts \\
\hline
A. Britten & 15 pts
\end{tabular}

```

```

\hline
\multicolumn{2}{|c|}{16 to 17} \\
\hline
B. Simpson & 13 pts \\
\hline
C. Griffen & 9 pts \\
\hline
\end{tabular}

```

18 and over	
C. Anderson	20 pts
D. Smith	15 pts
A. Britten	15 pts
16 to 17	
B. Simpson	13 pts
C. Griffen	9 pts

### 3 Conclusion

The `tabular` can be tricky since a lot is going on in large tables. There are a few key items:

- You must know the number of columns being used.
- `&` starts a new cell in that row.
- Number of cells in a row must equal number of columns you chose—add more if necessary.
- A row must be ended with `\\` —hitting the enter key won't work.
  - You do not need to use `\\` with `\hline`.

If a table is not showing up right, subtract any `\hline`, `\cline`, or `\multicolumn` commands. In my experience, tables are often confused by not having the right values in `\cline` and `\multicolumn` arguments (e.g. values do not sum up to the number of columns).