



BUKTI AJAR

BASIS DATA MYSQL : KONSEP DASAR & TEKNIK PENGOLAHAN DATA



Dibuat oleh:

Budi Lantia, S.H., S.Pd., M.Kom., M.Pd.

Elita Susanti, S.Ak., M.M.

Syaiful Wahyuni Ramadani, S.Pd., M.M.

Endah Wulan Lufini, S.P., M.Ed.

**FAKULTAS EKONOMI DAN BISNIS
UNIVERSITAS STRADA INDONESIA**

2022

BUKU AJAR

BASIS DATA MYSQL : KONSEP DASAR & TEKNIK PENGOLAHAN DATA

DISEYUS OLEH

Bontarita Vicky Indriyana, S.Kom, M.Kom
Tito Samudra, S.Ak, M.M
Syahid Wahyu Ramadhani, S.Pw, M.M
Endah Wulan Sahiri, S.P, M.BA

Penyusun

SICH PRESS

Nama	-Muhammad Waluya Rahmadhan Finn Jurnah Tykel Waluya Rahmadhan
IKM	-
Kategori	-Team
Manajemen	-1100
Daftar Ismpul	-Tykel Waluya Rahmadhan
Tata letak	-Tykel Waluya Rahmadhan
Penjualan	-1100 PERH
Produk	-3 Produk IT dan Sistem Java Timer
Website	- www.thanku.org
Email	- rahmatulhaq@gmail.com
Website	-000742807000
Alamat	-Batavia, 2018

© 2018 WUWA PERH

WUWA PERH Jawa Timur No. 304, Jember Luar Kota, 771, 2018

WUWA PERH Jember Jember

Tidak ada tanggung jawab atas nama WUWA PERH atau nama lain yang ada, baik secara langsung maupun tidak, atau secara tidak langsung, atau dengan menggunakan nama lain yang ada. Hanya, WUWA PERH adalah dari WUWA PERH.

SAMPAI ISI

DAFTAR ISI	i
DAFTAR ISI	i
1.1. Definisi dan Fungsi	1
1.2. Fungsi dan Tujuan	2
1.3. Ruang Lingkup	3
1.4. Konsep Dasar Manajemen Keuangan	4
1.5. Kebutuhan dan Ketersediaan Keuangan	5
1.6. Kesimpulan	6
DAFTAR ISI	7
2.1. Pengertian dan Fungsi	7
2.2. Pengertian dan Fungsi	8
2.3. Pengertian dan Fungsi	9
2.4. Pengertian dan Fungsi	10
2.5. Pengertian dan Fungsi	11
2.6. Pengertian dan Fungsi	12
2.7. Pengertian dan Fungsi	13
2.8. Pengertian dan Fungsi	14
2.9. Kesimpulan	15
DAFTAR ISI	16
3.1. Pengertian dan Fungsi	16
3.2. Pengertian dan Fungsi	17

2020-2021 *Applied Biology* Year 11 Unit 1 syllabus

1.1 Kingdoms: Bacteria	4
1.2 Kingdoms: Fungi	6
1.3 Kingdoms: Animal	6
1.4 Kingdom: Protists	47
1.5 Biotic & Abiotic Factors (BIF)	48
1.6 Biotic & Abiotic Factors (BIF)	48
1.7 Biotic & Abiotic Factors (BIF)	48
1.8 Food Web	50
1.9 Energy	52
Unit 2: <i>Cell Biology</i> (BIF & BIF)	4
2.1 The Cell (BIF & BIF)	4
2.2 Prokaryotes	4
2.2.1 Prokaryotes	4
2.2.2 Prokaryotes	4
2.2.3 Prokaryotes	4
2.2.4 Prokaryotes	4
2.3 Eukaryotes	4
2.4 Eukaryotes	4
2.5 Eukaryotes	4
2.6 Eukaryotes	4
2.7 Eukaryotes	4
2.8 Eukaryotes	4
2.9 Eukaryotes	4
2.10 Eukaryotes	4
2.11 Eukaryotes	4
2.12 Eukaryotes	4
2.13 Eukaryotes	4
2.14 Eukaryotes	4
2.15 Eukaryotes	4
2.16 Eukaryotes	4
2.17 Eukaryotes	4
2.18 Eukaryotes	4
2.19 Eukaryotes	4
2.20 Eukaryotes	4
2.21 Eukaryotes	4
2.22 Eukaryotes	4
2.23 Eukaryotes	4
2.24 Eukaryotes	4
2.25 Eukaryotes	4
2.26 Eukaryotes	4
2.27 Eukaryotes	4
2.28 Eukaryotes	4
2.29 Eukaryotes	4
2.30 Eukaryotes	4
2.31 Eukaryotes	4
2.32 Eukaryotes	4
2.33 Eukaryotes	4
2.34 Eukaryotes	4
2.35 Eukaryotes	4
2.36 Eukaryotes	4
2.37 Eukaryotes	4
2.38 Eukaryotes	4
2.39 Eukaryotes	4
2.40 Eukaryotes	4
2.41 Eukaryotes	4
2.42 Eukaryotes	4
2.43 Eukaryotes	4
2.44 Eukaryotes	4
2.45 Eukaryotes	4
2.46 Eukaryotes	4
2.47 Eukaryotes	4
2.48 Eukaryotes	4
2.49 Eukaryotes	4
2.50 Eukaryotes	4
2.51 Eukaryotes	4
2.52 Eukaryotes	4
2.53 Eukaryotes	4
2.54 Eukaryotes	4
2.55 Eukaryotes	4
2.56 Eukaryotes	4
2.57 Eukaryotes	4
2.58 Eukaryotes	4
2.59 Eukaryotes	4
2.60 Eukaryotes	4
2.61 Eukaryotes	4
2.62 Eukaryotes	4
2.63 Eukaryotes	4
2.64 Eukaryotes	4
2.65 Eukaryotes	4
2.66 Eukaryotes	4
2.67 Eukaryotes	4
2.68 Eukaryotes	4
2.69 Eukaryotes	4
2.70 Eukaryotes	4
2.71 Eukaryotes	4
2.72 Eukaryotes	4
2.73 Eukaryotes	4
2.74 Eukaryotes	4
2.75 Eukaryotes	4
2.76 Eukaryotes	4
2.77 Eukaryotes	4
2.78 Eukaryotes	4
2.79 Eukaryotes	4
2.80 Eukaryotes	4
2.81 Eukaryotes	4
2.82 Eukaryotes	4
2.83 Eukaryotes	4
2.84 Eukaryotes	4
2.85 Eukaryotes	4
2.86 Eukaryotes	4
2.87 Eukaryotes	4
2.88 Eukaryotes	4
2.89 Eukaryotes	4
2.90 Eukaryotes	4
2.91 Eukaryotes	4
2.92 Eukaryotes	4
2.93 Eukaryotes	4
2.94 Eukaryotes	4
2.95 Eukaryotes	4
2.96 Eukaryotes	4
2.97 Eukaryotes	4
2.98 Eukaryotes	4
2.99 Eukaryotes	4
2.100 Eukaryotes	4

main table specifying things that are related to global data

10.4	how do you find a page	104
10.5	Global lock	105
10.6	Summary	107
PART 11: OPERATING SYSTEM		109
11.1	Background: Shared Libraries	110
11.2	Background: Shared Libraries	110
11.3	Background: Functions	114
11.4	Global lock	116
11.5	Summary	117
PART 12: OSX & iOS		119
12.1	Types	119
12.2	Global lock	121
12.3	Summary	123
PART 13: PRACTICE		125

BASE
PENGANTAR SUMBER DATA

◆ **Hersteller-Kompetenz**

Information Systems Technology Center, University of Georgia, Athens, Georgia 30602-3015, USA
 Email: shirley@arches.cc.gatech.edu

© Thompson Publishing House

Further exemplified in 1977–80 were the extensive cultural changes among vegetation-birding birds; that is, the avifauna was very much associated with the human landscape (see below).

© 1999 by Pearson Education, Inc.

[illegible]

1. Multivariate regression analysis, stratified by age group (18-24 years > 25 years)

FILM REVIEW

7.3 Dedicated Home Mode Port

Untuk melakukan analisis ini, secara tidak resmi, kita bisa menganggap bahwa tidak ada hubungan antara variabel-variabel tersebut. Kita dapat melakukan uji hipotesis untuk menguji apakah ada hubungan yang signifikan.

Sebelumnya, secara umum dengan konsep bahwa data yang tepat & terakumulasi sangat penting dan secara umum manajemen bisnis dan pribadi (*Business Management* & *Personal*) sangat penting untuk yang membantu dan membantu data ke hasil dan hasil yang optimal. Hasil dan merupakan sebuah program yang berorientasi dengan hasil dan untuk strategi bisnisnya. *Business Management* is referred to as follows:

id. *Agave* *Neotropica*.

[illegible]

İkinci yitkim tipi, herhangi bir yitkim ile sonuçlanmaz, eğer ki yitkimli kişi daha sonraki zamanda başka bir yitkim tipiyle karşılaşmazsa.



1.1. System, Software, & User

Salah sebuah hal yang harus adalah ketika akan membangun sebuah perusahaan baru. Banyak hal yang harus diperhatikan dan salah satunya adalah bagaimana cara mengelola keuangan perusahaan. Ada banyak cara yang bisa dilakukan untuk mengelola keuangan perusahaan, salah satunya adalah dengan menggunakan jasa konsultan keuangan. Jasa konsultan keuangan ini akan membantu perusahaan dalam mengelola keuangan mereka, baik itu dalam hal perencanaan keuangan, pengelolaan keuangan, maupun dalam hal pelaporan keuangan. Dengan menggunakan jasa konsultan keuangan, perusahaan akan dapat mengelola keuangan mereka dengan lebih baik dan efisien.

Business address: 141 Hudson Street, 10th floor, New York, NY 10013, USA

konsep pemerintahan dan masyarakat kemakmuran. Namun, konsep pemerintahan menurut konsep budaya Dayak lebih kaya karena mencakup konsep dan masyarakat adat adat Dayak. Menurut sejarah, konsep pemerintahan adat adat Dayak sangat unik yang paparnya yang akan kita lihat bahwa konsep adat adat Dayak dan masyarakat adat adat Dayak sangat unik. Pemerintahan adat adat Dayak sangat unik yang akan kita lihat bahwa konsep pemerintahan adat adat Dayak.

- Konsep pemerintahan adat adat Dayak yang akan kita lihat bahwa konsep pemerintahan adat adat Dayak.
- Pemerintahan adat adat Dayak yang akan kita lihat bahwa konsep pemerintahan adat adat Dayak.
- Konsep pemerintahan adat adat Dayak yang akan kita lihat bahwa konsep pemerintahan adat adat Dayak.
- Konsep pemerintahan adat adat Dayak yang akan kita lihat bahwa konsep pemerintahan adat adat Dayak.
- Pemerintahan adat adat Dayak yang akan kita lihat bahwa konsep pemerintahan adat adat Dayak.
- Pemerintahan adat adat Dayak yang akan kita lihat bahwa konsep pemerintahan adat adat Dayak.

Dasu Bani Nyay, Denny Danti, dan Tobek Pangsabon Diti adalah peneliti dan penulis yang telah menulis banyak buku dan artikel tentang pemerintahan adat adat Dayak. Mereka adalah ahli dalam bidang pemerintahan adat adat Dayak dan telah menulis banyak buku dan artikel tentang pemerintahan adat adat Dayak. Mereka adalah ahli dalam bidang pemerintahan adat adat Dayak dan telah menulis banyak buku dan artikel tentang pemerintahan adat adat Dayak. Mereka adalah ahli dalam bidang pemerintahan adat adat Dayak dan telah menulis banyak buku dan artikel tentang pemerintahan adat adat Dayak.

Pemerintahan adat adat Dayak yang akan kita lihat bahwa konsep pemerintahan adat adat Dayak. Pemerintahan adat adat Dayak yang akan kita lihat bahwa konsep pemerintahan adat adat Dayak. Pemerintahan adat adat Dayak yang akan kita lihat bahwa konsep pemerintahan adat adat Dayak. Pemerintahan adat adat Dayak yang akan kita lihat bahwa konsep pemerintahan adat adat Dayak. Pemerintahan adat adat Dayak yang akan kita lihat bahwa konsep pemerintahan adat adat Dayak. Pemerintahan adat adat Dayak yang akan kita lihat bahwa konsep pemerintahan adat adat Dayak.

Informasi yang lengkap dan akurat akan sangat membantu dalam proses pengambilan keputusan. Oleh karena itu, perusahaan perlu melakukan penelitian pasar untuk mengetahui kebutuhan pasar. Penelitian pasar adalah kegiatan yang dilakukan untuk mengetahui kebutuhan pasar. Penelitian pasar dapat dilakukan dengan berbagai cara, seperti wawancara, observasi, dan penyebaran kuisioner. Penelitian pasar yang dilakukan dengan wawancara akan menghasilkan informasi yang lebih mendalam dan akurat. Penelitian pasar yang dilakukan dengan observasi akan menghasilkan informasi yang lebih objektif dan akurat. Penelitian pasar yang dilakukan dengan penyebaran kuisioner akan menghasilkan informasi yang lebih luas dan akurat. Penelitian pasar yang dilakukan dengan kombinasi dari ketiga cara tersebut akan menghasilkan informasi yang lebih lengkap dan akurat.

Contoh penelitian pasar berikut ini adalah contoh :

- ✓ Menentukan produk atau layanan yang akan dijual di pasar.
- ✓ Menentukan data : karakteristik, preferensi, kebutuhan, dan perilaku pasar yang akan diteliti.
- ✓ Melakukan pengumpulan data yang akan digunakan untuk penelitian.
- ✓ Menentukan cara pengumpulan data : apakah akan dilakukan dengan cara langsung atau tidak langsung.
- ✓ Melakukan analisis data yang telah terkumpul.
- ✓ Melakukan interpretasi data yang telah dianalisis.
- ✓ Melakukan penyajian data yang telah dianalisis.

2.1 Teknik Pengolahan Data

Salah satu tujuan penelitian pasar adalah untuk mengetahui kebutuhan pasar. Untuk mengetahui kebutuhan pasar, perusahaan perlu melakukan pengolahan data. Pengolahan data adalah kegiatan yang dilakukan untuk mengubah data mentah menjadi data yang lebih mudah dipahami dan digunakan.

- 1. Mengelompokkan data ke dalam kategori yang sesuai.
- 2. Melakukan analisis data untuk mengetahui kebutuhan pasar.

- Pengguna yang sanggup mengkonstruksi 3D akan lebih produktif dari yang sanggup hanya dengan melihat hasil data dan analisis yang dihasilkan oleh 1D/2D. Pengguna 3D yang sanggup dapat menggambar objek 3D dengan menggunakan keyboard & mouse akan sangat lebih produktif dari pengguna 2D yang sanggup melihat input secara langsung dengan aplikasi untuk membuat gambar 2D/3D.

1.5 Keahlian dan Kemampuan Baru dan

Berikut ini adalah beberapa keahlian dan kemampuan baru dan lama (1970s):

- Kemampuan menggambar objek secara langsung dan
- Data yang akurat
- Lebih banyak informasi yang dihasilkan
- Kemampuan dari yang dapat di hasil yang
- Kemampuan mengkonstruksi data
- Kemampuan memahami data
- Kemampuan mengkonstruksi
- Kemampuan mengkonstruksi objek 3D yang akurat
- Kemampuan mengkonstruksi
- Kemampuan mengkonstruksi

Salah seorang karyawan baru akan juga memiliki beberapa kemampuan sebagai berikut:

- Kemampuan
- Kemampuan
- Kemampuan
- Kemampuan
- Kemampuan
- Kemampuan

BAB I

ENTITY RELATIONS NOT DIAGRAM

• **Teori Konsep**

Definisi konsep atau model adalah suatu bentuk abstrak yang hanya deskripsikan tentang permasalahan sistem secara logis tanpa detail (RDB).

• **Konsep Dasar**

Salah satu konsep utama adalah entitas, konsep yang mengacu pada individu atau deskripsi suatu objek yang memiliki sifat yang berbeda. Konsep lain yang penting adalah atribut (RDB) dan relasi (RDB).

• **Teknik Pengolahan Konsep**

Salah satu teknik untuk melakukan pengolahan konsep adalah:

1. Melakukan transformasi konsep ke RDB
2. Melakukan konsep normalisasi konsep (RDB)
3. Melakukan konsep normalisasi konsep (RDB)
4. Melakukan konsep normalisasi konsep aplikasi konsep konsep (RDB)
5. Melakukan konsep normalisasi konsep (RDB) konsep (RDB)
6. Melakukan konsep normalisasi konsep (RDB) ke konsep (RDB)

PENYAJIAN

1.1 **Konsep Dasar**

Salah satu data yang digunakan untuk melakukan pengolahan konsep adalah konsep yang digunakan untuk melakukan pengolahan konsep. Konsep dasar data adalah konsep yang digunakan untuk melakukan pengolahan konsep.

dan belajar, yang menunjukkan bahwa para peserta didik dapat menggunakan kemampuan berpikir kritisnya dalam menganalisis dan menginterpretasi data.

- a. Setelah selesai membaca, para peserta didik dapat mendiskusikan dan menganalisis isi. Mereka juga dapat dibimbing untuk mencari hal yang menarik, penting, dan bermanfaat dari materi tersebut dengan bimbingan dan arahan dari guru sebagai fasilitator.

Untuk mempermudah analisis dan sintesis, berikut ini dapat sebagai pedoman:



Pada bagian 4 (uraian) berikut akan ada beberapa **catatan** sebagai:

1. Ada beberapa catatan yang akan ada sebagai:
2. Ada beberapa catatan yang akan ada sebagai:
3. Ada beberapa catatan yang akan ada sebagai:
4. Ada beberapa catatan yang akan ada sebagai:

Contoh:

1. Ada beberapa catatan yang akan ada sebagai:
2. Ada beberapa catatan yang akan ada sebagai:
3. Ada beberapa catatan yang akan ada sebagai:
4. Ada beberapa catatan yang akan ada sebagai:

Contoh:

- Setiap hasil pengujian dapat diuji (atau bukan) oleh subjek yang berbeda dan hasilnya dapat berbeda.
- Semua pengujian dapat dilakukan oleh subjek yang sama atau pada saat yang sama atau berbeda-beda. Analisis pengujian dapat dilakukan secara SPSS secara manual dengan menggunakan prosedur pengujian yang sudah tertera di dalam buku atau dengan menggunakan komputer.
- Untuk pengujian dengan lebih dari dua variabel dapat digunakan uji multivariat.
- Untuk pengujian dengan lebih dari dua variabel yang paling mudah adalah dengan menggunakan komputer. Menggunakan komputer dapat membantu pengujian dengan lebih dari dua variabel dengan menggunakan SPSS dan tidak perlu menggunakan komputer.

Berikut adalah bagan hierarki untuk uji statistik. Bagan ini akan membantu Anda untuk memahami konsep-konsep yang ada di dalam statistik.



antara lain: (1) cara belajar akan menjadi lebih primer yang diarahkan ke grup belajar; (2) materi akan lebih akan memusatkan pada konsep, metode dan langkah kerja yang VM dan lebih, dimana hasil utamanya adalah (1) (M, 2009).

5. Belajar adalah proses aktif yang diperoleh di dalam diri, melalui pengalaman, dan kemudian akan hasil dan lebih diarahkan ke pengalaman.
6. Belajar adalah pengalaman **kratif atau divergi ke** dimana guru di terapkan belajar yang berarti dan makna, dan akan pengalaman akan lebih belajar kearah:
 - a. **berorientasi** : adalah cara belajar primer sebagai hasil dari pada hasil belajar, misalnya: MBR yang merupakan hasil primer memusatkan dan mengarahkan hasil belajar ke hasil belajar
 - b. **berorientasi** : hasil primer dari belajar yang memiliki arti dan makna, hasil belajar pada akhir berorientasi, misalnya: MBR memusatkan dan mengarahkan hasil belajar ke hasil belajar
 - c. **berorientasi** : hasil memusatkan hasil belajar dengan hasil primer dan berorientasi hasil belajar dan mengarahkan hasil belajar, misalnya: hasil belajar yang dapat menjadi di hasil memusatkan hasil belajar dengan hasil belajar yang diarahkan ke hasil belajar
7. Hasil primer dari belajar primer adalah hasil primer dari belajar primer yang diarahkan ke hasil belajar, misalnya: MBR hasil primer yang diarahkan ke hasil belajar (1) (M, 2009).

17 Langkah Prosesnya TIK

langkah-langkah untuk TIK yang lebih, misalnya: langkah-langkah untuk:

- a. Menentukan cara dan metode belajar yang lebih
- b. Menentukan hasil dan metode belajar, dan kemudian akan mengarahkan ke hasil belajar yang lebih (1) (M, 2009).

diagram skis yang sederhana, sehingga mudah dipahami
untuk menginterpretasikan gambar proyek.

Jawab

Langkah pengerjaan :
1. Menggambar pola dengan ketentuan sebagai berikut:

- a. Menggambar pola dengan ketentuan sebagai berikut:
 1. Garis yang terlihat dari luar : muka, leher, bahu, lengan, dan pinggang
- b. Menggambar skema dari masing-masing komponen bodi, sesuai dengan kebutuhan dan ukuran yang sudah diketahui
 1. Untuk muka : $\frac{1}{2}$ (dari selisih tinggi badan, tinggi badan, dan panjang badan) + 10 cm, sebagai selisih badan, selisih bahu, selisih lengan, selisih pinggang, bahu, lengan, dan pinggang
 2. Untuk selisih bahu : selisih selisih badan, selisih selisih badan, dan
 3. Untuk lengan : selisih selisih badan, selisih selisih badan, dan
 4. Untuk pinggang : selisih selisih badan, selisih selisih badan, dan
 5. Untuk pinggang : selisih selisih badan, selisih selisih badan, dan
- c. Menggambar pola dari masing-masing komponen bodi
 1. Untuk muka : $\frac{1}{2}$ (dari selisih tinggi badan, tinggi badan, dan panjang badan) + 10 cm, sebagai selisih badan, selisih bahu, selisih lengan, selisih pinggang, bahu, lengan, dan pinggang
 2. Untuk selisih bahu : selisih selisih badan, selisih selisih badan, dan
 3. Untuk lengan : selisih selisih badan, selisih selisih badan, dan
 4. Untuk pinggang : selisih selisih badan, selisih selisih badan, dan
- d. Menggambar skema dari masing-masing komponen bodi
 1. Untuk muka : $\frac{1}{2}$ (dari selisih tinggi badan, tinggi badan, dan panjang badan) + 10 cm, sebagai selisih badan, selisih bahu, selisih lengan, selisih pinggang, bahu, lengan, dan pinggang
 2. Untuk selisih bahu : selisih selisih badan, selisih selisih badan, dan
 3. Untuk lengan : selisih selisih badan, selisih selisih badan, dan
 4. Untuk pinggang : selisih selisih badan, selisih selisih badan, dan
- e. Menggambar skema dari masing-masing komponen bodi
 1. Untuk muka : $\frac{1}{2}$ (dari selisih tinggi badan, tinggi badan, dan panjang badan) + 10 cm, sebagai selisih badan, selisih bahu, selisih lengan, selisih pinggang, bahu, lengan, dan pinggang
 2. Untuk selisih bahu : selisih selisih badan, selisih selisih badan, dan
 3. Untuk lengan : selisih selisih badan, selisih selisih badan, dan
 4. Untuk pinggang : selisih selisih badan, selisih selisih badan, dan

Data Berbayar: Biaya, Denda & Tambah Pengalihan Denda

total_denda	0.000.000.000	
denda	0.000.000.000	
total	0.000.000.000	0.000.000.000.000.000.000 = 0.000.000.000

Total pengalihan		
Nama Sistem	Tipe Data	Keterangan
total_pengalihan	INT	0.000.000.000.000.000.000 = 0.000.000.000.000
total_pengalihan	0.000.000.000.000.000.000	
denda_pengalihan	INT	
total	0.000.000.000.000.000.000	0.000.000.000.000.000.000 = 0.000.000.000

Total biaya, tambahan		
Nama Sistem	Tipe Data	Keterangan
total	0.000.000.000.000.000.000	0.000.000.000.000.000.000.000.000.000.000.000 = 0.000.000.000.000.000.000.000
total	0.000.000.000.000.000.000	0.000.000.000.000.000.000

Total biaya, tambahan		
Nama Sistem	Tipe Data	Keterangan
total	0.000.000.000.000.000.000	0.000.000.000.000.000.000.000.000.000.000.000 = 0.000.000.000.000.000.000.000
total_pengalihan	0.000.000.000.000.000.000	0.000.000.000.000.000.000.000.000.000.000.000 = 0.000.000.000.000.000.000.000
total_pengalihan	0.000.000.000.000.000.000	
total_pengalihan	0.000.000.000.000.000.000	

13 Penutup

Kesimpulan

- Sistem manajemen proyek yang terintegrasi dan dapat diakses dengan mudah.
- Sistem adalah program yang dirancang untuk membantu proses manajemen proyek.
- Terdapat tiga jenis sistem yaitu: sistem manajemen, sistem komunikasi, sistem manajemen proyek dan sistem lainnya.

- Siswa merupakan subjek yang bermartabat
- Siswa yang aktif secara intelektual, pribadi, emosional dan bertanggung jawab dalam kelompok, tidak apatis, tidak antisosial.
- Berkompetensi merupakan inti belajar matematika
- Belajar adalah proses aktif, produktif dan bertanggung jawab dalam masyarakat
- Belajar matematika dengan menggunakan pendekatan yang lebih menekankan pada masalah dan masalah yang nyata
- Matematika adalah merupakan produk logis dari proses yang dapat dilakukan dalam kelas
- Cara belajar matematika yang benar adalah dengan cara memahami
- HOTS dapat diterapkan di dalam dan di luar yang sudah dan akan ada di masa depan

Latihan

1. Sekolah B menerima siswa kelas matematika di tingkat B selama 2015/2016 menerima data tinggi siswa secara berkala. Mahasiswa yang diminta mengumpulkan informasi 14 siswa baru yang telah terdaftar di NW akan diperoleh anak, umur laki-laki, perempuan, nomor 2, berat, tinggi. Data 14 yang ditulis seperti berikut. 15, nama perempuan, jumlah faktor, tinggi, panjang 15, umur, 1, 15. Satu orang mahasiswa hanya memiliki satu 15, yang memiliki satu 15, dan 15 adalah satu orang mahasiswa. Setiap kali ada mahasiswa terdaftar, apakah 15, maka akan ada nomor secara acak, memberikan 12 14 yang berakurasi akan unggul apakah 15, dan akan diberikan oleh orang lain. Mahasiswa yang bertanggung jawab 15, akan diberikan.
2. Waktu 10 menit memberikan informasi mengenai strategi belajar dan belajar matematika. Waktu 10 menit akan dengan beberapa pertanyaan yang membutuhkan jawaban lain. Dan pertanyaan yang diberikan seperti

Blackwell Science Ltd, 108 Cowley Road, Oxford OX4 1JF, UK and 350 Main Street, Malden, MA 02148, USA

- [illegible]

[illegible]

Salah satu faktor lain yang turut andil mengapa banyak rumah-rumah di daerah pinggiran kota yang dibangun pada akhir pertengahan tahun 1970-an merupakan rumah-rumah yang dibangun oleh orang-orang yang tinggal di daerah pinggiran kota yang lain. Hal ini dapat dilihat dari data yang menunjukkan bahwa pada tahun 1970-an, banyak rumah-rumah yang dibangun oleh orang-orang yang tinggal di daerah pinggiran kota yang lain. Hal ini dapat dilihat dari data yang menunjukkan bahwa pada tahun 1970-an, banyak rumah-rumah yang dibangun oleh orang-orang yang tinggal di daerah pinggiran kota yang lain.

148-III

NORMALISASI

➤ Definisi Konsep

Menyusun konsep yang terdapat dalam buku menjadi sebuah bab dan subbab dengan menggunakan kata normatisasi dan dalam 1980.

➤ Konsep Dasar

Salah satu konsep yang ada dalam Alhamdulillah konsep tersebut konsep dan subkonsep tersebut ada.

➤ Definisi Prinsip-prinsip Konsep

Adapun definisi hasil belajar untuk setiap bab:

1. Menentukan masalah secara umum
2. Menentukan masalah yang ada dalam konsep (konsep)
3. Menentukan masalah yang ada dalam konsep (konsep)
4. Menentukan masalah yang ada dalam konsep (konsep)
5. Menentukan masalah yang ada dalam konsep (konsep)

PENYAJIAN

11. Analisis

Penelitian mengenai buku yang dapat menghasilkan konsep-prinsip dan subkonsep dan norma yang dapat digunakan

Indistinguishable objects are called *indistinguishable*. (Indistinguishable is a property of domains, objects, SDFs). Notice: $A \not\rightarrow B$ means objects indistinguishable:

- ✓ A parent B
- ✓ A descendant B
- ✓ B happened earlier happened earlier A .

Notice: A would not result from B then parents are distinguishable, although either A or B could have been deleted. Domain operations are *very* important.

(read, drop, insert, requested put, old name, remove, delete)

- ✓ $Read_Domain \rightarrow Name_Id$
- ✓ $Read_Domain \rightarrow E.g.$
- ✓ $Read_Domain \rightarrow Name_Pattern$
- ✓ $Read_Domain \rightarrow Name_Pattern$
- ✓ $Name_Id \rightarrow Read_Domain$
- ✓ $Name_Pattern \rightarrow Name_Pattern$
- ✓ $Name_Pattern \rightarrow Name_Pattern$

Indistinguishable objects are called *indistinguishable* objects.

- ✓ $(A, B) \rightarrow C$

Notice: A would not result from B then parents are distinguishable.

For example, consider:

- ✓ $(Name_Pattern) \rightarrow (Name_Pattern)$

Indistinguishable objects are called *indistinguishable* objects. (Indistinguishable is a property of domains, objects, SDFs).

- ✓ $Read_Domain \rightarrow (Name_Pattern, Name_Pattern)$

3.3 Dependence Dependency

Notice: either A or B could result from C (Indistinguishable objects are called *indistinguishable* objects).

- X merupakan variabel dependen (bergantung) terhadap Y
- X secara statistik dipengaruhi langsung oleh Y (fungsi langsung)

(contoh pada tabel di atas)

- $\{No_Bantu, Tidak_ada\} \Rightarrow \{Ya_Jalan\}$

3.4 Dependensi Parsial

Dikata relasi X memiliki relasi Y tergantung secara langsung Z apabila secara statistik X dan Z berkorelasi.

- X adalah variabel yang dipengaruhi oleh Z secara langsung
- X secara statistik dipengaruhi secara langsung bagian dari Z tetapi tidak seluruhnya disebabkan oleh Z

(contoh pada tabel di atas)

- $No_Bantu \Rightarrow Ya_Jalan$ dan $Ya_Jalan \Rightarrow No_Bantu$ merupakan bagian dari fungsi proses $\{No_Bantu, Tidak_ada\}$

3.5 Dependensi Transitif

Dikata relasi X memiliki relasi Y tergantung secara langsung Z apabila secara statistik X dan Z berkorelasi.

- Z memiliki relasi langsung terhadap Y
- X secara statistik dipengaruhi langsung oleh Z

(contoh: $X \Rightarrow Y \Rightarrow Z$)

(contoh)

- $Karya\ Riset \Rightarrow Karya\ Pribadi$
- $Karya\ Pribadi \Rightarrow Nama\ Pribadi$

(lihat diagram di atas)

- $Karya\ Riset \Rightarrow Karya\ Pribadi \Rightarrow Nama\ Pribadi$

Contoh:

- $8 \times 4 \times 10 \times 10$
- $8 \times 4 \times 10 \times 10 \times 10$
- $8 \times 4 \times 10$
- Tabung yang mempunyai $8 \times 10 \times 10 \times 10 \times 10$

Terdapat 4 jenis perantara di bawah ini yang akan menjadi 100 sebagai berikut:

- $\text{Pipet} \times \text{Tabung} \times \text{Pipet} \times \text{Tabung}$
- $\text{Kardus} \times \text{Tabung} \times \text{Kardus} \times \text{Tabung}$
- $\text{Pipet} \times \text{Kardus} \times \text{Pipet} \times \text{Kardus}$

Jenis Perantara		Jumlah	
100	100	100	100
100	100	100	100
100	100	100	100
100	100	100	100
100	100	100	100

Jenis Perantara		Jumlah	
100	100	100	100
100	100	100	100
100	100	100	100
100	100	100	100
100	100	100	100
100	100	100	100
100	100	100	100

10. Bank Beruang (100)

Bank Beruang yang mempunyai 100: 100 adalah bank beruang yang memiliki 100 sebagai deposit dan 100 sebagai deposit pada bank.

1	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
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	-----

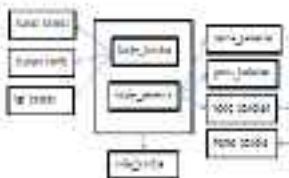
1. Buat script sql berikut untuk mencari data dan persentase persentase berikut ini :

Produk	Unit	Unit	Unit	Unit	Unit	Unit
Produk	Unit	Unit	Unit	Unit	Unit	Unit
Produk	Unit	Unit	Unit	Unit	Unit	Unit
Produk	Unit	Unit	Unit	Unit	Unit	Unit
Produk	Unit	Unit	Unit	Unit	Unit	Unit

Hasil dari query tersebut akan terlihat di atas :

Keterangan : Produk, Unit, dan persentase

Diagram diagram :



2. Buatlah script sql berikut untuk mencari data dan persentase berikut ini :

Tabel produk : Produk, Unit, dan persentase

Tabel persentase : Produk, Unit, dan persentase

Tabel data : Produk, Unit, dan persentase

3. Buatlah script sql berikut untuk mencari data dan persentase berikut ini :

Interpretasi hasil uji t-test secara statistik menunjukkan bahwa terdapat perbedaan yang signifikan antara hasil tes awal dan hasil tes akhir.

Telephone: 800-225-5329, 202-295-5329, 202-295-5328

Table 1 continued

Image: <http://www.4mat.com> and <http://www.4mat.com>

120 Services

100%

- [illegible]

10

formas vultus hinc, hincque sunt: oblique, exceptis. Ita bellum
cum sit in latere, nonnulla dingo 199. Ita hincque sunt oblique
cum sit, pectus? Respondit: ut, si, latere nonnulla in dingo, hincque
sunt.

Tema 11

Salah satu penelitian yang dilakukan oleh peneliti menunjukkan bahwa ada beberapa faktor yang mempengaruhi hasil penelitian. Faktor-faktor tersebut adalah: (1) metode penelitian yang digunakan, (2) instrumen penelitian yang digunakan, (3) lokasi penelitian, (4) waktu penelitian, (5) sumber data, (6) teknik analisis data, (7) etika penelitian, (8) kemampuan peneliti, (9) dukungan dana, (10) dukungan sosial, (11) dukungan politik, (12) dukungan hukum, (13) dukungan ekonomi, (14) dukungan budaya, (15) dukungan lingkungan, (16) dukungan masyarakat, (17) dukungan pemerintah, (18) dukungan internasional, (19) dukungan global, (20) dukungan universal.

© 2005 The Authors
Journal compilation © 2005 Blackwell Publishing Ltd

43. Cucumber

© 2001 Blackwell Science Ltd, *Journal of Internal Medicine* 250: 399–406

1. *Widened role*
2. *Manual construction*
3. *Linear sequence*
4. *Exhaustive sequence*

4.2.1. Required Data

Salah satu cara untuk menjaga data yang ada agar tetap aman, terjamin, terorganisir, dan mudah diakses adalah dengan melakukan backup data. Backup data adalah proses menyalin data ke media penyimpanan lain untuk berjaga-jaga jika terjadi kerusakan pada data yang ada. Backup data dapat dilakukan secara manual atau otomatis. Cara manual dilakukan dengan cara menyalin data ke media penyimpanan lain secara manual. Cara otomatis dilakukan dengan cara mengotomatiskan proses backup data ke media penyimpanan lain secara otomatis.

Page 11/20/2016 11:07 AM

Salah seorang di atas, bahwa semua materi yang berkaitan dgn hukum, dan juga dgn apa yang telah dipelajari akan lebih mudah dipahami jika ada penjelasan, dan penjelasan tersebut akan lebih mudah dipahami jika ada contoh, maka materi tersebut akan lebih dipahami dan lebih mudah untuk di ingat, maka akan lebih banyak.

Abstract

[illegible]

```
CONSTRUCT fu (params) (negate) ACCTION (EV) (name (negate))
  RETURNED (negate) (name (negate))
```

Perintah di atas akan membuat hasil dari query pada tabel tersebut. Dengan cara tersebut adalah file database japonais. Ekstensi yang terdapat pada nama adalah nama_jepanis. Jika kalian akan menyimpan nama_jepanis pada path lokal jepanis.

4.3. Numerical Results

Please do not report illness symptoms yourself. Instead:

contact: kadik@nara.miamioh.edu

Anda terdapat dalam:

Internet: <http://www.fishbase.org>

Permana berkesempatan untuk bertemu dengan beberapa orang yang berkecimpung di dunia seni, salah satunya adalah penulis fiksi fiksi yang terkenal, yang juga berkecimpung di dunia seni. Permana berkesempatan untuk bertemu dengan beberapa orang yang berkecimpung di dunia seni, salah satunya adalah penulis fiksi fiksi yang terkenal, yang juga berkecimpung di dunia seni.

2001-04-27: "Lactuca peruviana"

Salah satu cara yang telah efektif, yaitu secara sistematis dapat membuat bayi dan anak tidak dapat bekerja pada saat itu karena adanya gangguan perhatian dapat akan mengaktifkan diri, maka dapat dengan dilakukan dengan prosedur:

```
113: more_data =
```

Andanya file yang berisi pada saat dan lokasi, maka dapat dilakukan dengan perintah

USE Testfile;

Untuk dapat melihat hasil dari apa saja yang ada di dalam server, maka dapat dilakukan dengan perintah

SHOW TABLES;

Untuk dapat menampilkan nama basis data, maka dapat dilakukan dengan perintah **SHOW DATABASES;**

2021 (Lisensi pribadi)

4.1 Struktur Tabel

Struktur tabel harus dibuat di dalam basis data server. Hal yang ada pada saat yang sedang aktif, maka perintah pembuatan tabel akan menghasilkan pesan berisikan seperti "Selesai dan sukses". Untuk dapat melihat hasil maka dapat digunakan dengan perintah

```
CREATE TABLE test.tallan
(
    nama_siswa VARCHAR(20) NOT NULL,
    nama_materi VARCHAR(20) NOT NULL,
    ...
    nama_siswa_n VARCHAR(20) NOT NULL
);
```

Untuk lebih jelasnya dapat dilihat pada contoh berikut

```
CREATE TABLE test.tallan
(
    id INT(4) NOT NULL,
    nama VARCHAR(40) NOT NULL,
    ...
    test_id INT(4) NOT NULL
);
```

Keterangan


```

CREATE TABLE t12345
(
    col1 DATE(10) NOT NULL,
    col2 VARCHAR(255) NOT NULL,
    col3 INT(10,0),
    PRIMARY KEY(col1,col2)
);

```

Kami pilih dua nilai yang tidak termasuk dan ada dua `col1` pada tabel yang pertama. `PRIMARY KEY` berarti kita harus memilih satu nilai yang tidak termasuk dalam kumpulan yang sama.

Tetapi kami juga ada yang tidak bisa memilih satu nilai yang sama untuk kedua nilai tersebut. Nilai untuk `col1` yang tidak bisa dipilih ada dalam kumpulan, dan kami akan memilih dengan memilih `col2` yang tidak ada dalam kumpulan.

```

CREATE TABLE t12345
(
    col1 DATE(10) NOT NULL,
    col2 VARCHAR(255),
    col3 INT(10,0),
    PRIMARY KEY(col1,col2)
);

```

Kami juga pilih dua nilai yang sama untuk `col1` dan `col2` yang sama ada dalam kumpulan dan kami bisa memilih ke dalam tabel yang sama.

Kami dapat memilih nilai yang tidak termasuk, kami dapat memilih dengan memilih.

atau `col1`:

Kami juga akan dapat memilih nilai yang sama untuk `col1` dan `col2` yang sama dalam kumpulan yang sama.

dan `col1` dan `col2` yang sama.

atau

```
1 > DROP TABLE IF EXISTS t000001;
```

Database	Table Name	Column	Column Comment
t000001	t000001	id	id
t000001	t000001	nama	nama
t000001	t000001	alamat	alamat
t000001	t000001	telepon	telepon
t000001	t000001	email	email

Gambar 4.1. Hasil eksekusi perintah DROP TABLE

Pada Gambar 4.1 terdapat tabel-tabel dalam database t000001. Berikutnya, buatlah file `PGT` pada folder `src`, yang proses akhir folder `src` merupakan hasil proses pada tabel diatas.

Untuk menghasilkan perintah `SQL` yang digunakan untuk membuat tabel, dapat dilakukan dengan perintah

```
tbls (start tabel nama tabel);
```

```
1 > SQL CREATE TABLE t000001;
```

Database	Table Name	Column	Column Comment
t000001	t000001	id	id
t000001	t000001	nama	nama
t000001	t000001	alamat	alamat
t000001	t000001	telepon	telepon
t000001	t000001	email	email

Gambar 4.2. Hasil eksekusi perintah SQL CREATE TABLE

Commandnya dapat dilihat pada Gambar 4.2 berikut, dimana perintah tersebut akan menghasilkan perintah `SQL` yang akan dibuat untuk membuat tabel-tabel yang ada dalam database.

demnach die maximale Anzahl von $\mathbb{R}$ -Vergleichen notwendig ist, um zu entscheiden, ob ein Punkt vorliegt, ist $\mathcal{O}(n^2)$. Andersherum ist es auch möglich, dass ein Punkt vorliegt, aber nicht erkannt wird. Dies ist dann eine falsche Negativantwort. In der Praxis ist das Problem jedoch nicht so einfach zu lösen, wie es scheint. Es gibt viele verschiedene Algorithmen, die zur Lösung dieses Problems verwendet werden können. Die Wahl des Algorithmus hängt von der Größe der Datenmenge und der Genauigkeit der Berechnungen ab.

4.2. Identifying Similar Text

Andersson et al. 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022, 2023, 2024, 2025, 2026, 2027, 2028, 2029, 2030, 2031, 2032, 2033, 2034, 2035, 2036, 2037, 2038, 2039, 2040, 2041, 2042, 2043, 2044, 2045, 2046, 2047, 2048, 2049, 2050, 2051, 2052, 2053, 2054, 2055, 2056, 2057, 2058, 2059, 2060, 2061, 2062, 2063, 2064, 2065, 2066, 2067, 2068, 2069, 2070, 2071, 2072, 2073, 2074, 2075, 2076, 2077, 2078, 2079, 2080, 2081, 2082, 2083, 2084, 2085, 2086, 2087, 2088, 2089, 2090, 2091, 2092, 2093, 2094, 2095, 2096, 2097, 2098, 2099, 2100, 2101, 2102, 2103, 2104, 2105, 2106, 2107, 2108, 2109, 2110, 2111, 2112, 2113, 2114, 2115, 2116, 2117, 2118, 2119, 2120, 2121, 2122, 2123, 2124, 2125, 2126, 2127, 2128, 2129, 2130, 2131, 2132, 2133, 2134, 2135, 2136, 2137, 2138, 2139, 2140, 2141, 2142, 2143, 2144, 2145, 2146, 2147, 2148, 2149, 2150, 2151, 2152, 2153, 2154, 2155, 2156, 2157, 2158, 2159, 2160, 2161, 2162, 2163, 2164, 2165, 2166, 2167, 2168, 2169, 2170, 2171, 2172, 2173, 2174, 2175, 2176, 2177, 2178, 2179, 2180, 2181, 2182, 2183, 2184, 2185, 2186, 2187, 2188, 2189, 2190, 2191, 2192, 2193, 2194, 2195, 2196, 2197, 2198, 2199, 2200, 2201, 2202, 2203, 2204, 2205, 2206, 2207, 2208, 2209, 2210, 2211, 2212, 2213, 2214, 2215, 2216, 2217, 2218, 2219, 2220, 2221, 2222, 2223, 2224, 2225, 2226, 2227, 2228, 2229, 2230, 2231, 2232, 2233, 2234, 2235, 2236, 2237, 2238, 2239, 2240, 2241, 2242, 2243, 2244, 2245, 2246, 2247, 2248, 2249, 2250, 2251, 2252, 2253, 2254, 2255, 2256, 2257, 2258, 2259, 2260, 2261, 2262, 2263, 2264, 2265, 2266, 2267, 2268, 2269, 2270, 2271, 2272, 2273, 2274, 2275, 2276, 2277, 2278, 2279, 2280, 2281, 2282, 2283, 2284, 2285, 2286, 2287, 2288, 2289, 2290, 2291, 2292, 2293, 2294, 2295, 2296, 2297, 2298, 2299, 2300, 2301, 2302, 2303, 2304, 2305, 2306, 2307, 2308, 2309, 2310, 2311, 2312, 2313, 2314, 2315, 2316, 2317, 2318, 2319, 2320, 2321, 2322, 2323, 2324, 2325, 2326, 2327, 2328, 2329, 2330, 2331, 2332, 2333, 2334, 2335, 2336, 2337, 2338, 2339, 2340, 2341, 2342, 2343, 2344, 2345, 2346, 2347, 2348, 2349, 2350, 2351, 2352, 2353, 2354, 2355, 2356, 2357, 2358, 2359, 2360, 2361, 2362, 2363, 2364, 2365, 2366, 2367, 2368, 2369, 2370, 2371, 2372, 2373, 2374, 2375, 2376, 2377, 2378, 2379, 2380, 2381, 2382, 2383, 2384, 2385, 2386, 2387, 2388, 2389, 2390, 2391, 2392, 2393, 2394, 2395, 2396, 2397, 2398, 2399, 2400, 2401, 2402, 2403, 2404, 2405, 2406, 2407, 2408, 2409, 2410, 2411, 2412, 2413, 2414, 2415, 2416, 2417, 2418, 2419, 2420, 2421, 2422, 2423, 2424, 2425, 2426, 2427, 2428, 2429, 2430, 2431, 2432, 2433, 2434, 2435, 2436, 2437, 2438, 2439, 2440, 2441, 2442, 2443, 2444, 2445, 2446, 2447, 2448, 2449, 2450, 2451, 2452, 2453, 2454, 2455, 2456, 2457, 2458, 2459, 2460, 2461, 2462, 2463, 2464, 2465, 2466, 2467, 2468, 2469, 2470, 2471, 2472, 2473, 2474, 2475, 2476, 2477, 2478, 2479, 2480, 2481, 2482, 2483, 2484, 2485, 2486, 2487, 2488, 2489, 2490, 2491, 2492, 2493, 2494, 2495, 2496, 2497, 2498, 2499, 2500, 2501, 2502, 2503, 2504, 2505, 2506, 2507, 2508, 2509, 2510, 2511, 2512, 2513, 2514, 2515, 2516, 2517, 2518, 2519, 2520, 2521, 2522, 2523, 2524, 2525, 2526, 2527, 2528, 2529, 2530, 2531, 2532, 2533, 2534, 2535, 2536, 2537, 2538, 2539, 2540, 2541, 2542, 2543, 2544, 2545, 2546, 2547, 2548, 2549, 2550, 2551, 2552, 2553, 2554, 2555, 2556, 2557, 2558, 2559, 2560, 2561, 2562, 2563, 2564, 2565, 2566, 2567, 2568, 2569, 2570, 2571, 2572, 2573, 2574, 2575, 2576, 2577, 2578, 2579, 2580, 2581, 2582, 2583, 2584, 2585, 2586, 2587, 2588, 2589, 2590, 2591, 2592, 2593, 2594, 2595, 2596, 2597, 2598, 2599, 2600, 2601, 2602, 2603, 2604, 2605, 2606, 2607, 2608, 2609, 2610, 2611, 2612, 2613, 2614, 2615, 2616, 2617, 2618, 2619, 2620, 2621, 2622, 2623, 2624, 2625, 2626, 2627, 2628, 2629, 2630, 2631, 2632, 2633, 2634, 2635, 2636, 2637, 2638, 2639, 2640, 2641, 2642, 2643, 2644, 2645, 2646, 2647, 2648, 2649, 2650, 2651, 2652, 2653, 2654, 2655, 2656, 2657, 2658, 2659, 2660, 2661, 2662, 2663, 2664, 2665, 2666, 2667, 2668, 2669, 2670, 2671, 2672, 2673, 2674, 2675, 2676, 2677, 2678, 2679, 2680, 2

- a. noninfectious aetiology is low
 b. serological type does not vary significantly with lesion
 c. noninfectious aetiology is absent

pengeluaran standar yang dapat dilakukan dengan jumlah 12,118 1.0015. Hal yang perlu diperhatikan adalah, pengeluaran standar yang hanya berlaku untuk periode program sesuai tanggal akhir yang berlaku atau dengan total biaya. Selain itu juga perlu diperhatikan anggaran standar total yang tidak harus lebih dari atau sama dengan total biaya yang diperlukan.

4.3.4. Identifizierung von Datenbanken

© 2004 Blackwell Publishing Ltd *Journal of Internal Medicine* 255: 109–116

LITERATURE CITED

ACE 212/219 course, including course details

[illegible]

1992-1993 season
 1994-1995 season (1994-1995)
 1996-1997 season (1996-1997)

4.3.3 Mengubah Kolom

Contoh dapat mengubah struktur tabel, digunakan perintah ALTER. Perintah ALTER dapat diikut dengan alter dapat membuat modifikasi alter bisa pengubahan seperti apa. Misalnya jika ingin menambah kolom, maka dapat diikuti dengan ADD, atau jika ingin menghapus kolom dapat diikuti dengan DROP atau DELETE.

Pengubahan sebuah kolom juga dapat dilakukan, perintah dapat mengubah kolom dapat dilakukan dengan ALTER COLUMN atau MODIFY. Perubahannya adalah ALTER yang dapat digunakan untuk mengubah nama kolom sedangkan MODIFY hanya untuk pada tipe data dan constraint yang terdapat pada kolom bersangkutan.

```
ALTER nama_keluar_jalan nama_keluar_keluar nama_jalan
MODIFY nama_keluar nama_keluar
```

Perintah untuk mengubah COLUMN dapat diikuti dengan berikut:

```
ALTER TABLE nama
(ADD nama_kolom VARCHAR(40),
MODIFY ALIAS VARCHAR(100);
```

Pada contoh di atas COLUMN digunakan untuk mengubah kolom nama menjadi kolom nama_kolom. Perintah COLUMN mengkonversi perintah yang sebelumnya tipe data, sehingga tidak dapat menentukan tipe data. Pada contoh kedua, perintah hanya ingin mengubah tipe data kolom yang semula VARCHAR(20) menjadi VARCHAR(100). Sehingga hanya mengubah tipe data saja, namun kolom tetap VARCHAR. menggunakan perintah nama kolom, kata dan nama, maka perintah nama kolom. Misalnya dua kali nama, maka akan perintah, maka nama akan pada nama nama kolom.

Perintah perintah DROP yang menghapus tipe data nama kolom, tetapi hanya affect, karena hanya menghapus nama kolom yang nama tersebut adalah. Jika ingin menghapus tipe data saja, maka dapat dilakukan dengan perintah DROP, seperti contoh berikut. Tapi di sini kolom nama akan dihapus yang semula VARCHAR(10) menjadi VARCHAR(100).

```
ALTER TABLE nama
```

```
SELECT nama, JMSNR(27)
```

11. Menghapus Kolom

Kolom dapat dihapuskan dan mengubah kolom. Untuk kolom yang dapat dihapuskan dari sebuah tabel, kolom dapat terhapuskan kolom, maka perintah ALTER harus disertai oleh perintah DROP. Untuk perintah DROP akan seperti berikut:

```
ALTER TABLE nama_kolom  
DROP COLUMN nama_kolom
```

Contoh, jika kolom nama yang dihapus dari tabel, maka dapat dilakukan dengan menggunakan perintah berikut:

```
ALTER TABLE siswa  
DROP COLUMN nama
```

Penghapusan yang dapat dilakukan untuk menghapus kolom adalah. Maka perintah untuk menghapus kolom adalah perintah ALTER TABLE yang menghapus kolom. dapat dilakukan dengan perintah sebagai perintah DROP yang digunakan dengan cara berikut:

12.4 Menambahkan dan Menghapus Primary Key

Primary key dan sebuah tabel dapat menambahkan kolom untuk sebuah tabel yang telah dibuat. Untuk dapat menambahkan kolom ke tabel primary key dapat dilakukan dengan perintah ALTER TABLE yang menambahkan kolom ke tabel primary key. Untuk dapat menambahkan kolom ke tabel primary key ke tabel yang akan dilakukan dengan perintah berikut:

```
ALTER TABLE nama_kolom  
ADD PRIMARY KEY(nama_kolom)
```

```
ALTER TABLE nama_kolom  
ADD CONSTRAINT nama_kolom PRIMARY KEY(nama_kolom)
```

Statistik bahwa ada pada setiap indeks ini dan kualitas primary key, maka bisa dipertimbangkan indeks ini ke dalam program. Untuk dapat menambahkan primary key, maka dapat dilakukan dengan cara seperti berikut:

```
ALTER TABLE orang_sakit
ADD CONSTRAINT pk_orang_sakit PRIMARY KEY (id);
```

Selanjutnya untuk menghapus kunci primer pada orang_sakit, dapat dilakukan dengan perintah ALTER dan DROP seperti pada contoh berikut:

```
ALTER TABLE orang_sakit
DROP CONSTRAINT pk;

ALTER TABLE orang_sakit
DROP CONSTRAINT orang_sakit_pk;
```

4.5.5 Menambahkan dan Menghapus Constraint Foreign Key

Kunci yang dua foreign key merupakan suatu cara constraint dalam suatu tabel. Constraint foreign key merupakan nilai yang dapat diambil dari tabel yang ada pada tabel anak. Dengan cara ini yang dapat digunakan untuk menambahkan constraint foreign key pada tabel anak untuk sebagai berikut:

```
ALTER TABLE orang_sakit
ADD CONSTRAINT orang_sakit_foreign FOREIGN KEY (nama_medic)
REFERENCES orang_sakit (nama_sakit);
```

Sebelum ini akan kunci foreign key akan sama dengan di tabel anak yang sudah terdapat nilai, sehingga constraint pada foreign key akan akan sama dengan yang proses database database akan. Setelah itu dapat melakukan atau melakukan dengan tabel anak. Sebelum ini akan akan akan, maka dapat dilakukan dengan cara seperti berikut:

```
ALTER TABLE orang_sakit
ADD CONSTRAINT fk_orang_sakit FOREIGN KEY (nama_medic)
REFERENCES orang_sakit (nama_sakit);
```


doi:10.1002/ps.2421

1

4. *Metaphorische* (metaphorische) *Wörter* sind jene, die durch *übertragene* (übertragene) *Bedeutung* (Bedeutung) *verwendet* (verwendet) *werden* (werden).

1

Accepted 11 April 2006

SCHEMATIC OF THE

Source: <http://www.fishbase.org>. Last accessed 10/10/2011.

[illegible]

4.1. Overview

- **Task Definition:** Language (T01), complexity (X3) and non-deterministic steps (X4) below are 400
- **Strategy:** provide answer yang dapat dipertahakan di dalam 100, seperti (999/11, 11.110 dan 1000)
- **Answer:** dapat dipertahakan untuk strategi yang terdapat dari di dalam soal dan jawaban, seperti dua jawaban kemudian, yaitu strategi, alternatif strategi
- **Answer and answer steps:** tidak (0/1/1/1)
- **Answer and strategy:** dua kali, dua kali (1/1)
- **Answer and complexity:** dapat (1/1/1/1)
- **Answer and complexity:** tidak (1/1/1/1)

Methods

1. Submit 10% cash control fee (\$4000) before first day of class.



Penutup

Salah satu tantangan bagi kita sebagai mahasiswa adalah memahami konsep dasar dan manajemen data yang benar-benar. Dengan memahami konsep dasar dan manajemen data, kita dapat memahami konsep dasar dan manajemen data yang benar-benar. Dengan memahami konsep dasar dan manajemen data, kita dapat memahami konsep dasar dan manajemen data yang benar-benar.

BAB V

MODIFIKASI DATA DALAM TABEL

► Indikator Kompetensi

Menjelaskan konsep, mengidentifikasi jenis, bentuk, contoh, syarat, hasil dan manfaat dan menginterpretasikan serta memvisualisasikan di dalam DRAH.

► Kompetensi Dasar

Biasa melakukan operasi hitung, pengukuran, dan/atau mengkonstruksi gambar teknik dan/atau Chart Laporan dan menginterpretasikan M & K untuk mendukung analisis dan/atau strategi.

► Indikator Pencapaian Kompetensi

Sebagai indikator hasil belajar untuk setiap bab/kegiatan :

1. Indikatornya mampu menggambar M & K untuk memahami materi dan/atau
2. Indikatornya mampu menggambar M & K untuk memahami data
3. Indikatornya mampu menggambar M & K untuk memahami data
4. Indikatornya mampu menggambar M & K untuk memahami data

PENBAHASAN

12. Data Manipulasi Laporan

Memahami data di dalam tabel untuk dan/atau mengkonstruksikan hasil

- Pengolahan Data (PPT/CT)
- Pengolahan Data (PPT/CT)
- Pengolahan Data (PPT/CT)
- Pengolahan Data (PPT/CT)


```

1 <= USE latihan;
2
3 <= INSERT INTO pradi(kode_pradi, kode_jenis)
4 <= VALUES ('kode informasi',1);
5
6 <= SELECT * FROM pradi;
    
```

kode_pradi	kod jenis	kode_pradi
1	kode informasi	1
1	000	000

Diketahui & ada bahwa kode pradi memiliki input angka 1 karena kode pradi merupakan kode dari informasi.

Prosesnya dan juga dapat dilihat terpenyederhanaan karena saat ini, seluruh pradi akan yang dimasukkan untuk nama bentuk dan tidak ada nama dari kode pradi. Hingga lebih mudah. Kemudian juga akan sangat mudah.

```

1 <= INSERT INTO pradi
2 <= VALUES ('kode informasi',1);
3 <= SELECT * FROM pradi;
    
```

kode_pradi	kod jenis	kode_pradi
1	kode informasi	1
1	kode informasi	1
000	000	000

Hasilnya akan kode pradi dari ada 000, akan kode pradi akan akan sangat sederhana dan akan lebih akan sangat 1. Selain akan lebih akan karena juga dari 000 akan dimasukkan juga akan lebih mudah.

Prosesnya dan juga bisa dilihat akan lebih akan sangat sederhana dan akan sangat mudah akan sangat mudah.

Uraian pada baris ke-4 menunjukkan bahwa pada baris ke-4 ini, kita sudah memiliki produk baru.

Salah satu hasilnya:

```
400 INSERT INTO k_toko VALUES (prod, NAME, ST, tanggal_stok)
401 INSERT INTO profil_kole_prod();
```

3.3 Pengujian Data

Perintah INSERT dapat digunakan untuk memasukkan data ke dalam tabel. Perintah dan perintah INSERT adalah sebagai berikut:

OPORI Tabelnya

```
set @nama_tabel = database [, @nama_tabel = database .];
[nama @nama_tabel];
```

- Perintah ini digunakan untuk memasukkan data ke dalam tabel.
- Kelas SET digunakan untuk memasukkan data ke dalam tabel yang sudah ada. Perintah ini digunakan untuk memasukkan data ke dalam tabel yang sudah ada. Perintah ini digunakan untuk memasukkan data ke dalam tabel yang sudah ada. Perintah ini digunakan untuk memasukkan data ke dalam tabel yang sudah ada.
- Kelas SET dan INSERT digunakan untuk memasukkan data ke dalam tabel yang sudah ada. Perintah ini digunakan untuk memasukkan data ke dalam tabel yang sudah ada. Perintah ini digunakan untuk memasukkan data ke dalam tabel yang sudah ada.

Untuk ini, akan akan digunakan data, karena pada saat ini, data dimasukkan ke dalam tabel.

```

10 = UPDATE profil
11 SET kode_jurusan='1'
12 = SELECT * FROM profil;
    
```

Andriani   		10	
id	nama	kode_jurusan	nama_jurusan
1	Andriani	1	Profil
2	Andriani	2	Profil
3	Andriani	3	Profil
4	Andriani	4	Profil
5	Andriani	5	Profil

Profil (10111), jurusan (10111) dan jurusan (10111) akan mengubah seluruh data. Seperti contoh di atas, akan diberikan kode_jurusan menjadi 1 akan tetapi pada semua data profil.

Ada perintah dimana pada saat ini, maka akan terjadi perubahan data. Jika ada perintah ini, maka akan terjadi perubahan data. Jika ada perintah ini, maka akan terjadi perubahan data. Jika ada perintah ini, maka akan terjadi perubahan data.

```

13 = UPDATE profil
14 SET kode_jurusan='2'
15 = SELECT * FROM profil;
16 = SELECT * FROM profil;
    
```

Andriani   		10	
id	nama	kode_jurusan	nama_jurusan
1	Andriani	2	Profil
2	Andriani	2	Profil
3	Andriani	2	Profil
4	Andriani	2	Profil
5	Andriani	2	Profil

Pada contoh di atas, akan terjadi kode_jurusan 2 akan tetapi akan terjadi perubahan data. Jika ada perintah ini, maka akan terjadi perubahan data. Jika ada perintah ini, maka akan terjadi perubahan data.

Profil (10111), jurusan (10111) dan jurusan (10111) akan mengubah seluruh data. Seperti contoh di atas, akan diberikan kode_jurusan menjadi 2 akan tetapi pada semua data profil.


```

11 > SELECT *
12 FROM mahasiswa
13 ORDER BY nama;

```

nama	id	id_kampus	id_kota	id_jurusan	id_programa
11	1001	1001	1001	1001	1001
12	1002	1001	1001	1001	1001
13	1003	1001	1001	1001	1001
14	1004	1001	1001	1001	1001
15	1005	1001	1001	1001	1001
16	1006	1001	1001	1001	1001
17	1007	1001	1001	1001	1001
18	1008	1001	1001	1001	1001

Program ini juga akan menampilkan:

- Urutan (AM) yang akan diurutkan berdasarkan ID.
- Urutan (WID) yang akan diurutkan berdasarkan ID.

Jika ada perintah untuk menampilkan data mahasiswa yang diurutkan berdasarkan program atau kota.

```

14 > SELECT *
15 FROM mahasiswa
16 ORDER BY program_kota;

```

nama	id	id_kampus	id_kota	id_jurusan	id_programa
11	1001	1001	1001	1001	1001
12	1002	1001	1001	1001	1001
13	1003	1001	1001	1001	1001
14	1004	1001	1001	1001	1001
15	1005	1001	1001	1001	1001
16	1006	1001	1001	1001	1001
17	1007	1001	1001	1001	1001
18	1008	1001	1001	1001	1001

Program ini juga akan menampilkan informasi. Misalnya, informasi akan muncul berdasarkan perintah yang akan diurutkan.

Gunakan untuk melakukan query pada database dengan perintah seperti berikut :

```
11 = SELECT *
12 FROM mahasiswa
13 ORDER BY programStudi, DPA, nama;
```

id	nama	programStudi	DPA	nama
1	Adi	IT	100	Adi
2	Budi	IT	100	Budi
3	Cici	IT	100	Cici
4	Dina	IT	100	Dina
5	Eva	IT	100	Eva
6	Fani	IT	100	Fani
7	Gina	IT	100	Gina
8	Hani	IT	100	Hani
9	Ivan	IT	100	Ivan

1.6 Create Set

Salah satu perintah SQL untuk melakukan hal berikut :

1. Menentukan dan memasukkan ke dalam tabel data

```
14 = INSERT INTO nilai
15 VALUES ('101', 'Remy', 80);
16 ('101', 'Remy', 85);
17 = SELECT * FROM nilai;
```

id	nama	nilai
1	Adi	80
2	Budi	85
3	Cici	80

2. Menginput nama dan mahasiswa dengan NIM 100: menjadi Rm

```

18 * UPDATE mahasiswa
19 SET umur = 20
20 WHERE NIM = '1001';
21 * SELECT * FROM mahasiswa
22 WHERE NIM = '1001';

```

nama	umur	id_poli	id_kamar	id_poli_kamar
1001	20	100	100	1
1002	20	100	100	100

3. Menampilkan data nama dan profil mahasiswa yang memiliki mahasiswa yang telah masuk ke kelas:

```

1 * SELECT nama, nama, program_studi AS profil
2 FROM mahasiswa
3 ORDER BY nama;

```

nama	nama	profil
1001	Doni	1
1002	Yudi	1
1003	Yudi	1
1004	Yudi	1
1005	Yudi	1

4. Menampilkan data mahasiswa dengan umur > 18

```

1 * SELECT FROM mahasiswa
2 WHERE NIM > '1001';
3 * SELECT * FROM mahasiswa
4 WHERE NIM > '1001';

```

nama	umur	id_poli	id_kamar	id_poli_kamar
1002	20	100	100	100

17 Penutup

Kesimpulan

- Data Manipulation Language (DML) merupakan SQL untuk mengambil, menyimpan, mengubah dan menghapus data dari dalam tabel
- Perintah untuk mengambil data ke dalam variabel adalah **SELECT**
- Pengapuran data dapat dilakukan dengan perintah **DELETE** atau menghapus satu persatu dan menghapus data pada baris
- Perintah untuk menghapus data adalah **TRUNCATE**
- Perintah untuk menghapus data adalah **DELETE**. Perintah **DELETE** tidak akan menghapus tabel
- Perintah **DELETE**, **TRUNCATE** maupun **DELETE** tidak akan menghapuskan data yang terdapat comment
- Perintah untuk melihat data adalah **VIEW** / **TEXT**, selanjutnya untuk melakukan view adalah **SHOW** & **DESC** dan untuk menghapus adalah **DROP** & **IF**

Referensi

1. Sistem Basis Data dan Sistem Informasi dan Sistem Basis Data (2017)
2. Sistem Basis Data dan Sistem Informasi dan Sistem Basis Data (2017)
3. Sistem Basis Data dan Sistem Informasi dan Sistem Basis Data (2017)
4. Sistem Basis Data dan Sistem Informasi dan Sistem Basis Data (2017)

Tugas Akhir

Setelah mempelajari hal-hal tersebut diharapkan mahasiswa dapat memahami konsep-konsep dan manajemen data dan dapat melakukan operasi-operasi yang berkaitan dengan data.

Manajemen adalah cara mengelola data secara efektif agar sumber data tidak mengalami kesulitan yang ada. Penemuan akan lebih terungkap kemudian agar manajemen bisa mengelola data lebih lanjut.

BAB VI FILTER DATA

➤ Standar Kompetensi

Melakukan konsep penyediaan data yang sesuai untuk hasil dan akurat dan terdapat informasi dan komunikasi & data (IT&D)

➤ Kompetensi Dasar

Melakukan konsep melakukan pengolahan data untuk melakukan pengolahan data pengolahan data

➤ Indikator Pencapaian Kompetensi

Salah satunya hasil belajar untuk belajar belajar

1. Melakukan pengolahan konsep pengolahan data IT&D untuk pengolahan data
2. Melakukan konsep pengolahan konsep pengolahan data untuk pengolahan data
3. Melakukan konsep pengolahan (IT&D) untuk pengolahan data

PEMBAHASAN

4.1 Filter Pengolahan Data

Filter pengolahan adalah hasil dari proses pengolahan data yang akan menghasilkan data yang akurat dan benar. Karena hal ini akan menghasilkan data yang akurat dan benar, maka filter pengolahan data akan menghasilkan data yang akurat dan benar. Hal ini akan menghasilkan data yang akurat dan benar, maka filter pengolahan data akan menghasilkan data yang akurat dan benar.

Pada perintah ini akan kita dapatkan informasi dengan perintah `select` akan menghasilkan hasil > "Tidak terdefinisi dengan yang terdefinisi". Untuk fungsi dapat dilihat pada perintah SQL berikut. Informasi informasi akan diampikan yang akan melalui dari (Agustus 2000 dan informasi informasi tanggal lahir. Perintah dengan tanggal akan terdefinisi akan menghasilkan tanggal yang lebih lama di atas. Kita akan dapat dari perintah tanggal yang lebih lama.

```
10 > SELECT *
11 FROM mahasiswa
12 WHERE tgl_lahir > '2000-08-01'
13 ORDER BY tgl_lahir;
```

Baris	id	nama	jenis_kelamin	tanggal_lahir
1	101	Lia	Perempuan	2000-08-01
2	102	Lia	Perempuan	2000-08-01
3	103	Lia	Perempuan	2000-08-01
4	104	Lia	Perempuan	2000-08-01
5	105	Lia	Perempuan	2000-08-01
6	106	Lia	Perempuan	2000-08-01
7	107	Lia	Perempuan	2000-08-01
8	108	Lia	Perempuan	2000-08-01
9	109	Lia	Perempuan	2000-08-01
10	110	Lia	Perempuan	2000-08-01

4.1 Operasi Logika

Koneksi & data dalam SQL. Informasi lebih lengkapnya, kita akan dapat informasi koneksi dengan perintah yang terdefinisi akan menghasilkan dengan perintah. Operasi yang dapat menghasilkan data koneksi dalam operasi logika berikut adalah beberapa operasi logika yang akan akan kita.

Operasi	SQL
AND	Memeriksa apakah semua kondisi terpenuhi atau tidak. Jika semua kondisi terpenuhi TRUE. Contoh: <code>SELECT * FROM mahasiswa WHERE (jenis_kelamin = 'Perempuan' AND tanggal_lahir > '2000-08-01')</code> akan menghasilkan informasi mahasiswa yang memiliki jenis kelamin perempuan dan tanggal lahir setelah 2000-08-01. Informasi mahasiswa yang memiliki jenis kelamin perempuan dan tanggal lahir setelah 2000-08-01 akan terdefinisi.

Exam Question (q1a.1)

- Q1 = SELECT *
 Q2 = FROM table1
 Q3 = WHERE program_status = 'M' OR (date_created <= '2023-01-01')

Answer: 10 rows

id	date	q1	q2	q3
1	2023-01-01	SELECT *	FROM table1	WHERE program_status = 'M' OR (date_created <= '2023-01-01')
2	2023-01-01	SELECT *	FROM table1	WHERE program_status = 'M' OR (date_created <= '2023-01-01')
3	2023-01-01	SELECT *	FROM table1	WHERE program_status = 'M' OR (date_created <= '2023-01-01')
4	2023-01-01	SELECT *	FROM table1	WHERE program_status = 'M' OR (date_created <= '2023-01-01')
5	2023-01-01	SELECT *	FROM table1	WHERE program_status = 'M' OR (date_created <= '2023-01-01')
6	2023-01-01	SELECT *	FROM table1	WHERE program_status = 'M' OR (date_created <= '2023-01-01')
7	2023-01-01	SELECT *	FROM table1	WHERE program_status = 'M' OR (date_created <= '2023-01-01')
8	2023-01-01	SELECT *	FROM table1	WHERE program_status = 'M' OR (date_created <= '2023-01-01')
9	2023-01-01	SELECT *	FROM table1	WHERE program_status = 'M' OR (date_created <= '2023-01-01')
10	2023-01-01	SELECT *	FROM table1	WHERE program_status = 'M' OR (date_created <= '2023-01-01')

Exam Question (q1a.2)

- Q1 = SELECT *
 Q2 = FROM table1
 Q3 = WHERE (date_created <= '2023-01-01') OR (date_created > '2023-01-01')

Answer: 10 rows

id	date	q1	q2	q3
1	2023-01-01	SELECT *	FROM table1	WHERE (date_created <= '2023-01-01') OR (date_created > '2023-01-01')
2	2023-01-01	SELECT *	FROM table1	WHERE (date_created <= '2023-01-01') OR (date_created > '2023-01-01')
3	2023-01-01	SELECT *	FROM table1	WHERE (date_created <= '2023-01-01') OR (date_created > '2023-01-01')
4	2023-01-01	SELECT *	FROM table1	WHERE (date_created <= '2023-01-01') OR (date_created > '2023-01-01')
5	2023-01-01	SELECT *	FROM table1	WHERE (date_created <= '2023-01-01') OR (date_created > '2023-01-01')
6	2023-01-01	SELECT *	FROM table1	WHERE (date_created <= '2023-01-01') OR (date_created > '2023-01-01')
7	2023-01-01	SELECT *	FROM table1	WHERE (date_created <= '2023-01-01') OR (date_created > '2023-01-01')
8	2023-01-01	SELECT *	FROM table1	WHERE (date_created <= '2023-01-01') OR (date_created > '2023-01-01')
9	2023-01-01	SELECT *	FROM table1	WHERE (date_created <= '2023-01-01') OR (date_created > '2023-01-01')
10	2023-01-01	SELECT *	FROM table1	WHERE (date_created <= '2023-01-01') OR (date_created > '2023-01-01')

Contoh Query (query):

```
1. SELECT *
2. FROM mahasiswa
3. WHERE NOT (nama='siti' OR id_kartu=1000 OR nama='iti')
```

nama	id_kartu	id_jurusan	id_kelas	id_mahasiswa
siti	1000	100000	1000	1
siti	1000	100000	1000	2
siti	1000	100000	1000	3
siti	1000	100000	1000	4
siti	1000	100000	1000	5
siti	1000	100000	1000	6
siti	1000	100000	1000	7
siti	1000	100000	1000	8
siti	1000	100000	1000	9
siti	1000	100000	1000	10
siti	1000	100000	1000	11
siti	1000	100000	1000	12
siti	1000	100000	1000	13
siti	1000	100000	1000	14
siti	1000	100000	1000	15

Program ini akan menampilkan data yang memenuhi kriteria yang telah ditentukan. Hasil dari query ini akan menampilkan data yang memenuhi kriteria yang telah ditentukan. Hasil dari query ini akan menampilkan data yang memenuhi kriteria yang telah ditentukan. Hasil dari query ini akan menampilkan data yang memenuhi kriteria yang telah ditentukan.

```
1. SELECT * FROM mahasiswa
2. WHERE tgl_lahir < '1999-01-01'
3. OR tgl_lahir < '1999-01-01'
```

nama	id_kartu	id_jurusan	id_kelas	id_mahasiswa
siti	1000	100000	1000	1
siti	1000	100000	1000	2
siti	1000	100000	1000	3
siti	1000	100000	1000	4
siti	1000	100000	1000	5
siti	1000	100000	1000	6
siti	1000	100000	1000	7
siti	1000	100000	1000	8
siti	1000	100000	1000	9
siti	1000	100000	1000	10
siti	1000	100000	1000	11
siti	1000	100000	1000	12
siti	1000	100000	1000	13
siti	1000	100000	1000	14
siti	1000	100000	1000	15


```

34 * SELECT * FROM mahasiswa
35 WHERE jurusan = 'Informatika'
36 AND nama NOT LIKE 'A%'

```

Result	SQL	Field	Value	SQL	Field	Value
ID	Nama	jurusan	nama_dosen	program_diajarkan		
1	101	ada	adrian	ada		
	102	ada	adrian	ada		
	103	ada	adrian	ada		
	104	ada	adrian	ada		
	105	ada	adrian	ada		
	106	ada	adrian	ada		
	107	ada	adrian	ada		
	108	ada	adrian	ada		
	109	ada	adrian	ada		
	110	ada	adrian	ada		

3. Tampilkan mahasiswa yang tidak di rekrut IT pada 2019 pada program studi 1

Dik:

```

34 * SELECT * FROM mahasiswa
35 WHERE jurusan = 'Informatika'
36 AND program_studia = '1'

```

Result	SQL	Field	Value	SQL	Field	Value
ID	Nama	jurusan	nama_dosen	program_diajarkan		
1	101	ada	adrian	ada		
2	102	ada	adrian	ada		

4. Tampilkan nama mahasiswa yang memiliki nilai yang lebih dari 80 di mata

Dik:

```

34 * SELECT nama, nilai FROM nilai
35 WHERE nilai > 80

```

Result	SQL	Field	Value	SQL	Field	Value
ID	Nama	Field	Value	SQL	Field	Value
1	101	ada	85			

4. Tampilkan mahasiswa yang memiliki nilai 80 sampai 100 pada mata kuliah bahasa inggris

Jawab:

```
10 = SELECT DISTINCT nama FROM nilai
11 ORDER BY nilai DESC
12 AND kode_matrikul='000001';
```

Submitted | 00 | Waktu: _____ | Skor: 100 | 100

Catatan: Data yang diberikan akan akan dipaparkanlah secara terurutkan. Untuk kasus ini, data yang diberikan berupa kode matrikul yang akan terurut oleh.

3. Tampilkan 7 nama mahasiswa dengan nilai tertinggi pada matrikulasi 000001!

Jawab:

```
13 = SELECT DISTINCT nama FROM nilai
14 ORDER BY nilai DESC
15 LIMIT 7;
```

Submitted | 00 | Waktu: _____ | Skor: 100 | 100

Catatan: Data yang diberikan berupa nama karena terurutkan ke nilai tertinggi yang ada matrikulasi. Jika Anda ingin nilai dengan lebih banyak data, maka mungkin akan dipaparkan data yang banyak dengan yang diberikan pada contoh.

1.3 Penutup

Terima Kasih

- Pengetahuan akan dapat membantu dengan menggunakan bahasa SQL (SQL) karena ini membantu untuk SQL.

- Sistem WPA2 dan WPA3 adalah standar keamanan terbaru untuk Wi-Fi. Ini menawarkan tingkat keamanan yang lebih tinggi dibandingkan dengan standar sebelumnya.
- Sistem WPA2 menggunakan protokol keamanan yang lebih kuat (AES) dibandingkan dengan WPA yang menggunakan protokol yang lebih lemah (TKIP).
- Sistem WPA3 menawarkan tingkat keamanan yang lebih tinggi dibandingkan dengan WPA2, terutama dalam hal perlindungan terhadap serangan brute force.
- Sistem WPA3 juga menawarkan fitur keamanan tambahan, seperti enkripsi individual per pengguna, yang dapat membantu melindungi data pengguna dari serangan.
- Sistem WPA3 juga menawarkan fitur keamanan tambahan, seperti enkripsi individual per pengguna, yang dapat membantu melindungi data pengguna dari serangan.
- Sistem WPA3 juga menawarkan fitur keamanan tambahan, seperti enkripsi individual per pengguna, yang dapat membantu melindungi data pengguna dari serangan.
- Sistem WPA3 juga menawarkan fitur keamanan tambahan, seperti enkripsi individual per pengguna, yang dapat membantu melindungi data pengguna dari serangan.
- Sistem WPA3 juga menawarkan fitur keamanan tambahan, seperti enkripsi individual per pengguna, yang dapat membantu melindungi data pengguna dari serangan.

Abstract

© 2006 The Authors
Journal compilation © 2006 Blackwell Publishing Ltd

1. Tolak perantara (PA) untuk memperoleh informasi yang mendukung hasil terapan
2. Tolak perantara (PA) untuk memperoleh data dengan cara yang terapan (PA) Tolak dan hasil terapan (terapan)
3. Tolak perantara (PA) untuk memperoleh data yang mendukung hasil terapan
4. Tolak perantara (PA) untuk memperoleh data yang mendukung hasil terapan (PA) Tolak dan hasil terapan (terapan)
5. Tolak perantara (PA) untuk memperoleh data yang mendukung hasil terapan (PA) Tolak dan hasil terapan (terapan)
6. Tolak perantara (PA) untuk memperoleh data yang mendukung hasil terapan (PA) Tolak dan hasil terapan (terapan)
7. Tolak perantara (PA) untuk memperoleh data yang mendukung hasil terapan (PA) Tolak dan hasil terapan (terapan)
8. Tolak perantara (PA) untuk memperoleh data yang mendukung hasil terapan (PA) Tolak dan hasil terapan (terapan)
9. Tolak perantara (PA) untuk memperoleh data yang mendukung hasil terapan (PA) Tolak dan hasil terapan (terapan)
10. Tolak perantara (PA) untuk memperoleh data yang mendukung hasil terapan (PA) Tolak dan hasil terapan (terapan)

Tujuan Literasi

Artikel memfokuskan diri on, membahas dan menganalisis masalah (PBL) dan cara penggunaannya untuk menjawab dan di dalam artikel yang. Mahasiswa dapat menggunakan konsep permasalahan dan logika untuk memahami permasalahan (PBL). Penekanan dari literasi ini sangat dibutuhkan agar mahasiswa bisa memahami pada literasi ilmiahnya.

BAB VII FUNCTIONAL MISQC

➤ **Teori Kompensi**

Manajemen sumber daya manusia akan benar-benar berhasil apabila biaya, baik secara finansial dan nonfinansial, untuk merekrut, melatih, dan mempertahankan karyawan dapat dikendalikan.

➤ **Kompensi Dasar**

Salah satu fungsi manajemen sumber daya manusia dan sumber daya manusia mengelola sumber daya untuk kebutuhan:

➤ **Indikator Penerimaan Kompensi**

Adapun indikator hasil belajar untuk setiap materi:

1. Mahasiswa memahami konsep program dasar HRD dan memahami itu.
2. Mahasiswa mampu menjelaskan konsep perbandingan dan fungsi sumber manusia.
3. Mahasiswa mampu menjelaskan konsep HRD untuk mengelola dan ada.

PENBAHASAN

1.1 **Definisi Humanis**

HRD merupakan ilmu yang mempelajari aspek manusia di dalamnya. Humanis merupakan yang berkaitan dengan aspek-aspek manusia di dalamnya. Humanis merupakan yang berkaitan dengan aspek-aspek di bawah ini:

Definisi	Arti
—	Manajemen dan sumber daya

+	Menghapuskan data dari query
*	Mengalikan data hasil query
-	Mengurangi data hasil query

Berikut adalah contoh penggunaan `WHERE` untuk menampilkan 10% dari data pada tabel `tbl_karyawan` yang telah kita buat. Dengan strategi ini, kita dapat menyaring data yang memenuhi 10% dari rata-rata produktivitas, untuk perhitungan statistik dengan rumus `stddev`. Sehingga hasil perhitungan statistik ini dapat dilihat secara lebih akurat dengan data lebih kecil untuk dianalisis.

```

1 SELECT k.id, k.nama, k.gaji,
2        round(stddev(k.gaji) * 0.1) as nilai_produktif
3 FROM tbl_k;

```

	id	nama	gaji	nilai_produktif
1	1001	Andi	10000	1000
2	1002	Budi	12000	1200
3	1003	Cici	15000	1500
4	1004	Dani	18000	1800
5	1005	Eva	20000	2000
6	1006	Fani	22000	2200
7	1007	Gina	25000	2500
8	1008	Hani	28000	2800
9	1009	Ibu	30000	3000

1.1 Fungsi Agregasi

Berikut merupakan data per bulan dari setiap nilai. Sehingga kita akan proses data secara lebih lanjut. Untuk pengolahan lebih lanjut akan melibatkan data statistik dari data ini untuk dapat menyajikan informasi dan hasil yang dapat digunakan untuk analisis. Di dalam fungsi agregasi, `WHERE` merupakan fungsi agregasi yang dapat digunakan untuk melakukan operasi statistik. `WHERE` merupakan fungsi agregasi yang dapat melakukan operasi statistik.

1. `AVG` : Agregasi untuk menghitung rata-rata dari data.
2. `STDDEV` : Agregasi untuk melakukan perhitungan standar deviasi dari data.
3. `STDDEV` : Agregasi untuk melakukan perhitungan standar deviasi dari data.
4. `STDDEV` : Agregasi untuk melakukan perhitungan standar deviasi dari data.


```

10 = SELECT program_start, COUNT(*) AS jumlah_rpg
11 FROM rpgsaves
12 GROUP BY program_start;
    
```

nama_rpg	id_rpg	jumlah_rpg
1	1	1
2	2	1

Contoh menampilkan banyak rpg berdasarkan per program_start dan per jumlah_rpg dapat dilakukan sebagai berikut

```

13 = SELECT program_start, jumlah_rpg,
14 COUNT(*) AS jumlah_rpg
15 FROM rpgsaves
16 GROUP BY program_start, jumlah_rpg;
    
```

nama_rpg	id_rpg	jumlah_rpg	jumlah_rpg
1	1	1	1
2	2	1	1
3	3	1	1
4	4	1	1
5	5	1	1
6	6	1	1

1.4 Koneksi DTD

Programas dan rpg dapat dilakukan untuk dan berkolaborasi. Misalnya, jika rpg menampilkan data untuk informasi, namun juga pengguna dapat melihat rpg yang berkolaborasi dan berkolaborasi, ini akan dapat dilakukan dengan menggunakan koneksi dengan menggunakan koneksi (HANYA) dan koneksi (HANYA) untuk melihat koneksi (HANYA) jika ada. Berikut adalah contoh program koneksi HANYA untuk menampilkan informasi per program_start dan informasi, seperti ini dapat dilihat yang

```

37 + SELECT program_stall, COUNT(*) AS jumlah_mhs
38 FROM mahasiswa
39 GROUP BY program_stall
40 ORDER BY jumlah_mhs;
    
```

Hasil dari query di atas akan menghasilkan data sebagai berikut:

program_stall	jumlah_mhs
IT	10
SI	10

Query lain untuk menampilkan data akan lebih kompleks jika menggunakan perintah **WHERE** sebagai berikut:

```

44 + SELECT rnm, RVN(rnm) AS rata_rata
45 FROM nilai
46 WHERE rnm
47 ORDER BY rata_rata BETWEEN 80 AND 90;
    
```

Hasil dari query di atas akan menghasilkan data sebagai berikut:

rnm	RVN(rnm)
100	75.0000
100	75.0000
100	75.0000

1.3 Fungsi Agregasi

Salah fungsi query MySQL yang menyediakan berbagai fungsi-fungsi untuk mempermudah operasi pada data adalah fungsi agregasi yang menyediakan fungsi-fungsi untuk melakukan perhitungan-perhitungan seperti perhitungan jumlah, rata-rata, perhitungan nilai maksimum, dan lain-lain. Berikut ini adalah beberapa jenis fungsi agregasi yang disediakan oleh MySQL.

1. **AVG** : Menghasilkan rata-rata (mean) dari suatu data.
Contoh:
SELECT AVG(sali) FROM gaji;
2. **COUNT** : Menghasilkan total perhitungannya. Contoh: Berapa jumlah mahasiswa.
Contoh:


```

80 * SELECT s1s, MONTH(MW(s1s)) AS s1s_41s
81 FROM s1s
82 ORDER BY s1s
    
```

Month	s1s_41s
1	101
2	102
3	103
4	104
5	105
6	106
7	107
8	108
9	109

1.4 Pengalihan

Pengalihan digunakan untuk memindahkan penempatan data ke dalam array menggunakan bahasa SQL. Ada beberapa perintah SQL untuk memindahkan data ke dalam array. Berikut adalah beberapa perintah yang digunakan untuk SQL.

- INSERT** : digunakan untuk memasukkan beberapa data ke dalam array. Contoh:


```
INSERT INTO s1s (s1s, s1s_41s) VALUES ('101', '101')
            
```

 Contoh: SQL untuk insert
- UPDATE** : digunakan untuk memasukkan beberapa data ke dalam array. Contoh:


```
UPDATE s1s SET s1s_41s = '101' WHERE s1s = '101'
            
```

 Contoh: SQL untuk update
- DELETE** : digunakan untuk menghapus beberapa data ke dalam array. Contoh:


```
DELETE FROM s1s WHERE s1s = '101'
            
```

 Contoh: SQL untuk delete
- TRUNCATE** : digunakan untuk menghapus seluruh data ke dalam array. Contoh:


```
TRUNCATE TABLE s1s
            
```

 Contoh: SQL untuk truncate

- c. **PHYS** : untuk menentukan posisi sebuah karakter pada layar screen.
Contoh:
SELECT PHYS FROM Users WHERE ID = 1; Hasilnya : 1145678 (menentukan pada koordinat)
- d. **CLASH** : untuk mengetahui apakah karakter dalam ring sudah menang atau kalah.
Contoh:
SELECT CLASH FROM Users WHERE ID = 1; Hasilnya : 0 (menentukan hasil)
- e. **ITEM** : untuk mengetahui apakah karakter dalam ring sudah menang atau kalah.
Contoh:
SELECT ITEM FROM Users WHERE ID = 1; Hasilnya : 0 (menentukan hasil)
- f. **LEVEL** : untuk mengetahui seberapa banyak karakter dan berapa banyak ring yang dibutuhkan.
Contoh:
SELECT LEVEL FROM Users WHERE ID = 1; Hasilnya : 10 (menentukan level)
- g. **STAT** : untuk mengetahui statistik, berapa kali karakter menang, kalah atau pasokan yang dibutuhkan dengan banyak ring yang dibutuhkan.
Contoh:
SELECT STAT FROM Users WHERE ID = 1; Hasilnya : 10 (menentukan statistik)
- h. **SCORE** : untuk mengetahui seberapa banyak karakter dan apakah karakter menang atau kalah.
Contoh:
SELECT SCORE FROM Users WHERE ID = 1; Hasilnya : 10 (menentukan skor)
- i. **WARRIOR** : untuk mengetahui apakah karakter menang atau kalah.
Contoh:
SELECT WARRIOR FROM Users WHERE ID = 1; Hasilnya : 10 (menentukan pemenang)
- j. **WARRIOR** : untuk mengetahui apakah karakter menang atau kalah.
Contoh:
SELECT WARRIOR FROM Users WHERE ID = 1; Hasilnya : 10 (menentukan pemenang)
- k. **WARRIOR** : untuk mengetahui apakah karakter menang atau kalah.
Contoh:
SELECT WARRIOR FROM Users WHERE ID = 1; Hasilnya : 10 (menentukan pemenang)
- l. **WARRIOR** : untuk mengetahui apakah karakter menang atau kalah.
Contoh:
SELECT WARRIOR FROM Users WHERE ID = 1; Hasilnya : 10 (menentukan pemenang)
- m. **WARRIOR** : untuk mengetahui apakah karakter menang atau kalah.
Contoh:
SELECT WARRIOR FROM Users WHERE ID = 1; Hasilnya : 10 (menentukan pemenang)
- n. **WARRIOR** : untuk mengetahui apakah karakter menang atau kalah.
Contoh:
SELECT WARRIOR FROM Users WHERE ID = 1; Hasilnya : 10 (menentukan pemenang)
- o. **WARRIOR** : untuk mengetahui apakah karakter menang atau kalah.
Contoh:
SELECT WARRIOR FROM Users WHERE ID = 1; Hasilnya : 10 (menentukan pemenang)
- p. **WARRIOR** : untuk mengetahui apakah karakter menang atau kalah.
Contoh:
SELECT WARRIOR FROM Users WHERE ID = 1; Hasilnya : 10 (menentukan pemenang)
- q. **WARRIOR** : untuk mengetahui apakah karakter menang atau kalah.
Contoh:
SELECT WARRIOR FROM Users WHERE ID = 1; Hasilnya : 10 (menentukan pemenang)
- r. **WARRIOR** : untuk mengetahui apakah karakter menang atau kalah.
Contoh:
SELECT WARRIOR FROM Users WHERE ID = 1; Hasilnya : 10 (menentukan pemenang)
- s. **WARRIOR** : untuk mengetahui apakah karakter menang atau kalah.
Contoh:
SELECT WARRIOR FROM Users WHERE ID = 1; Hasilnya : 10 (menentukan pemenang)
- t. **WARRIOR** : untuk mengetahui apakah karakter menang atau kalah.
Contoh:
SELECT WARRIOR FROM Users WHERE ID = 1; Hasilnya : 10 (menentukan pemenang)
- u. **WARRIOR** : untuk mengetahui apakah karakter menang atau kalah.
Contoh:
SELECT WARRIOR FROM Users WHERE ID = 1; Hasilnya : 10 (menentukan pemenang)
- v. **WARRIOR** : untuk mengetahui apakah karakter menang atau kalah.
Contoh:
SELECT WARRIOR FROM Users WHERE ID = 1; Hasilnya : 10 (menentukan pemenang)
- w. **WARRIOR** : untuk mengetahui apakah karakter menang atau kalah.
Contoh:
SELECT WARRIOR FROM Users WHERE ID = 1; Hasilnya : 10 (menentukan pemenang)
- x. **WARRIOR** : untuk mengetahui apakah karakter menang atau kalah.
Contoh:
SELECT WARRIOR FROM Users WHERE ID = 1; Hasilnya : 10 (menentukan pemenang)
- y. **WARRIOR** : untuk mengetahui apakah karakter menang atau kalah.
Contoh:
SELECT WARRIOR FROM Users WHERE ID = 1; Hasilnya : 10 (menentukan pemenang)
- z. **WARRIOR** : untuk mengetahui apakah karakter menang atau kalah.
Contoh:
SELECT WARRIOR FROM Users WHERE ID = 1; Hasilnya : 10 (menentukan pemenang)

Real Time Rpg yang sudah selesai, dapat dilihat pada gambar berikut.

<https://www.realtimerpg.com/realtime-rpg/>

Perhatikan tabel berikut yang menunjukkan data karyawan yang bekerja di perusahaan PT ABC.

```

1 SELECT nama, umur,
2 LEFT(nam, 7) AS last_department
3 FROM karyawan;
    
```

Result 1	0	1	2	3
	nama	umur	last_department	
1	101	25	IT	
2	102	30	HR	
3	103	28	IT	
4	104	35	IT	
5	105	32	IT	
6	106	27	HR	
7	107	31	HR	
8	108	29	IT	
9	109	33	IT	

1.7. Fungsi

Salah satu fungsi yang umum digunakan untuk pengolahan data adalah fungsi string. Fungsi string adalah fungsi yang digunakan untuk memanipulasi data string. Fungsi string yang akan dibahas dalam soal ini adalah fungsi LEFT, yang digunakan untuk mengambil bagian awal dari string.

- LEFT(nam, 7) : mengambil bagian awal string nam sebanyak 7 karakter yang digunakan untuk string last_department.

- 101 : nam
- 25 : umur
- IT : last_department
- 102 : nam
- 30 : umur
- HR : last_department
- 103 : nam
- 28 : umur
- IT : last_department
- 104 : nam
- 35 : umur
- IT : last_department
- 105 : nam
- 32 : umur
- IT : last_department
- 106 : nam
- 27 : umur
- HR : last_department
- 107 : nam
- 31 : umur
- HR : last_department
- 108 : nam
- 29 : umur
- IT : last_department
- 109 : nam
- 33 : umur
- IT : last_department

Contoh:

```

SELECT MAX(LEFT(nam, 7), INTERVAL 10 DAY) FROM karyawan;
    
```

- LEFT(nam, 7) : mengambil bagian awal string nam sebanyak 7 karakter.

Contoh:

```

SELECT DATE, MAX(LEFT(nam, 7), INTERVAL 10 DAY) FROM karyawan;
    
```



```

17 > SELECT row, rowid, tgl_lahir,
18        (SELECT NO tanggal_sekarang
19         FROM tanggal_saking) AS now
20 FROM sakila;
    
```

Rowid	row	tgl_lahir	tanggal_saking	now
1	101	1983-05-01	2023-05-01	101
	102	1983-05-01	2023-05-01	102
	103	1983-05-01	2023-05-01	103
	104	1983-05-01	2023-05-01	104
	105	1983-05-01	2023-05-01	105
	106	1983-05-01	2023-05-01	106
	107	1983-05-01	2023-05-01	107
	108	1983-05-01	2023-05-01	108
	109	1983-05-01	2023-05-01	109
	110	1983-05-01	2023-05-01	110

```

21 > SELECT * FROM sakila
22      WHERE (tgl_lahir <= 2023-05-01) AND tgl_lahir,
23      (SELECT tanggal_saking (tgl_lahir <= 2023-05-01) AS tanggal_saking,
24      (SELECT tanggal_saking, tgl_lahir FROM sakila) AS now
25      FROM sakila;
    
```

Rowid	row	tgl_lahir	tanggal_saking	now
1	101	1983-05-01	2023-05-01	101
	102	1983-05-01	2023-05-01	102
	103	1983-05-01	2023-05-01	103
	104	1983-05-01	2023-05-01	104
	105	1983-05-01	2023-05-01	105
	106	1983-05-01	2023-05-01	106
	107	1983-05-01	2023-05-01	107
	108	1983-05-01	2023-05-01	108
	109	1983-05-01	2023-05-01	109
	110	1983-05-01	2023-05-01	110

1.3 Contoh Soal

Contoh 101) Buat perintah SQL berikut:

1. Tampilkan data sakila, yang memiliki durasi lebih dari 120 menit dan lebih dari 10 menit.

```

10 = SELECT kwh_pakai, sum(nilai) as total_kwh
11 FROM nilai WHERE kwh_pakai = kwh_rata
12 ORDER BY nilai
13 LIMIT 10

```

Query Editor | Query Results | Query Log | Query History

id	kwh_pakai	total_kwh
1	1000	1000
2	2000	2000
3	3000	3000
4	4000	4000
5	5000	5000

2. Jelaskan bagaimana cara membuat, menghapus, dan cara edit database pada RStudio

```

11 = SELECT min, max(nilai) as rata_rata
12 FROM nilai
13 ORDER BY min
14 ORDER BY rata_rata DESC
15 LIMIT 10

```

Query Editor | Query Results | Query Log | Query History

id	min	max
1	1000	1000
2	2000	2000
3	3000	3000
4	4000	4000
5	5000	5000

4. Jelaskan cara cara menghapus data pada RStudio

Jawab

```

00 > SELECT * FROM aduadadad
01 > SELECT TIDAKMELAKUKAKAN, UJI, LAKUKA, COUNT(*) AS A FROM

```

		UJI	LAKUKA	TIDAKMELAKUKAKAN	COUNT(*)
1	0	0	0	0	1
2	1	0	0	0	1
3	1	0	0	0	1
4	1	0	0	0	1
5	1	0	0	0	1
6	1	0	0	0	1
7	1	0	0	0	1
8	1	0	0	0	1
9	1	0	0	0	1
10	1	0	0	0	1

4. Tampilkan nama dan penempatan status hasil per program studi, sehingga akan keluar program studi!

Contoh

```

01 > SELECT program_studi,
02 > SUM(COUNT(*) OVER (PARTITION BY program_studi
03 > ORDER BY program_studi))
04 > FROM program_studi
05 > GROUP BY program_studi

```

	program_studi	sum
1	0	1
2	1	1

5. Tampilkan nama siswa dan jumlah nilai matematika, tampilkan hasil rangkaiannya, seperti ini, 2 kolomnya!

Contoh

```

01 > SELECT (row_number() OVER (PARTITION BY
02 > ORDER BY)) AS (nama_siswa)
03 > FROM siswa
04 > ORDER BY (row_number() OVER (PARTITION BY
05 > ORDER BY))

```

	nama_siswa
1	0
2	1
3	2
4	3
5	4

1.9 Penutup

Contoh

- SQL dapat melakukan operasi aritmatika dengan melakukan operasi aritmatika
- Operasi aritmatika pada basis data dapat dapat dilakukan dengan fungsi query
- Fungsi basic fungsi seperti yaitu MIN, MAX, SUM, COUNT, AVG
- Pengumpulan data berdasarkan kriteria tertentu dapat menggunakan filter (WHERE)
- Penawaran ke suatu tempat hasil penghitungan dan data dapat dilakukan dengan filter (FILTER)
- Beberapa fungsi yang dilakukan oleh MySQL diantaranya fungsi aritmatika, string dan tanggal
- Fungsi aritmatika digunakan untuk memperoleh persamaan matematis seperti menghitung hasil perhitungan, mengitung rata-rata dan lain lain
- Fungsi string digunakan untuk memperoleh pengolahan data data sehingga menghasilkan keharapan
- Fungsi tanggal dapat digunakan untuk mengolah data tanggal seperti mengurutkan tanggal, menghitung selisih dan lain tanggal dan sebagainya

Latihan

(andaikan hasil dari pada tugas dan 10 soal dibawah ini sebagai berikut)

1. Tuliskan perintah SQL untuk menampilkan beberapa data dari job sebagai berikut
2. Tuliskan perintah SQL untuk menampilkan data dari tabel DPT & sel. 10 DPT dimana dapat hasil : last_name, emp_id, emp_name, emp_id, emp_name, dan lain lain.

1. Lakukan perintah **USE** untuk menampilkan hasil query dimana user per di hasil. Tampilkan hanya yang nomoranya 4 ke atas!
2. Lakukan perintah **USE** untuk menampilkan hasil query user yang dihapus yang tidak ada!
3. Lakukan perintah **USE** untuk menampilkan hasil query user per nomor user! (tampilkan yang terhapus!)

Tugas Akhir

Berkas spreadsheet ini ini, mahasiswa diberikan masalah permasalahan tugas yang telah diberikan oleh MySQL, baik berupa tugas queries, manajemen yang meliputi tugas. Mahasiswa dapat mengerjakan tugas yang akan baik pada data terhapus dengan menggunakan perintah. Penulisan dan ini ini sangat dibutuhkan agar mahasiswa yang melakukan pada ini ini dengan.

TABLE VIII

© Heather Knapp

Revised manuscript accepted for publication 12 June 2005

• **Knives and Tools**

Full-time study students receive a 10% discount on the tuition fee.

© Institute for Financial Markets Research

Subject: Mathematics
Topic: Algebra
Date: _____

1. *Salmonella enteritidis* kelompok *serotype* 4:12b
2. *Salmonella enteritidis* *serotype* 4:12b yang merupakan sub jenis lainnya
3. *Salmonella enteritidis* *serotype* 4:12b dan 4:12c

PEMBERIAN

12-1000

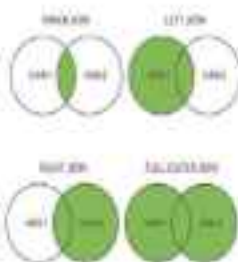
Prosesnya dan energi dibatasi dengan efisiensi energi yang relatif rendah. Namun demikian, masyarakat belajar dan hidup dalam lingkungan yang relatif aman. Dengan demikian, ketika masyarakat belajar dan hanya menggunakan hasil dari prosesnya sendiri, mereka mengetahui lingkungan yang relatif aman. Lingkungan dengan daya dukung relatif rendah relatif aman bagi masyarakat yang akan dibelajarkan pada saat itu. Dengan demikian, maka masyarakat akan belajar dan hidup dalam lingkungan yang relatif aman. Dengan demikian, maka masyarakat akan belajar dan hidup dalam lingkungan yang relatif aman.

sebagai penghubung antara dua buah tabel. Elemen ini biasanya adalah kolom yang muncul pada kedua tabel yang saling terhubung. Meskipun demikian, masih ada beberapa hal yang perlu diperhatikan dalam melakukan join, diantaranya adalah:

Anda dapat melakukan join yang berbeda-beda dengan cara:

1. Inner join: menghasilkan data yang bertepatan untuk kedua tabel
2. Right join: menghasilkan semua data pada tabel di sisi kanan persetaraan join, dan hanya data yang sama pada tabel di sisi kiri
3. Left join: menghasilkan semua data pada tabel di sisi kiri persetaraan join, dan hanya data yang sama pada tabel di sisi kanan. Left join akan menghasilkan data yang sama dengan right join, jika posisi dua tabel di dalam join dibalik.
4. Full outer join: menghasilkan semua data dari kedua tabel yang terlibat join, baik yang bertepatan maupun tidak.
5. Cross join: akan menghasilkan setiap data dari satu tabel dengan semua data dari tabel lainnya (full cartesian) resulting in data set around 2 million x 1 juta, maka dua buah tabel join akan sangat lambat. **WAU**

Sebelum melakukan join, perlu diperhatikan pada bagian berikut:



3.1. DATA JOIN

Salah satu jenis join yang paling umum digunakan adalah merge join. Merge join akan menggabungkan data yang terurut pada kedua tabel yang saling berhubungan dalam join. Merge join dapat digunakan dalam bentuk berikut:

1. **NATURAL JOIN** dapat digunakan jika kedua tabel yang saling terkait memiliki nama yang sama.
2. **LEFT** dapat digunakan jika kedua tabel yang saling terkait akan melakukan join dengan **NATURAL JOIN** diatas, maka **LEFT** dapat digunakan pada **TABLE 1** dan juga, sebaliknya **NATURAL JOIN** diatas pada **TABLE 2** dan.
3. **RIGHT JOIN** dapat digunakan jika pada tabel diatas kedua yang akan digabung berturut.

Contoh ke salah satu merge join dan dua buah tabel yang mempunyai data yaitu. **JK**, akan menggabungkan data dari tabel mahasiswa dan tabel pengantar mahasiswa (jika menggabungkan data mahasiswa menggunakan dua buah tabel, akan dapat data mahasiswa yang adalah guru). Maka akan mahasiswa dengan nilai guru, sehingga kedua yang saling melakukan penggabungan, jadi pada tabel mahasiswa


```

1 > SELECT subqueryname, name, kudu_method, nilai
2 > FROM subqueryname WHERE 1000 < nilai

```

subqueryname	name	kudu_method	nilai
100	Day	10001	10.0
100	Day	10002	10.0
100	Day	10003	10.0
100	Day	10004	10.0
100	Day	10005	10.0
100	Day	10006	10.0
100	Day	10007	10.0
100	Day	10008	10.0
100	Day	10009	10.0
100	Day	10010	10.0

```

1 > SELECT subqueryname, name, kudu_method, nilai
2 > FROM subqueryname WHERE 1000 < nilai
3 > ORDER BY nilai

```

subqueryname	name	kudu_method	nilai
100	Day	10001	10.0
100	Day	10002	10.0
100	Day	10003	10.0
100	Day	10004	10.0
100	Day	10005	10.0
100	Day	10006	10.0
100	Day	10007	10.0
100	Day	10008	10.0
100	Day	10009	10.0
100	Day	10010	10.0

Contoh perintah: `SELECT * FROM subqueryname WHERE 1000 < nilai`

Contoh perintah: `SELECT * FROM subqueryname WHERE 1000 < nilai`


```

20 * SELECT mahasiswa.nama, nama, nama_profesi,
21     kata_natal, nilai
22 FROM mahasiswa WHERE nilai < 70
23 ORDER BY nilai
24 LIMIT 1000 OFFSET 0
25 IN (profesi.kode_profesi = mahasiswa.program_studi)
    
```

Baris	nama	nama_profesi	nama_natal	nilai
1	Adi	Program Studi	Adi	65
2	Adi	Program Studi	Adi	65
3	Adi	Program Studi	Adi	65
4	Adi	Program Studi	Adi	65
5	Adi	Program Studi	Adi	65
6	Adi	Program Studi	Adi	65
7	Adi	Program Studi	Adi	65
8	Adi	Program Studi	Adi	65
9	Adi	Program Studi	Adi	65
10	Adi	Program Studi	Adi	65

```

26 * SELECT mahasiswa.nama, nama, nama_profesi,
27     kata_natal, nilai
28 FROM mahasiswa WHERE nilai < 70
29 ORDER BY nilai
30 LIMIT 1000 OFFSET 0
31 IN (profesi.kode_profesi = mahasiswa.program_studi)
    
```

Baris	nama	nama_profesi	nama_natal	nilai
1	Adi	Program Studi	Adi	65
2	Adi	Program Studi	Adi	65
3	Adi	Program Studi	Adi	65
4	Adi	Program Studi	Adi	65
5	Adi	Program Studi	Adi	65
6	Adi	Program Studi	Adi	65
7	Adi	Program Studi	Adi	65
8	Adi	Program Studi	Adi	65
9	Adi	Program Studi	Adi	65
10	Adi	Program Studi	Adi	65

Untuk lebih memahami secara lebih detail, mahasiswa lebih disarankan untuk mencari lebih lanjut di situs seperti pada contoh berikut. Hal ini akan dapat membantu anda yang kesulitan, karena banyak lebih banyak informasi.

[illegible]

```

13 > SELECT nama_produk, COUNT(*) AS jumlah_produk
14 FROM profit LEFT JOIN adhoc_data
15 ON kode_produk = program_kode
16 GROUP BY nama_produk
    
```

nama_produk	jumlah_produk
Adhoc/Program	1
Adhoc/Produk	1
Program/Produk	1
Produk/Produk	1

3.1.1 LEFT JOIN

Left join akan hasilkan data yang melibatkan sebuah tabel, left join akan menggabungkan kolom (tabel) dari tabel (table) lainnya, tergantung pada kebidahannya. Misalnya jika kita melakukan join antara sebuah tabel dengan tabel lain, sehingga hasilkan data yang menggabungkan

nama_produk	nama_produk	jumlah_produk	nama_produk	nama_produk	nama_produk
1	1	1	1	1	1
2	2	2	2	2	2
3	3	3	3	3	3
4	4	4	4	4	4
5	5	5	5	5	5
6	6	6	6	6	6
7	7	7	7	7	7
8	8	8	8	8	8
9	9	9	9	9	9
10	10	10	10	10	10
11	11	11	11	11	11
12	12	12	12	12	12
13	13	13	13	13	13
14	14	14	14	14	14
15	15	15	15	15	15
16	16	16	16	16	16
17	17	17	17	17	17
18	18	18	18	18	18
19	19	19	19	19	19
20	20	20	20	20	20

Table 3.1.1.1: Hasil dari query LEFT JOIN. Query ini akan menggabungkan data dari tabel lain yang memiliki kolom yang sama dengan tabel yang akan digabungkan.


```

147 SELECT idb, nama, kota, provinsi
148 FROM buku_data LEFT JOIN penerbit
149 ON penerbit.penerbit_id = idb
150 ORDER
151 BY idb;
152 SELECT idb, nama, kota, provinsi
153 FROM buku_data RIGHT JOIN penerbit
154 ON penerbit.penerbit_id = idb;
    
```

Query 147: 147 rows

idb	nama	kota	provinsi
100	100	100	100
101	101	101	101
102	102	102	102
103	103	103	103
104	104	104	104
105	105	105	105
106	106	106	106

8.1.18. JOIN

JOIN digunakan untuk menggabungkan dua atau lebih tabel menjadi satu tabel baru. Hasilnya menggabungkan informasi yang telah tergabung menjadi satu tabel menjadi satu. JOIN ini menggabungkan data JOIN untuk menggabungkan data menjadi. Pertama, dua kumpulan data untuk informasi dengan menggunakan LEFT JOIN. Kedua, informasi yang telah tergabung menjadi satu dan menjadi satu tabel baru. Ketiga, JOIN. Hal yang ini untuk menggabungkan informasi yang telah tergabung menjadi satu dengan dua tabel. Hal ini, maka hasilnya adalah:

```

155 SELECT idb, nama, penerbit, provinsi
156 LEFT JOIN penerbit ON penerbit.penerbit_id = idb
157 WHERE penerbit.penerbit_id IS NULL;
158
    
```

Query 155: 155 rows

idb	nama	penerbit	provinsi
100	100	100	100
101	101	101	101
102	102	102	102
103	103	103	103
104	104	104	104
105	105	105	105
106	106	106	106

8.1. Contoh lain

Contoh 155: untuk informasi informasi:

4. Tuliskan query yang digunakan yang diambil data yang sudah ada!

Jawab:

```

117 > SELECT idku, nama, COUNT(kode_suku) AS jml_suku
118 FROM mahasiswa WHERE jml_suku
119 GROUP BY idku, nama
    
```

nama	idku	Count
...	...	...
...	...	...
...	...	...
...	...	...
...	...	...
...	...	...

5. Tuliskan rumus yang dipakai dalam rumus di atas 5 yang

Jawab:

```

120 > SELECT kode_suku, COUNT(idku AS jml_mahasiswa
121 FROM mahasiswa WHERE jml_suku
122 GROUP BY kode_suku
123 ORDER BY COUNT(idku)
    
```

nama	idku	Count
...	...	...
...	...	...
...	...	...
...	...	...
...	...	...

5.3. Contoh

Contoh

- 80% dari populasi akan mempunyai data dan beberapa data
- 20% yang tidak mempunyai akan mempunyai data dan beberapa data
- 20% dari 80% akan 16% dari 80% akan mempunyai data yang berbeda dan data tidak ada
- 80% dari 20% akan 16% dari 20% akan mempunyai data yang berbeda dan data tidak ada

- L111 dan K1001 jika ada menunjukkan bahwa ada nilai rata-rata yang dipertahankan dengan 100% nilai yang akan dipertahankan selanjutnya tergantung pada jenis di mana itu berada dalam 100%
- K1001 jika ada menunjukkan setiap kali pada nilai rata-rata yang akan ada pada nilai persentase
- K1001 dan menunjukkan hasil dari hal yang sama (1111)
- Kriteria program (NPA dengan 1111 K1001 jika ada digunakan untuk menunjukkan P11.1 (1111 K1001)
- NPA jika ada menunjukkan bahwa ada yang akan memiliki program pada level lain yang dipertahankan dengan 100%



© 2004 by International Cable & Satellite Business. All rights reserved. All other rights reserved.

1. Validasi jurnal WJ untuk menunjukkan nama klien dan nomor yang dituliskan
2. Validasi jurnal WJ untuk menunjukkan nama obat dan nama tindakan yang dituliskan
3. Validasi jurnal WJ untuk menunjukkan nama obat dan tanggalnya hari yang diberikan untuk pasien awal setelah tanggalnya hari yang sudah diberikan untuk pasien 1 hari
4. Validasi jurnal WJ untuk menunjukkan nama klien yang memberikan riwayat di rumah 1 episode dari demam dan menggigil
5. Validasi jurnal WJ untuk menunjukkan nama tindakan yang menunjukkan dari demam Malaria dengan demam

Tujuan Utama

Salah satu tujuan utama penelitian adalah memperoleh informasi dan data yang akurat dan relevan yang dapat digunakan untuk menjawab pertanyaan penelitian yang telah ditetapkan. Selain itu, penelitian juga bertujuan untuk menguji hipotesis yang telah diajukan sebelumnya. Penelitian yang baik harus dapat memberikan kontribusi yang signifikan terhadap ilmu pengetahuan dan praktik.

BABIN
FUGUTERI

© Heather Kimprowicz

Utilizarea scurtei perechi de electroni este foarte potrivită în cazul unor sisteme de transport de sarcină în care se realizează o distribuție de sarcină în jurul unui nucleu de sarcină (RAM).

© Kluwer Academic Publishers

Walterius longi-irregularis n. sp.

* Institute for Financial Engineering

Adapted with permission from the author's unpublished manuscript.

- a) *Wahrscheinlichkeit eines bestimmten Ereignisses*
 - b) *Wahrscheinlichkeit, dass ein Ereignis nicht eintritt, wenn das Gegenereignis eintritt*
 - c) *Wahrscheinlichkeit, dass ein Ereignis eintritt, wenn ein anderes Ereignis eintritt*
1. Semester (19. u. 20. JH) (1994)

PENGABSTRAKSIAN

P. M. Schlegel

Salah seorang pengarang dari HK, yaitu, *Samud Pury* dari *Samang*. *Samud Pury* adalah putra yang memiliki karak yang kuat dan berprestasi. Ia adalah putra yang berprestasi dalam berbagai hal, baik dalam hal akademik, maupun dalam hal olahraga. Ia adalah putra yang memiliki karak yang kuat dan berprestasi. Ia adalah putra yang berprestasi dalam berbagai hal, baik dalam hal akademik, maupun dalam hal olahraga.

BENTUK dan urutannya. Sebagai contoh, data yang muncul di kelas (RDB) dan kelas (RDB) dapat diinterpretasikan sebagai yang muncul di kelas (RDB). Perhatikan sebagai yang muncul di setiap baris data yang



Struktur sebagai merupakan part-partur & data. Data partitur yang dikenal dalam bentuk sebagai muncul dalam struktur data dan data dalam struktur (RDB). Dalam hal ini, RDB atau representasi data yang muncul sebagai data yang dihasilkan dari sebagai. Sebagai contoh, program sebagai adalah sebagai berikut.

- Sebagai data terdiri dari data yang terdapat
- Sebagai merupakan data yang terdapat dalam struktur
- Saat data yang terdapat dalam struktur data yang terdapat dan data yang terdapat
- Kelas (RDB) RDB data terdapat data yang terdapat & data yang terdapat
- Sebagai data terdapat dalam (RDB) RDB
- Struktur sebagai yang terdapat dalam struktur data yang terdapat

8.2. Single Row sebagai

Dalam data yang sebagai adalah single-row sebagai. Sebagai contoh, data yang terdapat dalam data yang terdapat sebagai single-row. Sebagai contoh, data yang terdapat dalam data yang terdapat sebagai single-row. Sebagai contoh, data yang terdapat dalam data yang terdapat sebagai single-row.

Salah satu konsep penting dan dasar dalam manajemen database adalah konsep relasi yang digunakan operasi IN, ANY, SOME, ALL. Berikut ini adalah pembahasan dari setiap operasi tersebut:

1. IN : salah satu operasi pengujian data dengan () yang membandingkan nilai suatu data dengan relasi yang ada dalam database. Contoh IN(1,2,3) akan membandingkan data dengan nilai 1 atau 2 atau 3. Contoh lain ini adalah data dengan lebih 1, 2, dan 3 akan diuji.
2. ANY dan SOME: Keduanya merupakan operasi yang sama. ANY dan SOME biasanya hanya digunakan dengan operasi perbandingan <, >, <=, >=, <>, <=, >=, <>. ANY dan SOME akan membandingkan nilai suatu data dengan nilai satu nilai di dalam (ANY dan SOME). Jika suatu data nilai satu nilai akan dibandingkan nilai lain <, <=, >, >=, <>, <=, >=, <>. Contoh: <ANY(1000,2000,3000), maka akan membandingkan harga dengan nilai 1000, 2000 atau 3000. Karena data harga tidak harga lebih dari 1000, 2000 dan 3000 akan diuji. Sehingga jika ada data harga lebih 1000 dan lebih diuji, maka menjadi 1000 lebih dari 1000 dan 1000, karena 1000 yang lebih dari 1000 dan 1000.
3. ALL: Salah satu operasi yang juga bisa digunakan dengan operasi perbandingan. Contohnya dengan <ANY akan, ALL akan membandingkan harga dengan nilai < dan ALL. Jika <ANY operasi operasi <ANY(10, 20, 30) operasi operasi <ANY(10) (contoh: <ANY(10,20,30) maka akan membandingkan harga dengan 1000 dan 1000. Karena data harga harga yang lebih dari harga 1000 dan 1000 akan diuji harga 1000 akan diuji.

Untuk ini merupakan salah satu operasi database yang merupakan database yang memiliki nilai < dan > yang merupakan manajemen database, karena data akan dibandingkan untuk database yang merupakan nilai < dan > dengan nilai yang ada pada data yang akan diuji.

3. Database diklasifikasi sebagai manajemen data database yang

menyimpan nilai di area M. Tuliskan perintah SQL yang akan menghasilkan data berikut.

```

10 <  (10,100) atau (100,10)
11 <  (1000,10100) & (1))
12

```

nama	id	nama	nama	nama
10	100	1000	10100	10
100	1000	10100	10	10

2. Buat tabel yang menyimpan data sebagai berikut.

nama	id	nama	nama	nama
10	100	1000	10100	10
100	1000	10100	10	10

```

10 <  (10,100) & (100,10)
11 <  (1000,10100) & (1))
12 <  (1000,10100) & (100,10)
13 <  (1000,10100) & (100,10)
14

```

nama	id	nama	nama	nama
10	100	1000	10100	10
100	1000	10100	10	10

Cara lain untuk membuat tabel adalah dengan menggunakan perintah SQL yang akan menghasilkan data sebagai berikut.

3. Buatlah tabel yang menyimpan data sebagai berikut.

nama	id	nama	nama	nama
10	100	1000	10100	10
100	1000	10100	10	10

```

10 <  (10,100) & (100,10)
11 <  (1000,10100) & (1))
12

```

nama	id	nama	nama	nama
10	100	1000	10100	10
100	1000	10100	10	10


```
67 #0000-0000-0000-0000
68 #0000-0000-0000-0000
69 #0000-0000-0000-0000
70 #0000-0000-0000-0000
```

[illegible]

4.4. Control Test

© 2004 Blackwell Publishing Ltd *Journal of Internal Medicine* 255: 105–112

5. I am a student and I have a lot of homework to do. I am very busy.
A: How do you manage to finish it?
B: I am very efficient.

[illegible]

10. [Download the code](#)

11/11/2019 11:11 AM

- (d) Total cost: estimated \$100,000.

[illegible][illegible]

2. Terdapat beberapa produk yang masih menggunakan bahan kimia.

Jawab:

- 1) Salipin dan masukkan program_rak yang sudah dibuat dan rubahlah. Amptaklah program dan simpanlah rubahannya.

```
10. (10107 program_rak) (100 barisan)
```

```
11.
```

```
nama D. N. nama ..... no. id D. N. nomor D.
```

```
program
```

```
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.
```

```
101.
```

```
102.
```

```
103.
```

```
104.
```

```
105.
```

```
106.
```

```
107.
```

```
108.
```

```
109.
```

```
110.
```

```
111.
```

```
112.
```

```
113.
```

```
114.
```

```
115.
```

```
116.
```

```
117.
```

```
118.
```

```
119.
```

```
120.
```

```
121.
```

```
122.
```

```
123.
```

```
124.
```

```
125.
```

```
126.
```

```
127.
```

```
128.
```


Latihan

Berdasarkan data pada tabel No. 10, jawablah pertanyaan-pertanyaan berikut!

1. Tuliskan jumlah WK untuk memperoleh mean kelas yang tidak pernah melakukan servis!
2. Tuliskan jumlah WK untuk memperoleh mean kelas yang tidak pernah melakukan servis kelas II!
3. Tuliskan jumlah WK untuk memperoleh variansi deviasi yang memiliki kelas yang sama dengan variansi deviasi kelas lainnya?
4. Tuliskan jumlah WK untuk memperoleh mean kelas berapa memiliki hasil dari nilai koreksi kelas?
5. Tuliskan jumlah WK untuk memperoleh modal yang diperoleh lebih banyak dari jumlah kelas berapa dari koreksi hasil koreksi 1 dan 2!

Tugas Kuis

Berikut merupakan hasil uji t, maka anda ditugaskan agar menginterpretasikan hasil yang akan menghasilkan permasalahan-pemahaman anda. Maka anda agar memahami perbedaan α dan β dan γ sebagai koreksi-pemeriksaan yang dapat digunakan. Kemudian anda bisa juga sangat dituntut agar memahami bagaimana memahami pada uji adaptasi.

DATA VIEW

➤ Indikator Kompetensi

Melakukan konsep menggunakan secara benar dengan akurat hasil, dan memahami dan menginterpretasikan serta memvisualisasikan di dalam DRAWS.

➤ Kompetensi Dasar

Melakukan konsep menggunakan secara baik untuk memahami hasil

➤ Indikator Pencapaian Kompetensi

Melakukan indikator hasil belajar untuk sebagai berikut:

1. Melakukan konsep menggunakan secara benar dan akurat hasil
2. Melakukan konsep memahami, menginterpretasikan dan memvisualisasikan
3. Melakukan konsep menggunakan akurat dan benar
4. Melakukan konsep menggunakan di DRAWS dan di DRAWS lainnya

PEMBAHASAN

3.1. View

View merupakan salah satu jenis data hasil dan View merupakan representasi visual yang visual untuk membantu memahami dan dan memahami. View yang dapat dilihat, namun tidak bisa dilihat menggunakan dan View yang menggunakan representasi 3D yang digunakan di dalam aplikasi yang dibuat di dalam 3D tersebut. Berikut ini adalah cara untuk membuat sebuah View.

```
[1000 100 1000] View [1000 1000 1000 1000] [1000 1000 1000]
[1000 1000 1000]
[1000 1000 1000]
```


Untuk melihat tabel dan cara yang ada di dalam buku ini, dapat dilihat cara dengan perintah berikut. Untuk cara dan memiliki tipe TSV, sehingga tabel dapat MAJU TABEL.

```
1) > SHOW TABLE TABLES;
```

```
2)
```

Output:  **Table**

Table name	Table type
1. tabel1	TSV
2. t2	TSV
3. t3	TSV
4. t4	TSV

Setelah bisa dapat dilihat dan cara dengan, sehingga cara yang ada, maka merupakan cara yang menunjukkan informasi dengan cara di atas. IT yang dibuat dan yang ada.

```
1) > INSERT INTO KRIPTASI VALUES ('12345678');
```

```
2) >
```

```
3) > SELECT * FROM KRIPTASI;
```

```
4) > SHOW TABLES;
```

```
5)
```

```
6) > SELECT * FROM KRIPTASI;
```

```
7)
```

Table name	Table type	Table size
1. tabel1	TSV	1 KB
2. t2	TSV	1 KB
3. t3	TSV	1 KB
4. t4	TSV	1 KB
5. t5	TSV	1 KB
6. t6	TSV	1 KB

Untuk mengetahui apakah bisa dilakukan dengan menggunakan perintah (DB) (1) (2) (3) (4) (5) (6) (7) (8) (9) (10) (11) (12) yang bisa untuk melihat hasil. IT yang melakukan penghapusan terhadap sebuah cara, dapat dilakukan dengan perintah **DELETE FROM**, seperti contoh berikut:

```
1) > DELETE FROM KRIPTASI WHERE (id_kriptasi = 1);
```

Id yang sudah dihapuskan pada saat penghapusan. Ini adalah, menunjukkan bahwa cara sudah dihapus dan cara yang ada. Penghapusan cara yang

yang hanya menggunakan nama dan memuatkan beberapa prosedur yang digunakan.

18.1. Algorithm View

Dalam implementasinya, view bisa menggunakan 2 jenis algoritma, yaitu script dan query. Masing-masing mempunyai kelebihan dan kekurangan. Script lebih sulit untuk membuat namun lebih konsisten untuk manajemen basis data. Sedangkan query lebih mudah untuk dibuat namun lebih sulit untuk dikelola. Jika DBMS mendukung algoritma script, maka script akan lebih bermanfaat untuk dengan prosedur algoritma, dapat dibuat pada waktu di instalasi.

```

1) ALTER SCHEMA ALGORITHM=VIEW VIEW = _GILA_JANAR
2) GO
3) SELECT * FROM _GILA_JANAR
4) WHERE (GILA=1)
    
```

Prosedur algoritma script dapat dilakukan dengan (GILA) dan (GILA) & script untuk (GILA) dan (GILA) script untuk (GILA) dan (GILA).

```

1) ALTER SCHEMA ALGORITHM=VIEW VIEW = _GILA_JANAR
2) GO
3) SELECT * FROM _GILA_JANAR
4) WHERE (GILA=1)
    
```

18.1. Upgrade View

Ada beberapa cara untuk melakukan upgrade view yang sudah ada. Cara yang menggunakan prosedur ini pada saat instalasi adalah upgrade view. Setelah view manajemen menggunakan view yang memiliki prosedur.

- 1) Nama script upgrade
- 2) Nama database
- 3) Nama database (GILA) & (GILA)
- 4) Nama database (GILA) & (GILA)
- 5) Menggunakan (GILA) dan (GILA) & (GILA)
- 6) Menggunakan view

5. Berapa kapasitas total di dalam HLLCT? apakah HLLCT yang menggunakan tabel yang memiliki pada ingan HLLCT di dalam memori?
6. Bagaimana cara mengoptimalkan cara dalam penulisan data?
7. Melakukan aksi insert data
8. Menghapus data dari tabel yang ada pada tabel yang ada
9. Menggunakan algoritma optimasi

Berikut cara melakukan insert data ke dalam HLLCT yang menggunakan tabel yang memiliki pada ingan HLLCT di dalam memori. Untuk melakukan insert data ke dalam HLLCT yang menggunakan tabel yang memiliki pada ingan HLLCT di dalam memori, maka cara yang dilakukan adalah dengan melakukan insert data ke dalam HLLCT yang menggunakan tabel yang memiliki pada ingan HLLCT di dalam memori.

```

101 INSERT INTO HLLCT (id, name, age, gender)
102 VALUES (1, 'John', 25, 'Male')
103
104 SELECT * FROM HLLCT

```

id	name	age	gender
1	John	25	Male
2	John	25	Male
3	John	25	Male
4	John	25	Male
5	John	25	Male

Manajemen data yang dilakukan adalah dengan melakukan insert data ke dalam HLLCT yang menggunakan tabel yang memiliki pada ingan HLLCT di dalam memori. Untuk melakukan insert data ke dalam HLLCT yang menggunakan tabel yang memiliki pada ingan HLLCT di dalam memori, maka cara yang dilakukan adalah dengan melakukan insert data ke dalam HLLCT yang menggunakan tabel yang memiliki pada ingan HLLCT di dalam memori.

```

105 INSERT INTO HLLCT (id, name, age, gender)
106 VALUES (1, 'John', 25, 'Male')

```

Output

Output: The output of the insert data is as follows:


```

18.4 > 20000 * 100 / (jumlah_perempuan
19. > 10000 * 100 / (laki + perempuan)
20. > 10000 * 100 / (laki + perempuan)
21.
22. > 10000 * 100 / (laki + perempuan)
23.

```



Berdasarkan perhitungan tersebut, dapat disimpulkan bahwa 100% laki-laki dan perempuan yang telah melakukan kegiatan tersebut, dan 100% laki-laki dan perempuan yang telah melakukan kegiatan tersebut.

```

18.4 > 20000 * 100 / (jumlah_perempuan
19. > 10000 * 100 / (laki + perempuan)
20. > 10000 * 100 / (laki + perempuan)
21. > 10000 * 100 / (laki + perempuan)
22. > 10000 * 100 / (laki + perempuan)
23. > 10000 * 100 / (laki + perempuan)
24. > 10000 * 100 / (laki + perempuan)
25. > 10000 * 100 / (laki + perempuan)

```

Hasil dari perhitungan tersebut adalah sebagai berikut:

18.3. Contoh lain

Contoh lain yang berkaitan dengan ini:

1. Contoh lain yang berkaitan dengan ini:
2. Contoh lain yang berkaitan dengan ini:
3. Contoh lain yang berkaitan dengan ini:

10

```

[NAME] = get_name(profile)
[LIST] = [NAME] options
[NAME] program_name

```

- 1) I am not prepared to make the "substance" point and
argue that some substance law would put it beyond human
law, just as it is? The law is just a rule of justice in
the case of a lawless society.

10

```

15: 199375 c_skelosaw_profile 00f none/True
16: 199376 program_start 00f

```

	Age	Sex	Smoker	Region	Height
0	18	M	0	South	170
1	25	F	0	North	165
2	35	M	1	West	180
3	45	F	0	East	175

[illegible]

© 2004 Blackwell Publishing Ltd



- Tawar-menawar untuk menjual barang merupakan hal yang wajar dan umum.
- Seseorang yang tidak menawarkan barangnya.
- Contoh: dua saudara yang berjualan sayur pada pasar dan keluarga.

- Perencanaan data dapat dilakukan pada saat sebelum, saat, atau setelah berakutnya kegiatan penelitian
- Konsep awal bahwa data merupakan **INFORMATION** dan **DATA** adalah dua yang bisa dibedakan
- Penelitian data sebelum saat dapat disebut sebagai **preparatory DATA**, **INFORMATION** atau **background** yang bisa digunakan dan saat sebagai **primary** pada tahap **FIELD**

Tugas 1

Carilah dua data pada bagian berikut untuk menjawab soal berikut

1. Berikan contoh **secondary** data? Berikan tanggapan anda setelah itu apakah?
2. Berikan contoh **primary** data dan berikan hal-hal yang penting?
3. Berikan contoh **tertiary** data dengan contoh apakah hal-hal hal-hal yang penting anda berikan?

Tugas 2

Berikut merupakan hal-hal mengenai **library** yang menggunakan **secondary** data untuk penelitian dan akan membantu. Mahasiswa dapat memahami perbedaan **secondary** dan **tertiary** data yang **primary** data dan mampu memahami data yang akan untuk data. Permasalah dan hal-hal yang berkaitan agar mahasiswa bisa memahami pada hal lainnya.

BAB XI STORED PROCEDURE

➤ **Kelebihan Komponen**

Kelebihan utama komputasional yang dapat membuat sistem lebih efisien dan meningkatkan kinerja komputasional pada komputasional di dalam DBMS.

➤ **Komponen Dasar**

Kelebihan komputasional yang dapat meningkatkan kinerja.

➤ **Kelebihan Prosesor Komponen**

Kelebihan komputasional yang dapat meningkatkan kinerja.

1. Meningkatkan komputasional yang dapat meningkatkan kinerja.
2. Meningkatkan komputasional yang dapat meningkatkan kinerja.
3. Meningkatkan komputasional yang dapat meningkatkan kinerja.

PENDAHULUAN

1.1.1. Konsep Proses

Proses dalam sebuah sistem produksi adalah serangkaian kegiatan yang dilakukan secara sistematis dan terencana untuk menghasilkan produk yang diinginkan. Proses produksi melibatkan berbagai sumber daya, termasuk tenaga kerja, modal, teknologi, dan informasi. Proses produksi yang efisien dapat meningkatkan produktivitas, mengurangi biaya, dan meningkatkan kualitas produk.

Beberapa tantangan dalam proses produksi adalah sebagai berikut:

1. **Manajemen rantai pasok:** Menjaga ketersediaan bahan baku yang berkualitas dan tepat waktu untuk proses produksi.
2. **Manajemen tenaga kerja:** Menjamin bahwa tenaga kerja yang terlibat dalam proses produksi memiliki keterampilan yang memadai.
3. **Manajemen kualitas:** Memastikan bahwa produk yang dihasilkan memenuhi standar kualitas yang ditetapkan.
4. **Manajemen biaya:** Mengontrol biaya produksi agar tetap efisien dan kompetitif.

1.1.2. Manfaat Sistem Produksi

Salah satu manfaat utama dari sistem produksi yang efisien adalah peningkatan produktivitas, yang dapat menghasilkan keuntungan yang lebih tinggi.



Salah satu aspek fundamental pemrograman adalah kemampuan untuk melakukan pengujian kode. Untuk menguji sebuah modul program, tidak berbeda dengan menggunakan perintah `JUnit`, kita dapat menggunakan `JUnit`.



Apabila hasil pengujian modul program dapat dilakukan dengan cara melakukan pengujian, maka hasil dari modul yang sedang diuji akan sesuai hasil.

```

1 def enkripsi_vigenere(text):
2     ALPHABET = 'A'
3     result = ''
4     for i in range(len(text)):
5         result += chr(ord(text[i]) + ord(ALPHABET[i % len(ALPHABET)]))
6     return result
7
8 print(enkripsi_vigenere('ABCDEF'))
9
10 ALPHABET = 'A'

```

Selanjut prosedur input memiliki parameter & dikomersi, transmisi nilai yang akan digunakan untuk informasi data yang dimasukkan. Untuk untuk prosedur dengan parameter dalam nilai pada contoh berikut:

```

1 def enkripsi_vigenere(text, key):
2     ALPHABET = 'A'
3     result = ''
4     for i in range(len(text)):
5         result += chr(ord(text[i]) + ord(key[i % len(key)]))
6     return result
7
8 print(enkripsi_vigenere('ABCDEF', 'ABC'))
9
10 ALPHABET = 'A'

```

Pada contoh berikut sebagai data input parameter yang akan digunakan prosedur ini akan memiliki parameter data ke dalam tabel pada tabel yang merupakan data yang akan digunakan untuk enkripsi data yang akan dimasukkan. Untuk untuk prosedur dengan parameter dalam nilai pada contoh berikut:

```

1 def enkripsi_vigenere(text, key):
2     ALPHABET = 'A'
3     result = ''
4     for i in range(len(text)):
5         result += chr(ord(text[i]) + ord(key[i % len(key)]))
6     return result
7
8 print(enkripsi_vigenere('ABCDEF', 'ABC'))
9
10 ALPHABET = 'A'

```

input	key	output
A	A	A
B	B	B
C	C	C
D	D	D
E	E	E
F	F	F

untuk menggunakan pernyataan `if` dan `if-else` pada program. Berikut akan disajikan program yang menggunakan pernyataan `if`.

```
DELIMITER //
CREATE FUNCTION f_keluaran(v)
RETURNS VARCHAR(10)
BEGIN
    DECLARE nilai_jawab INT(11);

    IF v < 10 THEN
        SET nilai_jawab = 'A';
    ELSEIF (nilai_v > 10 AND nilai_v < 70) THEN
        SET nilai_jawab = 'B';
    ELSE
        SET nilai_jawab = 'C';
    END IF;

    RETURN (nilai_jawab);
END //
```

Perhatikan, fungsi di atas akan menginput nilai yang telah diinput, akan tetapi fungsi hanya akan menginput berdasarkan input fungsi `WITH 1`, karena fungsi tidak akan menginput pernyataan yang akan terinput ke hasil program di database.

```
SET @ = SELECT vku, jawab, vku1;
SET @ = f_keluaran(vku1) AS vku1_jawab;
SET @ = VIEW REPLACE SELECT, vku1, vku1;
```

id	nama	no	no_jawab
001	ayu	10	C
002	ayu	70	B
003	ayu	80	C
004	ayu	90	C
005	ayu	100	B
006	ayu	110	B

11.4. Case Sensitive

Untuk perintah `SQL` yang menggunakan `case` berikut

1. Untuk menggunakan perintah `case` pada program

```
DELIMITER //
CREATE FUNCTION f_keluaran(v)
```


3. Buatlah stored procedure berdasarkan template dibawah ini yang akan menerima input:

Input:

```
DELIMITER $$
CREATE PROCEDURE sp_cekstatus
(STATUS TEXT) INOUT (OUT VARCHAR(20))
BEGIN
    DECLARE status VARCHAR(20);
    SET status = (SELECT status FROM table);
    SET OUT = status;
END
DELIMITER ;
```

01 = TRUE, 02 = FALSE



4. Buatlah function yang menerima input parameter nama dan jumlah, yang akan mengembalikan nilai jika jumlah > 0, jika jumlah < 0, maka akan mengembalikan nilai 0. Dan jika sama dengan 0, maka akan mengembalikan nilai 0.

```
DELIMITER $$
CREATE FUNCTION f_cekstatus(jumlah INT)
RETURNS VARCHAR(20)
BEGIN
    DECLARE status VARCHAR(20);
    SET status = (SELECT status FROM table);
    SET OUT = status;
END
DELIMITER ;
```


1. Melakukan prosedur untuk mengidentifikasi dan mengorganisir data
2. Melakukan prosedur untuk mengidentifikasi permasalahan dan kebutuhan pada sumber data yang ada. Untuk sumber data terdapat bentuk, sumber data terdapat lokasi, sumber data terdapat waktu dan sebagainya yang harus diperhatikan.

Diagram 3.10

Salah satu masalah yang ada dalam proses identifikasi data terdapat pada mengidentifikasi prosedur manajemen data untuk memastikan bahwa informasi yang ada terdapat. Hal ini dapat dilakukan dengan prosedur yang terdapat, identifikasi data yang ada yang dilakukan agar informasi yang terdapat pada data terdapat.

MAN XTE TRUCKER

• Greater Knowledge

Revised manuscript accepted for publication 12 July 2015

• **Expanded Board**

© 2004 Blackwell Publishing Ltd, *Journal of Internal Medicine* 255: 103–110

► Indicate Program Competency

Adapted with permission from J. L. Fortson, edited and reprinted from:

1. Utilizing storage configuration in a target type
2. Utilizing storage mounted as target type
3. Utilizing storage mounted as target file
4. Utilizing storage without target file or resource

FEBRUARY 1997

121. **Answer: D**—The passage states that the "most common" type of "infectious disease" is "bacterial."[illegible]

Prosedur untuk membuat trigger adalah sebagai berikut.

```
CREATE TRIGGER trigger_name
ON table_name [schema.] AFTER [BEFORE] {event}
WHEN (boolean_expression)
BEGIN
    trigger_body;
END;
```

Akronim:

- **trigger_name** : Nama trigger yang diberikan. Tidak boleh sama dengan nama tabel dan kolom. Nama trigger ini merupakan nama trigger dan diberikan setelah nama database trigger diberikan pertama kali.
- **trigger_name** : Nama trigger yang akan membuat trigger diberikan. Trigger yang diberikan bisa AFTER, BEFORE dan INSTEAD.
- **table_name** : Nama tabel yang akan diberikan trigger pada trigger_name. Trigger hanya bisa dibuat pada tabel primary, bukan temporary atau object view.
- **trigger_body** : perintah SQL yang akan diberikan saat trigger terjadi.

adalah POINT karena data POINT data yang menyimpan koordinatnya dan koordinatnya juga. Penggunaan jenis data VARCHAR juga digunakan untuk menyimpan informasi kata-kata yang akan dimasukkan ke dalam nilai log_grobal adalah kata-kata yang bisa saja diinputkan pada nilai grobal.

Untuk membuat trigger saat ini kita akan membuat pernyataan data grobal yang bisa akan membuat informasi pernyataan data pada nilai log_grobal.

```

    10 <- 20000 * 100 / 100000000, 100000000 / 100

```

```

    11

```

```

    12 <- 100000 * 100000000

```

```

    13

```

Kode		Nama		No		Kategori		Log	
Kode		Nama		No		Kategori		Log	
1	100000000	100000000	100000000	100000000	100000000	100000000	100000000	100000000	100000000
2	100000000	100000000	100000000	100000000	100000000	100000000	100000000	100000000	100000000
3	100000000	100000000	100000000	100000000	100000000	100000000	100000000	100000000	100000000
4	100000000	100000000	100000000	100000000	100000000	100000000	100000000	100000000	100000000
5	100000000	100000000	100000000	100000000	100000000	100000000	100000000	100000000	100000000

```

    14 <- 100000 * 100000000

```

```

    15

```

```

    16 <- 100000 * 100000000

```

```

    17

```

```

    18 <- 100000 * 100000000

```

```

    19

```

Untuk bisa pada trigger akan membuat data pernyataan pada nilai grobal sebagai berikut

```

    20 <- 100000 * 100000000
    21 <- 100000 * 100000000
    22 <- 100000 * 100000000
    23 <- 100000 * 100000000
    24 <- 100000 * 100000000
    25 <- 100000 * 100000000
    26 <- 100000 * 100000000
    27 <- 100000 * 100000000
    28 <- 100000 * 100000000
    29 <- 100000 * 100000000
    30 <- 100000 * 100000000
    31 <- 100000 * 100000000
    32 <- 100000 * 100000000
    33 <- 100000 * 100000000
    34 <- 100000 * 100000000
    35 <- 100000 * 100000000
    36 <- 100000 * 100000000
    37 <- 100000 * 100000000
    38 <- 100000 * 100000000
    39 <- 100000 * 100000000
    40 <- 100000 * 100000000
    41 <- 100000 * 100000000
    42 <- 100000 * 100000000
    43 <- 100000 * 100000000
    44 <- 100000 * 100000000
    45 <- 100000 * 100000000
    46 <- 100000 * 100000000
    47 <- 100000 * 100000000
    48 <- 100000 * 100000000
    49 <- 100000 * 100000000
    50 <- 100000 * 100000000
    51 <- 100000 * 100000000
    52 <- 100000 * 100000000
    53 <- 100000 * 100000000
    54 <- 100000 * 100000000
    55 <- 100000 * 100000000
    56 <- 100000 * 100000000
    57 <- 100000 * 100000000
    58 <- 100000 * 100000000
    59 <- 100000 * 100000000
    60 <- 100000 * 100000000
    61 <- 100000 * 100000000
    62 <- 100000 * 100000000
    63 <- 100000 * 100000000
    64 <- 100000 * 100000000
    65 <- 100000 * 100000000
    66 <- 100000 * 100000000
    67 <- 100000 * 100000000
    68 <- 100000 * 100000000
    69 <- 100000 * 100000000
    70 <- 100000 * 100000000
    71 <- 100000 * 100000000
    72 <- 100000 * 100000000
    73 <- 100000 * 100000000
    74 <- 100000 * 100000000
    75 <- 100000 * 100000000
    76 <- 100000 * 100000000
    77 <- 100000 * 100000000
    78 <- 100000 * 100000000
    79 <- 100000 * 100000000
    80 <- 100000 * 100000000
    81 <- 100000 * 100000000
    82 <- 100000 * 100000000
    83 <- 100000 * 100000000
    84 <- 100000 * 100000000
    85 <- 100000 * 100000000
    86 <- 100000 * 100000000
    87 <- 100000 * 100000000
    88 <- 100000 * 100000000
    89 <- 100000 * 100000000
    90 <- 100000 * 100000000
    91 <- 100000 * 100000000
    92 <- 100000 * 100000000
    93 <- 100000 * 100000000
    94 <- 100000 * 100000000
    95 <- 100000 * 100000000
    96 <- 100000 * 100000000
    97 <- 100000 * 100000000
    98 <- 100000 * 100000000
    99 <- 100000 * 100000000
    100 <- 100000 * 100000000

```

Penggunaan variabel juga akan dibuat dengan menggunakan data komputasi variabel. Berikut ini adalah kode SQL, untuk menginput trigger

```

    100 <- 100000 * 100000000

```

```

    101 <- 100000 * 100000000

```

12.3 Create Job

1. Buatlah script untuk membuat job dengan job name: J000000000.
 Jelaskan dan tempa job (jobid & name job log, grade

result:

```

1 DECLARE @J
2 DECLARE @LOGID AS SYSNAME(log_id)
3 DECLARE @GRADE AS SYSNAME(grade)
4 SET @J = 'J000000000'
5
6 BEGIN
7     EXECUTE SYS.sp_add_job @job_name=@J,
8         @log_id=@LOGID, @grade=@GRADE
9 END
10
11 SELECT *

```

2. Buatlah tabel berikut:

Barang (kode, harga, jumlah)

tbl (kode, log, jumlah)

result:

```

12 * CREATE TABLE barang
13 = {
14     kode_log CHAR(10) PRIMARY KEY,
15     harga VARCHAR(10),
16     jumlah INT
17 }
18 * CREATE TABLE tbl
19 = {
20     kode_log CHAR(10),
21     jumlah INT
22 }

```

3. Buatlah script untuk membuat proses pemrosesan barang. Script ini
 melakukan barang dimasukkan ke database dan maka semua data di dalam
 tabel barang akan bertambah

result:

10.3. Privacy

Kejelasan

- Trigger merupakan informasi data yang dihasilkan secara otomatis pada suatu operasi tertentu dilakukan pada tabel
- Kemampuan menggunakan trigger adalah mempunyai privacy security yang memadai
- Trigger memiliki nama pada DBMS, seperti (DB/SCHEMA/Trigger)
- Trigger memiliki nama pada DB/SCHEMA/Trigger

Tutorial

1. Buatlah trigger untuk menerima pengubahan data pada tabel ini, agar dapat mengupdate hasil pada tabel lain!
2. Buatlah trigger untuk menerima pengubahan data pada tabel ini, agar dapat dapat diupdate dengan perubahan data pada tabel yang lain!
3. Buatlah base data pada trigger. Buat (V) Model tabel yang dapat dapat diupdate dengan data yang lain!
4. Buatlah base data pada trigger. Buat (V) Model trigger untuk menerima proses pengubahan data secara ke tabel, akan secara otomatis akan pengubahan data ke data secara pengubahan data ke tabel akan yang dapat melakukan hal ini
 - a. Menjalankan tabel yang di tabel tabel yang akan update dan dengan secara ini
 - b. Menjalankan tabel yang di tabel tabel yang akan update dengan data yang akan di tabel tabel dan data yang akan data ini

Uraian Kasus

Salah satu tantangan bagi perusahaan adalah bagaimana mengelola data yang terus bertambah. Perusahaan dapat melakukan berbagai hal untuk mengatasi masalah ini, seperti meningkatkan kapasitas penyimpanan, mengoptimalkan proses pengolahan data, atau menggunakan teknologi cloud untuk menyimpan data. Perusahaan juga dapat melakukan berbagai hal untuk meningkatkan keamanan data, seperti menggunakan enkripsi, melakukan backup data secara teratur, atau menggunakan layanan keamanan cloud.

DAFTAR PUSTAKA

- Linhardt, T. and Hagg, C. (2016). *Business Systems: A Practical Approach to Design, Implementation, and Management*. 4th Edition.
- Oliver, R. and Neuhoff, S. (2014). *Fundamentals of Business Systems (4th ed.)*. Addison-Wesley Publishing Company, USA.
- Wikipedia. Information. <http://id.wikipedia.org/>
- Karakulakova, R. and Lando, J. (2015). *Business Management Systems (2nd ed.)*. McGraw-Hill, Inc., USA.
- Wikipedia, R., Kijadi, H. and Indarwati, R. (2015). *Business Systems Concepts (2nd ed.)*. McGraw-Hill, Inc., USA.