

Membuat Dokumen Dengan L^AT_EX

(versi 0.3)



Membuat Dokumen Dengan

L^AT_EX

(versi 0.3)

Tulisan ini bebas dibaca dan disebarluaskan.
Copyright atas tulisan ini adalah milik saya sepenuhnya
©Tedy Tirtawidjaja
tedytirtawidjaja@gmail.com

created by : TEDY



Tulisan ini sengaja dibuat untuk semua orang yang berniat menggunakan \LaTeX saat membuat dokumen. Semua yang ditulis dalam buku petunjuk ini sudah pernah dipraktekkan oleh penulis dalam setiap kesempatan membuat dokumen, baik itu laporan kuliah, artikel, dll. Jika ada koreksi, kritik, atau saran tentang tulisan ini silakan menghubungi penulis via email ke alamat tedytirtawidjaja@gmail.com.

Daftar Isi

Copyright	1
Daftar Isi	2
Daftar Tabel	3
1 Pendahuluan	4
1.1 Sejarah Singkat \LaTeX	4
1.2 Media Instalasi	5
1.3 Langkah-Langkah Instalasi	5
2 Langkah Praktis	6
3 Perintah-Perintah \LaTeX	8
3.1 Format Perintah	8
3.2 Preamble, Deklarasi & Environment	9
3.3 Spasi Dalam \LaTeX	9
3.4 Hyphenation	10
3.5 Alignment	10
3.6 Bahasa	11
3.7 Keterangan	11
3.8 Karakter Khusus	11
3.9 Font Dalam \LaTeX	12
3.9.1 Jenis Font	12
3.9.2 Bentuk Font	13
3.9.3 Ukuran Font	13
3.10 Mode Verbatim	14
4 Struktur Dasar Sebuah Dokumen	15
4.1 Document Class	15
4.1.1 Document Class Options	15
4.2 Paket-Paket Dalam \LaTeX	16
4.3 Document Environment	17
4.4 Penulisan Judul	17
4.5 Abstrak	18
4.6 Sistematika Isi Dokumen	18
4.7 Daftar Berurut	19
4.8 Daftar Isi	20
4.9 Tabel & Gambar	21
4.9.1 Tabel	21
4.9.2 Gambar	22
4.10 Referensi / Rujukan	23
4.11 Daftar Pustaka	23
4.12 Indeks	24

5	Notasi Matematika Dalam L^AT_EX	25
5.1	Penulisan Notasi Matematika Dalam Paragraf	25
5.2	Paragraf Khusus Matematika	25
5.3	Font Dalam Matematika	25
5.4	Tanda Kurung Dalam Matematika	27
5.5	Penulisan Akar	28
5.6	Penulisan Pecahan	28
5.7	Penulisan Array & Matriks	29
5.8	Penulisan Vektor	29
5.9	Penulisan Fungsi Matematika	29
5.10	Simbol-Simbol Matematika	30
	Bibliografi	32

Daftar Tabel

3.1	Karakter-Karakter Khusus	12
3.2	Aksen & Simbol Bahasa Asing	12
4.1	Struktur Dokumen	18
4.2	Contoh Tabel	21
5.1	Huruf Kecil Yunani	27
5.2	Huruf Kapital Yunani	27
5.3	Delimiter	28
5.4	Fungsi Matematika	30
5.5	<i>Binary Operator</i>	30
5.6	Simbol-Simbol dengan Limit	31
5.7	Tanda Panah Matematis	31
5.8	Simbol Matematika Lainnya	31

Bab 1

Pendahuluan

1.1 Sejarah Singkat L^AT_EX

T_EX adalah bahasa pemrograman yang diciptakan khusus dan menjadi bagian utama dari sistem pengaturan teks hasil pengetikan (*typesetting system*) yang menghasilkan dokumen/publikasi teks dan matematis yang berkualitas. T_EX diciptakan oleh Prof. Donald Knuth sekitar tahun 1978. Awalnya pada akhir tahun 1970 Prof. Donald Knuth menciptakan T_EX untuk merivisi volume kedua dari *magnum opus*-nya¹ yaitu buku *The Art of Computer Programming*; karena banyak simbol matematika yang tidak dapat dicetak dengan sempurna pada masa itu.

Dalam pengembangannya, T_EX tersedia untuk digunakan dalam berbagai macam komputer dan sistem operasi dan dirancang secara spesifik sehingga file masukan (file yang mengandung perintah-perintah pengolah kata, ini bukan sistem WYSIWYG²) akan menjadi portabel, dan bisa menghasilkan keluaran yang sama di segala macam sistem operasi yang memrosesnya - pemenggalan kata yang sama, pergantian baris, pergantian halaman, dll.

Pada tahun 1985 Leslie Lamport di Digital Equipment Corporation menciptakan L^AT_EX . L^AT_EX merupakan *user interface* dari T_EX . Leslie menciptakan L^AT_EX untuk mengotomatisasi semua perintah umum untuk menyiapkan sebuah dokumen. Dengan menggunakan L^AT_EX antara penulis dengan *typesetter* dapat dengan mudah menggunakan kemampuan pengaturan tulisan yang dimiliki oleh T_EX tanpa harus memahami *underlying language*. L^AT_EX sangat dianjurkan untuk semua orang kecuali *professional typographic programmer* and peneliti komputer, yang ingin mempelajari T_EX .

Saat ini L^AT_EX dan T_EX tersedia bebas di Internet dan dapat digunakan oleh perorangan.

L^AT_EX pada dasarnya adalah program yang dibuat sebagai *typesetting tool*. Secara garis besar langkah pembuatan dokumen menggunakan L^AT_EX adalah sebagai berikut : Mula-mula dokumen diketik pada suatu *text editor*, kemudian ditambahkan perintah-perintah L^AT_EX untuk mengatur isi dokumennya. Setelah seluruh dokumen sudah dilengkapi dengan perintah-perintah L^AT_EX , dokumen tersebut harus di-*compile* untuk menghasilkan output berupa dokumen yang diinginkan. Dokumen hasilnya tersebut dapat berupa dokumen dalam format *.pdf³, *.dvi⁴, atau *.ps⁵.

¹(Latin) *magnum* = besar & *opus* = karya/pekerjaan

²Sistem WYSIWYG=*what you see is what you get*- merupakan program pengolah kata dimana hasil pengaturan tulisan langsung terlihat di display, misalnya program *word processor* seperti Microsoft Word, Open Office, dll.

³PDF = Portable Document Format

⁴File dalam format DVI adalah standar output L^AT_EX . DVI adalah singkatan dari DeVice-Independent, maksudnya adalah file dengan format *.dvi dapat dicetak dengan menggunakan hampir semua jenis peralatan cetak tipografi.

⁵PS = Post Script

1.2 Media Instalasi

Untuk dapat menggunakan \LaTeX dalam lingkungan Microsoft Windows ada beberapa hal yang perlu disiapkan sebelumnya; di antaranya adalah :

1. Paket instalasi \LaTeX – dalam hal ini digunakan `small-miktex-2.4.1705`⁶.
Paket instalasi ‘`small-miktex-2.4.1705`’ setelah diinstal nantinya akan dipakai sebagai generator dokumen yang sudah diformat menggunakan “bahasa pemrograman” (baca : perintah-perintah) yang dimiliki \LaTeX .
2. *Text editor* – misalnya yang sudah ada dalam Microsoft Windows yaitu Notepad.
Pada prinsipnya dokumen yang akan diatur isinya menggunakan \LaTeX cukup dituliskan menggunakan *text editor* saja. Semua dokumen beserta perintah pengaturan teksnya dituliskan dalam satu dokumen tunggal dengan ekstensi `*.tex`.
3. Antarmuka grafis *user* dengan \LaTeX – dalam hal ini digunakan ‘`TeXniCenter 1 ver.Beta 6.31`’.
Antarmuka grafis⁷ ini memudahkan *user* untuk menggunakan perintah-perintah \LaTeX . Dengan menggunakan TeXniCenter ini beberapa perintah sudah disediakan *shortcut*nya sehingga *user* tidak perlu lagi mengetikkan perintah tersebut di dalam dokumennya, cukup memilih perintah yang diinginkan maka TeXniCenter yang akan mengetikkan perintah tersebut. Jauh lebih efisien menggunakan TeXniCenter sebagai pengganti *text editor* biasa.

1.3 Langkah-Langkah Instalasi

Langkah-langkah instalasi :

1. Jalankan ‘`small-miktex-2.4.1705`’ dengan double klik pada filenya. Kemudian ikuti petunjuk instalasi yang disediakan. Setelah proses instalasi selesai Anda perlu menguji apakah \LaTeX telah terinstal dengan benar, caranya adalah :
 - Jalankan Command Prompt; caranya : Klik Start - Run, ketik `cmd`, kemudian klik OK.
 - Pada Command Prompt ketik `latex` kemudian tekan Enter.
 - Jika \LaTeX telah terinstal dengan benar, akan muncul tulisan seperti berikut ini :

```
This is e-TeX, Version 3.141592-2.2 <MiKTeX 2.4>
**
```

2. Jalankan ‘`TeXniCenter 1 ver.Beta 6.31`’ dengan double klik pada filenya. Kemudian ikuti petunjuk instalasi yang disediakan.

Saat TeXniCenter dijalankan untuk pertama kali jangan lupa memilih opsi “INTEGRATE `TEXNICE`NTER WITH INSTALLED `LATEX`”

Jika semua langkah di atas sudah dilakukan maka komputer Anda telah siap untuk digunakan menghasilkan dokumen yang diformat menggunakan \LaTeX .

Selamat Mencoba

\LaTeX

⁶Media instalasi ini dapat didownload secara gratis di <http://www.miktex.org/>

⁷GUI = *graphical user interface*

Bab 2

Langkah Praktis

Untuk menghasilkan dokumen dapat dilakukan dengan beberapa langkah latihan praktis berikut ini¹ :

1. Dengan menggunakan *text-editor* Anda, tuliskan contoh dokumen berikut ini selengkapnya :

```
\documentclass[12pt]{article}
\usepackage{palatino,url}
\begin{document}
\section*{My first document}
This is a short example of a \LaTeX\ document I wrote
on \today. It shows a few simple features of automated
typesetting, including

\begin{itemize}
\item setting the default font size to 12pt;
\item specifying 'article' type for formatting;
\item using the Palatino typeface;
\item adding special formatting for URIs;
\item formatting a heading in 'section' style;
\item using the \LaTeX\ logo;
\item generating today's date;
\item formatting a list of items;
\item centering and italicizing;
\item autonumbering the pages.
\end{itemize}

\subsection*{More information}
This example was taken from 'Formatting Information',
which you can download from
\url{http://www.ctan.org/tex-archive/info/beginlatex/}
and use as a teach-yourself guide.

\begin{center}
\textit{Have a nice day!}
\end{center}
\end{document}
```

2. Simpan dokumen tersebut dengan nama **latihan.tex**. Ingat bahwa semua dokumen \LaTeX menggunakan format ***.tex** .

¹Hal ini dengan asumsi \LaTeX telah terinstal dengan benar di komputer Anda.

3. Buka Command Prompt, lalu ketikkan perintah (asumsi file `latihan.tex` ada di direktori `d:\data`) : `c:\> latex d:\data\latihan.tex` Perintah ini akan menghasilkan keluaran berupa dokumen dalam format `*.dvi`.

4. Untuk melihat output dokumennya, masih pada Command Prompt ketikkan perintah :

```
c:\> latihan.dvi
```

Maka akan ditampilkan dokumen yang Anda buat dengan menggunakan YAP DVI-viewer. Hasilnya akan terlihat seperti ini :

My first document

This is a short example of a \LaTeX document I wrote on 11 Maret 2006. It shows a few simple features of automated typesetting, including

- setting the default font size to 12pt;
- specifying ‘article’ type for formatting;
- using the Palatino typeface;
- adding special formatting for URIs;
- formatting a heading in ‘section’ style;
- using the \LaTeX logo;
- generating today’s date;
- formatting a list of items;
- centering and italicizing;
- autonumbering the pages.

More information

This example was taken from ‘Formatting Information’, which you can download from <http://www.ctan.org/tex-archive/info/beginlatex/> and use as a teach-yourself guide.

Have a nice day!

5. Jika dokumen yang ingin menghasilkan dokumen dengan format `*.pdf` maka pada poin ketiga di atas kita ganti perintahnya menjadi :

```
c:\> pdflatex d:\data\latihan.tex
```

Dan untuk melihat file keluarannya maka poin keempat di atas kita ganti perintahnya menjadi seperti berikut ini : `c:\> latihan.pdf`

Bab 3

Perintah-Perintah L^AT_EX

3.1 Format Perintah

Semua perintah L^AT_EX diawali dengan tanda *backslash* (`\`). Tanda ini memberitahukan kepada L^AT_EX untuk melakukan hal tertentu pada bagian dokumen tersebut. Perintah-perintah dalam L^AT_EX biasanya sudah cukup menjelaskan apa yang akan dilakukan L^AT_EX pada dokumen kita. Misalnya:

`\tableofcontents` : perintah ini digunakan untuk menambahkan daftar isi sebuah dokumen.

Antara suatu perintah dengan isi dokumen harus dibeli 1 spasi kosong. Hal ini supaya L^AT_EX mampu membedakan mana bagian yang merupakan perintah dan mana bagian yang merupakan isi dokumen. Misalnya :

`\small` - *Ini adalah bagian isi dokumen* : tanda *underscore* tersebut maksudnya antara perintah dan isi dokumen harus diselipkan satu spasi kosong. Perintah ini akan membuat tulisan menjadi berukuran kecil.

Semua perintah L^AT_EX sifatnya case-sensitive (memperhatikan perbedaan huruf besar dan kecil), misalnya `\large` merupakan perintah yang berbeda dengan `\Large`. Ada 1 jenis perintah yang harus ada pada setiap dokumen L^AT_EX yaitu perintah `\documentclass`. Perintah jenis ini harus ada pada awal dokumen.

Sepenggal teks/kalimat bisa dikelompokkan dengan meletakkannya di antara tanda kurung kurawal `{` dan `}`. Dengan melakukan pengelompokan, perintah yang diberikan dalam kelompok itu hanya akan mempengaruhi apa yang ada di dalam kelompok dan tidak akan mempengaruhi bagian dokumen yang lain. Misalnya :

`{\bfseries ...}`

Perintah ini akan menebalkan teks yang ada di antara tanda kurung kurawal saja.

Beberapa jenis perintah L^AT_EX membutuhkan argumen. Argumen adalah informasi tambahan yang kita berikan kepada L^AT_EX sebelum menjalankan perintah yang bersangkutan. Ada 2 jenis argumen : *mandatory* dan *optional*. Ada banyak juga perintah L^AT_EX yang diikuti oleh satu atau lebih *argument*. Argumen *mandatory* harus ditulis dalam tanda kurung kurawal. Cara penulisannya adalah seperti berikut ini :

`\chapter{Pendahuluan}`

Argumen *optional* cirinya adalah dituliskan dalam tanda kurung kotak `[` dan `]`. Jenis argumen ini bisa tidak diisi. Misalnya :

`ini adalah baris pertama \\ [2cm]`
`ini adalah baris kedua`

Contoh di atas menampilkan argumen *optional* yaitu `[2cm]`, dengan argumen ini antara baris pertama dan baris kedua akan disediakan spasi vertikal 2 centimeter. Namun bila argumen `[2cm]` tidak dituliskan, L^AT_EX tetap mengatur baris kedua berada di bawah baris pertama hanya saja spasi keduanya adalah standar spasi L^AT_EX .

3.2 Preamble, Deklarasi & Environment

Yang dimaksud dengan *preamble*/pembukaan adalah bagian dari dokumen L^AT_EX di antara perintah `\documentclass` dan perintah `\begin{document}`. Hanya ada beberapa perintah yang hanya bisa diletakkan di bagian ini. Yang paling umum diletakkan dalam bagian preamble adalah deklarasi penggunaan paket-paket L^AT_EX .

Deklarasi dalam L^AT_EX artinya memberikan referensi kepada perintah L^AT_EX untuk melakukan pemformatan mulai dari titik referensi tersebut. Deklarasi itu sendiri tidak menghasilkan teks apapun. Efek yang dihasilkan oleh proses perintah tertentu bisa dilokalisasi (dibatasi cakupan efeknya) dengan menempatkan deklarasi di dalam sebuah kelompok. Misalnya :

```
\bfseries
```

Deklarasi di atas adalah sebuah deklarasi yang mengubah font (pada semua teks setelah bagian perintah tersebut) menjadi lebih tebal. Jika ingin membatasi area yang terkena efeknya, caranya adalah dengan meletakkan deklarasi tersebut di dalam tanda kurung kurawal bersama dengan teks yang ingin dikenai efek perintahnya.

Yang dimaksud dengan environment adalah teks yang berada di antara perintah `\begin {...}` dan `\end {...}`. Titik-titik tersebut adalah nama jenis dari environment yang digunakan. Kelompok teks yang berada di dalam environment tersebut akan diformat dengan metode yang dimiliki environment tersebut. Misalnya :

```
\begin {bfseries}
isi environment
\end {bfseries}
```

Environment `bfseries` akan mengubah isi dari environment menjadi memiliki font tebal semua.

3.3 Spasi Dalam L^AT_EX

Dalam dokumen L^AT_EX semua baris-baris kosong, spasi yang banyak, dan tabulasi dianggap sebagai 1 spasi atau 1 baris kosong saja selama proses pengaturan tulisan. L^AT_EX mengatur spasi dan perataan teks (*alignment*) berdasarkan perintah yang diterimanya, sehingga kita mampu mengaturnya secara tepat. Contohnya :

```
\chapter {Pendahuluan}
}
ini adalah contoh dokumen
```

Format penulisan di atas akan menghasilkan keluaran yang sama jika dituliskan seperti ini :

```
\chapter{Pendahuluan} ini adalah contoh dokumen
```

Ada perintah khusus untuk membuat spasi dengan panjang tertentu baik secara horizontal maupun vertikal, yaitu :

- Jika kita ingin membuat jarak dengan panjang tertentu antara 2 baris, kita dapat menggunakan tanda ‘\’ di akhir baris. Kita juga dapat menentukan sendiri panjang baris kosong dengan menggunakan perintah seperti contoh berikut ini :

```

baris 1 \\
\vspace{2cm}
baris 2 \\

```

Dengan perintah ini L^AT_EX akan membuat mengosongkan baris-baris sepanjang 2 centimeter. Tanpa perintah ini sejauh apapun kita membuat spasi dalam teks dokumen, L^AT_EX akan tetap menganggapnya 1 spasi.

- Jika kita ingin membuat spasi sejauh beberapa centimeter antara 2 kata dibutuhkan perintah :

```
kata 1 \hspace{2cm} kata 2
```

Dengan perintah ini L^AT_EX akan membuat spasi sejauh 2 centimeter. Sama seperti poin sebelumnya tanpa perintah ini sejauh kita membuat spasi dalam teks dokumen, L^AT_EX akan tetap menganggapnya 1 spasi.

Jadi secara umum aturan yang dapat dipakai adalah : akhiri paragraf dengan tanda ‘\\’ dan berikan 1 baris kosong antara tiap-tiap paragraf dan 1 spasi kosong antara masing-masing kata.

3.4 Hyphenation

Hyphenation/pemenggalan kata dalam L^AT_EX artinya adalah perintah yang memberitahukan kepada L^AT_EX bagaimana cara memenggal sebuah kata menurut ejaan yang benar. Kadangkala L^AT_EX tidak dapat memenggal kata dengan tepat sesuai ejaan bahasa Indonesia. Akibatnya ada kata-kata tertentu yang selalu terpotong dengan ejaan yang salah. Masalah ini dapat diatasi dengan mengatur bagaimana kata tersebut dieja. Perintah yang digunakan adalah `\hyphenation {}`. Misalnya :

```
\hyphenation {me-tro-po-lit-an}
```

Dengan perintah di atas, jika L^AT_EX menemukan kata metropolitan di bagian ujung baris dan sudah melebihi panjang baris tersebut maka L^AT_EX akan memotong kata tersebut menurut aturan yang telah kita tentukan tersebut.

Perintah `\hyphenation {}` diletakkan sebelum `\begin{document}`. Kita dapat menempatkan banyak aturan pemenggalan kata pada bagian tersebut.

Jika kita ingin mengatur sebuah kata saja (yang jarang muncul dalam dokumen yang kita buat) kita bisa langsung menentukan aturan pemenggalan katanya dengan cara *soft-hyphens*. Caranya adalah dengan menyisipkan tanda `_` dalam suku kata yang ingin kita atur pemenggalannya.

Sebaliknya jika kita ingin sebuah kata tidak terpotong sama sekali ketika terletak di bagian ujung baris, kita dapat menggunakan perintah :

```
\mbox{kata yang tidak ingin dipenggal}
```

3.5 Alignment

Alignment/ perataan baris pada L^AT_EX ada 3 jenis yaitu rata kiri, rata kanan, atau rata tengah. Semua dokumen dalam L^AT_EX secara default diatur memiliki perataan *justified* (rata kanan kiri). Jika kita ingin mengatur supaya dokumen rata kiri digunakan perintah berikut ini :

```

\begin{raggedright}
isi dokumen yang ingin diatur rata kiri
\end{raggedright}

```

Jika kita ingin mengatur supaya dokumen rata kanan digunakan perintah berikut ini :

```
\begin{raggedleft}
isi dokumen yang ingin diatur rata kanan
\end{raggedleft}
```

Cara membuat
dokumen
RATA
KANAN

Apabila kita ingin mengatur supaya dokumen rata tengah digunakan perintah berikut ini :

```
\begin{center}
isi dokumen yang ingin diatur rata tengah
\end{center}
```

Cara membuat
dokumen
RATA
TENGAH

3.6 Bahasa

L^AT_EX dapat mengatur tulisan mengikuti aturan ejaan yang dimiliki beberapa bahasa tertentu¹. Kemampuan ini diatur oleh `babel` package yang dimiliki L^AT_EX. Hal tersebut berpengaruh pada pemenggalan kata, spasi setiap kata, indentasi, dan beberapa judul bagian dokumen yang digunakan dalam heading². Mengubah pengaturan bahasa dengan menggunakan `babel` akan secara otomatis mengubah nama-nama dari unit struktur dokumen (seperti misalnya Abstract, Chapter, Index) menjadi terjemahannya.

Perintah yang mengatur L^AT_EX untuk menggunakan babel bahasa Indonesia adalah seperti berikut ini :

```
\documentclass [a4paper, 12pt]{report}
\usepackage[bahasa]{babel}
\begin{document}
.....
.....
\end{document}
```

Cara mengatur
bahasa

3.7 Keterangan

Jika kita ingin menambahkan keterangan pada file `*.tex` (yang tidak akan tercetak), caranya adalah dengan menambahkan tanda `%` di awal setiap baris keterangan. Contohnya :

```
\documentclass [a4paper, 12pt]{report}
\usepackage[bahasa]{babel}
\begin{document}
% ini adalah baris keterangan, baris ini tidak akan tercetak dalam file keluaran
.....
\end{document}
```

Cara
menuliskan
keterangan

3.8 Karakter Khusus

Ada beberapa karakter tertentu yang membutuhkan perintah khusus pula untuk menuliskannya, di antaranya ada pada tabel 3.1:

¹Bahasa Indonesia juga termasuk di dalamnya.

²Misalnya daftar isi secara default ditulis sebagai `Table of Contents`. Namun jika digunakan `babel` untuk bahasa Indonesia, daftar isi tidak akan ditulis lagi demikian melainkan langsung ditulis `Daftar Isi`.

Karakter	Penulisan	Karakter	Penulisan
\	<code>\textbackslash</code>	\$	<code>\\$</code>
%	<code>\%</code>	^	<code>\^{}{}</code>
-	<code>\-</code>	~	<code>\~{}{}</code>
{	<code>\{</code>	}	<code>\}</code>
>	<code>\textgreater</code> atau <code>\$ >\$</code>	<	<code>\textless</code> atau <code>\$ <\$</code>
	<code>\textbar</code>	“	<code>\textquotedblleft</code>
”	<code>\textquotedblright</code>	‘	<code>\textquoteleft</code>
,	<code>\textquoteright</code>	#	<code>\#</code>

Tabel 3.1: Karakter-Karakter Khusus

Dalam bahasa asing sering digunakan aksen dan simbol-simbol tertentu dalam penulisan bahasanya. Tabel 3.2 berikut ini menampilkan beberapa aksen dan simbol yang sering digunakannya :

Karakter	Penulisan	Karakter	Penulisan	Karakter	Penulisan
à	<code>\`{a}</code>	ã	<code>\~{a}</code>	ă	<code>\v{a}</code>
ç	<code>\c{a}</code>	á	<code>\' {a}</code>	ā	<code>\={a}</code>
á	<code>\H{a}</code>	ç	<code>\d{a}</code>	â	<code>\^ {a}</code>
à	<code>\. {a}</code>	â	<code>\t{a}</code>	ä	<code>\b{a}</code>
ä	<code>\' ' {a}</code>	ã	<code>\u{a}</code>	œ	<code>\oe</code>
â	<code>\aa</code>	æ	<code>\ae</code>	ø	<code>\o</code>
Œ	<code>\OE</code>	Å	<code>\AA</code>	Æ	<code>\AE</code>
Ø	<code>\O</code>	ı	<code>\l</code>	L	<code>\L</code>
ß	<code>\ss</code>				

Tabel 3.2: Aksan & Simbol Bahasa Asing

3.9 Font Dalam L^AT_EX

3.9.1 Jenis Font

Font standar yang ada dalam L^AT_EX ada 3 jenis, yaitu :

1. Roman , cara menggunakan jenis font seperti ini adalah :

```
{\rmfamily teks yang ingin diformat }
```

2. Sans serif , cara menggunakan jenis font seperti ini adalah :

```
{\sffamily teks yang ingin diformat }
```

Cara menentukan jenis font

3. Typewriter , cara menggunakan jenis font seperti ini adalah :

```
{\ttfamily teks yang ingin diformat }
```

3.9.2 Bentuk Font

L^AT_EX menyediakan 4 bentuk font, yaitu :

1. *Italic* , cara mengatur bentuk font seperti ini adalah :

```
{\itshape teks yang ingin diformat }
```

2. *Slanted* , cara mengatur bentuk font seperti ini adalah :

```
{\slshape teks yang ingin diformat }
```

Cara mengatur bentuk font

3. Vertical , cara mengatur bentuk font seperti ini adalah :

```
{\upshape teks yang ingin diformat }
```

4. SMALL CAPS , cara mengatur bentuk font seperti ini adalah :

```
{\scshape teks yang ingin diformat }
```

3.9.3 Ukuran Font

Ada beberapa macam ukuran font dalam dokumen L^AT_EX yaitu :

Tiny Scriptsize Footnotesize Small Normal

Large Larger Largest Huge Hugger

Contoh ukuran font

Untuk menggunakan ukuran-ukuran tersebut caranya adalah sebagai berikut :

- Tiny : `{\tiny teks yang ingin diformat }`
- Scriptsize : `{\scriptsize teks yang ingin diformat }`
- Footnotesize : `{\footnotesize teks yang ingin diformat }`
- Small : `{\small teks yang ingin diformat }`
- Normal : `{\normalsize teks yang ingin diformat }`
- Large : `{\large teks yang ingin diformat }`
- Larger : `{\Larger teks yang ingin diformat }`
- Largest : `{\LARGE teks yang ingin diformat }`
- Huge : `{\huge teks yang ingin diformat }`
- Huger : `{\Huge teks yang ingin diformat }`

Cara mengatur ukuran font

3.10 Mode Verbatim

Semua teks dalam dokumen L^AT_EX yang ingin ditampilkan pada file keluarannya, harus dilengkapi dengan perintah. Ada kalanya kita ingin menampilkan teks seperti apa yang kita ketik. Perhatikan contoh berikut :

```
ini adalah baris pertama \\  
  
ini adalah baris kedua      dari contoh.
```

Hasil dari baris-baris perintah di atas adalah seperti ini :

```
ini adalah baris pertama  
ini adalah baris kedua dari contoh.
```

Pada contoh di atas terlihat seberapa jauh pun kita membuat spasi horizontal atau vertikal, hasilnya tidak terpengaruh. Hasilnya hanya menampilkan sesuai yang diperintahkan dan menurut standar aturannya L^AT_EX . Supaya teks yang dihasilkan sama persis susunan dan formasinya seperti yang kita ketik, kita bisa menggunakan mode verbatim. Cara penggunaannya adalah sebagai berikut :

```
\begin{verbatim}  
teks yang ingin kita buat dalam mode verbatim  
.....  
\end{verbatim}
```

Semua teks yang ada dalam mode verbatim tidak membutuhkan perintah-perintah pemformatan L^AT_EX . Cukup ketikkan teks yang akan ditampilkan dan bagaimana posisi teks tersebut. Sebagai contoh kita tuliskan teks contoh di atas dalam mode verbatim :

```
\begin{verbatim}  
ini adalah baris pertama  
  
ini adalah baris kedua      dari contoh.  
\end{verbatim}
```

Maka hasilnya akan tampak seperti berikut ini :

```
ini adalah baris pertama  
  
ini adalah baris kedua      dari contoh.
```

Mode verbatim sangat cocok digunakan untuk penulisan *source code* atau dokumentasi pembuatan perangkat lunak.

Bab 4

Struktur Dasar Sebuah Dokumen

Pendekatan yang digunakan oleh \LaTeX untuk melakukan pengaturan teks memiliki tujuan untuk memperoleh konsistensi struktur dokumen. Jadi kita harus mengatur elemen-elemen dalam dokumen secara benar dan tepat. Konsistensi dalam struktur dokumen membuat dokumen menjadi lebih mudah dibaca dan dipahami. Elemen yang dimaksud di atas adalah komponen-komponen penyusun sebuah dokumen. Misalnya sebuah dokumen biasanya tersusun atas bab, subbab, tabel, daftar, dst. Untuk mencapai konsistensi tersebut, setiap dokumen \LaTeX dimulai dengan mendeklarasikan kelas dokumen (document class) yang akan digunakan

4.1 Document Class

Class file pada \LaTeX menentukan layout halaman, jenis heading, dan berbagai perintah dan environment yang diperlukan untuk mengatur style dokumen. Cara untuk mendeklarasikan *Document Class* adalah memulai dokumen dengan :

```
\documentclass {class}
```

Ada beberapa jenis *document class* yang bisa dipakai dalam sebuah dokumen, yaitu :

- **report** : kelas ini dapat digunakan untuk membuat laporan (*report*) baik dalam bidang bisnis, teknik, hukum, akademis, atau ilmu pengetahuan.
- **article** : kelas ini dapat digunakan untuk membuat paper, artikel sebuah jurnal atau majalah, review, paper untuk konferensi, atau catatan riset.
- **book** : kelas ini digunakan untuk membuat buku dan thesis.
- **letter** : kelas ini digunakan untuk membuat surat.

Biasanya kelas ‘**article**’ adalah yang paling sering digunakan untuk sembarang jenis dokumen.

Masing-masing kelas di atas memiliki strukturnya sendiri. Misalnya pada kelas **article** tidak ada elemen bab, tidak seperti pada kelas **report** dan **book**.

4.1.1 Document Class Options

Document Class Option maksudnya adalah pilihan yang tersedia pada kelas dokumen yang bisa kita tentukan sendiri isinya. Opsi pada suatu kelas dokumen dituliskan seperti berikut :

```
\documentclass [option1, option2] { class }
```

Seperti terlihat di atas, kita dapat menentukan beberapa opsi sekaligus dalam tanda kurung dengan dibatasi tanda koma.

Default opsi yang digunakan oleh \LaTeX antara lain :

- Ukuran kertas yang digunakan adalah A4.
- Ukuran font yang digunakan adalah 10pt untuk semua kelas dokumen.
- Layout halaman yang digunakan adalah *two-sided printing* khusus untuk kelas `book` dan `report`, dan *one-sided printing* khusus untuk kelas `article` dan `letter`.
- Halaman judul yang terpisah di bagian awal dokumen khusus untuk kelas `book` dan `report`.

Opsi di atas dapat modifikasi dengan beberapa opsi berikut :

- Ukuran kertas : Kita dapat menentukan sendiri ukuran kertasnya. Cara penulisannya :

```
\documentclass [ a3paper ] { class } atau
\documentclass [ letterpaper ] { class }
```

- Ukuran font : kita dapat memilih ukuran 10pt, 11pt, atau 12pt. Cara penulisannya :

```
\documentclass [ a4paper, 11pt ] { class }
```

Setelah kita menentukan ukuran font yang dipakai, semua font dalam dokumen akan diatur sedemikian sehingga memiliki ukuran sesuai dengan yang kita tentukan. Font yang dipakai pada header, footer disesuaikan secara proporsional dengan ukuran font tersebut.

- Layout halaman dapat kita tentukan sendiri dengan 2 pilihan berikut :
 - `oneside` : jika kita menginginkan layout *one-sided printing* saat menggunakan kelas `book` dan `report`.
 - `twoside` : jika kita menginginkan layout *two-sided printing* saat menggunakan kelas `article`.
 - `titlepage` : jika kita menginginkan kelas `article` untuk memiliki halaman judul yang terpisah di bagian awal dokumen.
 - `draft` : opsi ini mengatur L^AT_EX supaya menandai masalah-masalah yang timbul seperti masalah pemenggalan kata (pemenggalan kata tidak tepat) atau masalah perataan tulisan (ada baris tertentu yang melebihi batas kanan dokumen). Tanda yang akan digunakan L^AT_EX adalah sebuah persegi kecil di bagian kanan dokumen tempat terjadinya masalah.

4.2 Paket-Paket Dalam L^AT_EX

Yang dimaksud dengan paket dalam L^AT_EX adalah fungsi-fungsi yang dipakai untuk menambah kemampuan L^AT_EX melakukan pengaturan dokumen. Ada banyak sekali paket yang dimiliki L^AT_EX baik yang sudah terintegrasi bersamaan di dalam instaler L^AT_EX maupun yang belum. Paket-paket yang belum terinstal bisa didownload dari <http://www.ctan.org>.

Untuk menggunakan paket tertentu dalam dokumen yang kita buat, kita perlu mendeklarasikannya terlebih dulu pada bagian preamble¹. Cara menggunakan paket yang sudah tersedia/terintegrasi di dalam L^AT_EX adalah seperti ini :

```
\documentclass {class}
\usepackage [ option ] {nama paket}
\begin{document}
.....
.....
.....
\end{document}
```

¹Bagian antara deklarasi kelas dokumen dengan awal *document environment*, lihat juga penjelasan pada bab 3.2.

Beberapa paket yang terintegrasi dalam L^AT_EX antara lain :

1. `graphicx` : paket ini membuat L^AT_EX mampu menghasilkan gambar grafis dan juga membuat L^AT_EX mampu menampilkan gambar yang kita sertakan dalam dokumen.
2. `hyperref` : paket ini membuat L^AT_EX mampu menghasilkan dokumen yang memiliki *dynamic link*² ke alamat tertentu.
3. `babel` : paket ini membuat L^AT_EX mampu mengenali format bahasa yang digunakan seperti yang sudah dijelaskan pada subbab Bahasa di Bab Perintah-Perintah L^AT_EX .
4. `color` : paket ini membuat L^AT_EX mampu menghasilkan teks dokumen yang memiliki warna sesuai warna yang ditentukan.
5. `makeidx` : paket ini membuat L^AT_EX mampu menghasilkan indeks dari dokumen yang dibuat.

4.3 Document Environment

Yang dimaksud dengan *document environment*³ adalah bagian dalam sebuah dokumen L^AT_EX dimana isi sebenarnya dari dokumen itu sendiri ditempatkan.

```
\documentclass {class}
\begin{document}
.....
.....
.....
\end{document}
```

Semua teks isi dari dokumen harus dituliskan di bagian titik-titik tersebut di atas. Teks yang ditulis sebelum `\begin{document}` dan sesudah `\end{document}`, kelak tidak akan muncul pada dokumen hasil *compile*.

Struktur `\begin ... \end` inilah yang disebut dengan *environment*. Environment membatasi bagian teks yang akan diatur dengan aturan tertentu.

Bagian antara deklarasi kelas dokumen dengan awal *document environment* disebut *preamble*.

4.4 Penulisan Judul

Judul > dalam sebuah dokumen L^AT_EX diletakkan pada awal *document environment*. Cara penulisannya adalah sebagai berikut :

```
\documentclass [ a4paper, 12pt ] {report}
\begin{ document}
  \title{Judul Dokumen}
  \autor{Nama Penulis}
  \date{Tanggal Pembuatan}
  \maketitle
  .....
  .....
\end{document}
```

²Link dapat berupa alamat URL, email, maupun *bookmark* ke halaman tertentu dari dokumen yang kita buat.

³Lihat bagian 3.2 tentang *environment*

4.5 Abstrak

Pada dokumen kelas `article` dan `report` umumnya memiliki abstrak/ringkasan. \LaTeX memiliki cara khusus untuk menuliskan abstrak. Formatnya adalah sebagai berikut :

```
\documentclass [a4paper, 12pt] {report}
\begin{document}
  \title{Judul Dokumen}
  \author{Nama Penulis}
  \date{Tanggal Pembuatan}
  \maketitle

  \begin{abstract}
    isi abstract
  \end{abstract}

  .....
\end{document}
```

Cara
menggunakan
abstrak

Jika kita ingin mengubah judul abstrak digunakan perintah ini sebelum `\begin{abstract}` :

```
\renewcommand {\abstractname}{RINGKASAN LAPORAN}
```

Perintah di atas akan mengganti judul abstrak menjadi “RINGKASAN LAPORAN”.

4.6 Sistematika Isi Dokumen

\LaTeX memiliki kemampuan untuk membagi dokumen dalam suatu susunan struktural (bab, subbab, subsubbab, dst) sampai 7 tingkatan. Berikut ini adalah daftar struktur yang disediakan oleh \LaTeX :

Struktur	Perintah
Bagian (part)	<code>\part{...}</code>
Bab (chapter)	<code>\chapter{...}</code>
Subbab (section)	<code>\section{...}</code>
Subsubbab (subsection)	<code>\subsection{...}</code>
Subsubsubbab (subsubsection)	<code>\subsubsection{...}</code>
Paragraf berjudul (titled paragraph)	<code>\paragraph{...}</code>
Anak paragraf berjudul (titled subparagraph)	<code>\subparagraph{...}</code>

Tabel 4.1: Struktur Dokumen

Ada beberapa hal yang perlu diketahui tentang penggunaan struktur di atas :

- Hanya dokumen dengan kelas `book` dan `report` bisa menggunakan semua struktur di atas.

- Dokumen kelas `article` hanya bisa menggunakan kelas `\section{...}` dan struktur-struktur di bawahnya bawah.
- Dokumen kelas `letter` tidak dapat menggunakan semua struktur di atas.

Contoh penggunaannya adalah seperti berikut ini :

```
\documentclass [a4paper, 12pt] {report}
\begin{document}
  \title{Judul Dokumen}
  \author{Nama Penulis}
  \date{Tanggal Pembuatan}
  \maketitle

  \begin{abstract}
    isi abstract
  \end{abstract}

  \chapter{Pendahuluan}
    isi bab I pendahuluan
  \section{Latar Belakang}
    isi subbab latar belakang

  \chapter{Dasar Teori}
    isi bab II dasar teori
  \section{Tinjauan Pustaka}
    isi subbab tinjauan pustaka

\end{document}
```

Cara menggunakan struktur dokumen

Pada contoh di atas, dokumen hasilnya akan memiliki bab I dengan judul Pendahuluan, subbab pertamanya berjudul Latar Belakang; bab II dengan judul Dasar Teori, subbab pertamanya berjudul Tinjauan Pustaka.

4.7 Daftar Berurut

Ada 3 jenis cara penulisan daftar berurut yaitu :

1. **Daftar dengan penomoran menggunakan simbol (*Bulleted List*)**, contohnya seperti berikut ini :

- Apel
- Jeruk
- Semangka
- Durian

Cara membuat itemize list

Format penulisan daftar seperti ini adalah sebagai berikut :

```
\begin{itemize}
\item ...
\item ...
\item ...
\item ...
...
```

```
...
\end{itemize}
```

Isi daftar dituliskan setelah `\item`. Simbol yang dipakai dapat kita tentukan sendiri. Sebagai contoh jika kita ingin menggunakan tanda * sebagai penanda item, caranya adalah dengan menambahkan keterangan simbol yang digunakan seperti berikut : `\item[*] ...`

2. **Daftar dengan penomoran menggunakan angka (*Numbered List*)**, contohnya seperti berikut ini :

```
1 Apel
2 Jeruk
3 Semangka
4 Durian
```

Cara membuat
enumerate list

Format penulisan daftar seperti ini adalah sebagai berikut :

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

3. **Daftar deskripsi**, contohnya seperti berikut ini :

```
ITB Institut Teknologi Bandung
UI Universitas Indonesia
IPB Institut Pertanian Bogor
UGM Universitas Gajah Mada
```

Cara penulisannya adalah seperti berikut ini :

```
\begin{description}
\item [ Hal 1 ] penjelasan hal 1
\item [ Hal 2 ] penjelasan hal 2
\item [ Hal 3 ] penjelasan hal 3
\item [ Hal 4 ] penjelasan hal 4
...
\end{description}
```

Cara membuat
description list

4.8 Daftar Isi

Untuk menampilkan daftar isi digunakan perintah :

```
\tableofcontents
```

Perintah ini diletakkan pada bagian dimana daftar isi tersebut akan ditempatkan. Biasanya daftar isi ditempatkan tepat setelah abstrak/kata pengantar.

Untuk menampilkan daftar gambar digunakan perintah :

```
\listoffigures
```

Untuk menampilkan daftar tabel digunakan perintah :

```
\listoftables
```

L^AT_EX menghasilkan file berekstensi *.toc untuk menangani daftar isi, daftar gambar, dan daftar tabel. Jika daftar isi, daftar gambar, dan daftar tabel tidak menampilkan keseluruhan struktur dokumen dengan benar, kita dapat mengatur sendiri isinya dengan cara menambahkan perintah-perintah berikut :

```
\addcontentsline{toc}{struktur}{teks yang ingin ditampilkan pada daftar isi}
```

Struktur dapat diisi dengan `chapter`, `section`, `subsection`, `dst`, tergantung bagian dokumen yang ingin kita masukkan ke dalam daftar isi. Dengan perintah di atas L^AT_EX akan menghasilkan baris baru dalam daftar isi dan akan secara otomatis menentukan nomor halaman bagian tersebut.

4.9 Tabel & Gambar

4.9.1 Tabel

Untuk menempatkan sebuah tabel dalam dokumen L^AT_EX caranya adalah menggunakan *table environment* :

```
\begin{table}
...
\end{table}
```

Bagian titik-titik tersebut adalah bagian isi dari tabel itu sendiri. Cara mengisi bagian tersebut adalah seperti berikut :

```
\begin{center}

\begin{tabular}{|c|l|r|}

\hline
&Judul Kolom 1 & Judul Kolom 2 & Judul Kolom 3 \\ \hline
Isi Baris 1 Kolom 1 & Isi Baris 1 Kolom 2 & Isi Baris 1 Kolom 3 \\ \hline
Isi Baris 2 Kolom 1 & Isi Baris 2 Kolom 2 & Isi Baris 2 Kolom 3 \\ \hline
\end{tabular}

\caption{Contoh Tabel}
\end{center}
```

Cara Membuat
Tabel

Hasil dari perintah tersebut adalah sebagai berikut :

Judul Kolom 1	Judul Kolom 2	Judul Kolom 3
Isi Baris 1 Kolom 1	Isi Baris 1 Kolom 2	Isi Baris 1 Kolom 3
Isi Baris 2 Kolom 1	Isi Baris 2 Kolom 2	Isi Baris 2 Kolom 3

Tabel 4.2: Contoh Tabel

Ada beberapa hal yang perlu diketahui dari format perintah tersebut di atas :

- `{|c|l|r|}` adalah bagian yang menentukan banyaknya kolom yang akan dihasilkan. Huruf-huruf tersebut mewakili `center`, `left`, & `right`, yaitu menentukan *alignment* dari isi sel yang dibuat. Sementara garis `—` menentukan apakah tabel ingin dibatasi garis atau tidak. Jika antara kolom maupun tidak ingin diberi garis batas, kita tinggal menghilangkan `—` tersebut.

- Pengaturan posisi tabel dapat kita tentukan menurut 2 hal :
 - Perataan terhadap tepi dokumen : Dengan mengubah `\begin{center}` dan juga `\end{center}` kita bisa menentukan posisi tabel terhadap tepi dokumen. Silakan lihat penjelasan mengenai *alignment* pada subbab 3.5.
 - Huruf-huruf pada `\begin{table}[htbp]` juga berfungsi sebagai pengatur posisi tabel pada suatu halaman.
 - * **h** : tabel diletakkan persis di tempat perintah tersebut dituliskan dalam dokumen.
 - * **t** : tabel diletakkan di bagian atas halaman.
 - * **b** : tabel diletakkan di bagian bawah halaman.
 - * **p** : tabel diletakkan pada sebuah halaman khusus yang memuat hanya tabel itu saja.
- Untuk menuliskan isi dari masing-masing baris, digunakan format


```
isi kolom 1 & isi kolom 2 & isi kolom 3 dst
```

Perpindahan kolom saat mengisi sebuah baris ditandai dengan tanda `&` .
- Garis mendatar pada tabel (batas tiap baris) dihasilkan dengan perintah `\hline`

4.9.2 Gambar

Agar \LaTeX dapat menempatkan gambar di dalam dokumen, kita perlu mendeklarasikan penggunaan paket `graphicx` pada bagian preamble. Cara deklarasinya adalah :

```
\usepackage{graphicx}
```

Untuk menempatkan sebuah gambar dalam dokumen \LaTeX caranya adalah sebagai berikut :

```
\begin{figure}[htbp]
\caption{Nama Gambar}
\begin{center}
\includegraphics[width=3cm,height=3cm\columnwidth]{nama file gambar}
\end{center}
\end{figure}
```

Cara
Menyisipkan
Gambar

Ada beberapa hal yang perlu diketahui dari format perintah di atas :

- Panjang dan Lebar dari gambar yang akan ditampilkan dapat diubah sesuai keinginan kita. Isi dari `width` dapat kita isi dengan lebar gambar tersebut dan isi dari `height` dapat kita isi dengan tinggi gambar tersebut; keduanya harus dilengkapi dimensi dari ukuran panjang yang kita gunakan. Dengan mengatur `width` dan `height` kita bisa memasukkan gambar meskipun gambar tersebut memiliki ukuran dimensi yang besar.
- File gambar yang ingin kita masukkan dalam dokumen, harus diletakkan pada direktori yang sama dengan direktori file dokumen (`*.tex`) kita berada.
- Pengaturan posisi gambar dapat kita tentukan menurut 2 hal :
 - Perataan terhadap tepi dokumen : Dengan mengubah `\begin{center}` dan juga `\end{center}` kita bisa menentukan posisi gambar terhadap tepi dokumen. Silakan lihat penjelasan mengenai *alignment* pada subbab 3.5.
 - Huruf-huruf pada `\begin{figure}[htbp]` juga berfungsi sebagai pengatur posisi gambar pada suatu halaman.
 - * **h** : gambar diletakkan persis di tempat perintah tersebut dituliskan dalam dokumen.
 - * **t** : gambar diletakkan di bagian atas halaman.

- * **b** : gambar diletakkan di bagian bawah halaman.
- * **p** : gambar diletakkan pada sebuah halaman khusus yang memuat hanya gambar itu saja.

Saat menggunakan **h**, \LaTeX akan secara otomatis menempatkan gambar di halaman baru jika tidak ada cukup ruang untuk gambar tersebut di tempat perintah gambar dituliskan.

- Format gambar standar \LaTeX adalah `*.eps` (Encapsulated PostScript). Tetapi, gambar dengan format `*.jpg` juga bisa digunakan.

4.10 Referensi / Rujukan

Dalam suatu dokumen ada kalanya kita ingin merujuk ke suatu bagian dari dokumen, caranya adalah seperti berikut :

1. Pada bagian yang akan dijadikan rujukan kita beri `label` terlebih dahulu. Contohnya :

```
\chapter{Dasar Teori}
\label{dasar teori}
```

2. Pada bagian dokumen yang memerlukan rujukan kita tambahkan perintah `\ref{label}`. Contohnya :

```
... Untuk melihat teori ini silakan lihat Bab \ref{dasar teori} ...
```

Cara Membuat Rujukan

Dengan perintah tersebut akan memunculkan nomor bab yang bisa diakses langsung. Akan tetapi jika yang akan ditampilkan sebagai rujukan adalah nomor halamannya maka perintah perintahnya adalah seperti berikut ini :

```
... Untuk melihat teori ini silakan lihat Bab \ref{dasar teori} pada halaman \pageref{dasar teori} ...
```

Referensi yang dipakai dalam suatu dokumen kadangkala berbentuk alamat URL situs tertentu. Untuk menampilkan alamat URL sebagai referensi digunakan perintah berikut ini :

```
\url{alamat URL}
```

Yang perlu diingat untuk dapat menyisipkan referensi/rujukan yang dinamis⁴ kita perlu mendeklarasikan penggunaan paket `hyperref`. Cara pendeklarasian paket-paket \LaTeX bisa dilihat pada subbab 4.2.

4.11 Daftar Pustaka

Untuk menampilkan daftar pustaka atau bibliografi pada akhir sebuah dokumen \LaTeX digunakan format perintah seperti berikut ini :

```
\begin { thebibliography } { 99 }
\bibitem {label untuk referensi} { keterangan pustaka yang digunakan }
.....
.....
\end{thebibliography}
```

Cara membuat Daftar Pustaka

Beberapa hal yang perlu diketahui dari perintah di atas antara lain :

- Angka 99 memberitahu \LaTeX bahwa penomoran maksimal Daftar Pustaka adalah 99.

⁴baca : bisa langsung di-*klik*.

- Label untuk referensi diisikan *keyword* yang akan digunakan saat membuat rujukan ke pustaka yang bersangkutan.
- Keterangan pustaka diisi informasi mengenai : penulis, judul pustaka, edisi, penerbit, kota penerbit, tahun penerbitan.

Cara untuk membuat rujukan ke salah satu pustaka yang sudah kita tuliskan dalam daftar pustaka adalah menggunakan perintah seperti ini :

```
~\cite{label referensinya}
```

Penggunaannya pada sebuah dokumen contohnya sebagai berikut :

```
.....
.....
Untuk mempelajari \LaTeX\ dapat membaca ebook Begin \LaTeX\ ~\cite{pustaka1}
.....
.....

\begin { thebibliography } { 99 }
\bibitem {pustaka1} { Peter Flynn : Begin \LaTeX\, Silmaril Consultants, (1999)}
.....
.....
\end{thebibliography}
```

Cara Merujuk
Daftar Pustaka

4.12 Indeks

Dalam sebuah buku/dokumen, indeks memiliki peran penting antara lain memudahkan pembaca mencari kata tertentu dalam buku/dokumen tersebut. \LaTeX memiliki kemampuan untuk menghasilkan indeks secara otomatis hanya dengan beberapa perintah. Untuk dapat menyisipkan indeks pada sebuah dokumen \LaTeX kita perlu mendeklarasikan penggunaan paket `makeidx`⁵.

Setelah kita mendeklarasikan penggunaan paket `makeidx`, di bagian preamble juga kita tuliskan `\makeindex`. Untuk menandai kata-kata mana saja yang akan dimasukkan ke dalam daftar indeks, kita menambahkan perintah `\index {kata tertentu}`. Perintah tersebut diletakkan berulang-ulang kali setiap kali menjumpai kata yang sama.

Untuk menampilkan bab khusus yang berisi indeks, kita gunakan perintah `\printindex` di tempat di mana bab tersebut akan diletakkan (biasanya di bagian akhir dokumen).

⁵Lihat subbab 4.2.

Bab 5

Notasi Matematika Dalam L^AT_EX

L^AT_EX memiliki kemampuan yang hebat dalam menghasilkan dokumen yang berisi notasi-notasi matematis. Untuk bisa menghasilkan dokumen berisi notasi matematis kita harus berada dalam *matematic environment*¹.

5.1 Penulisan Notasi Matematika Dalam Paragraf

Untuk menyisipkan notasi matematika dalam suatu kalimat/paragraf digunakan perintah berikut ini :

```
\begin{math} ..... \end{math} atau  
$ ..... $
```

Titik-titik merah tersebut di atas diisi dengan notasi matematis yang akan disisipkan.

5.2 Paragraf Khusus Matematika

Untuk menuliskan suatu notasi matematika yang cukup panjang, kita bisa memilih untuk menuliskannya dalam suatu paragraf baru. Perintah yang digunakan adalah sebagai berikut :

```
\begin{displaymath}  
.....  
\end{displaymath}
```

Titik-titik merah tersebut di atas diisi dengan notasi matematis yang akan disisipkan.

5.3 Font Dalam Matematika

Ada beberapa perintah yang dapat digunakan untuk mengubah jenis font yang dipakai dalam notasi matematis, di antaranya adalah :

1. `\mathrm{...}`
2. `\mathsf{...}`
3. `\mathtt{...}`
4. `\mathit{...}`
5. `\mathbf{...}`
6. `\mathcal{...}`

¹Lihat bab 3.2 tentang *environment*.

Berikut adalah contoh hasil notasi matematis dengan masing-masing jenis font di atas :

1. Perintah `\mathrm{x y x}` akan menghasilkan :

xyz

2. Perintah `\mathsf{x y x}` akan menghasilkan :

xyz

3. Perintah `\mathtt{x y x}` akan menghasilkan :

xyz

4. Perintah `\mathit{x y x}` akan menghasilkan :

xyz

5. Perintah `\mathbf{x y x}` akan menghasilkan :

xyz

6. Perintah `\mathcal{X Y Z}` akan menghasilkan :

\mathcal{XYZ}

Untuk menuliskan font matematika dalam bentuk *superscripts* dan *subscripts* digunakan aturan berikut ini :

- Superscripts , cara penulisannya adalah dengan perintah `\sp{...}` atau dengan tanda \wedge .
- Subscript , cara penulisannya adalah dengan perintah `\sb{...}` atau dengan tanda $_$.

Contoh pemakaiannya sebagai berikut :

```
\begin{displaymath}
y = x\sb{1}\sp{2} + x\sb{2}\sp{2}
\end{displaymath}
```

Perintah di atas akan menghasilkan keluaran seperti berikut :

$$y = x_1^2 + x_2^2$$

Contoh lainnya :

```
\begin{displaymath}
f(x)=e^{\wedge}(x\_1)
\end{displaymath}
```

Perintah di atas akan menghasilkan keluaran seperti berikut :

$$f(x) = e^{x_1}$$

Notasi matematika sering menggunakan huruf-huruf Yunani. Tabel 5.1 berikut ini memuat daftar huruf kecil Yunani dan cara penulisannya dalam L^AT_EX :

Karakter	Penulisan	Karakter	Penulisan	Karakter	Penulisan
α	<code>\alpha</code>	*	<code>\beta</code>	γ	<code>\gamma</code>
δ	<code>\delta</code>	ϵ	<code>\epsilon</code>	ε	<code>\varepsilon</code>
ζ	<code>\zeta</code>	η	<code>\eta</code>	θ	<code>\theta</code>
ϑ	<code>\vartheta</code>	ι	<code>\iota</code>	κ	<code>\kappa</code>
λ	<code>\lambda</code>	μ	<code>\mu</code>	ν	<code>\nu</code>
ξ	<code>\xi</code>	π	<code>\pi</code>	ϖ	<code>\varpi</code>
ρ	<code>\rho</code>	ϱ	<code>\varrho</code>	σ	<code>\sigma</code>
ς	<code>\varsigma</code>	τ	<code>\tau</code>	υ	<code>\upsilon</code>
ϕ	<code>\phi</code>	φ	<code>\varphi</code>	χ	<code>\chi</code>
ψ	<code>\psi</code>	ω	<code>\omega</code>		

Tabel 5.1: Huruf Kecil Yunani

Tabel 5.2 berikut ini memuat huruf kapital Yunani dan cara penulisannya dalam \LaTeX :

Karakter	Penulisan	Karakter	Penulisan	Karakter	Penulisan
Γ	<code>\Gamma</code>	Δ	<code>\Delta</code>	Θ	<code>\Theta</code>
Λ	<code>\Lambda</code>	Ξ	<code>\Xi</code>	Π	<code>\Pi</code>
Σ	<code>\Sigma</code>	Υ	<code>\Upsilon</code>	Φ	<code>\Phi</code>
Ψ	<code>\Psi</code>	Ω	<code>\Omega</code>		

Tabel 5.2: Huruf Kapital Yunani

5.4 Tanda Kurung Dalam Matematika

Penulisan tanda kurung dalam notasi matematis tidak bisa² menggunakan tanda kurung biasa. Cara penulisan yang akan mengeluarkan notasi matematika yang baik adalah sebagai berikut :

`\right DELIMITER` : untuk menghasilkan tanda kurung sebelah kanan
`\left DELIMITER` : untuk menghasilkan tanda kurung sebelah kiri

Delimiter sendiri adalah tanda kurung biasa yang penulisannya tentunya sesuai standar perintah \LaTeX . Beberapa delimiter yang biasa digunakan dalam notasi matematika ditunjukkan dalam Tabel 5.3 :

²Tidak bisa disini berarti tidak akan menghasilkan keluaran yang bagus.

Karakter	Penulisan	Karakter	Penulisan	Karakter	Penulisan
(())	{	\{
}	\}	[[]]
			\	/	/
\	\textbackslash	<	\langle	>	\rangle
⌊	\lfloor	⌋	\rfloor	⌈	\lceil
⌉	\rceil	↑	\uparrow	↓	\downarrow
↕	\Updownarrow	↕	\Downarrow	↕	\updownarrow
↕	\Updownarrow				

Tabel 5.3: Delimiter

5.5 Penulisan Akar

Format perintah untuk menghasilkan akar matematik adalah sebagai berikut :

```
\sqrt{pangkat}{bilangan yang diakar}
```

Contoh pemakaiannya adalah sebagai berikut :

```
\begin{displaymath}
\sqrt{2}{a+b}
\end{displaymath}
```

Perintah di atas akan menghasilkan notasi seperti berikut :

$$\sqrt[2]{a+b}$$

5.6 Penulisan Pecahan

Format perintah untuk menghasilkan notasi pecahan adalah sebagai berikut :

```
\frac{numerator}{denominator}
```

Contoh pemakaiannya adalah sebagai berikut :

```
\begin{displaymath}
\frac{12x}{x+1}
\end{displaymath}
```

Perintah di atas akan menghasilkan notasi seperti berikut :

$$\frac{12x}{x+1}$$

5.7 Penulisan Array & Matriks

Sebuah array/matriks dituliskan dalam environment `tabular` sama seperti cara pembuatan tabel (lihat subbab 4.9.1). Perintah untuk menghasilkan sebuah array atau matriks adalah seperti berikut :

```
\begin{displaymath}
\left (
\begin{array}{rrr}
0 & 45 & 23 \\
34 & -93 & 68 \end{array}
\right )
\end{displaymath}
```

Contoh perintah di atas akan menghasilkan matriks seperti berikut ini :

$$\begin{pmatrix} 0 & 45 & 23 \\ 34 & -93 & 68 \end{pmatrix}$$

Beberapa hal yang perlu diketahui dari format perintah tersebut di atas :

- Sama seperti cara penulisan tabel, huruf `r` di bagian belakang `\begin{array}{rrr}` fungsinya adalah menentukan posisi dari masing-masing komponen matriks tersebut. Dalam hal ini masing-masing komponen matriks dibuat menjadi rata kanan.
- Tanda kurung yang digunakan adalah berupa tanda kurung kurawal. Bagian kurung buka dan kurung tutup didefinisikan masing-masing.

5.8 Penulisan Vektor

Penulisan vektor dalam \LaTeX menggunakan perintah seperti berikut ini :

```
\begin{displaymath}
\vec{\text{variabel}}
\end{displaymath}
```

Misalnya :

```
\begin{displaymath}
\vec{x}
\end{displaymath}
```

akan menghasilkan vektor \vec{x} .

5.9 Penulisan Fungsi Matematika

Ada cukup banyak fungsi matematika yang memiliki perintah khusus untuk menuliskannya dalam dokumen \LaTeX seperti misalnya `sinus`, `cosinus`, dll. Berikut adalah Tabel 5.4 yang menampilkan beberapa fungsi tersebut :

<code>\arccos</code>	<code>\arcsin</code>	<code>\arctan</code>
<code>\arg</code>	<code>\cos</code>	<code>\cosh</code>
<code>\cot</code>	<code>\coth</code>	<code>\csc</code>
<code>\deg</code>	<code>\det</code>	<code>\dim</code>
<code>\exp</code>	<code>\gcd</code>	<code>\hom</code>
<code>\inf</code>	<code>\ker</code>	<code>\lg</code>
<code>\lim</code>	<code>\liminf</code>	<code>\limsup</code>
<code>\ln</code>	<code>\log</code>	<code>\max</code>
<code>\min</code>	<code>\Pr</code>	<code>\sec</code>
<code>\sin</code>	<code>\sinh</code>	<code>\sup</code>
<code>\tan</code>	<code>\tanh</code>	

Tabel 5.4: Fungsi Matematika

5.10 Simbol-Simbol Matematika

Untuk dapat menggunakan berbagai simbol matematika, kita harus mendeklarasikan penggunaan paket³ `amsmath` pada bagian preamble. Tabel 5.5, Tabel 5.6, Tabel 5.7, dan Tabel 5.8 berikut menunjukkan simbol-simbol matematika serta perintah penulisannya dalam \LaTeX ⁴.

Karakter	Penulisan	Karakter	Penulisan	Karakter	Penulisan
\amalg	<code>\amalg</code>	$*$	<code>\ast</code>	\bullet	<code>\bullet</code>
\bigcirc	<code>\bigcirc</code>	∇	<code>\bigtriangledown</code>	\triangle	<code>\bigtriangleup</code>
\cap	<code>\cap</code>	\cdot	<code>\cdot</code>	\circ	<code>\circ</code>
\cup	<code>\cup</code>	\dagger	<code>\dagger</code>	\circ	<code>\ddagger</code>
\diamond	<code>\diamond</code>	\div	<code>\div</code>	\mp	<code>\mp</code>
\odot	<code>\odot</code>	\ominus	<code>\ominus</code>	\oplus	<code>\oplus</code>
\oslash	<code>\oslash</code>	\otimes	<code>\otimes</code>	\pm	<code>\pm</code>
\setminus	<code>\setminus</code>	\sqcap	<code>\sqcap</code>	\sqcup	<code>\sqcup</code>
\star	<code>\star</code>	\times	<code>\times</code>	\triangleleft	<code>\triangleleft</code>
\triangleright	<code>\triangleright</code>	\uplus	<code>\uplus</code>	\vee	<code>\vee</code>
\wedge	<code>\wedge</code>	\wr	<code>\wr</code>		

Tabel 5.5: Binary Operator

Daftar simbol-simbol matematika selengkapnya dapat dilihat di situs Comprehensive Tex Archive Network (<http://www.ctan.org/>).

³Lihat subbab 4.2 tentang cara penggunaan paket-paket \LaTeX

⁴Daftar simbol-simbol matematika selengkapnya dapat dilihat di situs Comprehensive TEX Archive Network (<http://www.ctan.org/>).

Karakter	Penulisan	Karakter	Penulisan	Karakter	Penulisan
Σ	<code>\sum</code>	\int	<code>\int</code>	\oint	<code>\oint</code>
\prod	<code>\prod</code>	\coprod	<code>\coprod</code>	\bigcap	<code>\bigcap</code>
\bigcup	<code>\bigcup</code>	\bigsqcup	<code>\bigsqcup</code>	\bigvee	<code>\bigvee</code>
\bigwedge	<code>\bigwedge</code>	\bigodot	<code>\bigodot</code>	\bigotimes	<code>\bigotimes</code>
\bigoplus	<code>\bigoplus</code>	\biguplus	<code>\biguplus</code>		

Tabel 5.6: Simbol-Simbol dengan Limit

Karakter	Penulisan	Karakter	Penulisan
\downarrow	<code>\downarrow</code>	\Downarrow	<code>\Downarrow</code>
\hookleftarrow	<code>\hookleftarrow</code>	\hookrightarrow	<code>\hookrightarrow</code>
\leftarrow	<code>\leftarrow</code>	\Leftarrow	<code>\Leftarrow</code>
\leftharpoonupdown	<code>\leftharpoonupdown</code>	\leftarrow	<code>\leftharpoonup</code>
\leftrightarrow	<code>\leftrightarrow</code>	\Leftrightarrow	<code>\Leftrightarrow</code>
\longleftarrow	<code>\longleftarrow</code>	\Longleftarrow	<code>\Longleftarrow</code>
\longleftrightarrow	<code>\longleftrightarrow</code>	\Longleftrightarrow	<code>\Longleftrightarrow</code>
\longmapsto	<code>\longmapsto</code>	\longrightarrow	<code>\longrightarrow</code>
\Longrightarrow	<code>\Longrightarrow</code>	\mapsto	<code>\mapsto</code>
\nearrow	<code>\nearrow</code>	\nwarrow	<code>\nwarrow</code>
\rightarrow	<code>\rightarrow</code>	\Rightarrow	<code>\Rightarrow</code>
\rightharpoonupdown	<code>\rightharpoonupdown</code>	\rightarrow	<code>\rightharpoonup</code>
\rightrightarrows	<code>\rightrightarrows</code>	\searrow	<code>\searrow</code>
\swarrow	<code>\swarrow</code>	\uparrow	<code>\uparrow</code>
\Uparrow	<code>\Uparrow</code>	\Updownarrow	<code>\Updownarrow</code>
\Downarrow	<code>\Downarrow</code>		

Tabel 5.7: Tanda Panah Matematis

Karakter	Penulisan	Karakter	Penulisan	Karakter	Penulisan
\dots	<code>\ldots</code>	\cdots	<code>\cdots</code>	\vdots	<code>\vdots</code>
\ddots	<code>\ddots</code>	\forall	<code>\forall</code>		

Tabel 5.8: Simbol Matematika Lainnya

Bibliografi

- [1] Peter Flynn : *Formatting information A beginners introduction to typesetting with L^AT_EX*, Silmaril Consultants (1999)
- [2] Tobias Oetiker Hubert Partl, Irene Hyna and Elisabeth Schlegl : *The Not So Short Introduction to L^AT_EX 2_ε*, Free Software Foundation, Inc., (2005)
- [3] Scott Pakin : *The Comprehensive L^AT_EX Symbol List*, Comprehensive T_EX Archive Network, (2005)
- [4] Nicola Talbot : *Creating a PDF document using PDFLaTeX*, (2004)
- [5] Michael Doob : *A Gentle Introduction to T_EX : A Manual for Self-study*, Department of Mathematics, Manitoba University, (2005)
- [6] Nicola Talbot : *L^AT_EX for Complete Novice*, (2004)

“Siapa mencintai didikan,
mencintai pengetahuan”