

Belajar L^AT_EX 2_ε Dari Awal

Isro' Hidayatulloh

06 Juli 2015

Daftar Isi

I	Tahap Pengenalan	2
1	Apa itu L^AT_EX?	3
1.1	Instalasi	3
1.1.1	Instalasi di Linux	3
2	Dasar	5
2.1	Sintak L ^A T _E X	5
2.1.1	Spasi	5
2.1.2	Karakter Khusus	6
2.1.3	Group	7
2.1.4	Environment	7
2.1.5	Commands	7
2.1.6	Komentar	8
2.2	Membuat Dokumen Pertama	8
2.2.1	Apa Arti dari Kode Sumber?	9
2.3	File	9
2.3.1	Membuat Nama File yang Baik	9
2.3.2	File Tambahan	9

Bagian I

Tahap Pengenalan

BAB 1

Apa itu L^AT_EX?

T_EX adalah bahasa pemrograman yang dibuat oleh Donald Knuth¹ fungsinya untuk membuat dokumen yang baik dengan mudah. Lalu apa hubungannya dengan L^AT_EX, L^AT_EX adalah pengembangan dari T_EX dengan penambahan package atau style. Sehingga memberikan kemudahan untuk pembuatan dokumen yang kompleks, seperti untuk mengubah font, penambahan gambar, dan masih banyak lagi.

1.1 Instalasi

Instalasi L^AT_EX menggunakan sistem distribusi, distribusi L^AT_EX ini terdiri dari sekumpulan paket dan program. Berikut adalah distribusi L^AT_EX untuk kebanyakan sistem operasi:

- TeX Live² adalah kebanyakan distribusi T_EX untuk BSD, GNU/Linux, Mac OS X, dan Windows.
- MiKTeX³ adalah distribusi khusus Windows.
- MacTeX⁴ distribusi khusus Mac OS turunan dari TeX Live.

Paket distribusi ini tidak mengikutsertakan editor. Kita bisa menggunakan text editor biasa untuk membuat kode sumber L^AT_EX.

1.1.1 Instalasi di Linux

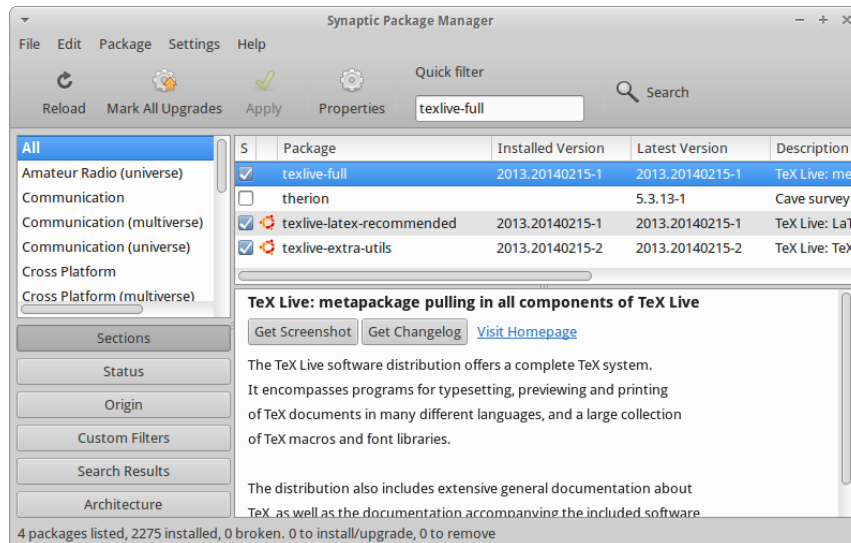
Linux terdiri dari banyak distro/distribusi, ada Ubuntu, Debian, Fedora, dsb. Kebanyakan distro telah mengikutsertakan distribusi TeX Live pada repositorinya. Pada Ubuntu kita bisa menggunakan aplikasi Synaptic Package Manager.

¹<http://en.wikipedia.org/wiki/Donald%20Knuth>

²<http://www.tug.org/texlive/>

³<http://www.miktex.org/>

⁴<http://www.tug.org/mactex/>



Gambar 1.1: Install Texlive-full Menggunakan Synaptic

BAB 2

Dasar

Pada bab ini kita akan mulai belajar dasar-dasar LaTeX, jadi anda harus sudah menginstall aplikasi LaTeX pada komputer. Kami menyarankan menggunakan Texmaker sebagai editor LaTeX. Karena kelengkapan dan juga kemudahan penggunaannya. Sedangkan pada sistem operasi Windows teman-teman bisa menggunakan Texwork.

2.1 Sintak \LaTeX

\LaTeX mengkonversi teks sumber dikombinasikan dengan markup kedalam dokumen dengan kualitas tinggi. File input untuk \LaTeX adalah plain text¹. Bentuk umum paling sederhana adalah sebagai berikut:

```
1 \documentclass{article}
2
3 \begin{document}
4 Hello world!
5 \end{document}
```

2.1.1 Spasi

Kompiler LaTeX menormalisasi spasi, baik itu berupa spasi atau tab. Meski kita membuat spasi sebanyak apapun LaTeX akan menganggapnya sebagai spasi tunggal. Begitu juga dengan ganti baris tunggal juga dianggap sebagai spasi, jika ganti baris ini dilakukan dua kali atau membuat baris kosong antar dua baris. Hal ini baru menjadikan Paragraf baru pada output.

Berikut contoh yang akan membantu anda memahamai bagaimana kerja spasi di LaTeX:

¹<http://en.wikipedia.org/wiki/plain%20text>

```

1 \documentclass[12pt]{article}
2
3 \begin{document}
4 Tidak peduli berapa pun anda
5 memasukkan satu atau beberapa      spasi
6 setelah sebuah kata.
7
8 Sebuah baris kosong memulai sebuah
9 baris baru.
10
11 \end{document}

```

Listing 2.1: spasi.tex

Sehingga output yang didapat dari kode sumber diatas adalah:

Tidak peduli berapa pun anda memasukkan satu atau beberapa spasi
 setelah sebuah kata.
 Sebuah baris kosong memulai sebuah baris baru.

Gambar 2.1: Spasi Pada LaTeX

Untuk selanjutnya silahkan anda beruji coba sendiri yah.. :-). Selanjutnya adalah

2.1.2 Karakter Khusus

Karakter-karakter ini adalah karakter/huruf yang memiliki suatu arti bagi kompiler LaTeX. Jika kita tuliskan karakter-karakter ini secara langsung pada kode, maka akan menghasilkan hal yang tidak sesuai keinginan. Karakter-karakter ini adalah :

\$ % ^ & _ { } ~ \

```

1 \# \$ \% \^{} \& \_ \{ \} \~{} \textbackslash{}

```

Listing 2.2: karakterkhusus.tex

Karakter backslash(\) tidak bisa ditambahkan dengan cara menambahkan backslash lagi setelahnya (\\). Cara ini digunakan untuk mengakhiri baris. Sedangkan pada karakter ^ dan ~ harus ditambahkan kurung kurawal, karena kalau tidak perintah ini \n akan menghasilkan ñ.

2.1.3 Group

Sebuah *group* pada dasarnya didefinisikan dengan sepasang kurung kurawal `{}`. Dan perintah-perintah yang ada didalamnya hanya berkerja pada group tersebut. Perintah `\begin{group}` dan `\endgroup` merupakan perintah alternatif dari kurung kurawal. Contoh:

```
1 {
2 \bf Yang ini BOLD
3 }
4 di sini tidak lagi
```

Yang ini BOLD di sini tidak lagi

group berguna ketika kita menginginkan suatu perintah hanya berpengaruh pada beberapa baris atau kata saja.

2.1.4 Environment

Environment di LaTeX punya aturan yang mirip dengan perintah/command, biasanya environment punya pengaruh lebih luas. Sintaks umumnya:

```
1 \begin{namaenvironment}
2 di sini teks yang terpengaruh oleh environment
3 \end{namaenvironment}
```

Diantara `\begin` dan `\end` kita dapat menambahkan perintah lain dan juga bisa nested environment(environment bertumpuk).

2.1.5 Commands

(*Commands*) bersifat case sensitive dan mengikuti dua aturan berikut:

- diawali dengan backslash `\` dan selanjutnya nama command yang hanya berisi hanya huruf. Command diterminasi dengan sebuah spasi, sebuah angka atau apapun selain huruf.
- Berisi sebuah backslash `\` dan tentunya sebuah non-huruf.

Beberapa commands butuh argumen yang diletakkan di dalam kurung kurawal `{}` setelah nama command. Juga terkadang butuh opsi tambahan yang diletakkan di dalam kurung siku `[]`. Sintaks sebagai berikut:

```
1 \namacommand[ opsi 1 , opsi 2 , ... ] { argumen1 } argumen2 } ...
```


Kebanyakan perintah standar LaTeX punya *switch*. Switch ini adalah sebuah ke-samaan dari command/perintah hanya saja tidak punya argumen dan berada di dalam kurung kurawal. Jangan memanggilnya diluar kurung karena akan berpengaruh ke seluruh dokumen.

Contoh:

```

1 % \emph adalah command/baris perintah, \em adalah switch
2 \emph{text tercetak miring}, bagian ini normal % Format Benar
3 {\em text tercetak miring}, bagian ini normal % Format benar
4
5 \em text tercetak miring, bagian normal % Format salah
6 \emph{text tercetak miring}, bagian normal % Format salah

```

2.1.6 Komentar

Jika pada input file terdapat karakter % maka baris setelahnya tidak ikut dieksekusi, dan tidak berpengaruh pada hasil output. Biasanya digunakan untuk membuat catatan tentang perintah atau keterangan dari dokumen yang dibuat.

```

1
2 Saya sedang % mangan telo
3 % Jadi ini lah komentar, bukan komentator
4 menulis % huruf apa saja ya..?%^&
5     menggunakan % Pisau tajam
6 \LaTeXe

```

Listing 2.3: komentar.tex

Saya sedang menulis menggunakan L^AT_EX 2_ε

Sebagai catatan bahwa karakter % bisa digunakan untuk memisahkan kata yang panjang tanpa ada spasi atau ganti baris.

2.2 Membuat Dokumen Pertama

Sekarang saatnya membuat dokumen sederhana untuk langkah awal. Kita bisa membuat kode sumber LaTeX ini menggunakan text editor seperti Notepad++, vim, dan gedit. Namun kami sarankan untuk menggunakan editor LaTeX khusus seperti Texwork untuk Windows sedangkan untuk Linux menggunakan Texmaker.

```

1 % assalam.text - Contoh pertama LaTeX
2 \documentclass{article}
3 \begin{document}
4 Assalamualaikum Warahmatulloh... % Salam Rek..! Biar beda :-)

```

```
5 | \end{document}
```

Listing 2.4: assalam.tex

2.2.1 Apa Arti dari Kode Sumber?

Mari kita pelajari tiap baris dari kode sumber di atas:

- `% assalam.text` - Contoh pertama LaTeX
Baris pertama adalah baris komentar, baris ini tidak akan nampak dan tidak menjalankan perintah apapun. Bisa digunakan untuk memberi keterangan mengenai isi kode sumber, dan informasi tentang pengarang, dsb.
- `\documentclass{article}`
Ini adalah baris perintah atau command yang memberitahukan LaTeX untuk menggunakan format dokumen `article`.
- `\begin{document}`
Ini adalah gerbang untuk lingkungan dokumen, Isi dokumen diletakkan setelah perintah ini. Sedangkan perintah-perintah yang berada di atasnya dinamakan *preamble*.
- `Assalamualaikum Warahmatulloh... % Salam Rek..! Biar beda :-)`
Nah ini adalah baris yang akan nampak pada hasil akhir.
- `\end{document}`
Ini adalah baris penutup dari perintah `\begin{document}`. Apa pun setelah perintah ini tidak akan dipedulikan.

2.3 File

2.3.1 Membuat Nama File yang Baik

Tidak semua nama file bisa digunakan, ada tata caranya. Jangan menggunakan nama file atau folder menggunakan spasi, kita bisa membuat nama yang panjang tapi hindari spasi. Gunakan huruf (a-z) angka (1-9) dan tanda minus (-) dan sebuah titik untuk pemisah ekstensi file. Jangan mencampur antara huruf besar dan kecil dalam satu nama, karena beberapa sistem operasi tidak memperdulikan namun yang lainnya membedakannya.

2.3.2 File Tambahan

Selain file `.tex` ada beberapa file pendukung lain, file-file ini ada yang bersifat semetara. Berikut ekstensi umum file-file di LaTeX:

Ekstensi	Penjelasan
.aux	File ini digunakan untuk menyimpan informasi yang berhubungan dengan cross-reference
.bbl	File bibliography, merupakan output dari BiBTeX dan digunakan oleh LaTeX
.bib	File database bibliography
.blg	File log dari BiBTeX
.bst	File Tipe/model BiBTeX
.cls	File class yang mendefinisikan seperti apa dokumen anda. Dipilih dengan menggunakan perintah <code>\documentclass</code> .
.dtx	TeX terdokumentasi