

# **CERDAS MENGUASAI LATEX**



---

# CERDAS MENGUASAI LATEX

## Dalam 24 Jam

---

**Rolly M. Awangga**  
Informatics Research Center



**Kreatif Industri Nusantara**

***Penulis:***

Rolly Maulana Awangga

ISBN : 978-602-53897-0-2

***Editor:***

M. Yusril Helmi Setyawan

***Penyunting:***

Syafrial Fachrie Pane

Khaera Tunnisa

Diana Asri Wijayanti

***Desain sampul dan Tata letak:***

Deza Martha Akbar

***Penerbit:***

Kreatif Industri Nusantara

***Redaksi:***

Jl. Ligar Nyawang No. 2

Bandung 40191

Tel. 022 2045-8529

Email : awangga@kreatif.co.id

***Distributor:***

Informatics Research Center

Jl. Sariasih No. 54

Bandung 40151

Email : irc@poltekpos.ac.id

Cetakan Pertama, 2019

Hak cipta dilindungi undang-undang

Dilarang memperbanyak karya tulis ini dalam bentuk dan dengan cara  
apapun tanpa ijin tertulis dari penerbit

*‘Jika Kamu tidak dapat  
menahan lelahnya  
belajar, Maka kamu harus  
sanggup menahan  
perihnya Kebodohan.’  
Imam Syafi’i*

# CONTRIBUTORS

---

ROLLY MAULANA AWANGGA, Informatics Research Center., Politeknik Pos Indonesia, Bandung, Indonesia



# CONTENTS IN BRIEF

---





# DAFTAR ISI

---



# DAFTAR GAMBAR

---



# DAFTAR TABEL

---



# Listings

---





# FOREWORD

---

Sepatah kata dari Kaprodi, Kabag Kemahasiswaan dan Mahasiswa



# KATA PENGANTAR

---

Buku ini diciptakan bagi yang awam dengan git sekalipun.

R. M. AWANGGA

*Bandung, Jawa Barat  
Februari, 2019*



# ACKNOWLEDGMENTS

---

Terima kasih atas semua masukan dari para mahasiswa agar bisa membuat buku ini lebih baik dan lebih mudah dimengerti.

Terima kasih ini juga ditujukan khusus untuk team IRC yang telah fokus untuk belajar dan memahami bagaimana buku ini mendampingi proses Intership.

R. M. A.



# ACRONYMS

---

ACGIH	American Conference of Governmental Industrial Hygienists
AEC	Atomic Energy Commission
OSHA	Occupational Health and Safety Commission
SAMA	Scientific Apparatus Makers Association





# GLOSSARY

---

git	Merupakan manajemen sumber kode yang dibuat oleh linus torvald.
bash	Merupakan bahasa sistem operasi berbasiskan *NIX.
linux	Sistem operasi berbasis sumber kode terbuka yang dibuat oleh Linus Torvald



# SYMBOLS

---

$A$  Amplitude

$\&$  Propositional logic symbol

$a$  Filter Coefficient

$\mathcal{B}$  Number of Beats



# INTRODUCTION

---

ROLLY MAULANA AWANGGA, S.T., M.T.

Informatics Research Center  
Bandung, Jawa Barat, Indonesia

Pada era disruptif saat ini. git merupakan sebuah kebutuhan dalam sebuah organisasi pengembangan perangkat lunak. Buku ini diharapkan bisa menjadi penghantar para programmer, analis, IT Operation dan Project Manajer. Dalam melakukan implementasi git pada diri dan organisasinya.

Rumusnya cuman sebagai contoh aja biar keren[?].

$$ABCDEF\alpha\beta\Gamma\Delta\sum_{def}^{abc} \tag{I.1}$$



# BAB 1

---

## EDITOR DAN COMPILER

---

### 1.1 Mengenal .tex

Pertama pahami dulu bagaimana badan isi file .tex yang akan kita kerjakan. Download atau lihat salah satu file latex yang akan kita kerjakan. Untuk mengisi latex kita harus mengisinya di dalam komponen yang merupakan tag dengan pembuka begin dan diakhiri dengan end. Kemudian kenali bagian buku terdiri dari part, chapter dan section. Part itu bisa kita andaikan bab, chapter sub bab, dan section adalah bagian.

Kita bisa memisahkan isi dari latex dengan perintah input kemudian di dalam kurung kurawal letak file .tex yang akan kita masukkan kedalam file utama latex tersebut.

LATEX merupakan program pengolahan kata atau sistem persiapan pembuatan dokumen untuk pengetikan sistem TeX, yang dinamakan berdasarkan gaya penulisannya sebagai LaTeX. Nama LaTeX itu sendiri hanya mengacu pada bahasa penulisan yang digunakan pada sebuah dokumen, bukan pada editor yang digunakan untuk menulis dokumen tersebut. Untuk membuat dokumen dalam format LaTeX, sebuah file berformat .tex harus dibuat menggunakan semacam text editor. Walaupun, banyak text editor yang dapat digunakan untuk membuat dokumen LaTeX, beberapa text editor sengaja dibuat khusus untuk menggunakan bahasa LaTeX.



### 1.1.1 Keuntungan Latex

1. Tersedianya beberapa program untuk melihat hasil pemrosesan latex yang dapat menampilkannya persis seperti hasil cetakan dengan printer
2. Penulisan rumus matematis dapat dilakukan dengan cara sangat mudah dan profesional
3. Banyak jurnal internasional yang menerima artikel artikel dalam format tex
4. Pemakai hanya perlu belajar sedikit perintah yang mudah dipahami yang menyatakan struktur logis sebuah dokumen
5. Latex mendorong pengarang untuk menulis naskah yang tersusun dengan baik

### 1.1.2 Perbedaan LateX dari program pengolahan data lainnya

1. Latex menggunakan perintah yang diawali dengan backslash sedangkan untuk program olah data lain tidak ada.
2. Peraturan diLateX umumnya bersifat otomatis
3. Peraturan simbol matematika cukup dengan menekan epsilon yang sudah tersedia

### 1.1.3 Kekurangan pada LateX

1. Tidak memiliki user-friendly seperti MS Word
2. Tidak cocok untuk dokumen yang singkat karena hanya biasanya penuh dengan simbol matematika
3. latex menggunakan perintah yang sudah disesuaikan dengan penggunaannya.

## 1.2 Compiler

Kemudian untuk dapat menuliskan kode LaTeX kita harus menggunakan editor LaTeX. Oleh karena itu pastikan kita sudah meng-install aplikasi editor LaTeX seperti texworks, texmaker, winedt dll. Untuk dapat melihat perintah yang sudah kita lakukan, kita harus melakukan compile. *Compile* merupakan proses eksekusi yang dilakukan untuk melihat hasil dari perintah yang telah kita buat dalam LaTeX yang dapat kita preview dalam bentuk file dengan ekstensi PDF. Editor LaTeX yang saya praktekan disini adalah texmaker. Kemudian untuk dapat menuliskan kode LaTeX kita harus menggunakan editor LaTeX. Oleh karena itu pastikan kita sudah meng-install aplikasi editor LaTeX seperti texworks, texmaker, winedt dll. Untuk dapat melihat perintah yang sudah kita lakukan, kita harus melakukan compile. *Compile* merupakan proses eksekusi yang dilakukan untuk melihat hasil dari perintah yang telah kita buat

dalam LaTeX yang dapat kita preview dalam bentuk file dengan ekstensi PDF. Editor LaTeX yang saya praktekan disini adalah texmaker. TexMaker merupakan salah satu perangkat lunak IDE, yang berlisensi GNU/GPL yang dibuat untuk mempermudah proses sebuah dokumen seperti tesis, penelitian dll. Kita bisa melakukan kompilasi dengan perintah yang ada di listing 1.1.

```

1 pdflatex -shell-escape -interaction=nonstopmode -file-line-error git.
  tex | grep "\.:[0-9]*:.*|LaTeX Warning:"
2
3 pdflatex -shell-escape -interaction=nonstopmode -file-line-error git.
  tex | grep "\.:[0-9]*:.*"
4
5 pdflatex -shell-escape -interaction=nonstopmode -file-line-error git.
  tex | grep -i "\.:[0-9]*:.*\|warning"

```

**Listing 1.1** Perintah kompilasi latex keluaran pdf

### 1.3 Hello LATEX

Sekarang Kamu sudah mempunyai distribusi LATEX yang sudah terinstal. Mari kita coba jalankan! Pertama buka editor Latex dan simpan file dengan nama hello.tex seperti pada listing 1.2.

```

1 \documentclass{article}
2 % Say hello
3 \begin{document}
4 Hello , World!
5 \end{document}

```

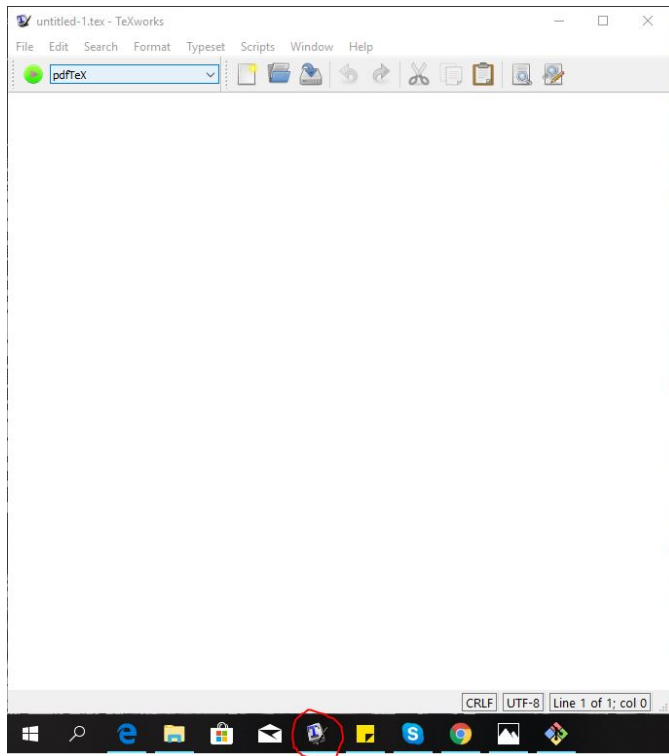
**Listing 1.2** Contoh listing kode hello Latex

[?]

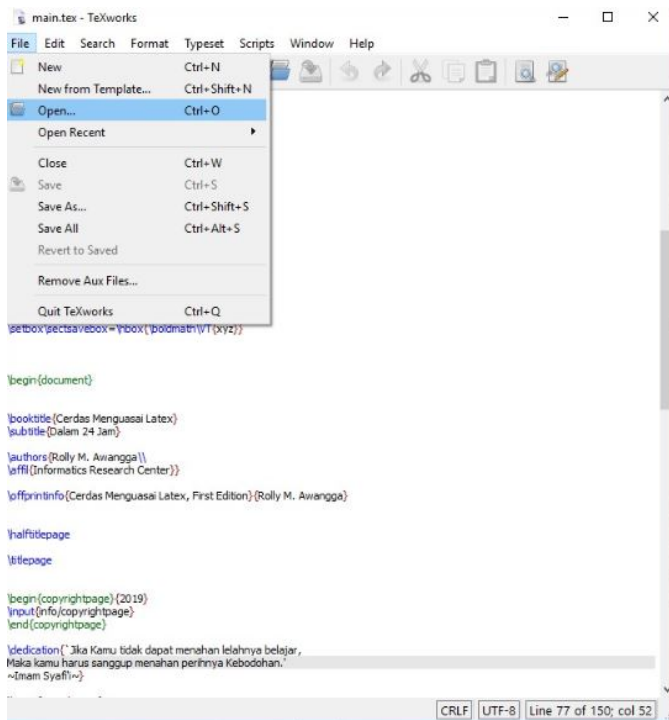
### 1.4 Mengedit Menggunakan Editor LaTeX

Editor khusus pada LaTeX dibutuhkan agar kita dapat mengedit sebuah dokumen. Dengan menggunakan Editor Latex kita dapat mengedit sebuah file dokumen lalu kita dapat melihat langsung hasil dari perintah yang telah dilakukan dalam bentuk PDF. Bagaimana cara mengedit sebuah file pada LaTeX? Berikut penjelasannya :

1. Pertama pastikan anda telah meng-install Editor latex yang akan digunakan (Editor latex yang digunakan disini adalah TeXworks)
2. Buka Editor latex yang sudah kalian install 1.1
3. Setelah membuka editor latex, pastikan terlebih dulu anda telah memiliki file atau dokumen yang akan anda edit.
4. Kemudian, pilih menu file dan buka menu open file seperti pada gambar 1.2

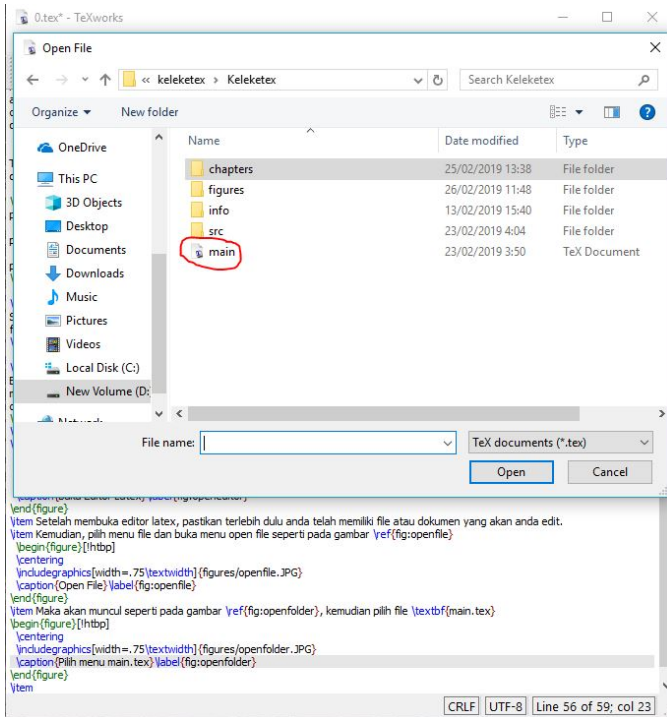


**Gambar 1.1** Buka Editor Latex



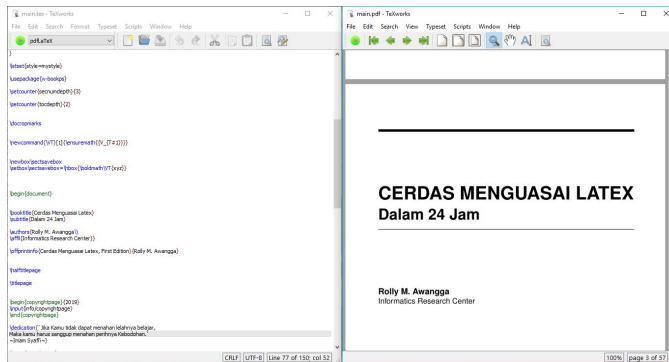
**Gambar 1.2** Open File

5. Setelah memilih menu file kemudian buka file main.tex seperti pada gambar 1.3, kemudian pilih file **main.tex**

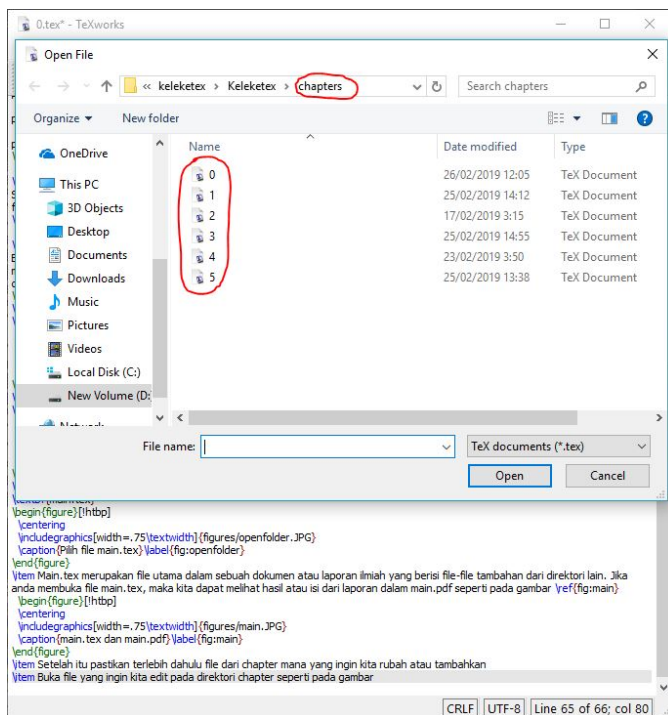


**Gambar 1.3** Pilih file main.tex

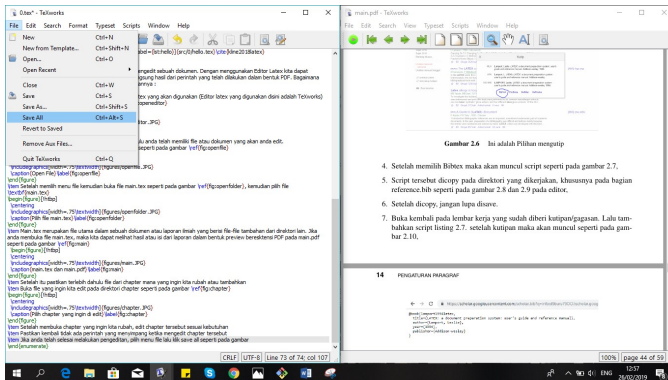
6. Main.tex merupakan file utama dalam sebuah dokumen atau laporan ilmiah yang berisi file-file tambahan dari direktori lain. Jika anda membuka file main.tex, maka kita dapat melihat hasil atau isi dari laporan dalam bentuk preview beres-tesensi PDF pada main.pdf seperti pada gambar 1.4
7. Setelah itu pastikan terlebih dahulu file dari chapter mana yang ingin kita rubah atau tambahkan
8. Buka file yang ingin kita edit pada direktori chapter seperti pada gambar 1.5
9. Setelah membuka chapter yang ingin kita rubah, edit chapter tersebut sesuai kebutuhan
10. Pastikan kembali tidak ada perintah yang menyimpang ketika mengedit chapter tersebut
11. Jika anda telah selesai melakukan pengeditan, pilih menu file lalu klik save all seperti pada gambar 1.6



Gambar 1.4 main.tex dan main.pdf

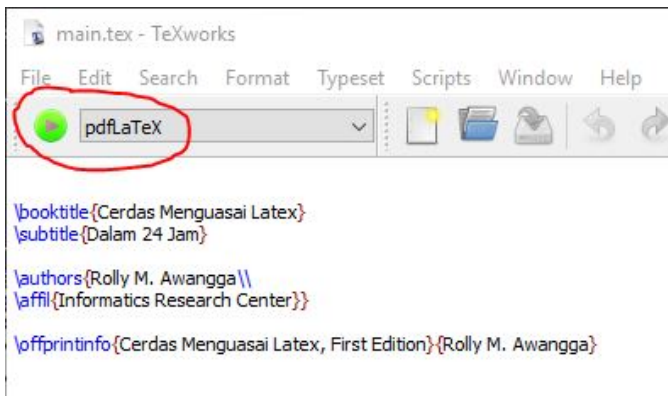


Gambar 1.5 Pilih chapter yang ingin di edit



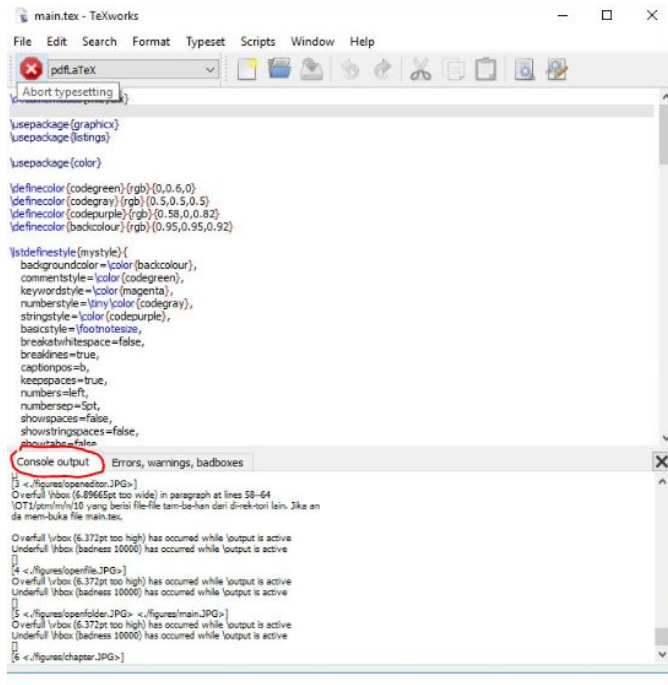
**Gambar 1.6** Pilih Save all

12. Setelah itu agar perintah yang telah kita buat dapat dijalankan, buka kembali main.tex lalu pilih button *typeset* dengan format *pdflatex* untuk melakukan compile seperti pada gambar 1.7



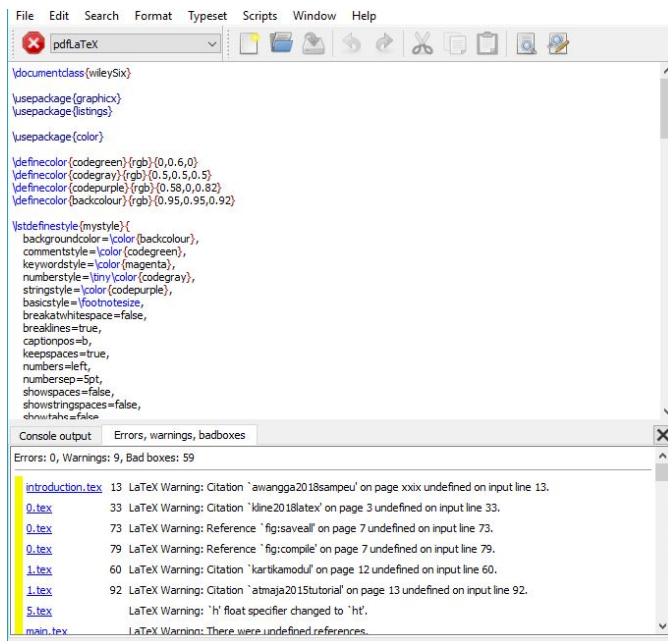
**Gambar 1.7** Proses Melakukan Compile

13. Setelah melakukan compile pastikan kembali tidak ada perintah yang error pada *console outputnya* seperti pada gambar 1.8
14. Jika terjadi error kita dapat melihat error tersebut dan memperbaikinya pada bagian *Error*, *Warnings*, *Badboxes* seperti pada gambar 1.9



**Gambar 1.8** Console Output





Gambar 1.9 Errors

## BAB 2

---

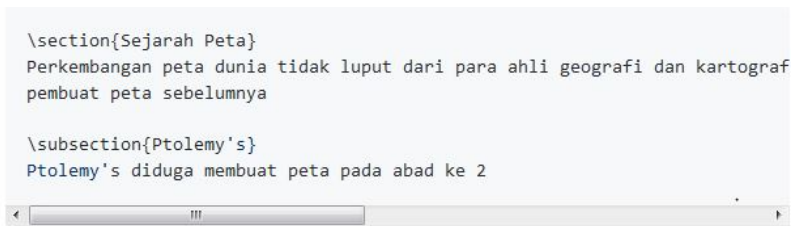
# PENGATURAN PARAGRAF

---

### 2.1 Pembagian bab

Secara default pembagian bab pada latex menggunakan perintah *section*, *subsection*, *subsubsection* dan *subsubsubsection*. Untuk mengatur kedalaman suatu dokumen pada bab bab tertentu, kita dapat menggunakan perintah berikut ini pada bagian Preamble : `setcounter{secnumdepth}` `setcounter{tocdepth}`

Opsi yang digunakan pada syntax `secnumdepth` pada perintah `verbcounter=` seperti perintah diatas, berarti Anda telah merubah kedalaman bab yang Anda perbaharui sampai dengan level 5 yaitu *section* – *subsection* – *subsubsection* – *paragraph* – *subparagraph*. Sedangkan pada perintah dari opsi `tocdepth` berfungsi untuk membuat table of contents atau menampilkan kedalaman bab sampai dengan level 5, namun jika tidak di setel maka pada bagian level 3 kebawah tidak akan dapat ditampilkan pada bagian toc 2.1.



**Gambar 2.1** Pembagian Bab.

## 2.2 Format Cetak

Pada format LaTeX teks mempunyai bentuk plaintext, yang artinya teks tersebut belum diformat. Pada proses formatting teks dapat dilakukan dengan bahasa tersendiri yaitu bahasa markup. Hal paling mendasar antara lain cetak tebal, miring dan garis bawah. Cetak tebal menggunakan perintah *textbf*, cetak miring menggunakan perintah *textit* dan garis bawah menggunakan perintah *underline*.

## 2.3 Tanda petik

Tanda petik di Latex menggunakan petik miring dan petik satu. Petik miring biasanya berada pada sebelah angka satu di keyboard dan diakhiri petik satu. Ingat fungsi tanda petik hanya untuk melakukan quote atau pengutipan langsung. Untuk istilah bahasa Inggris gunakan miring atau italic.

```
1 'kalimat dalam tanda petik'
```

**Listing 2.1** Contoh kalimat dalam tanda petik pada Latex

## 2.4 Penomoran

Perintah penomoran pada latex biasanya menggunakan format *Numbering* atau format *Bullets*. Perintah yang digunakan pada format *Numbering* adalah *enumerate* sedangkan untuk *Bullets* yang menyerupai poin menggunakan *itemize*.

*Numbering* merupakan perintah yang digunakan untuk membuat daftar berurut dengan penomoran menggunakan angka (numbered list), yang biasanya diberikan pada awal baris baru. Sedangkan *Bullets* atau poin adalah perintah yang digunakan untuk membuat daftar berurut dengan penomoran berupa symbol atau poin (bulleted list). Pada listing 2.3 adalah perintah untuk memasukan listing pada penomoran .

```

1 \begin{enumerate}
2   \item 1
3   \item 2

```

```
4 \end{enumerate}
```

### Listing 2.2 Memberikan Perintah Numbering

Sedangkan *Bullets* atau poin adalah perintah yang digunakan untuk membuat daftar berurut dengan penomoran berupa symbol atau poin (bulleted list) 2.3.

```
1 \begin{itemize}
2 \item
3 \item
4 \end{itemize}
```

### Listing 2.3 Menambahkan kode perintah bullets

## 2.5 Kode Program

Agar kita dapat memasukan kode program, kita dapat menggunakan perintah *lstlisting*. Perintah ini berfungsi untuk memasukkan atau menambahkan kode program apapun ke dalam file yang terpisah. Untuk memasukan perintah *lstlisting* kita perlu menulis parameter *caption* dan *label* untuk memberikan penjelasan keterangan kode program dan sebagai sumber referensi dari label kode program.

```
1 \begin{lstlisting}[caption=Contoh kalimat dalam tanda petik di Latex,
   label={lst:tandapetik}]
2 'kalimat dalam tanda petik'
3 \end{lstlisting}
```

### Listing 2.4 Menambahkan kode program

## 2.6 Tabel

Untuk dapat membuat tabel kita harus menggunakan perintah *table*. Selain itu kita juga perlu menambahkan referensi pada tabel yang terdapat dalam kalimat berdasarkan labelnya 2.5

```
1 \begin{table}[h]
2 \caption{Small Table}
3 \centering
4 \begin{tabular}{ccc}
5 \hline
6 one&two&three\\
7 \hline
8 C&D&E\\
9 \hline
10 \end{tabular}
11 \label{table:contoh}
12 \end{table}
```

### Listing 2.5 Menambahkan perintah table

Untuk dapat membuat tabel kita harus menggunakan perintah *table*. Selain itu kita juga perlu menambahkan referensi pada tabel yang terdapat dalam kalimat berdasarkan labelnya.

Untuk dapat membuat tabel kita harus menggunakan perintah *table*. Selain itu kita juga perlu menambahkan referensi pada tabel yang terdapat dalam kalimat berdasarkan labelnya [?].

## 2.7 Document class

Pada dokumen Latex terdapat atau mempunyai beberapa struktur yang dicirikan dengan blok yang diberi apit oleh perintah *begin* dan *end*. Latex memberikan pilihan Class dokumen yang bisa di pakai, antara lain adalah Book, Report, Article dan lain sebagainya. Class document book merupakan Class Document yang paling tepat untuk menulis, karena dapat mendukung table of contents yang dapat berfungsi langsung untuk generate daftar isi secara langsung.

Format ukuran font pada latex 2.2

Dalam memberikan penulisan judul pada format latex biasanya di letakkan pada awal document, untuk cara penulisan nya dapat dilakukan sebagai berikut:

1. *backslash* document class kurung kurawal a4papper, ukuran yang di inginkan tutup kurawal lalu report
2. *backslash* begin buka kurawal document tutup kurawal
3. *backslash* begin buka kurawal judul document tutup kurawal
4. *backslash* autor buka kurawal nama penulis tutup kurawal
5. *backslash* date buka kurawal tanggal pembuatan tutup kurawal
6. *backslash* maketitle
7. *backslash* and buka kurawal document tutup kurawal

## 2.8 Costum Command

Sesuai dengan namanya Costum Command, dimana ke unggulan latex ada fitur yang satu ini, Pembuat dokumen ini dapat membuat macro untuk kebutuhan yang sifatnya spesifik dan berulang-ulang, dimana costum cummad dapat melakukan tanda bintang berjejer sebagai penanda garis.

## 2.9 Membuat Penomoran Referensi

Untuk menambahkan referensi atau melakukan sanitasi pada latex kita dapat menggunakan berbagai macam cara. Salah satu cara sederhana yang dapat kita gunakan

1. Tiny  
`{\tiny teks yang ingin diformat }`
2. Scriptsize  
`{\scriptsize teks yang ingin diformat }`
3. Footnotesize  
`{\footnotesize teks yang ingin diformat }`
4. Small  
`{\small teks yang ingin diformat }`
5. Normal  
`{\normalsize teks yang ingin diformat }`
6. Large  
`{\large teks yang ingin diformat }`
7. Larger  
`{\LARGE teks yang ingin diformat }`
8. Largest  
`{\LARGE teks yang ingin diformat }`
9. Huge  
`{\huge teks yang ingin diformat }`
10. Huger  
`{\Huge teks yang ingin diformat }`

**Gambar 2.2** ukuran font

adalah dengan menggunakan environment yang di sebut *thebibliography*. Namun, kebanyakan orang saat ini menggunakan *BibTeX* untuk melakukan sanitasi sebagai acuan referensi. Dengan menggunakan *BibTeX* kita dapat mengatur sitasi sendiri secara terpisah dalam format file .bib [?]. Disaat mengutip maupun menggunakan sanitasi diperkenankan untuk memberi keterangan referensi atau sumber asal suatu kutipan dan gagasan. Untuk mengetahui bagaimana menambahkan referensi pada latex, kita dapat melihat langkah-langkahnya seperti pada gambar 2.3.

Bagaimana cara membuatnya di Latex? berikut cara membuatnya:

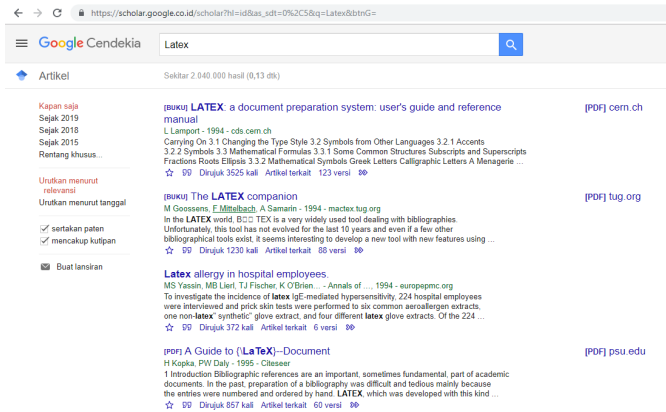
1. Cari materi yang akan dikutip melalui Google Scholar seperti pada gambar 2.4

## 7.1 ASCII

### 7.1.1 Definisi ASCII

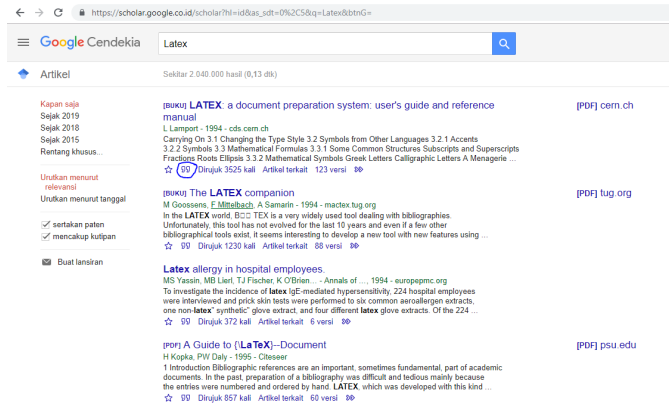
Berdasarkan artikel yang ditulis oleh hieronymus [2] ASCII atau American Standard Code for Information Interchange merupakan sebuah pengkodean berstandar Internasional yang berupa kode huruf dan simbol, seperti Hex dan Unicode dan juga merupakan simbol tambahan dari database. ASCII bersifat universal contohnya 124

### Gambar 2.3 Ini adalah Contoh Penomoran Referensi



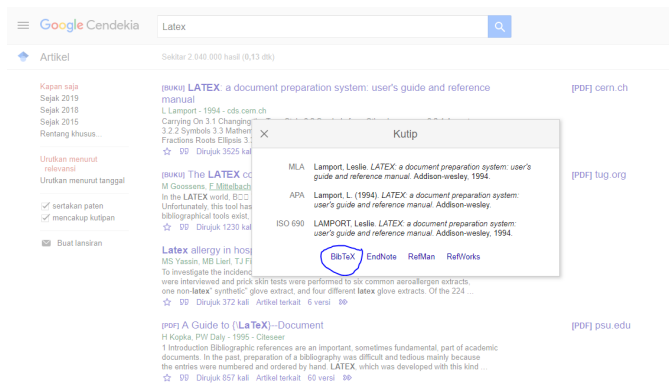
### Gambar 2.4 Ini adalah Halaman Google Scholar

- Setelah selesai mengutip jangan lupa untuk mengambil script bibtexnya dengan cara klik pada tanda kutip seperti pada gambar 2.5,



**Gambar 2.5** Ini adalah Tanda proses awal mengambil reference

- Maka akan muncul seperti gambar 2.6, lalu pilih Bibtex.



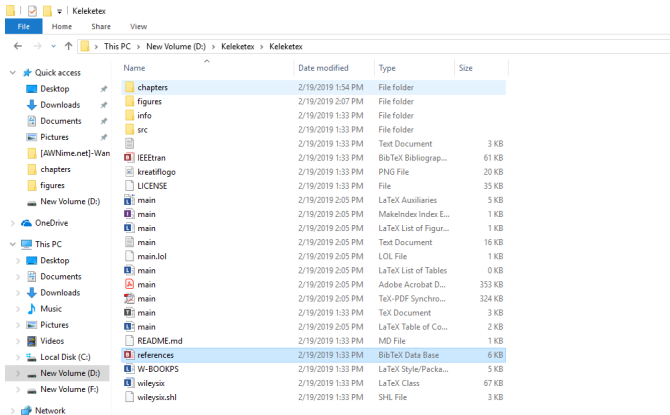
**Gambar 2.6** Ini adalah Pilihan mengutip

- Setelah memilih Bibtex maka akan muncul script seperti pada gambar 2.7,
- Script tersebut dicopy pada direktori yang dikerjakan, khususnya pada bagian reference.bib seperti pada gambar 2.8 dan 2.9 pada editor,
- Setelah dicopy, jangan lupa disave.
- Buka kembali pada lembar kerja yang sudah diberi kutipan/gagasan. Lalu tambahkan script listing 2.6. setelah kutipan maka akan muncul seperti pada gambar 2.10,

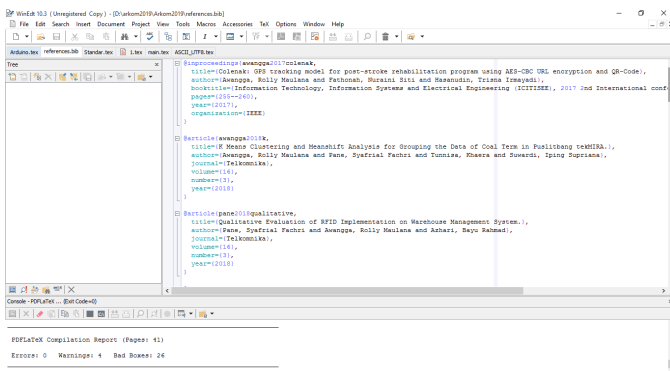


```
@book{lampport1994latex,
  title={LATEX: a document preparation system: user's guide and reference manual},
  author={Lampport, Leslie},
  year={1994},
  publisher={Addison-wesley}
}
```

**Gambar 2.7** Ini adalah Script BibTex



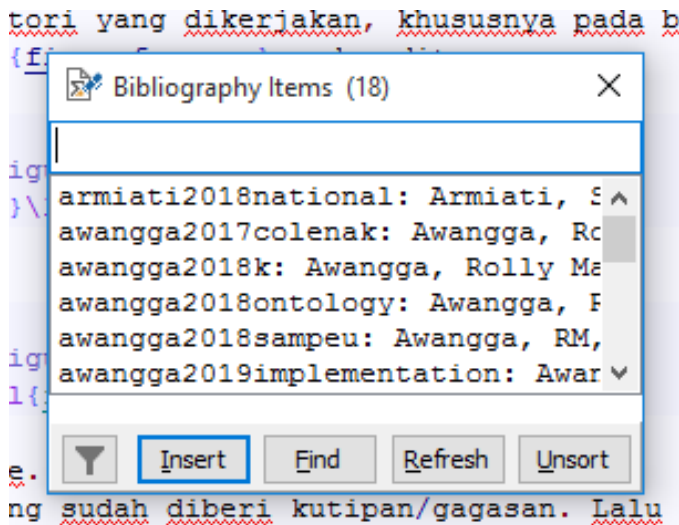
**Gambar 2.8** Ini adalah Direktori pekerjaan



**Gambar 2.9** Ini adalah Reference.bib

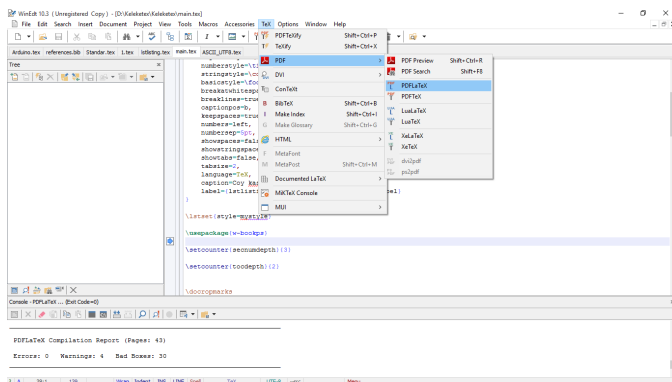
- 1 `\cite{armiati2018national}`
- 2 'di dalam kurung kurawal diisi sesuai sumber'

**Listing 2.6** Penggunaan perintah cite untuk reference

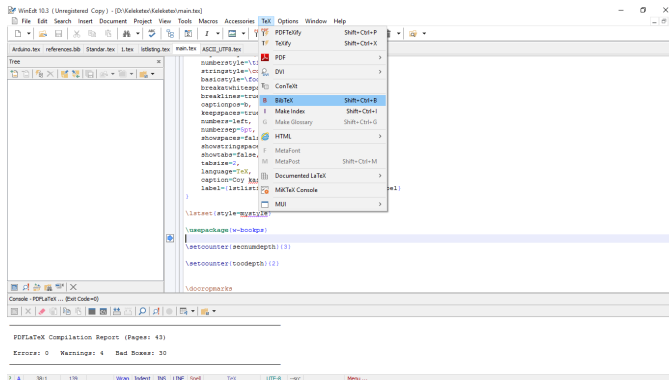


**Gambar 2.10** Ini adalah Proses pemilihan sumber

8. Pilih insert dan save.
9. Untuk proses compilenya dilakukan 2 kali yaitu pada main.tex pilih Tex lalu pilih pdflatex dan Bibtex, dilakukan berulang minimal 3 kali compile. Seperti pada gambar 2.11 untuk pdflatex dan 2.12 untuk BibTex.



**Gambar 2.11** Ini adalah Compile pdflatex



**Gambar 2.12** Ini adalah Compile BibTeX

## 2.10 Menambahkan Spesial Karakter

Untuk menambahkan karakter spesial pada LaTeX kita dapat menggunakan tanda *backslash* didepan karakter yang ingin kita tandai. Terdapat beberapa karakter yang tidak bisa langsung digunakan seperti tanda *ampersand*. Selain itu format pemberian kutipan pada LaTeX berbeda dengan pemberian kutipan pada editor lainnya, cara memasukkan karakter spesial menggunakan listing 2.7

```

1 \&
2 \%
3 \$
4 \#
5 \{ \}
6 \_
7 \"dalam petik\"
8 'dalam petik'
9 jika spesial karakter menjadi banyak atau satu baris gunakan verb
10 contoh :
11 \verb |%$'%&$$'%'%'&'%|

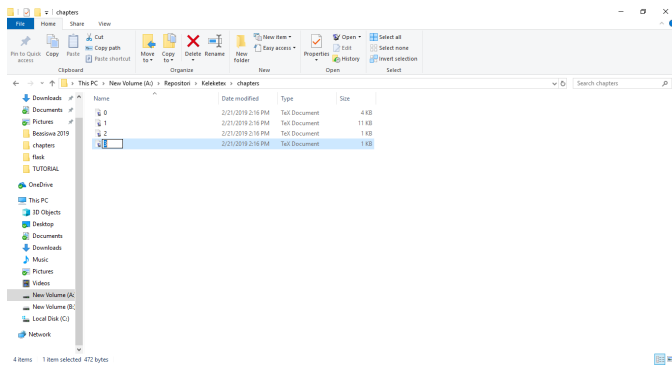
```

**Listing 2.7** Contoh kode untuk menambahkan karakter spesial

## 2.11 Menambahkan Chapter

Berikut ini merupakan langkah-langkah untuk menambahkan *chapter* baru.

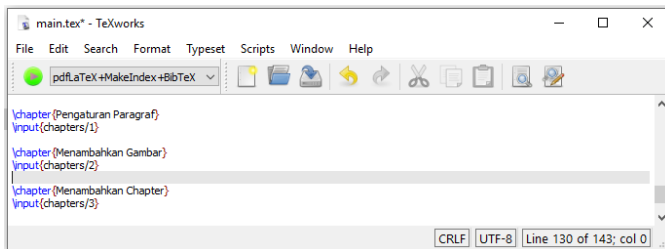
1. Pertama kita buat *chapter* baru pada repositori kita di folder *chapters*, seperti pada gambar ??.
2. Kemudian kita tambahkan kode seperti pada *listing* ?? yang berfungsi untuk memanggil *chapter* yang baru kita tambahkan pada file *main.tex* seperti pada gambar ??.



**Gambar 2.13** Menambahkan Chapter Baru

```
1 \chapter{Judul Bab}
2 \input{lokasi_chapter}
```

**Listing 2.8** Penggunaan perintah input untuk menambahkan chapter



**Gambar 2.14** Menambahkan Perintah Input Chapter

3. Terakhir, compile file `main.tex` untuk melihat chapter baru yang telah kita tambahkan pada file `main.pdf`.



## BAB 3

---

# MENAMBAHKAN GAMBAR

---

### 3.1 Gambar

Untuk dapat menambahkan gambar pada latex biasanya hal yang harus dilakukan adalah mendeklarasikan penggunaan paket `graphicx` pada bagian preamble agar latex dapat menempatkan gambar sesuai dengan yang kita inginkan di dalam sebuah dokumen. Cara menambahkan gambar dapat dilihat seperti pada listing ??.

```
1 \begin{figure}[!htbp]
2   \includegraphics[width=0.75\textwidth]{figures/namagambar.JPG}
3   \caption{penjelasan keterangan gambar.}
4   \label{labelgambar}
5 \end{figure}
```

**Listing 3.1** Contoh kode untuk menambahkan gambar

Beberapa hal yang harus kita perhatikan dari format perintah penambahan gambar :

1. File gambar yang ingin kita masukkan kedalam dokumen harus diletakkan pada direktori yang sama dengan direktori file dokumen ( .tex) yang telah kita buat.

2. Panjang dan Lebar suatu gambar dapat diubah sesuai dengan yang kita inginkan. Perintah *width* berfungsi untuk mengatur lebar gambar, sedangkan perintah *height* berfungsi untuk mengatur tinggi gambar tersebut.
3. Dengan mengatur **width** dan **height** kita dapat memasukkan gambar meskipun gambar tersebut memiliki ukuran dimensi yang besar.
4. Format gambar standar latex adalah **.eps** (*Encapsulated PostScript*) namun kita juga dapat menggunakan format **.jpg**.

### 3.1.1 Posisi Gambar

Pengaturan posisi gambar dapat kita tentukan melalui 2 hal :

- Perataan tepian dokumen : Dari contoh yang sudah kita lihat pada listing kita bisa menambahkan perintah tertentu untuk mengubah posisi dari suatu gambar, misalkan kita menambahkan perintah *center* maka posisi gambar akan berubah ketengah sesuai dengan perintah yang diberikan.
- Huruf-huruf seperti [**htbp**] yang terdapat pada perintah diatas juga memiliki fungsi untuk mengatur gambar pada suatu halaman.
  - h** : berfungsi untuk meletakkan gambar persis ditempat perintah tersebut dituliskan didalam dokumen.
  - t** : berfungsi untuk meletakkan gambar dibagian atas halaman.
  - b** : berfungsi untuk meletakkan gambar dibagian bawah halaman.
  - p** : berfungsi untuk meletakkan gambar pada sebuah halaman khusus yang memuat hanya gambar itu saja.
- Saat kita menggunakan **h**, latex akan secara otomatis menempatkan gambar di-halaman baru jika tidak ada cukup ruang untuk memuat gambar tersebut dengan perintah yang telah dituliskan pada gambar.

## BAB 4

---

# NOTASI MATEMATIKA PADA LATEX

---

### 4.1 Membuat Rumus dengan LaTeX

Sebagai aplikasi editor pengolah dokumen, LATEX memiliki kemampuan yang mampu menghasilkan dokumen berisi notasi-notasi matematis. Agar dapat menghasilkan dokumen yang berisikan notasi-notasi matematis maka kita harus berada dalam *Mathematics Environment*.

### 4.2 Penulisan Notasi Matematika

Pada latex kita dapat menuliskan suatu notasi matematika yang cukup panjang dalam suatu paragraf baru. Penulisan Notasi Matematika dalam suatu paragraf dapat dilihat pada listing ??.

```
1 \begin{displaymath}  
2 .....  
3 \end{displaymath}
```

**Listing 4.1** Notasi Matematika Dalam Paragraf



### 4.3 Font Dalam Notasi Matematika

Ada beberapa perintah pada yang dapat digunakan untuk mengubah jenis font notasi matematis dalam latex. Beberapa perintah tersebut dapat kita lihat pada listing ??.

```
1 1. $\mathrm{x y z}$
2 2. $\mathsf{x y z}$
3 3. $\mathtt{x y z}$
4 4. $\mathit{x y z}$
5 5. $\mathbf{x y z}$
```

**Listing 4.2** Jenis Font Matematis

Hasil output :

xyz  
xyz  
xyz  
*xyz*  
**xyz**

### 4.4 Rumus Dasar

Rumus dasar ini terdiri dari 3 notasi yaitu penjumlahan, pengurangan, dan perkalian. Contoh kode untuk rumus dasar bisa dilihat pada listing ??.

```
1 $$ a+b$$ %Notasi untuk penjumlahan
2 $$ a-b$$ %Notasi untuk pengurangan
3 $$ a \times b$$ %Notasi untuk perkalian
```

**Listing 4.3** Penggunaan Rumus Dasar

Hasil output:

$$a + b$$

$$a - b$$

$$a \times b$$

#### 4.4.1 Rumus Pecahan

Rumus pecahan yang dimaksud adalah notasi per pada pembagian. Contoh kode untuk rumus pecahan bisa dilihat pada listing ??.

```
1 $$ a/b$$ %Notasi per dengan output satu baris
2 $$ \frac {a}{b}$$ %Notasi per dengan output dua baris
```

**Listing 4.4** Penggunaan Rumus Pecahan

Hasil output:

$$a/b$$

$$\frac{a}{b}$$

#### 4.4.2 Rumus Akar

Rumus akar dapat dilihat pada listing ??.

```
1 $$ \sqrt[a]{b} $$ %Notasi akar dengan output b akar a
2 $$ \sqrt{\sqrt{a}} $$ %Notasi akar dengan output a akar di dalam akar
```

**Listing 4.5** Penggunaan Rumus Akar

Hasil output:

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

$$\sqrt{\sqrt{a}}$$

### 4.5 Perumusan Menggunakan Superscripts dan Subscripts

Penulisan *Superscripts* dan *Subscripts* biasanya digunakan untuk membuat sebuah rumus dengan menghasilkan pangkat diatas dan pangkat dibawah pada suatu rumus. Cara penulisan penggunaan ini adalah dengan menggunakan perintah **sp** dan perintah **sb**. Untuk contoh penerapan perintah *Superscripts* dan *Subscripts* dapat kita lihat pada listing ??.

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

**Listing 4.6** Penggunaan Superscripts dan Subscripts

Hasil output :

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

Atau kita juga dapat menggunakan perintah lain seperti pada listing ??.

```
1 \begin{displaymath}
2 f(x) = e^{x_1}
3 \end{displaymath}
4 %Penggunaan tanda " ^ " digunakan sebagai Superscripts
5 %Penggunaan tanda " _ " digunakan sebagai Subscripts
```

**Listing 4.7** Perintah Pada Superscripts dan Subscripts

Hasil output :

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

## 4.6 Perumusan Array dan Matriks

Dalam LaTeX, kita dapat menuliskan rumus sebuah array pada environment **tabular**. Perintah untuk membuat array dan matriks dapat kita lihat pada listing ??.

```

1 \begin{displaymath}
2 \left(
3 \begin{array}{rrr}
4 0 & 55 & 23 \\
5 34 & -83 & 68 \end{array}
6 \right)
7 \end{displaymath}

```

**Listing 4.8** Penulisan Array atau Matriks

Hasil output :

$$\begin{pmatrix} 0 & 55 & 23 \\ 34 & -83 & 68 \end{pmatrix}$$

Ada beberapa hal yang perlu kita ketahui dalam penulisan rumus array atau matriks ini :

- Penulisan array memiliki kesamaan seperti saat membuat format tabel
- Perintah ”**rrr**” berfungsi untuk menentukan posisi dari masing-masing komponen matriks tersebut
- Tanda kurung kurawal ”( )” berfungsi untuk mendefinisikan bagian kurung buka dan kurung tutup pada sebuah matriks

## 4.7 Perumusan Vektor

Dalam LaTeX, perumusan dengan format *vektor* kita dapat menuliskannya dengan perintah seperti pada listing ??.

```

1 \begin{displaymath}
2 \vec{\text{variabel}}
3 \end{displaymath}

```

**Listing 4.9** Penulisan Vektor

Contoh kita akan mengubah variabel  $x$  kedalam satuan vektor. Maka hasil outputnya adalah :

$$\vec{x} = a + b$$

## BAB 5

---

# SISTEMATIKA DOKUMEN

---

### 5.1 Sistematika Dokumen

Dalam format latex memiliki kemampuan untuk membuat suatu susunan yang struktural atau tertata dengan rapi seperti *bab*, *subbab*, dan sebagainya.

**Tabel 5.1** Small Table

Struktur	penulisan
Bagian	part
Bab	chapter
subbab	section
subsubbab	subsection
subsubsubbab	subsubsection
paragraf berjudul	paragraph
anak paragraf berjudul	subparagraph



## BAB 6

---

# ALIGNMENT

---

### 6.1 Alignment

Alignment adalah perataan baris pada LaTeX. terdapat 3 jenis pada perataan tersebut yaitu rata kiri, rata kanan atau rata tengah. pada dokumen LaTeX memiliki perataan yang secara default sudah diatur dan memiliki perataan justified atau biasa disebut rata kanan.

### 6.2 Membuat Tabel

Latex memiliki banyak keunggulan dalam membuat dokumen selain membuat format penulisan dokumen menjadi akurat dan tertata dengan rapi, latex juga mempermudah pengguna dalam penulisan dokumen yakni tidak perlu memperhatikan penulisan karena latex secara otomatis dapat memperbaharuinya. salah satu keunggulan latex yaitu dapat membuat tabel yakni seperti ini

**Tabel 6.1**    LaTeX Table

<b>Bagian I</b>	<b>Bagian II</b>
Cover	judul
Kata pengantar	abstrak
daftar isi	si
kesimpulan	penutup