

Introduction to Scientific Typesetting

Lesson 3: Lists and Tables

Ryan Higginbottom

January 12, 2011

An Overview

Lists

Tables

Lists

Tables

An Overview

Lists

The `itemize`
environment

The `enumerate`
Environment

The `description`
Environment

Nesting Lists

Customizing `itemize`

The `outlines`
package

More with `outlines`

Practice

Tables

Lists

The `itemize` environment

An Overview

Lists

The `itemize`
environment

The `enumerate`
Environment

The `description`
Environment

Nesting Lists

Customizing `itemize`

The outlines
package

More with outlines

Practice

Tables

\LaTeX provides three list environments: `itemize`, `enumerate`, and `description`.

Within each environment, `\item` gives you the next item. Also, `\item` must be the first thing in the environment.

Some things to notice about the `itemize` environment:

- It is a list with bullet points.
- It has no numbers.

```
\begin{itemize}
\item It is a list with bullet points.
\item It has no numbers.
\end{itemize}
```

The enumerate Environment

An Overview

Lists

The `itemize`
environment

The `enumerate`
Environment

The `description`
Environment

Nesting Lists

Customizing `itemize`

The outlines
package

More with outlines

Practice

Tables

Things I like about the `enumerate` environment:

1. It also produces a list.
2. It enumerates the list.

```
\begin{enumerate}  
\item It also produces a list.  
\item It enumerates the list.  
\end{enumerate}
```

The description Environment

An Overview

Lists

The `itemize`
environment

The `enumerate`
Environment

The `description`
Environment

Nesting Lists

Customizing `itemize`

The `outlines`
package

More with `outlines`

Practice

Tables

The `description` environment is generally used for stating definitions.

Spanish The language spoken in Spain.

French The language spoken in France.

```
\begin{description}
\item[Spanish] The language spoken in Spain.
\item[French] The language spoken in France.
\end{description}
```

Note: the optional argument is the whole point here.

It is easy to *nest* the listing environments.

1. This is the first level in `enumerate`.

(a) This is the second level.

```
\begin{enumerate}
\item This is the first level.
  \begin{enumerate}
  \item This is the second level.
  \end{enumerate}
\end{enumerate}
```

This can be done with `itemize` as well.

The standard “bullets” that come in the `itemize` environment are dictated by the document class that you use.

- Here is the first level.
 - Here is the second level.
 - Here is the third level.

These symbols can be changed on a case-by-case basis:

- ★ This item begins with a star.
- † This item begins with a dagger.

```
\item[ $\star$ ]
```

```
\item[ $\dagger$ ]
```

The outlines package

An Overview

Lists

The `itemize`
environment

The `enumerate`
Environment

The `description`
Environment

Nesting Lists

Customizing `itemize`

The `outlines`
package

More with `outlines`

Practice

Tables

Nesting lists is made easy with the `outlines` package. Everything goes inside of an `outline` environment.

```
\begin{outline}
\1 This is the first level.
\2 Easy to move to level two.
\3 Level three is easy too.
\1 Back to level one.
\end{outline}
```

- This is the first level.
 - Easy to move to level two.
 - Level three is easy too.
- Back to level one.

By default, the `outline` environment uses `itemize`. We can easily make it use `enumerate`.

```
\begin{outline}[enumerate]
\1 This is the first level.
\2 Easy to move to level two.
\3 Level three is easy too.
\1 Back to level one.
\end{outline}
```

1. This is the first level.
 - (a) Easy to move to level two.
 - i. Level three is easy too.
2. Back to level one.

An Overview

Lists

The `itemize`
environment

The `enumerate`
Environment

The `description`
Environment

Nesting Lists

Customizing `itemize`

The `outlines`
package

More with `outlines`

Practice

Tables

Let's practice!

Open up the first example PDF file from Sakai, and reproduce it.

An Overview

Lists

Tables

Getting Started with
Tables

Column Specifications

More Control

Even More Control

Practice

More Column
Specifications

Column Width
Example

Intercolumn Command
Example

Extra Column
Specifiers

Practice

Changing the Line
Thickness

The `multirow`
Package

Practice

Tables on Multiple
Pages

An Example

Tables

Getting Started with Tables

An Overview

Lists

Tables

Getting Started with
Tables

Column Specifications

More Control

Even More Control

Practice

More Column
Specifications

Column Width
Example

Intercolumn Command
Example

Extra Column
Specifiers

Practice

Changing the Line
Thickness

The `multirow`
Package

Practice

Tables on Multiple
Pages

An Example

A table is generated by the `tabular` environment. \LaTeX treats the table as one big symbol.

Consequences:

- Tables (usually) cannot stretch across pages.
- Usually tables should be *displayed*, like in the center environment.

The syntax for `tabular`:

```
\begin{tabular}{column specifications}  
table cells, separate columns with &  
separate rows with \\  
draw horizontal lines with \hline  
\end{tabular}
```

Column Specifications

An Overview

Lists

Tables

Getting Started with
Tables

Column Specifications

More Control

Even More Control

Practice

More Column
Specifications

Column Width
Example

Intercolumn Command
Example

Extra Column
Specifiers

Practice

Changing the Line
Thickness

The `multirow`
Package

Practice

Tables on Multiple
Pages

An Example

For each column, you type `l`, `c`, or `r`, depending on the justification you want. You can put in lines between columns with `|`.

duck	vulture
humming bird	bee

```
\begin{center}
\begin{tabular}{|l|r|} \hline
duck & vulture \\ \hline
humming bird & bee \\ \hline
\end{tabular}
\end{center}
```

An Overview

Lists

Tables

Getting Started with
Tables

Column Specifications

More Control

Even More Control

Practice

More Column
Specifications

Column Width
Example

Intercolumn Command
Example

Extra Column
Specifiers

Practice

Changing the Line
Thickness

The `multirow`
Package

Practice

Tables on Multiple
Pages

An Example

Instead of `\hline`, the command `\cline{1-2}` draws a horizontal line through columns 1 and 2 only (for example).

duck	vulture	goose
humming bird	bee	eagle

```
\begin{center}
\begin{tabular}{|l|r|c|}\hline
duck & vulture & goose\\ \cline{1-2}
humming bird & bee & eagle \\ \hline
\end{tabular}
\end{center}
```

An Overview

Lists

Tables

Getting Started with
Tables

Column Specifications

More Control

Even More Control

Practice

More Column
Specifications

Column Width
Example

Intercolumn Command
Example

Extra Column
Specifiers

Practice

Changing the Line
Thickness

The `multirow`
Package

Practice

Tables on Multiple
Pages

An Example

The command `\multicolumn` in tables allows you to stretch an entry across columns.

duck	vulture	goose
humming bird	bee	

```
\begin{tabular}{|l|r|c|}\hline  
duck & vulture & goose\\ \hline  
humming bird & \multicolumn{2}{c|}{bee} \\ \hline  
\end{tabular}
```

Syntax:

```
\multicolumn{#1}{#2}{#3}
```

#1 — number of columns to span

#2 — alignment of the new super-column

#3 — text of column

An Overview

Lists

Tables

Getting Started with
Tables

Column Specifications

More Control

Even More Control

Practice

More Column
Specifications

Column Width
Example

Intercolumn Command
Example

Extra Column
Specifiers

Practice

Changing the Line
Thickness

The `multirow`
Package

Practice

Tables on Multiple
Pages

An Example

Try to reproduce this table:

one	two	three	four
five	six	seven	eight
nine	wide ten		
longer words	longer words	longer words	longer words

More Column Specifications

An Overview

Lists

Tables

Getting Started with
Tables

Column Specifications

More Control

Even More Control

Practice

More Column
Specifications

Column Width
Example

Intercolumn Command
Example

Extra Column
Specifiers

Practice

Changing the Line
Thickness

The `multirow`
Package

Practice

Tables on Multiple
Pages

An Example

Within the `tabular` environment, so far we know the column commands `l`, `c`, and `r`. Two more are provided by default.

- `p{size}` — This makes a column of a fixed width. Everything in that column is left justified. Entries in other rows are justified vertically at the top of the cell.
- `@{command}` — This is a way of putting commands between columns.

Column Width Example

An Overview

Lists

Tables

Getting Started with
Tables

Column Specifications

More Control

Even More Control

Practice

More Column
Specifications

Column Width
Example

Intercolumn Command
Example

Extra Column
Specifiers

Practice

Changing the Line
Thickness

The `multirow`
Package

Practice

Tables on Multiple
Pages

An Example

Note the difference:

This entry goes on and on and on for a while	Make it stop
--	--------------

This entry goes on and on and on for a while	Make it stop
---	--------------

```
\begin{tabular}{|l|c|} \hline
```

...

```
\begin{tabular}{|p{1.5in}|c|} \hline
```

Intercolumn Command Example

An Overview

Lists

Tables

Getting Started with
Tables

Column Specifications

More Control

Even More Control

Practice

More Column
Specifications

Column Width
Example

Intercolumn Command
Example

Extra Column
Specifiers

Practice

Changing the Line
Thickness

The `multirow`
Package

Practice

Tables on Multiple
Pages

An Example

The `@{...}` command is especially useful when you want to align along decimal points. Notice:

A nice number	2.5177
A better number	3.14
A super number	554.8

```
\begin{tabular}{|c|r@{.}l|} \hline  
A nice number & 2&5177 \\ \hline  
A better number & 3&14 \\ \hline  
A super number & 554&8 \\ \hline  
\end{tabular}
```

Extra Column Specifiers

An Overview

Lists

Tables

Getting Started with
Tables

Column Specifications

More Control

Even More Control

Practice

More Column
Specifications

Column Width
Example

Intercolumn Command
Example

Extra Column
Specifiers

Practice

Changing the Line
Thickness

The `multirow`
Package

Practice

Tables on Multiple
Pages

An Example

If you load the `array` package in your preamble, you get two other column specifiers.

- `m{size}` — The same as `p{size}` except other entries in the row are vertically center justified.
- `b{size}` — The same as `p{size}` except other entries in the row are vertically bottom justified.

An Overview

Lists

Tables

Getting Started with
Tables

Column Specifications

More Control

Even More Control

Practice

More Column
Specifications

Column Width
Example

Intercolumn Command
Example

Extra Column
Specifiers

Practice

Changing the Line
Thickness

The `multirow`
Package

Practice

Tables on Multiple
Pages

An Example

Let's practice!

Open up the second example PDF file from Sakai, and reproduce it.

Changing the Line Thickness

Sometimes you'd like a thicker line width in your tables (default is 0.4 pt). We can change the `arrayrulewidth` to do this.

left	right
left left	right right

left	right
left left	right right

```
\setlength{\arrayrulewidth}{2pt}
\begin{tabular}{|l|r|} \hline
left & right \\ \hline
left left & right right \\ \hline
\end{tabular}
```

Note: This will make a “global” change unless it is inside of some environment. One solution is to put `{` before `\setlength...` and `}` after `\end{tabular}`.

An Overview

Lists

Tables

Getting Started with
Tables

Column Specifications

More Control

Even More Control

Practice

More Column
Specifications

Column Width
Example

Intercolumn Command
Example

Extra Column
Specifiers

Practice

Changing the Line
Thickness

The `multirow`
Package

Practice

Tables on Multiple
Pages

An Example

The multirow Package

An Overview

Lists

Tables

Getting Started with
Tables

Column Specifications

More Control

Even More Control

Practice

More Column
Specifications

Column Width
Example

Intercolumn Command
Example

Extra Column
Specifiers

Practice

Changing the Line
Thickness

The multirow
Package

Practice

Tables on Multiple
Pages

An Example

With `\usepackage{multirow}` in your preamble, you can have entries that span multiple rows, just the same way that we can have entries that span multiple columns now.

```
\multirow{nrows}{width}{contents}
```

`nrows` number of rows to span

`width` width of entry, can type *

`contents` contents of the entry

one	two	three
four	five	
six	seven	eight

```
\begin{tabular}{|1|1|1|} \hline  
one & two & \multirow{2}{*}{three} \\ \cline{1-2}  
four & five & \\ \hline
```

An Overview

Lists

Tables

Getting Started with
Tables

Column Specifications

More Control

Even More Control

Practice

More Column
Specifications

Column Width
Example

Intercolumn Command
Example

Extra Column
Specifiers

Practice

Changing the Line
Thickness

The `multirow`
Package

Practice

Tables on Multiple
Pages

An Example

Try to reproduce this table:

span #1		duck
owl	goose	swan
sparrow	dove	finch
vulture		porcupine

Tables on Multiple Pages

An Overview

Lists

Tables

Getting Started with
Tables

Column Specifications

More Control

Even More Control

Practice

More Column
Specifications

Column Width
Example

Intercolumn Command
Example

Extra Column
Specifiers

Practice

Changing the Line
Thickness

The `multirow`
Package

Practice

Tables on Multiple
Pages

An Example

We said before that a table could not be split across pages. Without extra packages, that's true. We can make it happen with the `supertabular` package.

Since this splits tables across pages, it has a way to specify the *table header* that shows up on every page.

`\tablehead{rows}` — repeated on every page

`\tablefirsthead{rows}` — table header for only first page

`\tabletail{rows}` — repeated on every page

`\tablelasttail{rows}` — table footer for only last page

An Overview

Lists

Tables

Getting Started with
Tables

Column Specifications

More Control

Even More Control

Practice

More Column
Specifications

Column Width
Example

Intercolumn Command
Example

Extra Column
Specifiers

Practice

Changing the Line
Thickness

The `multirow`
Package

Practice

Tables on Multiple
Pages

An Example

Access the third example file (.tex) on Sakai.

Use the LaTeX \Rightarrow PDF profile.

Build and view.