

PANDUAN CERDAS MENGENAL APLIKA BANK SAMPAH

PANDUAN CERDAS MENGENAL APLIKASI BANK SAMPAH

Berbasis Web serta terdapat UI Android

Arrizal Furqona Gifary dan Bakti Qilan Mufid
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

1	Pendahuluan	1
2	Judul Bagian Kedua	3
3	Notasi Matematika Pada Latex	7
4	Sistematika Dokumen	15
5	Referens Pada Latex	19

DAFTAR ISI

Daftar Gambar	xi
Daftar Tabel	xiii
Foreword	xvii
Kata Pengantar	xix
Acknowledgments	xxi
Acronyms	xxiii
Glossary	xxv
List of Symbols	xxvii
Introduction	xxix
<i>Rolly Maulana Awangga, S.T., M.T.</i>	
1 Pendahuluan	1
1.1 Pengenalan Bank Sampah	1
1.2 Definisi Bank Sampah	2
2 Judul Bagian Kedua	3
	ix

2.1	Coba	3
3	Notasi Matematika Pada Latex	7
3.1	Membuat Rumus dengan LaTeX	7
3.2	Penulisan Notasi Matematika	7
3.3	Jenis Font Dalam Notasi Matematika	8
3.4	Rumus Dasar	8
3.4.1	Rumus Pecahan	8
3.4.2	Rumus Akar	9
3.5	Perumusan Menggunakan Superscripts dan Subscripts	9
3.6	Perumusan Array dan Matriks	10
3.7	Perumusan Vektor	10
3.8	Kombinasi Penggunaan Rumus	11
3.9	Penulisan Tata Letak Rumus	12
3.10	Penulisan Simbol	12
4	Sistematika Dokumen	15
4.1	Sistematika Dokumen	15
4.2	Alignment	16
4.3	Document class	16
4.4	Costum Command	17
4.5	Renewcommand	17
4.6	Penomoran	18
4.7	Daftar Pustaka	18
5	Referens Pada Latex	19
5.1	Membuat Penomoran Referensi	19

DAFTAR GAMBAR

5.1	Ini adalah Contoh Penomoran Referensi	20
5.2	Ini adalah Halaman Google Scholar	20
5.3	Ini adalah Tanda proses awal mengambil reference	20
5.4	Ini adalah Pilihan mengutip	21
5.5	Ini adalah Script BibTex	21
5.6	Ini adalah Direktori pekerjaan	22
5.7	Ini adalah Reference.bib	22
5.8	Ini adalah Proses pemilihan sumber	23
5.9	Ini adalah Compile pdflatex	23
5.10	Ini adalah Compile BibTex	24

DAFTAR TABEL

4.1	Small Table	15
-----	-------------	----

Listings

3.1	Notasi Matematika Dalam Paragraf	7
3.2	Jenis Font Matematis	8
3.3	Penggunaan Rumus Dasar	8
3.4	Penggunaan Rumus Pecahan	9
3.5	Penggunaan Rumus Akar	9
3.6	Penggunaan Superscripts dan Subscripts	9
3.7	Perintah Pada Superscripts dan Subscripts	9
3.8	Penulisan Array atau Matriks	10
3.9	Penulisan Vektor	10
3.10	Contoh Kombinasi Rumus Sigma	11
3.11	Contoh Kombinasi Rumus	11
3.12	Contoh Penulisan Tata Letak Dalam Rumus	12
3.13	Contoh Penulisan Simbol	12
4.1	Contoh Perintah Membuat Rata Kiri	16
4.2	Contoh Perintah Membuat Rata Kanan	16
4.3	Format Ukuran Font	16
4.4	Struktur Awal Document Pada Latex	17
4.5	Perintah Section pada New Command	17

4.6	Perintah Renewcommand	17
4.7	Memberikan Perintah Numbering	18
4.8	Menambahkan kode perintah bullets	18
4.9	Memberikan Perintah Daftar Pustaka	18
5.1	Penggunaan perintah cite untuk reference	21

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

PENDAHULUAN

1.1 Pengenalan Bank Sampah

Bank sampah ialah suatu terobosan yang berfokus pada pengelolaan sampah rumah tangga. Dimana sampah rumah tangga tersebut mampu di kelola dan memiliki timbal balik yang menguntungkan baik bagi rumah tangga yang mengeluarkan sampah, maupun pemerintah atau pengelola sampah. dengan hadirnya terobosan bank sampah ini diharapkan sampah-sampah rumah tangga dapat menguntungkan berbagai pihak.

Berdirinya bank sampah ini terpicu karena keprihatinan dan ketidaknyamanan masyarakat terhadap lingkungannya yang dari hari ke hari makin banyak sampah yang menumpuk, dikarenakan pembersihan yang dilakukan lebih sedikit dengan sampah yang dihasilkan, terutama dalam sampah rumah tangga. Oleh karena itu bank sampah hadir dengan maksud untuk memilah dan memilih(yang masih bisa digunakan, di daur ulang, atau masih bermanfaat) sampah-sampah rumah tangga tersebut.

1.2 Definisi Bank Sampah

Bank sampah adalah suatu tempat yang digunakan untuk mengumpulkan sampah yang sudah di *filter*. Hasil dari pengumpulan tersebut akan disetorkan ke tps (untuk sampah yang tidak dapat digunakan kembali) atau ke tempat pembuatan kerajinan dari barang-barang bekas. Bank sampah dikelola menggunakan sistem seperti perbankan yang dilakukan oleh petugas. Penyetor atau nasabah ialah warga masyarakat yang berada di sekitar lokasi bank yang dimana nasabah akan mendapatkan rekening seperti halnya rekening pada bank.[?]

Menurut peraturan menteri negara lingkungan hidup RI nomor 13 tahun 2012 tentang pedoman pelaksanaan *Reduce*, *Reuse*, dan *Recycle* melalui bank sampah, yang di atur dalam pasa 1 ayat 2 dengan bunyi yaitu: Bank sampah adalah tempat pemilahan dan pengumpulan sampah yang dapat didaur ulang dan/atau diguna ulang yang memiliki nilai ekonomi.[?]

BAB 2

JUDUL BAGIAN KEDUA

2.1 Coba

BAB II LANDASAN TEORI 2.1 Tinjauan Pustaka 2.1.1 Sistem Sistem adalah suatu kelompok dari komponen-komponen yang saling berhubungan dengan satu sama lain dengan tujuan yang sama untuk mencapai tujuan tertentu. Sedangkan Informasi adalah berupa data yang diolah menjadi bentuk lebih berguna dan berarti bagi pemakainya [1]. Sehingga dapat disimpulkan bahwa system informasi adalah suatu komponen yang saling berhubungan yang bertujuan mengumpulkan, mengubah, dan menyebarkan informasi dalam sebuah organisasi. 2.1.2 Pengendalian Aplikasi adalah penggunaan dalam suatu computer, intruksi (instruction) atau pernyataan (statement) yang disusun sedemikian rupa sehingga computer dapat memproses input menjadi output. Aplikasi dapat diartikan juga sebagai program yang dibuat untuk menolong manusia dalam melakukan tugas tertentu . [2] Berdasarkan pernyataan diatas, dapat diambil kesimpulan bahwa aplikasi adalah suatu program computer yang dibuat sedemikian rupa dengan suatu fungsi bagi pengguna jasa aplikasi yang dapat memproses input dan output oleh suatu sasaran yang akan dituju. 2.1.3 Bank Sampah Bank sampah adalah suatu tempat yang digunakan untuk mengumpulkan sampah yang sudah dipilah-pilah. Sampah yang dikumpulkan adalah sampah yang mem-

punyai nilai ekonomis. Cara kerja bank sampah umumnya hampir Sama dengan bank lainnya, ada nasabah, pencatatan pembukuan, dan manajemen pengelolaannya, apabila dalam bank yang biasa kita kenal yang disetorkan nasabah adalah uang akan tetapi dalam bank sampah yang disetorkan adalah sampah yang mempunyai nilai ekonomis. Sedangkan pengelola bank sampah harus orang kreatif dan inovatif serta memiliki jiwa kewirausahaan agar dapat meningkatkan pendapatan masyarakat. Sistem kerja bank sampah pengelolaan sampahnya berbasis rumah tangga, dengan memberi reward kepada yang berhasil memilah dan menyetorkan sejumlah sampah. (Clean (Ulfah, 2016)) [7]

2.1.4 Website Website adalah sebuah situs yang mana sekumpulan halaman yang menampilkan informasi data teks, data gambar diam atau gerak, data animasi, suara, video atau gabungan dari semuanya, baik yang bersifat statis atau dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (hyperlink). [3] Berdasarkan pengertian diatas dapat kita ketahui bahwa web adalah sekumpulan halaman yang di dalamnya terdapat multimedia yang saling berhubungan, dan dapat diakses melalui koneksi jaringan.

2.1.5 Codeigniter CodeIgniter adalah kerangka kerja aplikasi web untuk membangun aplikasi web yang sifatnya open source yang digunakan untuk membangun aplikasi PHP yang dinamis. CodeIgniter memiliki Karakteristik yang sangat fleksibel dan ringan dalam memodifikasi dan mengintegrasikan suatu website. Dalam CodeIgniter dapat juga digunakan pola Model View Controller (MVC) sehingga struktur kode yang di dihasilkan lebih terstruktur dan memiliki standar yang jelas. [4]

2.1.6 PHP PHP adalah singkatan dari Hyper-text Preprocessor yang merupakan bahasa pemrograman yang bersifat open source yang berada didalam server yang diproses di server. PHP salah satu bahasa pemrograman yang memiliki script yang terintegrasi dengan HTML dan berada pada server (server side HTML embedded scripting). [5] PHP bisa berinteraksi dengan database, file dan folder, PHP ini termaksud bahasa cross-platform, yang mana PHP ini bisa berjalan disistem operasi yang berbeda-beda (Windows, Linux, ataupun MAC).[5] Berdasarkan pernyataan diatas dapat disimpulkan bahwa PHP adalah bahasa adalah sebuah bahasa pemrograman yang sangat fleksibel untuk digunakan dalam membuat sebuah website, dan juga PHP dapat dijalankan di bawah sistem operasi LINUX dan Windows.

2.1.7 MySQL (My Structure Query Language) MySQL adalah sebuah server database open source yang terkenal yang digunakan berbagai aplikasi terutama untuk server atau membuat web. MySQL juga adalah sebuah implementasi dari sistem manajemen basis data relasional (RDBMS). Kehandalan suatu sistem basis data (DBMS) dapat diketahui dari cara kerja pengoptimasinya dalam melakukan proses perintah-perintah SQL yang dibuat oleh pengguna maupun program-program aplikasi yang memanfaatkannya. Berdasarkan dari pengertian diatas kesimpulan dari MySQL adalah sekumpulan sistem database yang banyak digunakan untuk pengembangan suatu aplikasi berbasis web dan bersifat open source.

2.1.8 UML (Unified Modeling Language) The Unified Modeling Language (UML) adalah standar de facto untuk analisis perangkat lunak berorientasi objek dan pemodelan desain. [9] UML membantu memodelkan berbagai aspek sistem melalui berbagai diagram yang di dukungnya. Setiap aspek dari suatu sistem disajikan dengan menggunakan jenis diagram UML tertentu dan satu set diagrama disebut seba-

gai model. 2.1.9 Database Database didefinisikan sebagai kumpulan data yang saling berkaitan secara teknis, database juga sebagai tempat media penyimpanan data kita dalam membuat sebuah program yang berisikan tabel, field, record, yang diselimiuti dengan DBMS (Database Management System). [7] Berdasarkan dari uraian di atas dapat disimpulkan bahwa database adalah sekumpulan data yang beris tabel, field, dan record yang disimpan secara sistematis. BAB III ANALISIS DAN PERANCANGAN 3.1 Analisis Analisis merupakan langkah awal untuk pengembangan sebuah aplikasi, karena perancangan dan bahkan pengembangan implementasi aplikasi tidak akan berjalan dengan baik tanpa adanya analisa terhadap aplikasi yang akan digunakan. Analisis juga dapat didefinisikan sebagai penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi masalah-masalah, kesempatan-kesempatan, hambatan-hambatan yang terjadi serta kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan agar mendapat hasil yang maksimal. Analisis yang dilakukan terhadap Aplikasi Bank Sampah Menggunakan Codelgniter ini dibuat menggunakan flowmap dan metode Object Oriented yang memberikan gambaran mengenai proses yang terdapat di dalam aplikasi tersebut. Sistem ini dibangun dengan menggunakan model MVC atau Model View Controller, dimana MVC merupakan model pengembangan dari Object Oriented Programming, dimana setiap baris kodenya dipisahkan menjadi tiga bagian, yaitu ada pada view sebagai form, lalu controller untuk menyimpan fungsi dan class dan model untuk menyimpan database. 3.1.1 Analisis sistem yang sedang berjalan Sistem yang berjalan saat ini terdiri dari satu prosedur yaitu alur pemasukan barang yang sedang berjalan.

Gambar 3.1 Prosedur yang sedang berjalan Keterangan : 1. Masyarakat membuang sampah pada TPS yang sudah tersedia di depan rumahnya masing-masing 2. Masyarakat Melakukan pembayaran ke kepala desa setiap sebulan sekali 3. Kepala desa mengkonfirmasi ke pada petugas kebersihan untuk melakukan pengambilan sampah setiap seminggu 2 kali. 4. Petugas kebersihan akan mengambil sampah setiap 2 kali dalam seminggu. 5. Petugas kebersihan akan mengelola sampah, sampah apa saja yang akan di daur ulang dan sampah apa saja yang akan di bakar.

3.1.2 Analisis Sistem yang akan di bangun Pada analisis sistem yang akan dibangun ini, dibuat beberpa pembaruan dari yang sebelumnya. Pada prosedur ini masyarakat di haruskan mendaftarkan terlebih dahulu untuk menjadi nasabah pada Bank Sampah. Prosedur yang akan dibangun pada Bank Sampah yaitu sebagai berikut :

Gambar 3.2 Prosedur yang akan dibangun Keterangan : 1. Masyarakat harus mendaftarkan diri terlebih dahulu untuk menjadi nasabah pada bank sampah 2. Setelah menjadi nasabah, masyarakat dapat menabung sampah pada bank sampah 3. Jika masyarakat menabung sampah, maka data-data sampah tersebut akan di kelola oleh admin 4. Admin akan mengkonfirmasi kepada petugas untuk melakukan pengambilan sampah pada nasabah yang telah menabung sampah 5. Petugas akan diberi alamat nasabah oleh admin 6. Petugas melakukan pengambilan sampah ke alamat nasabah

3.1.3 Analisis Sistem Login yang akan di bangun Pada analisis sistem yang akan dibangun ini, dibuat beberpa pembaruan dari yang sebelumnya. Pada prosedur ini masyarakat di haruskan mendaftarkan terlebih dahulu untuk menjadi nasabah pada

Bank Sampah. Prosedur login yang akan dibangun pada Bank Sampah yaitu sebagai berikut :

Gambar 3.3 Prosedur Login Keterangan : 1. User akan masuk ke halaman login 2. Jika sudah memiliki akun maka user akan langsung melakukan login dan masuk ke halaman user 3. Jika belum memiliki akun maka user akan melakukan registrasi terlebih dahulu 4. User disini yaitu nasabah atau petugas

3.1.3.1 Kebutuhan Fungsional (Functional Requirements) Kebutuhan fungsional adalah jenis kebutuhan yang berisi proses-proses apa saja yang nantinya dilakukan oleh sistem. Kebutuhan fungsional juga berisi informasi-informasi apa saja yang harus ada dan dihasilkan oleh sistem. Adapun kebutuhan fungsional yang akan dibuat yaitu: 1. Login 2. Kelola data user 3. Kelola data sampah 4. Menabung sampah 5. Pengambilan sampah Setiap proses memiliki representasi masing-masing pada sebuah tabel atau data yang terdapat pada database yang telah dirancang sebelumnya. Dan setiap proses berhubungan langsung dengan entitas atau user.

3.1.3.2 Kebutuhan Non-Fungsional (Non-Functional Requirement) Analisis kebutuhan non fungsional dilakukan untuk mengetahui spesifikasi kebutuhan untuk sistem. Spesifikasi kebutuhan melibatkan analisis perangkat keras/hardware, analisis perangkat lunak/software, analisis pengguna/user. Adapun kebutuhan non fungsional yang akan dibuat adalah sebagai berikut : A. Kebutuhan Perangkat Keras Pembuatan aplikasi ini menggunakan perangkat sebagai berikut : Tabel 3.1 Kebutuhan perangkat keras No. Jenis Keterangan 1 Processor : Intel® core™i3 2 Memory : 4 GB 3 Monitor : LCD 14,1 Inchi 4 Mouse dan keyboard : Standard B. Kebutuhan Perangkat Lunak Spesifikasi perangkat lunak yang digunakan adalah sebagai berikut :

Tabel 3. 2 Kebutuhan perangkat lunak No. Jenis Keterangan 1. Sistem Operasi Windows 10 Pro 64-Bit 2. Server Database Xampp 1.8.1 3. Bahasa Pemrograman PHP dan Android 4. Software Pendukung Visual Studio Code dan Android Studio 5. Browser Google Chrome

C. Analisis Pengguna/User Aplikasi yang akan dibuat ini digunakan ketika ingin menabung sampah dan melakukan penjemputan sampah, adapun User yang dilibatkan antara lain : Nasabah (Masyarakat) dan petugas kebersihan.

3.2 Perancangan 3.2.1 Use Case Diagram Use case diagram mendeskripsikan sebuah interaksi antara satu atau lebih actor dengan sistem informasi yang akan dibuat. Use case diagram digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. 3.2.1.1 Definisi Aktor Pada bagian ini akan dideskripsikan actor-aktor yang terlibat dalam Aplikasi Bank Sampah Tabel 3.3 Definisi Aktor No Aktor Deskripsi 1. Admin - Mengelola data sampah - Mengelola data user - Mengatur jadwal pengambilan sampah 2. Nasabah - Menabung sampah 3. Petugas - Kelola data sampah

Definisi Use Case Use case digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Tabel 3.4 Definisi Use Case

BAB 3

NOTASI MATEMATIKA PADA LATEX

3.1 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*. Terdapat beberapa perintah yang bisa digunakan dalam membuat rumus pada latex. Kita dapat menggunakan perintah *equation*, *displaymath* ataupun menggunakan $. Kita juga dapat menyelipkan rumus didalam suatu kalimat di sebuah paragraf dengan menggunakan perintah $$.$$

3.2 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 3.1.

```
1 \begin{displaymath}  
2 .....  
3 .....  
4 .....  
5 .....  
6 .....  
7 .....  
8 .....  
9 .....  
10 .....  
11 .....  
12 .....  
13 .....  
14 .....  
15 .....  
16 .....  
17 .....  
18 .....  
19 .....  
20 .....  
21 .....  
22 .....  
23 .....  
24 .....  
25 .....  
26 .....  
27 .....  
28 .....  
29 .....  
30 .....  
31 .....  
32 .....  
33 .....  
34 .....  
35 .....  
36 .....  
37 .....  
38 .....  
39 .....  
40 .....  
41 .....  
42 .....  
43 .....  
44 .....  
45 .....  
46 .....  
47 .....  
48 .....  
49 .....  
50 .....  
51 .....  
52 .....  
53 .....  
54 .....  
55 .....  
56 .....  
57 .....  
58 .....  
59 .....  
60 .....  
61 .....  
62 .....  
63 .....  
64 .....  
65 .....  
66 .....  
67 .....  
68 .....  
69 .....  
70 .....  
71 .....  
72 .....  
73 .....  
74 .....  
75 .....  
76 .....  
77 .....  
78 .....  
79 .....  
80 .....  
81 .....  
82 .....  
83 .....  
84 .....  
85 .....  
86 .....  
87 .....  
88 .....  
89 .....  
90 .....  
91 .....  
92 .....  
93 .....  
94 .....  
95 .....  
96 .....  
97 .....  
98 .....  
99 .....  
100 .....
```

```
3 \end{displaymath}
```

Listing 3.1 Notasi Matematika Dalam Paragraf

3.3 Jenis 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 3.2.

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

Listing 3.2 Jenis Font Matematis

Hasil output :

xyz

xyz

xyz

xyz

xyz

3.4 Rumus Dasar

Rumus dasar ini terdiri dari 3 notasi yaitu penjumlahan, pengurangan, dan perkalian. Contoh kode untuk rumus dasar bisa dilihat pada listing 3.3.

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

Listing 3.3 Penggunaan Rumus Dasar

Hasil output:

$$a + b$$

$$a - b$$

$$a \times b$$

3.4.1 Rumus Pecahan

Rumus pecahan yang dimaksud adalah notasi per pada pembagian. Contoh kode untuk rumus pecahan bisa dilihat pada listing 3.4.

```

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

```

Listing 3.4 Penggunaan Rumus Pecahan

Hasil output:

$$a/b$$

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

3.4.2 Rumus Akar

Rumus akar dapat dilihat pada listing 3.5. 3.5.

```

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 3.5 Penggunaan Rumus Akar

Hasil output:

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

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

3.5 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 3.6.

```

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

```

Listing 3.6 Penggunaan Superscripts dan Subscripts

Hasil output :

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

Atau kita juga dapat menggunakan perintah lain seperti pada listing 3.7.

```

1 \begin{displaymath}
2 f(x) = e^{\text{x}_1}
3 \end{displaymath}
4 %Penggunaan tanda " ^ " digunakan sebagai Superscripts
5 %Penggunaan tanda " _ " digunakan sebagai Subscripts

```

Listing 3.7 Perintah Pada Superscripts dan Subscripts

Hasil output :

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

3.6 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 3.8.

```
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 3.8 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

3.7 Perumusan Vektor

Dalam LaTeX, perumusan dengan format *vektor* kita dapat menuliskannya dengan perintah seperti pada listing 3.9.

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

Listing 3.9 Penulisan Vektor

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

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

3.8 Kombinasi Penggunaan Rumus

Pada section ini kita akan mempelajari bagaimana mengkombinasikan sebuah rumus dari penulisan dasar rumus Subscript, Superscript, Akar Pangkat, Pecahan dan sejenisnya. Contoh pertama dapat kita lihat pada listing 3.10.

```

1
2 $$\sum^{\infty}_{n=1} \frac{1}{n}$$
3
4 % \sum = Perintah yang menghasilkan output lambang Sigma
5 % " ^ " = Menempatkan perintah diatas rumus (superscripts)
6 % {infinity} = Perintah yang menghasilkan output lambang Infinity
7 % " _ " = Menempatkan perintah dibawah rumus (subscript)
8 % \frac = Memberikan nilai pecahan pada rumus

```

Listing 3.10 Contoh Kombinasi Rumus Sigma

Hasil output :

$$\sum_{n=1}^{\infty} \frac{1}{n}$$

Setelah melihat salah satu penggunaan kombinasi rumus diatas kita bisa melakukan kombinasi rumus lainnya. Seperti yang akan diperlihatkan pada listing 3.11

```

1 $\sqrt{\frac{a^2}{3b^3+1}}$
2 $\lim_{n \to \infty} \frac{1}{n}=0$
3 $\int^b_a x^2 \, dx$
4 $\lim \limits_{n \to \infty} \frac{1}{n}=0$
5 $\int \limits^b_a x^2 \, dx$
6 $\sum \limits^{\infty}_{n=1} \frac{1}{n}$

```

Listing 3.11 Contoh Kombinasi Rumus

Hasil output :

$$1. \sqrt{\frac{a^2}{3b^3+1}}$$

$$2. \lim_{n \rightarrow \infty} \frac{1}{n} = 0$$

$$3. \int_a^b x^2 dx$$

$$4. \lim_{n \rightarrow \infty} \frac{1}{n} = 0$$

$$5. \int_a^b x^2 dx$$

$$6. \sum_{n=1}^{\infty} \frac{1}{n}$$

3.9 Penulisan Tata Letak Rumus

Pada section ini kita kan mempelajari bagaimana menempatkan penulisan rumus sesuai dengan tata letak yang kita inginkan. Misalkan kita ingin membuat suatu rumus dan hasil dari rumus tersebut ingin kita letakkan di tengah, kiri atau pun kanan. Contoh penerapan tata letak dalam sebuah rumus dapat kita lihat pada listing 3.12.

```

1 
$$\begin{array}{ccc} (x+y)(x-y) & = & x^2 - xy + yx - y^2 \\ \text{\textbackslash\textbackslash} & & \\ (x+y)^2 & = & x^2 + 2xy + y^2 \end{array}$$

2  $\%$ {ccc} = center:center:center
3
4 
$$\begin{array}{lcr} (x+y)(x-y) & = & x^2 - xy + yx - y^2 \\ \text{\textbackslash\textbackslash} & & \\ (x+y)^2 & = & x^2 + 2xy + y^2 \end{array}$$

5  $\%$ {lcr} = left:center:right
6
7 
$$\begin{array}{rcr} (x+y)(x-y) & = & x^2 - xy + yx - y^2 \\ \text{\textbackslash\textbackslash} & & \\ (x+y)^2 & = & x^2 + 2xy + y^2 \end{array}$$

8  $\%$ {rcr} = right:center:left

```

Listing 3.12 Contoh Penulisan Tata Letak Dalam Rumus

Hasil output :

$$\begin{array}{lcl} 1. & \begin{array}{l} (x+y)(x-y) \\ \\ (x+y)^2 \end{array} & \begin{array}{l} = x^2 - xy + yx - y^2 \\ = x^2 - y^2 \\ = x^2 + 2xy + y^2 \end{array} \\ 2. & \begin{array}{l} (x+y)(x-y) \\ \\ (x+y)^2 \end{array} & \begin{array}{l} = x^2 - xy + yx - y^2 \\ = x^2 - y^2 \\ = x^2 + 2xy + y^2 \end{array} \\ 3. & \begin{array}{l} (x+y)(x-y) \\ \\ (x+y)^2 \end{array} & \begin{array}{l} = x^2 - xy + yx - y^2 \\ = x^2 - y^2 \\ = x^2 + 2xy + y^2 \end{array} \end{array}$$

3.10 Penulisan Simbol

Banyak penulisan simbol yang dapat kita implementasikan pada latex. Beberapa contoh perintah penggunaan simbol pada latex dapat kita lihat seperti pada listing 3.13

- 1 π
- 2 ϕ
- 3 ρ
- 4 σ
- 5 ϵ
- 6 δ
- 7 θ
- 8 κ

```

9  $\alpha$
10 $\beta$
11 $\gamma$
12 $\omega$
13 $\zeta$
14 $\eta$
15 $\iota$
16 $\lambda$
17 $\mu$
18 $\nu$
19 $\xi$
20 $\tau$
21 $\upsilon$
22 $\chi$
23 $\psi$

```

Listing 3.13 Contoh Penulisan Simbol

Hasil output :

1. π = Merupakan simbol **pi**
2. ϕ = Merupakan simbol **phi**
3. ρ = Merupakan simbol **rho**
4. σ = Merupakan simbol **sigma**
5. ϵ = Merupakan simbol **epsilon**
6. δ = Merupakan simbol **delta**
7. θ = Merupakan simbol **theta**
8. κ = Merupakan simbol **kappa**
9. α = Merupakan simbol **alpha**
10. β = Merupakan simbol **beta**
11. γ = Merupakan simbol **gamma**
12. ω = Merupakan simbol **omega**
13. ζ = Merupakan simbol **zeta**
14. η = Merupakan simbol **eta**
15. ι = Merupakan simbol **iota**
16. λ = Merupakan simbol **lambda**
17. μ = Merupakan simbol **mu**
18. ν = Merupakan simbol **nu**

- 19. ξ = Merupakan simbol **xi**
- 20. τ = Merupakan simbol **tau**
- 21. v = Merupakan simbol **upsilon**
- 22. χ = Merupakan simbol **chi**
- 23. ψ = Merupakan simbol **psi**

BAB 4

SISTEMATIKA DOKUMEN

4.1 Sistematika Dokumen

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

Tabel 4.1 Small Table

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

4.2 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. Misalkan saat ini kita ingin melakukan pengaturan dokumen rata kiri, maka perintah yang dimasukan yaitu :

```
1 \begin{raggedleft}
2
3 isi dokumen yang diatur dengan rata kanan
4
5 \end{raggedleft}
```

Listing 4.1 Contoh Perintah Membuat Rata Kiri

atau bisa juga kira masukan perintah yang sama untuk rata kiri.

```
1 \begin{raggedleft}
2
3 isi dokumen yang diatur dengan rata kanan
4
5 \end{raggedleft}
```

Listing 4.2 Contoh Perintah Membuat Rata Kanan

4.3 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 aadlah 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.

Perintah format ukuran font pada latex dapat kita lihat seperti pada listing 4.3

```
1 \tiny{teks yang ingin diformat}
2 \scriptsize{teks yang ingin diformat}
3 \footnotesize{teks yang ingin diformat}
4 \small{teks yang ingin diformat}
5 \normal{teks yang ingin diformat}
6 \large{teks yang ingin diformat}
7 \larger{teks yang ingin diformat}
8 \largest{teks yang ingin diformat}
9 \huge{teks yang ingin diformat}
10 \huger{teks yang ingin diformat}
```

Listing 4.3 Format Ukuran Font

Dalam memberikan penulisan judul pada format latex biasanya di letakkan pada awal document, untuk cara penulisan nya dapat dilakukan seperti pada listing ?? dibawah ini :

```

1 \documentclass{ A4 Papper / Ukuran yang di inginkan }
2 \begin{document}
3 \begin{Title} %Judul Dokumen
4 \author{Nama Penulis}
5 \date{Tanggal Pembuatan}
6 \maketitle{}
7 \end{document}

```

Listing 4.4 Struktur Awal Document Pada Latex

4.4 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. Pada Pengaturan huruf lateks dibuat dengan menggunakan tag atau perintah khusus yang menyediakan beberapa cara untuk memformat dokumen Anda. Kadang-kadang perintah standar tidak cukup untuk memenuhi beberapa kebutuhan spesifik.

4.5 Renewcommand

Perintah command merupakan perintah dasar dalam pembentukan format sebuah dokumen yang ada pada latex. Lalu apakah fungsi dari perintah *renewcommand*? Bagaimana mengubah perintah *section* pada command dalam suatu class? Atau bagaimana mengubah command latex pada bagian intinya daripada hanya dari command *section*-nya saja?

Perintah *section* tidak terdefinisi pada latex kernel namun dia terdefinisi oleh class file yang saling berhubungan. Contoh pengimplementasiannya dapat kita lihat pada dokumentasi listing 4.5 :

```

1 \newcommand\section{\@startsection {section}{1}{\z@}%
2 {-3.5ex \@plus -1ex \@minus -.2ex}%
3 {2.3ex \@plus .2ex}% Format Yang Ditandai
4 {\normalfont\Large\bfseries}}

```

Listing 4.5 Perintah Section pada New Command

Perintah *section* diatas menampilkan heading pada dokumen yang dibuat. Pada keterangan **Format Yang Ditandai** seperti pada listing diatas, dapat di artikan bahwa format heading memiliki space bernilai positive didalamnya, kita bisa mengubah format tersebut menjadi negative dengan perintah *renewcommand* seperti pada listing 4.6

```

1 \renewcommand\section{\@startsection {section}{1}{\z@}%
2 {-3.5ex \@plus -1ex \@minus -.2ex}%
3 {-1em}%
4 {\normalfont\Large\bfseries}}

```

Listing 4.6 Perintah Renewcommand

Preamble diatas memberikan efek yang berbeda pada dokumen. Package *titlesec* memberikan hasil yang lebih deklaratif untuk membuat perubahan yang sama, namun ia tetap harus mengasumsikan definisi asli kurang lebih seperti pada class dalam dokumen yang dibuat. Jadi pada dasarnya, perintah *renewcommand* dapat mengubah format suatu dokumen berdasarkan class file dari format *command* yang sudah ditentukan sebelumnya.

4.6 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 4.8 adalah perintah untuk memasukan listing pada penomoran .

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

Listing 4.7 Memberikan Perintah Numbering

Sedangkan *Bullets* atau poin adalah perintah yang digunakan untuk membuat daftar berurut dengan penomoran berupa symbol atau poin (bulleted list) 4.8.

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

Listing 4.8 Menambahkan kode perintah bullets

4.7 Daftar Pustaka

Untuk menampilkan daftar pustaka pada akhir sebuah dokumen LateX menggunakan format perintah sebagai berikut :

```
1 \begin{thebibliography}{99}
2 \bibitem{label untuk referensi}{keterangan pustaka yang digunakan}
3 .....
4 .....
5 \end{thebibliography}
```

Listing 4.9 Memberikan Perintah Daftar Pustaka

BAB 5

REFERENS PADA LATEX

5.1 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 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 5.1.

Bagaimana cara membuatnya di Latex? berikut cara membuatnya:

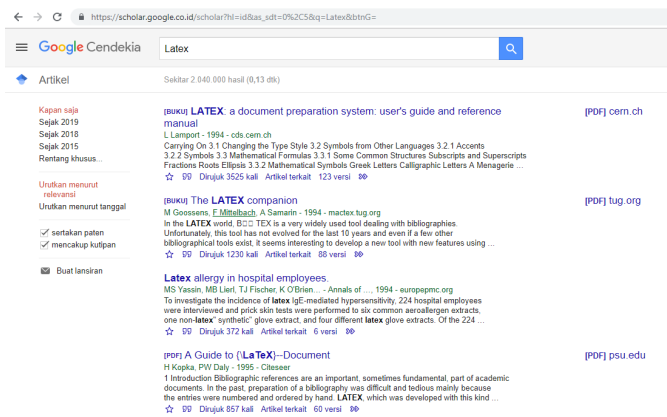
1. Cari materi yang akan dikutip melalui Google Scholar seperti pada gambar 5.2 ,
2. Setelah selesai mengutip jangan lupa untuk mengambil script bibtexnya dengan cara klik pada tanda kutip seperti pada gambar 5.3,

7.1 ASCII

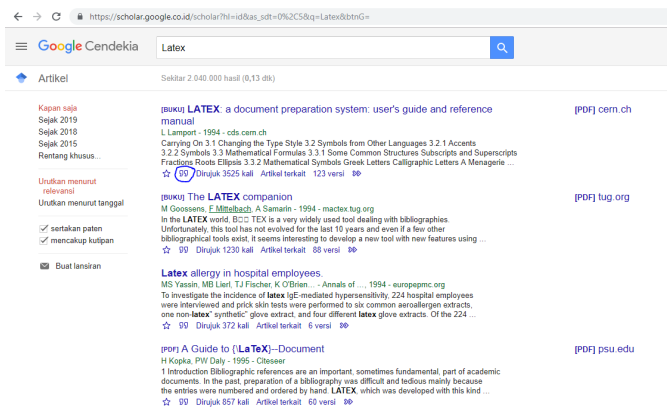
7.1.1 Definisi ASCII

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 5.1 Ini adalah Contoh Penomoran Referensi



Gambar 5.2 Ini adalah Halaman Google Scholar

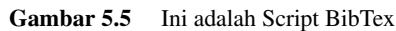


Gambar 5.3 Ini adalah Tanda proses awal mengambil reference

Gambar 5.4 Ini adalah Pilihan mengutip

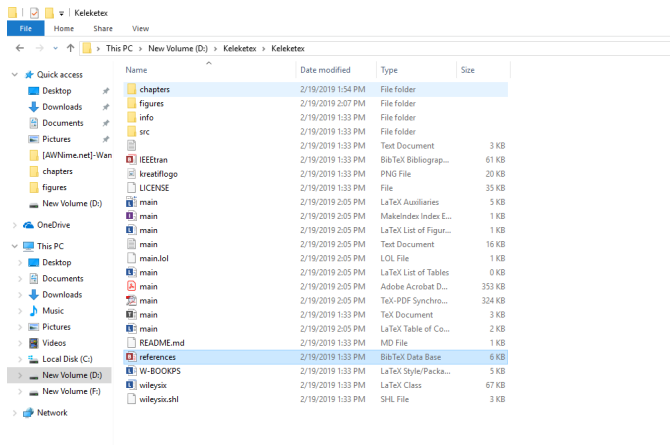


Gambar 5.5 Ini adalah Script BibTex

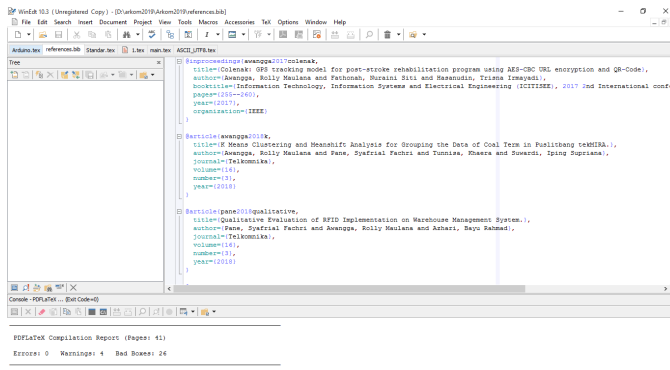


- ```
1 \cite{armiati2018national}
2 'di dalam kurung kurawal diisi sesuai sumber'
```

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 5.9 untuk pdflatex dan 5.10 untuk BibTex.



**Gambar 5.6** Ini adalah Direktori pekerjaan



**Gambar 5.7** Ini adalah Reference.bib



