

# The Basics

- Backslash “\” precedes commands
- Arguments are enclosed in braces “{ }”
- Everything after a percent “%” to the end of the line is a comment
- Paragraphs are separated by blank lines

```
% prospectus.tex
%
% This LaTeX source file provides a (somewhat trivial) template for a
% Programming Languages research paper prospectus.
%
%
% Use the standard article template.
%
\documentclass{article}

%
% Set the title, author, and date.
%
\title{My Insanely Great Paper Prospectus}
\author{Jimmy Neutron}
%
% If you want to datestamp your prospectus automatically, insert \today between
% the braces below.
%
\date{}

%
% The document proper.
%
\begin{document}

% Add the title section.
\maketitle

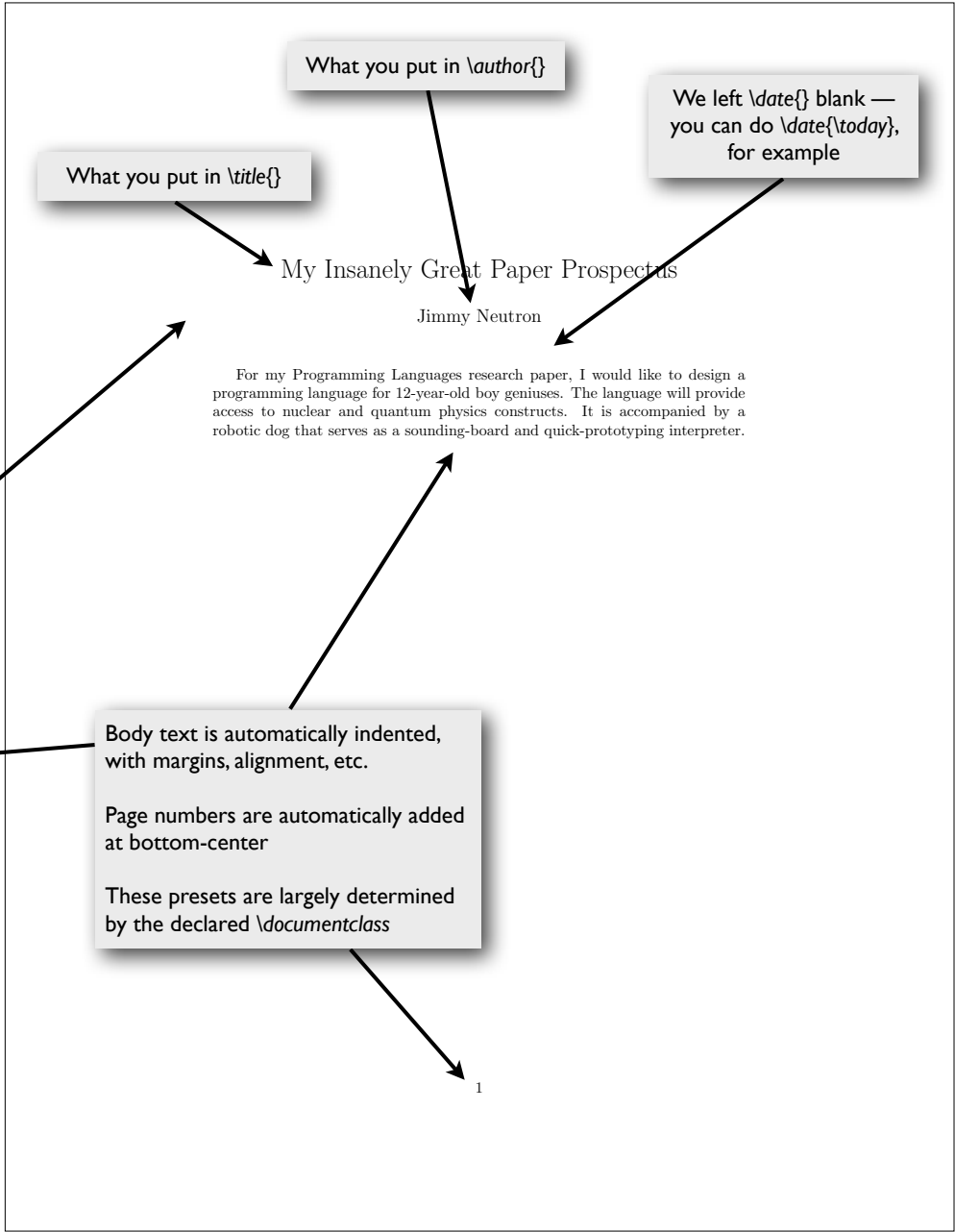
%
% Body text.
%
For my Programming Languages research paper, I would like to design a programming language for
12-year-old boy geniuses. The language will provide access to nuclear and quantum physics
constructs. It is accompanied by a robotic dog that serves as a sounding-board and quick-
prototyping interpreter.

\end{document}
```

Everything before `\begin{document}` is called the “preamble” — it defines properties of your document

The document actually starts here; `\maketitle` generates the title page, using properties you defined in the preamble

LaTeX is pre-HTML/XML, but it is conceptually similar; for example, certain blocks require a beginning and an ending



What you put in `\author{}`

We left `\date{}` blank — you can do `\date{\today}`, for example

What you put in `\title{}`

My Insanely Great Paper Prospectus  
Jimmy Neutron

For my Programming Languages research paper, I would like to design a programming language for 12-year-old boy geniuses. The language will provide access to nuclear and quantum physics constructs. It is accompanied by a robotic dog that serves as a sounding-board and quick-prototyping interpreter.

Body text is automatically indented, with margins, alignment, etc.

Page numbers are automatically added at bottom-center

These presets are largely determined by the declared `\documentclass`

# More Features

- LaTeX is all about focusing on *content*, not *presentation*

`\usepackage` resembles a module import — it makes new commands available

`\abstract` automatically adds a title and special font/indentation settings — you just provide the text

There are some commands that affect presentation — for example, the `\text*` commands will change the font of the enclosed text

The single `\tableofcontents` command creates all of this for you! `\pagebreak` does what you think it does

How does `\tableofcontents` work its magic? You explicitly state when sections, subsections, and subsubsections start

```
%
% paper.tex
%
% This LaTeX source file provides a template for a Programming Languages research paper.
%
%
% Use the standard article template.
%
\documentclass{article}

% Load up special logo commands.
\usepackage{doc}

%
% Set the title, author, and date.
%
\title{My Insanely Great Programming Language}
\author{Jimmy Neutron}
%
% If you want to datestamp your paper automatically, insert \today between
% the braces below.
%
\date{}

%
% The document proper.
%
\begin{document}

% Add the title section.
\maketitle

% Add an abstract.
\abstract{
Describe your paper in 100-200 words, give or take. The command-line \texttt{wc} utility is
really useful here!
}

% Add the table of contents on a new page.
\pagebreak
\tableofcontents

% Start the paper on a new page.
\pagebreak

%
% Body text.
%
\section{Introduction}

Describe the language in general terms; motivations behind this language; distinguishing
characteristics over other languages.
```

My Insanely Great Programming Language  
Jimmy Neutron  
**Abstract**  
Describe your paper in 100-200 words, give or take. The command-line  
we utility is really useful here!

Contents

1	Introduction	3
2	Previous Work	3
3	Syntax and Semantics Overview	3
4	Semantics of My Insanely Great Programming Language	3
4.1	Naming, Declarations, Scoping	3
4.2	Types	3
4.3	Expressions, Statements	3
4.4	Subroutines, Classes, Modules	3
4.5	Concurrency	3
5	Standard Library	3
6	Other Interesting Features	4
7	Conclusion	4
A	Sample Programs	4
A.1	Sample 1	4
A.2	Sample 2	4

```

\begin{document}

% Add the title section.
\maketitle

% Add an abstract.
\abstract{
Describe your paper in 100-200 words, give or take. The command-line \texttt{wc} utility is
really useful here!
}

% Add the table of contents on a new page.
\pagebreak
\tableofcontents

% Start the paper on a new page.
\pagebreak

%
% Body text.
%
\section{Introduction}

Describe the language in general terms; motivations behind this language;
characteristics over other languages.

```

Note how sections and subsections are automatically formatted and numbered for you

The `\BibTeX` command is available because we imported the `doc` package in the preamble

```

\section{Previous Work}

Summarize any existing precursors to this language, as well as any other prior work that is
relevant to this language. You will very likely hone your \BibTeX skills here. For example,
you would cite our textbook in this way \cite{Textbook}. I once wrote a paper on a visual
query language called MQuery \cite{MQuery}.

\section{Syntax and Semantics Overview}
\label{syntax}

Specify the micro (lexical) and macro (EBNF) syntax of your language. Highlight a few
semantic rules.

\section{Semantics of My Insanely Great Programming Language}

For the language description proper, include sample code and highlight distinguishing or
interesting points that are specific to the language in each section. \LaTeX makes it really
easy to refer to other sections, such as the one on syntax (Section-\ref{syntax}) or even a
section that is coming up, such as the one on concurrency (Section-\ref{concurrency}).

\subsection{Naming, Declarations, Scoping}

\subsection{Types}

\subsection{Expressions, Statements}

\subsection{Subroutines, Classes, Modules}

\subsection{Concurrency}
\label{concurrency}

Show how to define threads and how to do synchronization.

\section{Standard Library}

If you can list all of the library units, do so (for example, C++ has only 50 modules in its
library, but Java has 177). At least show the organization of library units within the
standard library. Describe the important ones.

\section{Other Interesting Features}

Include this section only if there are other interesting features that have not already been
discussed.

\section{Conclusion}

Provide a synopsis of the entire language, and touch on potential future work that can be
based on this language. Feel free to cite other work liberally.

\appendix

\section{Sample Programs}

```

Note how citations are created automatically — you just specify the citation identifier (see *paper.bib* on the next page)

You can assign labels (`\label`) that you can reference (`\ref`) elsewhere in the text; LaTeX converts these automatically into section numbers

## 1 Introduction

Describe the language in general terms; motivations behind this language; distinguishing characteristics over other languages.

## 2 Previous Work

Summarize any existing precursors to this language, as well as any other prior work that is relevant to this language. You will very likely hone your BibTeX skills here. For example, you would cite our textbook in this way [Sco00]. I once wrote a paper on a visual query language called MQuery [DC96].

## 3 Syntax and Semantics Overview

Specify the micro (lexical) and macro (EBNF) syntax of your language. Highlight a few semantic rules.

## 4 Semantics of My Insanely Great Programming Language

For the language description proper, include sample code and highlight distinguishing or interesting points that are specific to the language in each section.  $\LaTeX$  makes it really easy to refer to other sections, such as the one on syntax (Section 3) or even a section that is coming up, such as the one on concurrency (Section 4.5).

### 4.1 Naming, Declarations, Scoping

### 4.2 Types

### 4.3 Expressions, Statements

### 4.4 Subroutines, Classes, Modules

### 4.5 Concurrency

Show how to define threads and how to do synchronization.

## 5 Standard Library

If you can list all of the library units, do so (for example, C++ has only 50 modules in its library, but Java has 177). At least show the organization of library units within the standard library. Describe the important ones.

```
\subsection{Subroutines, Classes, Modules}
```

```
\subsection{Concurrency}  
\label{concurrency}
```

Show how to define threads and how to do synchronization.

```
\section{Standard Library}
```

If you can list all of the library units, do so (for example, C++ has only 50 modules in its library, but Java has 177). At least show the organization of library units within the standard library. Describe the important ones.

```
\section{Other Interesting Features}
```

Include this section only if there are other interesting features that have not already been discussed.

```
\section{Conclusion}
```

Provide a synopsis of the entire language, and touch on potential future work that can be based on this language. Feel free to cite other work liberally.

```
\appendix
```

```
\section{Sample Programs}
```

The `\appendix` command tells LaTeX that your main paper is done; the section numbering scheme then changes

## 6 Other Interesting Features

Include this section only if there are other interesting features that have not already been discussed.

## 7 Conclusion

Provide a synopsis of the entire language, and touch on potential future work that can be based on this language. Feel free to cite other work liberally.

## A Sample Programs

No "Hello World"s here, please.

### A.1 Sample 1

### A.2 Sample 2

## References

[DC96] John David N. Dionisio and Alfonso F. Cárdenas. MQuery: a visual query language for multimedia, timeline, and simulation data. *Journal of Visual Languages and Computing*, 7:377–401, December 1996.

[Sco00] Michael L. Scott. *Programming Language Pragmatics*. Morgan Kaufmann, 2000.

No "Hello World"s here, please.

```
\subsection{Sample 1}
```

```
\subsection{Sample 2}
```

```
% Generate the bibliography.
```

```
\bibliography{paper}
```

```
\bibliographystyle{alpha}
```

```
\end{document}
```

LaTeX has had smart quotes for decades: use grave accents for left quotes and apostrophes for right quotes

LaTeX has a companion program, BibTeX, that manages your bibliography automatically — this alone is worth virtually the entire learning curve

Like `\documentclass`, `\bibliographystyle` determines the overall formatting strategy for your references

```
%  
% paper.bib  
%  
% This is a sample BibTeX database file showing how you can manage your references  
% and citations in one place, with LaTeX taking care of formatting, numbering, and  
% collating them in your final work.  
%  
@book{  
  Textbook,  
  author = {Michael L Scott},  
  title = {Programming Language Pragmatics},  
  publisher = {Morgan Kaufmann},  
  year = {2000}  
}  
  
@article{  
  MQuery,  
  author = {{John David} N. Dionisio and Alfonso F. C\`{a}rdenas},  
  title = {{MQuery}: A Visual Query Language for Multimedia, Timeline, and Simulation  
Data},  
  journal = {Journal of Visual Languages and Computing},  
  year = {1996},  
  volume = {7},  
  month = {December},  
  pages = {377--401}  
}
```

You maintain your references in a separate file, usually named with a `.bib` extension — this allows you to reuse your references in other documents

The `.bib` file is a text-based database; each entry starts with its type (e.g., *book*, *article*, etc.) then states its identifier (which is what you use in your `\cite` commands in the LaTeX document); everything else is a name-value pair

Note how your bibliography entries are automatically labeled, indented, and formatted

BibTeX functionality is provided as a separate program, and requires several "passes" to see the final result — for the most consistent results, run *latex* twice, then *bibtex*, then *latex* a third time when processing your bibliography (i.e., *latex latex bibtex latex*)