



BUKU KERJA PRAKTIK MAHASISWA (BKPM)

WORKSHOP MANAJEMEN BASIS DATA

MIF120708

SEMESTER 1

OLEH

DIA BITARI MEI YUANA, S.ST.,M.Tr.Kom.

PROGRAM STUDI MANAJEMEN INFORMATIKA

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI JEMBER

TAHUN 2022

LEMBAR PENGESAHAN

WORKSHOP BASISDATA RELATIONAL

Mengetahui,

Koodinator Progam Studi,

Taufiq Rizaldi, S.ST., MT
NIP. 198903292019031007

Koordinator Mata Kuliah,

Dwi Putro Sarwo S, S.Kom, M.Kom
NIP. 198005172008121002

Penulis,


Dia Bitari Mei Y, S.ST.,M.Tr.Kom
NIP. 199305082022032013

Menyetujui,
Ketua Jurusan Teknologi Informasi

Hendra Yufit Riskiawan, S.Kom., M.Cs
NIP. 19830203 200604 1 003

KATA PENGANTAR

Puji Syukur kehadirat Tuhan Yang Maha Esa karena atas limpahan rahmat-Nya sehingga penulis dapat menyelesaikan Buku Kerja Praktikum Mahasiswa (BKPM) Workshop Basis Data. Materi Workshop terdiri dari pengenalan XAMPP dan PhpMyAdmin, relasi antar tabel, create statement pada DDL, alter statement, rename statement, drop table dan database statement, insert into statement, update statemnt pada DML, delete statement, tabel joins meliputi : inner join, cross join, natural join, clausa using, self join, fungsi agregate, like dan wildcards, IN dan Aliases, implementasi operator, implelentasi statement, implementasi clausa, dan studi kasus. BKPM ini disusun berdasarkan metode *Student Center Learning* yaitu menempatkan mahasiswa sebagai pusat kegiatan belajar. BKPM ini terdiri dari Capaian Pembelajaran Mata Kuliah (CPMK), Indikator Penilaian, Dasar Teori, Alat dan Bahan, Prosedur Kerja, Hasil dan Pembahasan, dan Rubrik Penilaian.

Penulis menyadari masih banyak kekurangan dalam penyusunan BKPM ini. Oleh karena itu, penulis sangat mengharapkan kritik dan saran demi perbaikan dan kesempurnaan modul ini. Penulis mengucapkan terima kasih kepada berbagai pihak yang telah membantu proses penyelesaian BKPM ini. Semoga BKPM ini dapat bermanfaat bagi kita semua.

Jember, 20 Agustus 2022

Penulis

DAFTAR ISI

Lembar Pengesahan	i
Kata Pengantar	ii
Daftar Isi	iii
ACARA 1: Pengenalan Xampp dan PhpMyAdmin	1
a. Capaian Pembelajaran Mata Kuliah (CPMK)	1
b. Indikator	1
c. Dasar Teori	2
d. Alat dan Bahan	3
e. Prosedur Kerja	4
f. Hasil dan Pembahasan	5
g. Kesimpulan	5
h. Rubrik Penilaian	5
ACARA 2: Pengenalan Xampp dan PhpMyAdmin	6
a. Capaian Pembelajaran Mata Kuliah (CPMK)	6
b. Indikator	6
c. Dasar Teori	6
d. Alat dan Bahan	6
e. Prosedur Kerja	6
f. Hasil dan Pembahasan	10
g. Kesimpulan	11
h. Rubrik Penilaian	11
ACARA 3: Merancang Entity Relationship Diagram (Studi Kasus)	12
a. Capaian Pembelajaran Mata Kuliah (CPMK)	12
b. Indikator	12
c. Dasar Teori	12
d. Alat dan Bahan	15
e. Prosedur Kerja	15

f. Hasil dan Pembahasan	15
g. Kesimpulan	16
h. Rubrik Penilaian	16
ACARA 4: Relasi Antar Tabel	17
a. Capaian Pembelajaran Mata Kuliah (CPMK)	17
b. Indikator	17
c. Dasar Teori	17
d. Alat dan Bahan	18
e. Prosedur Kerja	18
f. Hasil dan Pembahasan	21
g. Kesimpulan	21
h. Rubrik Penilaian	21
ACARA 5: DDL (Create Statement)	22
a. Capaian Pembelajaran Mata Kuliah (CPMK)	22
b. Indikator	22
c. Dasar Teori	22
d. Alat dan Bahan	24
e. Prosedur Kerja	24
f. Hasil dan Pembahasan	25
g. Kesimpulan	25
h. Rubrik Penilaian	26
ACARA 6: DDL (Alter Statement)	27
a. Capaian Pembelajaran Mata Kuliah (CPMK)	27
b. Indikator	27
c. Dasar Teori	27
d. Alat dan Bahan	30
e. Prosedur Kerja	30
f. Hasil dan Pembahasan	31
g. Kesimpulan	31
h. Rubrik Penilaian	31

ACARA 7: DDL (Rename Statement)	32
a. Capaian Pembelajaran Mata Kuliah (CPMK)	32
b. Indikator	32
c. Dasar Teori	32
d. Alat dan Bahan	33
e. Prosedur Kerja	33
f. Hasil dan Pembahasan	33
g. Kesimpulan	34
h. Rubrik Penilaian	34
ACARA 8: DDL (Drop Table n Database Statement)	35
a. Capaian Pembelajaran Mata Kuliah (CPMK)	35
b. Indikator	35
c. Dasar Teori	35
d. Alat dan Bahan	37
e. Prosedur Kerja	37
f. Hasil dan Pembahasan	38
g. Kesimpulan	38
h. Rubrik Penilaian	39
ACARA 9: DML (Insert Into Statement)	40
a. Capaian Pembelajaran Mata Kuliah (CPMK)	40
b. Indikator	40
c. Dasar Teori	40
d. Alat dan Bahan	43
e. Prosedur Kerja	43
f. Hasil dan Pembahasan	49
g. Kesimpulan	52
h. Rubrik Penilaian	52
ACARA 10: DML (Select Statement)	53
a. Capaian Pembelajaran Mata Kuliah (CPMK)	53
b. Indikator	53

c. Dasar Teori	53
d. Alat dan Bahan	55
e. Prosedur Kerja	55
f. Hasil dan Pembahasan	56
g. Kesimpulan	56
h. Rubrik Penilaian	56
ACARA 11: DML (Update Statement)	57
a. Capaian Pembelajaran Mata Kuliah (CPMK)	57
b. Indikator	57
c. Dasar Teori	57
d. Alat dan Bahan	58
e. Prosedur Kerja	58
f. Hasil dan Pembahasan	59
g. Kesimpulan	59
h. Rubrik Penilaian	60
ACARA 12: DML (Delete Statement)	61
a. Capaian Pembelajaran Mata Kuliah (CPMK)	61
b. Indikator	61
c. Dasar Teori	61
d. Alat dan Bahan	62
e. Prosedur Kerja	62
f. Hasil dan Pembahasan	63
g. Kesimpulan	63
h. Rubrik Penilaian	63
ACARA 13: Tabel Joins	64
a. Capaian Pembelajaran Mata Kuliah (CPMK)	64
b. Indikator	64
c. Dasar Teori	64
d. Alat dan Bahan	65

e. Prosedur Kerja	66
f. Hasil dan Pembahasan	66
g. Kesimpulan	66
h. Rubrik Penilaian	66
ACARA 14: Join Tabel (“,”, Cross Join & Inner Join).....	67
a. Capaian Pembelajaran Mata Kuliah (CPMK)	67
b. Indikator	67
c. Dasar Teori	67
d. Alat dan Bahan	69
e. Prosedur Kerja	69
f. Hasil dan Pembahasan	72
g. Kesimpulan	73
h. Rubrik Penilaian	73
ACARA 15: Join Tabel (Natural Join)	74
a. Capaian Pembelajaran Mata Kuliah (CPMK)	74
b. Indikator	74
c. Dasar Teori	74
d. Alat dan Bahan	74
e. Prosedur Kerja	74
f. Hasil dan Pembahasan	75
g. Kesimpulan	75
h. Rubrik Penilaian	75
ACARA 16: Join Tabel (CLAUSA USING).....	76
a. Capaian Pembelajaran Mata Kuliah (CPMK)	76
b. Indikator	76
c. Dasar Teori	76
d. Alat dan Bahan	77
e. Prosedur Kerja	77
f. Hasil dan Pembahasan	77
g. Kesimpulan	77

h. Rubrik Penilaian	77
ACARA 17: Join Tabel (Self Join)	78
a. Capaian Pembelajaran Mata Kuliah (CPMK)	78
b. Indikator	78
c. Dasar Teori	78
d. Alat dan Bahan	79
e. Prosedur Kerja	80
f. Hasil dan Pembahasan	80
g. Kesimpulan	81
h. Rubrik Penilaian	81
ACARA 18: Fungsi Agregat (Fungsi Dasar di Agregat)	82
a. Capaian Pembelajaran Mata Kuliah (CPMK)	82
b. Indikator	82
c. Dasar Teori	82
d. Alat dan Bahan	83
e. Prosedur Kerja	84
f. Hasil dan Pembahasan	86
g. Kesimpulan	87
h. Rubrik Penilaian	87
ACARA 19: Fungsi Agregat (Like dan Wildcards)	88
a. Capaian Pembelajaran Mata Kuliah (CPMK)	88
b. Indikator	88
c. Dasar Teori	88
d. Alat dan Bahan	89
e. Prosedur Kerja	89
f. Hasil dan Pembahasan	90
g. Kesimpulan	91
h. Rubrik Penilaian	91
ACARA 20: Fungsi Agregat (IN dan Aliases)	92
a. Capaian Pembelajaran Mata Kuliah (CPMK)	92

b. Indikator	92
c. Dasar Teori	92
d. Alat dan Bahan	94
e. Prosedur Kerja	94
f. Hasil dan Pembahasan	95
g. Kesimpulan	95
h. Rubrik Penilaian	95
ACARA 22: Implementasi Operator	96
a. Capaian Pembelajaran Mata Kuliah (CPMK)	96
b. Indikator	96
c. Dasar Teori	96
d. Alat dan Bahan	96
e. Prosedur Kerja	97
f. Hasil dan Pembahasan	97
g. Kesimpulan	98
h. Rubrik Penilaian	98
ACARA 23: Implementasi Statement	99
a. Capaian Pembelajaran Mata Kuliah (CPMK)	99
b. Indikator	99
c. Dasar Teori	99
d. Alat dan Bahan	103
e. Prosedur Kerja	103
f. Hasil dan Pembahasan	104
g. Kesimpulan	104
h. Rubrik Penilaian	104
ACARA 24: Implementasi Calusa	105
a. Capaian Pembelajaran Mata Kuliah (CPMK)	105
b. Indikator	105
c. Dasar Teori	105
d. Alat dan Bahan	108

e. Prosedur Kerja	108
f. Hasil dan Pembahasan	109
g. Kesimpulan	110
h. Rubrik Penilaian	110
ACARA 25: Mencari Studi Kasus.....	111
a. Capaian Pembelajaran Mata Kuliah (CPMK)	111
b. Indikator	111
c. Dasar Teori	111
d. Alat dan Bahan	113
e. Prosedur Kerja	113
f. Hasil dan Pembahasan	114
g. Kesimpulan	114
h. Rubrik Penilaian	114
ACARA 26: Presentasi Studi Kasus.....	115
a. Capaian Pembelajaran Mata Kuliah (CPMK)	115
b. Indikator	115
c. Dasar Teori	115
d. Alat dan Bahan	121
e. Prosedur Kerja	121
f. Hasil dan Pembahasan	122
g. Kesimpulan	122
h. Rubrik Penilaian	122
ACARA 27: Struktur Tabel	123
a. Capaian Pembelajaran Mata Kuliah (CPMK)	123
b. Indikator	123
c. Dasar Teori	123
d. Alat dan Bahan	123
e. Prosedur Kerja	123
f. Hasil dan Pembahasan	124
g. Kesimpulan	124

h. Rubrik Penilaian	124
ACARA 28: Memasukan data ke dalam tabel	125
a. Capaian Pembelajaran Mata Kuliah (CPMK)	125
b. Indikator	125
c. Dasar Teori	125
d. Alat dan Bahan	127
e. Prosedur Kerja	127
f. Hasil dan Pembahasan	128
g. Kesimpulan	128
h. Rubrik Penilaian	128
ACARA 29: Studi Kasus : Seleksi antar tabel (Join dua Tabel)	129
a. Capaian Pembelajaran Mata Kuliah (CPMK)	129
b. Indikator	129
c. Dasar Teori	129
d. Alat dan Bahan	129
e. Prosedur Kerja	129
f. Hasil dan Pembahasan	129
g. Kesimpulan	130
h. Rubrik Penilaian	130
ACARA 30: Seleksi antar tabel (Join Dua Tabel)	131
a. Capaian Pembelajaran Mata Kuliah (CPMK)	131
b. Indikator	131
c. Dasar Teori	131
d. Alat dan Bahan	131
e. Prosedur Kerja	131
f. Hasil dan Pembahasan	131
g. Kesimpulan	132
h. Rubrik Penilaian	132
ACARA 30: Seleksi antar tabel (Join Tiga Tabel).....	133
a. Capaian Pembelajaran Mata Kuliah (CPMK)	133

b. Indikator	133
c. Dasar Teori	133
d. Alat dan Bahan	136
e. Prosedur Kerja	136
f. Hasil dan Pembahasan	136
g. Kesimpulan	137
h. Rubrik Penilaian	137
ACARA 31: Studi Kasus : Seleksi antar tabel (Fungsi Agregat)	138
a. Capaian Pembelajaran Mata Kuliah (CPMK)	138
b. Indikator	138
c. Dasar Teori	138
d. Alat dan Bahan	138
e. Prosedur Kerja	138
f. Hasil dan Pembahasan	138
g. Kesimpulan	139
h. Rubrik Penilaian	139
ACARA 32: Studi Kasus : Seleksi antar tabel (Operator)	140
a. Capaian Pembelajaran Mata Kuliah (CPMK)	140
b. Indikator	140
c. Dasar Teori	140
d. Alat dan Bahan	140
e. Prosedur Kerja	141
f. Hasil dan Pembahasan	141
g. Kesimpulan	141
h. Rubrik Penilaian	141
ACARA 32: Studi Kasus : Seleksi antar tabel (Case, Limit, Having)	142
a. Capaian Pembelajaran Mata Kuliah (CPMK)	142
b. Indikator	142
c. Dasar Teori	142
d. Alat dan Bahan	142

e. Prosedur Kerja	142
f. Hasil dan Pembahasan	143
g. Kesimpulan	143
h. Rubrik Penilaian	143
ACARA 33: Store Procedures	144
a. Capaian Pembelajaran Mata Kuliah (CPMK)	144
b. Indikator	144
c. Dasar Teori	144
d. Alat dan Bahan	147
e. Prosedur Kerja	147
f. Hasil dan Pembahasan	150
g. Kesimpulan	150
h. Rubrik Penilaian	150
ACARA 34: Store Functions.....	151
a. Capaian Pembelajaran Mata Kuliah (CPMK)	151
b. Indikator	151
c. Dasar Teori	151
d. Alat dan Bahan	152
e. Prosedur Kerja	152
f. Hasil dan Pembahasan	152
g. Kesimpulan	153
h. Rubrik Penilaian	153
ACARA 35: Index dan View	154
a. Capaian Pembelajaran Mata Kuliah (CPMK)	154
b. Indikator	154
c. Dasar Teori	154
d. Alat dan Bahan	155
e. Prosedur Kerja	155
f. Hasil dan Pembahasan	156
g. Kesimpulan	157

h. Rubrik Penilaian	157
ACARA 35: Studi Kasus Index dan View	158
a. Capaian Pembelajaran Mata Kuliah (CPMK)	158
b. Indikator	158
c. Dasar Teori	158
d. Alat dan Bahan	158
e. Prosedur Kerja	158
f. Hasil dan Pembahasan	158
g. Kesimpulan	160
h. Rubrik Penilaian	160
ACARA 36: Trigger.....	161
a. Capaian Pembelajaran Mata Kuliah (CPMK)	161
b. Indikator	161
c. Dasar Teori	161
d. Alat dan Bahan	164
e. Prosedur Kerja	164
f. Hasil dan Pembahasan	165
g. Kesimpulan	167
h. Rubrik Penilaian	167
ACARA 37: Trigger ON Insert and Update	168
a. Capaian Pembelajaran Mata Kuliah (CPMK)	168
b. Indikator	168
c. Dasar Teori	168
d. Alat dan Bahan	168
e. Prosedur Kerja	168
f. Hasil dan Pembahasan	169
g. Kesimpulan	169
h. Rubrik Penilaian	169
ACARA 38: Studi Kasus Trigger	171
a. Capaian Pembelajaran Mata Kuliah (CPMK)	171

b. Indikator	171
c. Dasar Teori	171
d. Alat dan Bahan	171
e. Prosedur Kerja	171
f. Hasil dan Pembahasan	172
g. Kesimpulan	172
h. Rubrik Penilaian	172
ACARA 39: Studi Kasus Trigger	173
a. Capaian Pembelajaran Mata Kuliah (CPMK)	173
b. Indikator	173
c. Dasar Teori	173
d. Alat dan Bahan	173
e. Prosedur Kerja	173
f. Hasil dan Pembahasan	173
g. Kesimpulan	174
h. Rubrik Penilaian	174

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 1

Pokok Bahasan	: Pengenalan XAMPP dan phpMyAdmin
Acara Praktikum/Pertemuan	: Minggu 1/1
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami alat yang digunakan untuk membangun database
2. Mamahasiswa menerapkan alat yang digunakan untuk membangun database

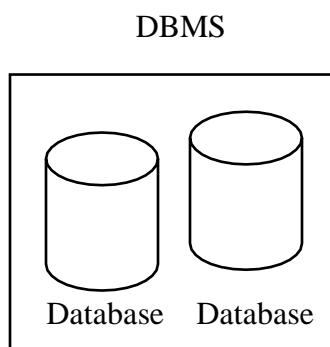
b. Indikator

Kemampuan mahasiswa dalam menggunakan alat untuk membangun database

c. Dasar Teori

Database merupakan sekumpulan data yang tersimpan dengan menggunakan manajemen sistem tertentu yang saling berhubungan sehingga dapat dikelolah dengan mudah. Database dapat disebut juga sebagai data yang berada didalam Database Manajemen Sistem yang biasa disingkat sebagai DBMS. DBMS berfungsi untuk menampung dan mengelolah banyak database dengan data, aturan, keamanan, dan hak akses berada didalam database dengan ilustrasi yang ditunjukkan pada Gambar 1. Berikut adalah macam-macam software untuk mengelolah Database diantaranya adalah :

1. Oracle
2. MySQL
3. SQL Server
4. PostgreSQL
5. MongoDB



Gambar1. Ilustrasi DBMS

DBMS terdiri dari beberapa jenis diantaranya adalah :

1. Relational DBMS
2. Hierarchical DBMS
3. Network DBMS
4. NoSQL DBMS

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi dan merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (localhost), yang terdiri atas program Apache HTTP Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. XAMPP dikembangkan dari sebuah tim proyek bernama Apache Friends, yang terdiri dari Tim Inti (Core Team), Tim Pengembang (Development Team) & Tim Dukungan (Support Team). Versi rilis terakhir XAMPP pada 12 September 2020 adalah versi 7.4.27 dan 8.1.2. Nama XAMPP merupakan singkatan dari :

X = Cross Platform (tempat sistem operasi apapun ditampung),

A = Apache,

M = MySQL,

P = PHP dan

P = Perl.

Program ini tersedia dalam GNU General Public License dan bebas, merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis. Bagian XAMPP yang biasa digunakan pada umumnya terdiri dari:

- a. htdoc merupakan folder tempat meletakkan berkas-berkas yang akan dijalankan, seperti berkas PHP, HTML dan skrip lain.
- b. phpMyAdmin merupakan bagian untuk mengelola basis data atau biasa disebut sebagai tools untuk mengelolah database MySQL yang ada di komputer.
- c. Kontrol Panel yang berfungsi untuk mengelola layanan (service) XAMPP. Seperti menghentikan (stop) layanan, ataupun memulai (start).

Salah satu fungsi pada XAMPP yang akan diterapkan pada pertemuan kali ini adalah sebagai setting Database phpMyadmin

d. Alat dan Bahan

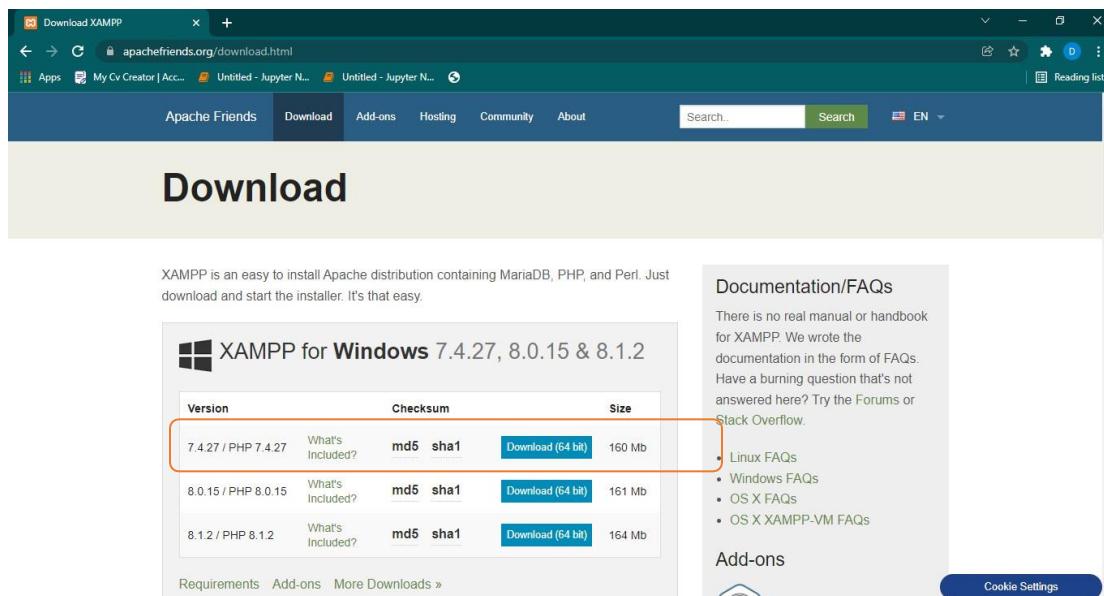
1. Client Server: Xampp
2. Tools : Phpmyadmin
3. Kertas A4 / Folio Bergaris
4. Pulpen

e. Prosedur Kerja

1. Hasil dan Pembahasan

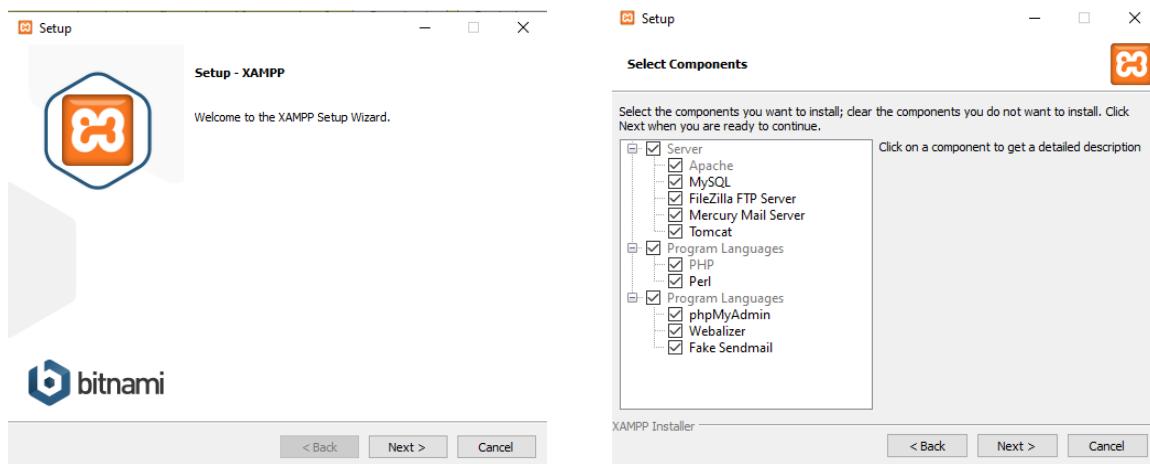
Mahasiswa menginstall XAMPP dengan langkah – langkah sebagai berikut :

1. Kunjungi situs <https://www.apachefriends.org/download.html> lalu download XAMPP versi 7.4.27

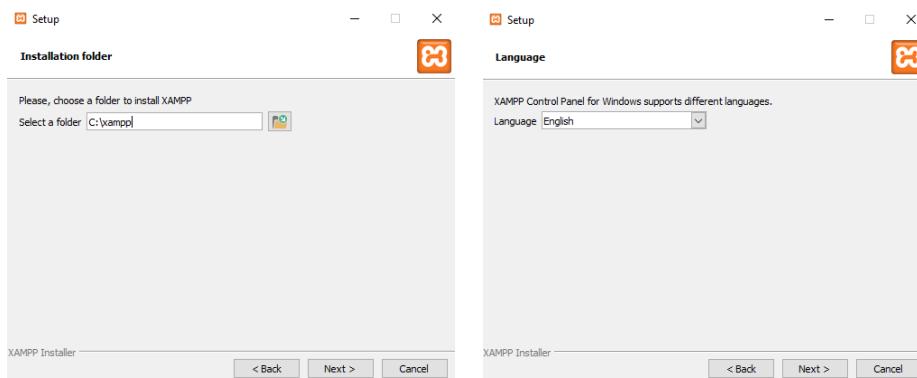


2. Double klik master XAMPP atau bisa klik kanan lalu pilih run as administrator klik install

3. Klik Next



4. Kemudian tentukan lokasi folder penyimpanan file-file dan folder XAMPP. Secara default akan diarahkan ke lokasi c:\xampp. Namun jika ingin menyimpannya di folder lain bisa klik browse dan tentukan secara manual folder yang ingin digunakan. Jika sudah selesai, lanjutkan dan klik tombol next dan ikuti prosesnya dengan menekan tombol next.



5. Tunggu beberapa menit hingga proses instalasi selesai. Jika sudah muncul jendela seperti di bawah ini, klik tombol Finish untuk menyelesaikannya.



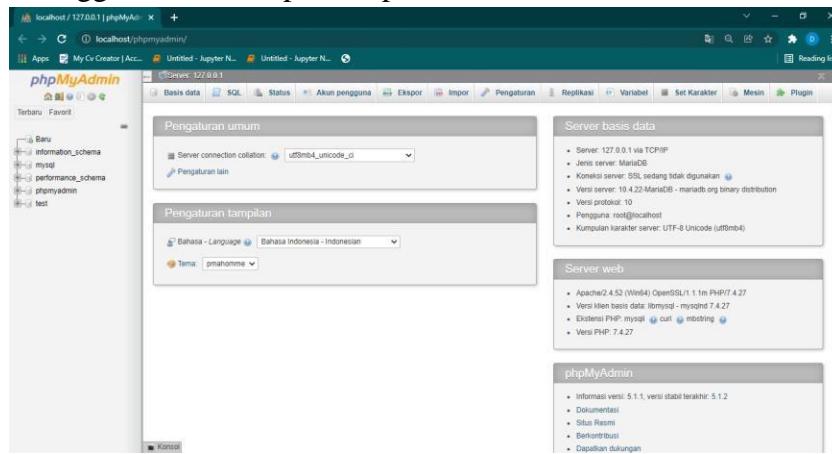
6. Untuk menjalankan aplikasi XAMPP terdapat beberapa langkah diantaranya sebagai berikut :

- 1) Buka aplikasi XAMPP, bisa melalui Start Menu atau Desktop, dan klik icon XAMPP. Atau, jika membuka begitu proses instalasi selesai maka klik Yes seperti yang terlihat pada gambar di atas.
- 2) Setelah terbuka, klik tombol Start pada kolom Action sehingga tombol tersebut berubah menjadi Stop. Dengan mengklik tombol tersebut, artinya itulah aplikasi yang dijalankan. Untuk database mysql menggunakan XAMPP, maka hanyalah aplikasi Apache dan MySQL yang di klik start. Lalu klik tombol admin

Module	Service	Module	PID(s)	Port(s)	Actions
	Apache				Start Admin Config Logs
	MySQL				Start Admin Config Logs
	FileZilla				Start Admin Config Logs
	Mercury				Start Admin Config Logs
	Tomcat				Start Admin Config Logs

Time	User	Message
9:34:05 AM	[main]	there will be a security dialogue or things will break! So think about running this application with administrator rights!
9:34:05 AM	[main]	XAMPP Installation Directory: "c:\xampp"
9:34:05 AM	[main]	Checking for prerequisites
9:34:05 AM	[main]	All prerequisites found
9:34:05 AM	[main]	Initializing Modules
9:34:05 AM	[main]	Starting Check-Timer
9:34:05 AM	[main]	Control Panel Ready
9:40:35 AM	[Apache]	Attempting to start Apache app...
9:40:36 AM	[Apache]	Status change detected. running
9:40:37 AM	[mysql]	Attempting to start MySQL app...
9:40:38 AM	[mysql]	Status change detected. running

- 3) Atau dapat membuka melalui browser dan ketik <http://localhost/phpmyadmin/> sehingga muncul tampilan seperti dibawah



f. Hasil dan Pembahasan

Membuat laporan dari penjelasan Sistem Manajemen Basis Data atau biasa disebut sebagai **DBMS (Database Management System)** dengan format laporan sebagai berikut :

- Terdapat cover dengan ketentuan “{Judul}”, Minggu I, logo Polije, nama, nim, golongan
- Berisi : Pendahuluan, isi makalah, kesimpulan, dan daftar Pustaka
- Format .pdf
- Simpan dengan Nama_NIM_Gol_MingguI
- Kumpulkan di <http://jti.polije.ac.id/elearning/>

g. Kesimpulan

Didapatkan sebuah pemahaman terkait installasi tools dan penjelasan dari DBMS untuk mengawali workshop manajemen basis data.

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak seusai	V			
		Total Skor*		Skor x 25	

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 2

Pokok Bahasan : Pengenalan XAMPP dan phpMyAdmin

Acara Praktikum/Pertemuan : Minggu 1/2

Tempat : Daring/Luring Politeknik Negeri Jember

Alokasi Waktu : 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep dari XAMPP
2. Mahasiswa mampu memahami dan menggunakan PhpMyadmin untuk membangun database

b. Indikator

Kemampuan mahasiswa dalam menggunakan alat untuk membangun database

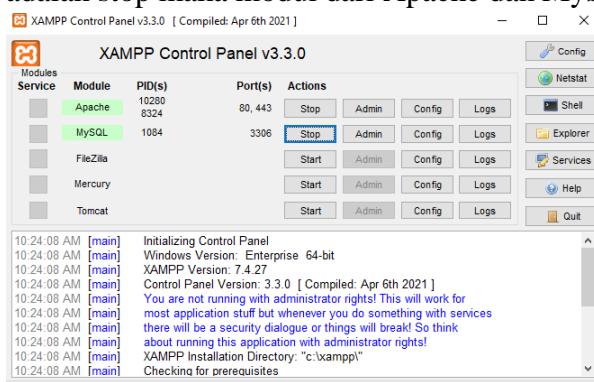
c. Dasar Teori

d. Alat dan Bahan

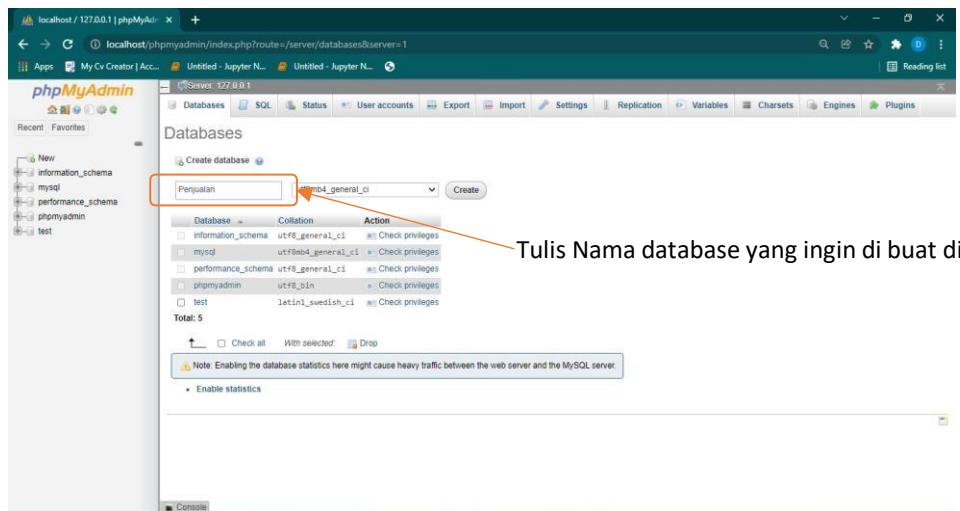
1. Client Server: Xampp
2. Tools : Phpmyadmin
3. Kertas A4 / Folio Bergaris
4. Pulpen

e. Prosedur Kerja

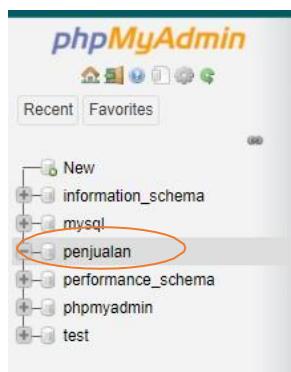
1. Jalankan aplikasi XAMPP dan klik start pada Apache dan MySQL. Jika status pada Action adalah stop maka modul dari Apache dan MySQL sudah berjalan.



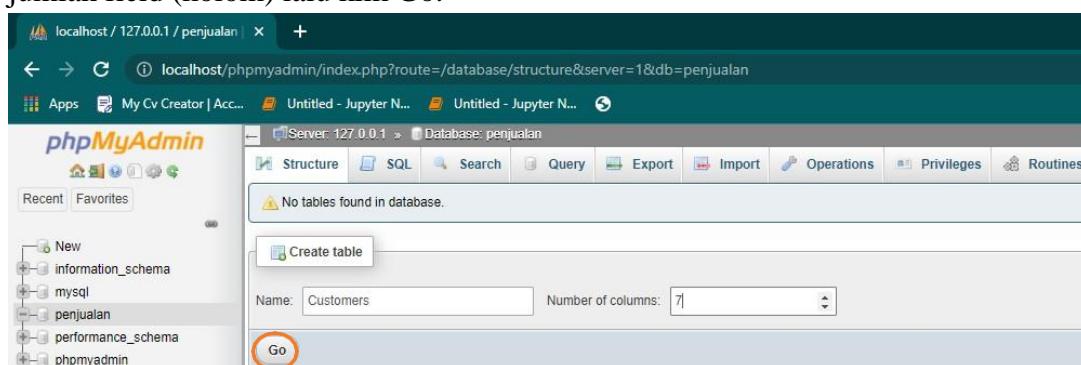
2. Klik Admin pada Modul MySQL atau buka browser dan ketik localhost/phpmyadmin/
3. Klik Basis data dan buat database baru dengan ketik “nama database” yang akan dibuat di kolom gambar lalu klik Buat/Create



4. Jika database berhasil dibuat maka akan tampil seperti gambar dibawah



5. Terdapat dua cara untuk membuat table yaitu dapat memanfaatkan tools pada phpMyAdmin atau dengan menulis syntax SQL. Jika menggunakan tools, maka tulis nama table pada kolom Name pada Create table dan Number of Column untuk menentukan jumlah field (kolom) lalu klik Go.



6. Untuk membuat table dengan ketentuan :

Field	Type	Null	Key	Default	Extra
CustomerID	int(11)	NO	PRI		auto_increment
CustomerName	varchar(255)	YES			
ContactName	varchar(255)	YES			
Address	varchar(255)	YES			
City	varchar(255)	YES			
PostalCode	varchar(255)	YES			
Country	varchar(255)	YES			

Maka terdapat dua cara yaitu :

- 1) Dengan tools phpMyAdmin seperti gambar dibawah. Jangan lupa untuk centang auto increment pada tanda A_I dan primary sebagai key unik pada attribute tabel CustomerID lalu klik Save

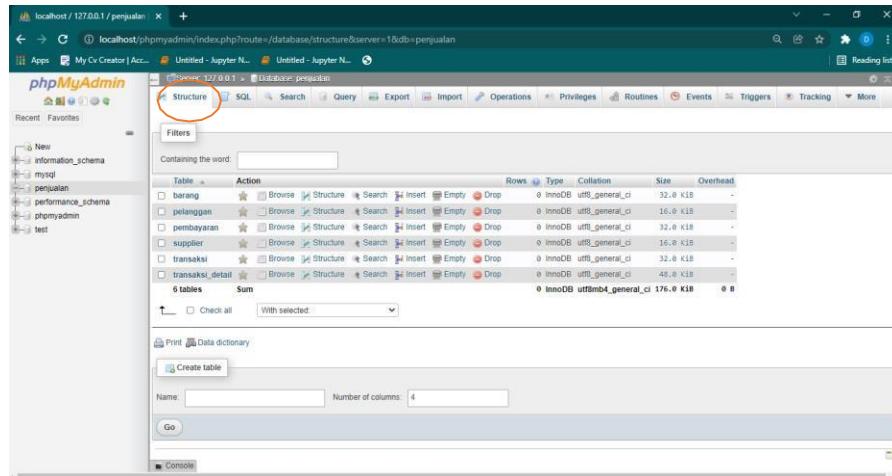
- 2) Dengan Syntax SQL

Gambar dibawah menandakan bahwa table berhasil dibuat.



```
CREATE TABLE `Customers` (
  `CustomerID` int(11) NOT NULL,
  `CustomerName` varchar(255) DEFAULT NULL,
  `Address` varchar(255) DEFAULT NULL,
  `City` varchar(255) DEFAULT NULL,
  `PostalCode` varchar(255) DEFAULT NULL,
  `Country` varchar(255) NOT NULL,
  PRIMARY KEY (`CustomerID`)
)
```

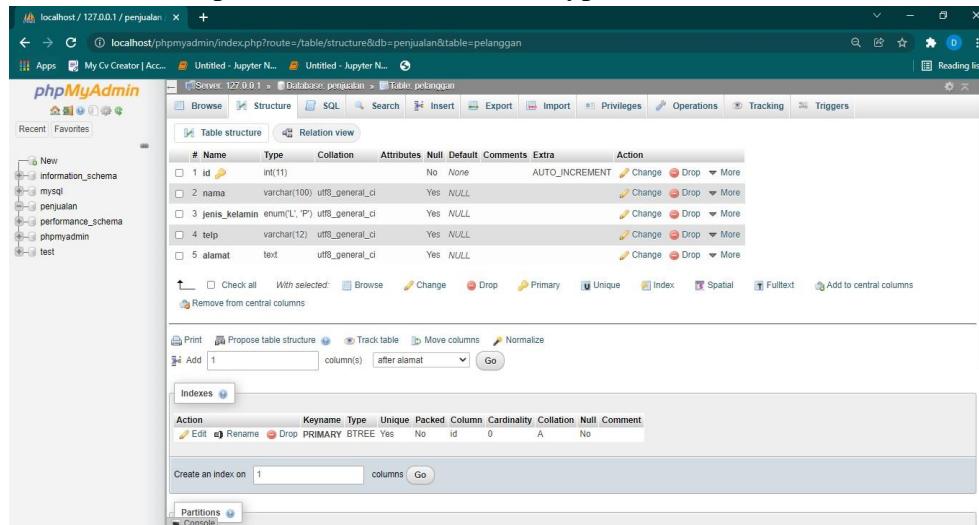
7. Untuk menampilkan tabel apa saja yang sudah dibuat dalam suatu database, dapat klik structure maka detail structure database akan terlihat.



The screenshot shows the phpMyAdmin interface for the 'penjualan' database. The 'Structure' tab is selected. A red circle highlights the 'Structure' tab in the top navigation bar. Below it, the table structure for 'pelanggan' is shown:

Table	Action	Rows	Type	Collation	Size	Overhead
barang	<input type="checkbox"/> Browse <input type="checkbox"/> Structure <input type="checkbox"/> Search <input type="checkbox"/> Insert <input type="checkbox"/> Empty <input type="checkbox"/> Drop	0	InnoDB	utf8_general_ci	32.0 Kib	-
pelanggan	<input type="checkbox"/> Browse <input type="checkbox"/> Structure <input type="checkbox"/> Search <input type="checkbox"/> Insert <input type="checkbox"/> Empty <input type="checkbox"/> Drop	0	InnoDB	utf8_general_ci	16.0 Kib	-
pembayaran	<input type="checkbox"/> Browse <input type="checkbox"/> Structure <input type="checkbox"/> Search <input type="checkbox"/> Insert <input type="checkbox"/> Empty <input type="checkbox"/> Drop	0	InnoDB	utf8_general_ci	32.0 Kib	-
supplier	<input type="checkbox"/> Browse <input type="checkbox"/> Structure <input type="checkbox"/> Search <input type="checkbox"/> Insert <input type="checkbox"/> Empty <input type="checkbox"/> Drop	0	InnoDB	utf8_general_ci	16.0 Kib	-
transaksi	<input type="checkbox"/> Browse <input type="checkbox"/> Structure <input type="checkbox"/> Search <input type="checkbox"/> Insert <input type="checkbox"/> Empty <input type="checkbox"/> Drop	0	InnoDB	utf8_general_ci	32.0 Kib	-
transaksi_detail	<input type="checkbox"/> Browse <input type="checkbox"/> Structure <input type="checkbox"/> Search <input type="checkbox"/> Insert <input type="checkbox"/> Empty <input type="checkbox"/> Drop	0	InnoDB	utf8_general_ci	48.0 Kib	-
6 tables	<input type="checkbox"/> Sum	0	InnoDB	utf8mb4_general_ci	176.0 Kib	0 B

8. Untuk melihat detail table dari salah satu table yang telah dibuat, maka klik salah satu table kemudian klik structure pada menu bar maka structure dari salah satu table yang sudah dibuat akan tampil mulai dari nama fileds, type data, collation, null, Extra, Action



The screenshot shows the phpMyAdmin interface for the 'penjualan' database, specifically viewing the 'pelanggan' table structure. The 'Structure' tab is selected. The table structure is displayed as follows:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)	utf8_general_ci		No	None	AUTO_INCREMENT	<input type="checkbox"/> Change <input type="checkbox"/> Drop <input type="checkbox"/> More	<input type="checkbox"/> Change <input type="checkbox"/> Drop <input type="checkbox"/> More
2	nama	varchar(100)	utf8_general_ci		Yes	NULL		<input type="checkbox"/> Change <input type="checkbox"/> Drop <input type="checkbox"/> More	<input type="checkbox"/> Change <input type="checkbox"/> Drop <input type="checkbox"/> More
3	jenis_kelamin	enum('L','P')	utf8_general_ci		Yes	NULL		<input type="checkbox"/> Change <input type="checkbox"/> Drop <input type="checkbox"/> More	<input type="checkbox"/> Change <input type="checkbox"/> Drop <input type="checkbox"/> More
4	telp	varchar(12)	utf8_general_ci		Yes	NULL		<input type="checkbox"/> Change <input type="checkbox"/> Drop <input type="checkbox"/> More	<input type="checkbox"/> Change <input type="checkbox"/> Drop <input type="checkbox"/> More
5	alamat	text	utf8_general_ci		Yes	NULL		<input type="checkbox"/> Change <input type="checkbox"/> Drop <input type="checkbox"/> More	<input type="checkbox"/> Change <input type="checkbox"/> Drop <input type="checkbox"/> More

Below the table structure, there are sections for 'Indexes' and 'Partitions'.

f. Hasil dan Pembahasan

Mahasiswa membuat database “pelanggan” dengan ketentuan tabel sebagai berikut :

Table	Action	Rows	Type	Collation	Size	Overhead
barang	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	-
pelanggan	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	16.0 KiB	-
pembayaran	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	-
supplier	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	16.0 KiB	-
transaksi	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	-
transaksi_detail	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	48.0 KiB	-
Sum		6 tables			176.0 KiB	0 B

1. Tabel **pelanggan** dengan ketentuan sebagai berikut :

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
nama	varchar(100)	YES		NULL	
jenis_kelamin	enum('L','P')	YES		NULL	
telp	varchar(12)	YES		NULL	
alamat	text	YES		NULL	

2. Tabel **barang** dengan ketentuan sebagai berikut :

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
kode_barang	varchar(10)	YES		NULL	
nama_barang	varchar(100)	YES		NULL	
harga	int(11)	YES		NULL	
stok	int(11)	YES		NULL	
supplier_id	int(11)	NO	MUL	NULL	

3. Tabel **supplier** dengan ketentuan sebagai berikut :

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
nama	varchar(100)	YES		NULL	
telp	varchar(12)	YES		NULL	
alamat	text	YES		NULL	

4. Tabel **transaksi** dengan ketentuan sebagai berikut :

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
waktu_transaksi	datetime	YES		NULL	
keterangan	text	YES		NULL	
total	int(11)	YES		NULL	
pelanggan_id	int(11)	NO	MUL	NULL	

5. Tabel **transaksi_detal** dengan ketentuan sebagai berikut :

Field	Type	Null	Key	Default	Extra
transaksi_id	int(11)	NO	PRI	<i>NULL</i>	
barang_id	int(11)	NO	PRI	<i>NULL</i>	
harga	int(11)	YES		<i>NULL</i>	
qty	int(11)	YES		<i>NULL</i>	

6. Tabel **pembayaran** dengan ketentuan sebagai berikut :

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	<i>NULL</i>	auto_increment
waktu_bayar	datetime	YES		<i>NULL</i>	
total	int(11)	YES		<i>NULL</i>	
metode	enum('TUNAI','TRANSFER','EDC')	YES		<i>NULL</i>	
transaksi_id	int(11)	NO	MUL	<i>NULL</i>	

g. Kesimpulan

Mahasiswa mampu memahami penggunaan dari XAMPP dan phpMyAdmin untuk membuat database dan table SQL dalam phpMyAdmin

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 3

Pokok Bahasan	: Merancang Entity Relationship Diagram (Studi Kasus)
Acara Praktikum/Pertemuan	: Minggu 2/1
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep ERD
2. Mamahasiswa mampu memahami cara penyajian ERD
3. Mahasiswa mampu membuat ERD

b. Indikator

1. Mahasiswa dapat memahami konsep ERD
2. Mamahasiswa dapat memahami cara penyajian ERD
3. Mahasiswa dapat membuat ERD

c. Dasar Teori

Entity Relationship Diagram (ERD) merupakan suatu model penyajian data yang menggunakan Entity dan Relasioship. ERD menggambarkan model konseptual dengan struktur yang logis dari basisdata berbasis grafis. Tujuan dari penyajian ERD adalah agar database dapat dipahami dan dirancang dengan mudah.

Symbol ERD :

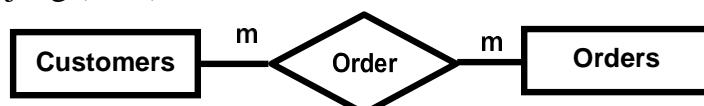
Entity

Entity adalah objek yang dapat dibedakan dalam dunia nyata. Sedangkan Entity Set adalah kumpulan

dari entitty yang sejenis. Entity set dapat berupa objek secara fisik (Rumah, Kendaraan, Peralatan) atau objek secara konsep(Pekerjaan, Perusahaan). Entity disimbolkan dengan persegi panjang ().

Relationship

Relationship adalah hubgan yang terjadi antara satu atau lebih entity. Sedangkan Relationship set adalah kumpulan relationship yang sejenis. Relationship disimbolkan dengan jajar genjang ().



Attribute

Attribute adalah karakteristik dari tiap entity atau relationship yang menyediakan penjelasan detail mengenai entity atau relationship tersebut. Nilai dari attribute adalah data aktual atau informasi yang disimpan pada suatu attribut di dalam entity atau relationship, dimana tiap attribute memiliki domain (value set) tersendiri. Domain

(value set) adalah batas-batas nilai yang diperbolehkan bagi suatu attribute. Attribute disimbolkan dengan jajar genjang ().

Jenis-jenis attribute yang digunakan dalam ERD adalah :

- *Key* : Atribut yang digunakan untuk menentukan suatu entity secara unik.
- *Atribut Simple* : Atribut sederhana yang tidak dapat dibagi dalam beberapa bagian.
- *Atribut Komposit* : Atribut yang dapat dibagi lagi dalam beberapa bagian. Contoh : Alamat yang dapat dibagi lagi menjadi Negara, Propinsi dan Kota
- *Atribut Single-valued* : Atribut yang memiliki paling banyak satu nilai untuk setiap baris data.
- *Multi-valued attributes* : Atribut yang dapat diisi dengan lebih satu nilai tetapi jenisnya sama. Contoh : Nomor Telp, Alamat, Gelar
- *Atribut Turunan* : Atribut yang diperoleh dari pengolahan dari atribut lain yang berhubungan. Contoh : Umur, IP
- *Attribute Key* : Atribut yang dapat dijadikan kunci untuk mencari data dalam relasi. Contoh: NIM Mahasiswa

Derajat Relasi

Derajat Relasi menunjukkan banyaknya himpunan entitas yang saling berelasi. Jenis derajat himpunan relasi adalah:

- Unary Degree (Derajat Satu) melibatkan sebuah entitas yang berelasi dengan dirinya sendiri
- Binary Degree(Derajat Dua) Himpunan relasi melibatkan dua himpunan entitas. Secara umum himpunan relasi dalam sistem basis data adalah binary
- Ternary Degree (Derajat Tiga) Himpunan relasi memungkinkan untuk melibatkan lebih dari dua himpunan entitas

Pemetaan Kardinalitas Relasi

Pemetaan kardinalitas relasi menggambarkan banyaknya jumlah maksimum entitas dapat berelasi dengan entitas pada himpunan entitas yang lain. Untuk Himpunan relasi biner pemetaan kardinalitasnya dapat merupakan salah satu dari tipe2 berikut :

1. Satu ke Satu (One to one)
Artinya, sebuah entity hanya dapat berelasi dengan satu buah object di entity yang lain. Kardinalitas ini disimbolkan dengan 1-1
2. Satu ke Banyak (One to many), disimbolkan dengan 1-m
Artinya, sebuah entity dapat berelasi dengan banyak object di entity yang lain. Kardinalitas ini disimbolkan dengan 1-m
3. Banyak ke Satu (Many to one), disimbolkan dengan m-1
Ini adalah kebalikan dari one to many, maksudnya banyak entity akan berelasi dengan satu objek yang sama pada entity yang lain. Kardinalitas ini disimbolkan dengan 1-m
4. Banyak ke Banyak (Many to many), disimbolkan dengan m-n
Artinya, akan ada banyak entity yang akan berelasi dengan banyak object di entity yang lain. Kardinalitas ini disimbolkan dengan m-n

Key

Penggunaan key merupakan cara untuk membedakan suatu entitas didalam himpunan entitas dengan entitas lain. Secara konsep, Masing-masing entitas (nilainya) berbeda, perbedaannya terlihat pada isi dari masing-masing atributnya. Oleh karena itu, dibutuhkan suatu atribut yang memiliki nilai yang menjadi pembeda dengan entitas lain. Key adalah satu atau gabungan dari beberapa atribut yang dapat membedakan semua row dalam relasi secara unik.

Ada 3 macam key yang dapat diterapkan pada suatu relasi :

1. Super Key

Merupakan satu atau lebih atribut (kumpulan atribut) yang dapat membedakan setiap baris data dalam sebuah relasi secara unik.

2. Candidate Key

Merupakan kumpulan atribut minimal yang dapat membedakan setiap baris data dalam sebuah relasi secara unik

3. Primary Key

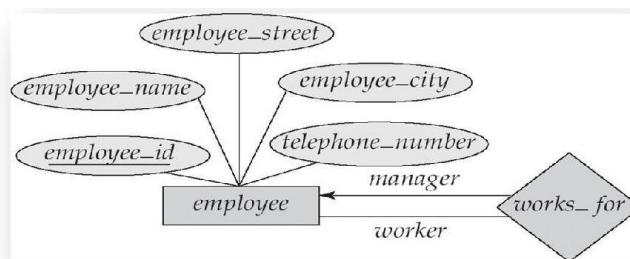
Merupakan salah satu dari candidate key yang terpilih

Pemilihan primary key dari sejumlah candidate key umumnya didasari oleh :

- a. Key tersebut lebih sering (lebih natural) untuk dijadikan sebagai acuan
- b. Key tersebut lebih ringkas
- c. Jaminan keunikan key tersebut lebih baik

Peran (Role)

Relasi himpunan entitas tidak harus dalam bentuk yang berbeda, contoh label “manager” dan “worker” disebut Roles (peran), yang menspesifikasi bagaimana entitas employee berinteraksi melalui relasi Works-for. Peran dalam ER diagram diindikasikan dengan memberikan label (nama) pada garis yang menghubungkan relasi dengan entitas. Label peran bersifat optional dan digunakan untuk mengklarifikasi semantik suatu relasi.



Dalam menggambarkan kardinalitas pada Diagram ER, digunakan :

- garis panah (\rightarrow) yang menunjukkan “Satu” atau
- garis biasa ($-$) yang menunjukkan “Banyak”, antara relasi dengan entitas

Weak Entity

Weak Entity adalah suatu entity dimana keberadaan dari entity tersebut tergantung dari keberadaan entity lain. Entity yang merupakan induknya disebut Identifying Owner dan relationship-nya Disebut Identifying Relationship. Weak Entity Selalu mempunyai Total

Participation Constraint dengan Identifying Owner. Contoh : entity tanggungan keberadaannya bergantung pada karyawan. Sehingga data pada entity tanggungan tidak dapat ada jika data dari entity karyawan tidak terisi.



e. Alat dan Bahan

1. Client Server: Xampp
2. Tools : Phpmyadmin
3. Kertas A4 / Folio Bergaris
4. Pulpen

f. Prosedur Kerja

Supermarket “SmartMarket” menjual berbagai produk seperti minuman, makanan, desserts, daging, dll. Berdasarkan skema table SmartMarket dibawah, buatlah rancangan database dalam bentuk ERD dengan relasi dan kardinalitas yang sesuai!



g. Hasil dan Pembahasan

Membuat laporan dari **ER-Diagram SmartMarket** dengan format laporan sebagai berikut :

- Terdapat cover dengan ketentuan “{Judul}”, Minggu I, logo Polije, nama, nim, golongan
- Berisi : Pendahuluan, isi makalah, kesimpulan, dan daftar Pustaka
- Format .pdf
- Simpan dengan Nama_NIM_Gol_MingguI
- Kumpulkan di <http://jti.polije.ac.id/elearning/>

h. Kesimpulan

Mahasiswa dapat merancang ERD berdasarkan studi kasus yang sudah ditentukan.

i. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 4

Pokok Bahasan	: Relasi Antar Tabel
Acara Praktikum/Pertemuan	: Minggu 2/2
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa memahami konsep relasi antar tabel
2. Mahasiswa mampu membuat relasi antar tabel
3. Mahasiswa mampu menentukan jenis foreign key (CASCADE, RESTRICT, SET NULL, dan NO ACTION) yang tepat

b. Indikator

Kemampuan mahasiswa dalam memahami dan mengimplementasikan relasi antar tabel menggunakan tools phpMyAdmin serta menentukan jenis foreign key yang tepat sesuai kebutuhan.

c. Dasar Teori

Menjelaskan cascade

Dalam merelasikan tabel, terdapat attribute yang bernilai primary key yang direlasikan dengan tabel lain dengan nama attribute yang sama namun bersifat foreign key. Primary key adalah suatu nilai dalam basis data yang digunakan untuk mengidentifikasi suatu baris tabel. Nilai dari primary key bersifat unik. Sedangkan Foreign Key merupakan attribute yang bukan merupakan atribut utama namun memiliki peran untuk menghubungkan antara tabel master dengan tabel turunannya. Jadi secara sederhana, penjelasan dari foreign key adalah sebuah attribute atau gabungan atribut yang terdapat dalam suatu tabel yang digunakan untuk menciptakan hubungan (relasi) antara dua tabel disebut juga sebagai **kunci asing**.

Terdapat 4 pilihan pengaturan (option) dalam foreign key diantaranya adalah :

1. CASCADE

Penggunaan CASCADE diperlukan jika kita menghapus atau merubah baris data dalam tabel A secara otomatis akan merubah baris pada tabel B menjadi NULL pada kolom yang terelasi. Hal ini dapat dilakukan jika kolom foreign key tidak memiliki pengaturan NOT NULL.

2. RESTRICT

Penggunaan RESTRICT diperlukan jika kita menghapus atau merubah baris data dalam tabel A maka tidak akan diperbolehkan jika pada tabel B masih ditemukan relasi datanya. InnoDB menolak perintah perubahan atau penghapusan tersebut.

3. SET NULL

Penggunaan SET NULL diperlukan jika kita menghapus atau merubah baris data dalam tabel A secara otomatis akan merubah baris pada tabel B menjadi NULL pada kolom yang terelasi. Hal ini dapat dilakukan jika kolom foreign key tidak memiliki pengaturan NOT NULL.

4. NO ACTION

Penggunaan NO ACTION dalam standar SQL, NO ACTION berarti tidak merubah apapun pada tabel anak jika kita merubah data pada salah satu tabelnya.

d. Alat dan Bahan

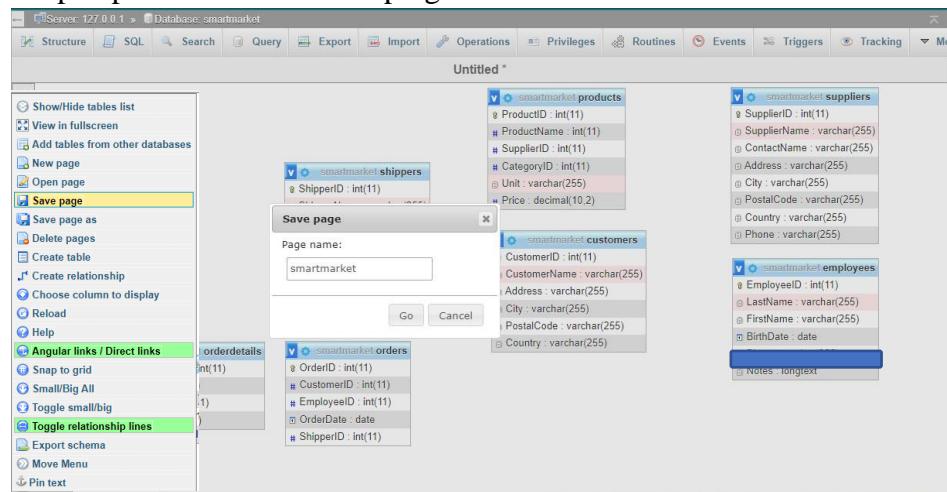
1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

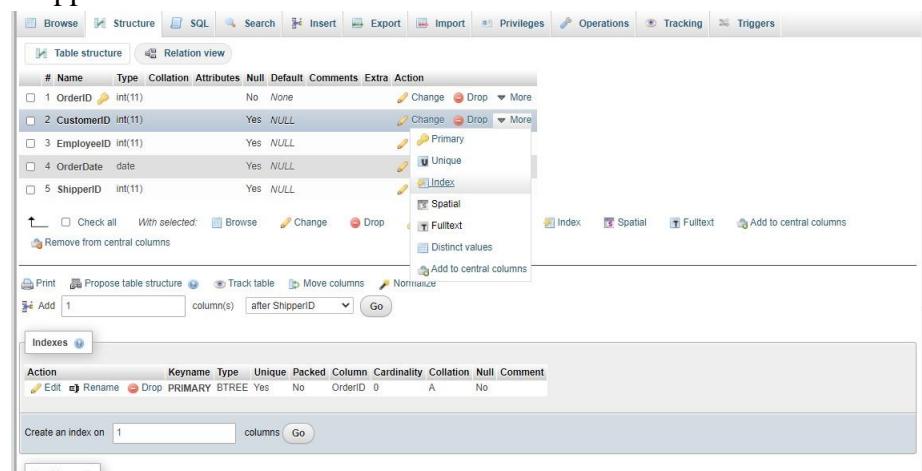
1. Buat Database dengan nama smartmarket menggunakan tools phpMyAdmin. Tabel – tabel dari database smartmarket dibuat berdasarkan ERD yang sudah dirancang pada pertemuan sebelumnya.
2. Ada beberapa cara untuk merelasikan tabel dengan phpMyAdmin, diantaranya dapat melalui Designer, dapat melalui relation view yang ada pada Structure tabel yang ingin direlasikan, dan dapat menggunakan syntax SQL
3. Untuk membuat relasi tabel melalui Designer, dapat dengan langsung klik Designer atau klik more lalu pilih Designer.

Table	Action	Rows	Type	Collation	Size	Overhead
categories	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	16.0 Kib	-
customers	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	16.0 Kib	-
employees	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	16.0 Kib	-
orderdetails	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	48.0 Kib	-
orders	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	64.0 Kib	-
products	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	48.0 Kib	-
shippers	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	16.0 Kib	-
suppliers	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	16.0 Kib	-
8 tables	Sum				240.0 Kib	0 B

4. Rapikan tabel tabel – tabel agar lebih mudah untuk direlasikan nantinya dan jangan lupa untuk menyimpannya terlebih dahulu dengan nama smartmarket dengan klik symbol simpan pada menu bar disamping.



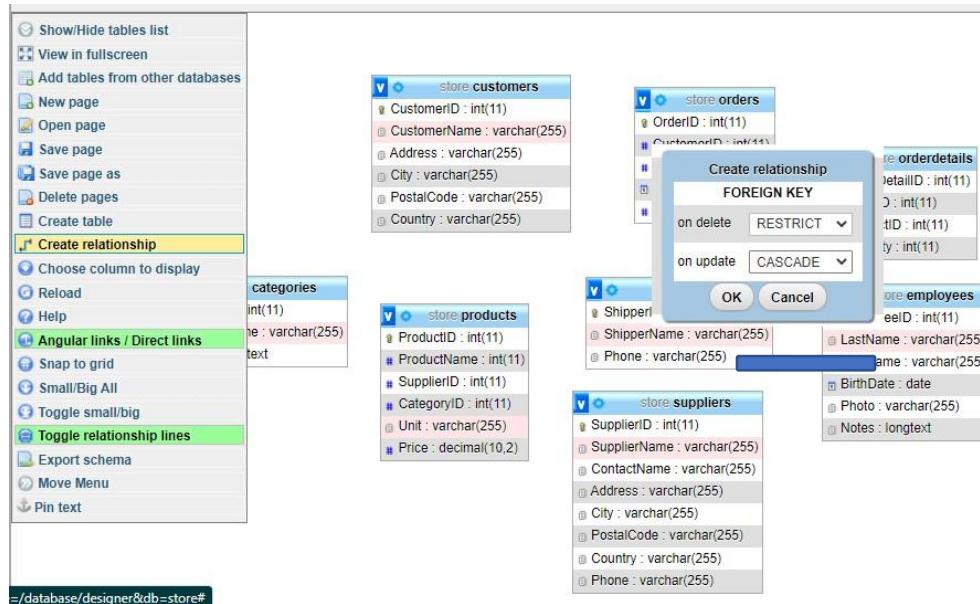
5. Buat index terlebih dahulu dari attribut yang akan dibuat foreign key agar tabel-tabel dapat direlasikan. Misal di table order lakukan index pada field CustomerID, EmployeeID, dan ShipperID.



Sehingga dapat dilihat hasil attribute yang telah di index seperti dibawah ini

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□ 1	OrderID	int(11)			No	None			
□ 2	CustomerID	int(11)			Yes	NULL			
□ 3	EmployeeID	int(11)			Yes	NULL			
□ 4	OrderDate	date			Yes	NULL			
□ 5	ShipperID	int(11)			Yes	NULL			

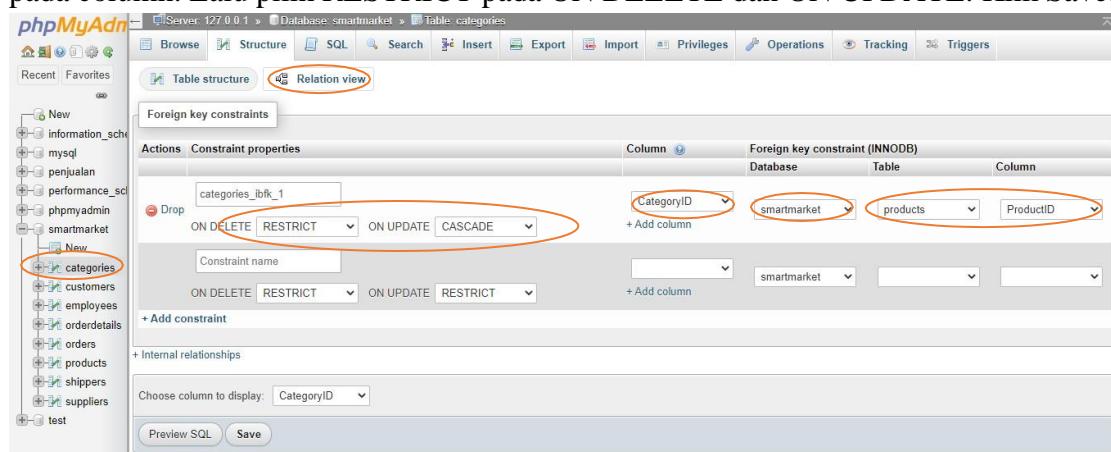
6. Klik Create relationship lalu klik pada attribute yang memiliki Primary Key pada tabel utama kemudian klik attribute foreign key yang sesuai pada tabel yang ingin direlasikan. Untuk keterangan jenis aturan foreign key dapat dilihat pada dasar teori.
Contoh pada tabel Customer akan direlasikan dengan tabel order, diatur foreign key dengan delete = “RESTRICT” dan update = “CASCADE”.



7. Terdapat cara lain untuk merelasikan tabel adalah dengan memilih tabel yang akan direlasikan lalu.

Contoh : tabel Categories akan direlasikan dengan tabel products dengan aturan foreign key untuk delete = RESTRICT dan update = RESTRICT

pilih tabel Categories lalu klik Structure – Relation view. Kemudian pilih tabel yang akan direlasikan yaitu tabel products, maka attribute ProductID akan secara otomatis terpilih pada column. Lalu pilih RESTRICT pada ON DELETE dan ON UPDATE. Klik Save.

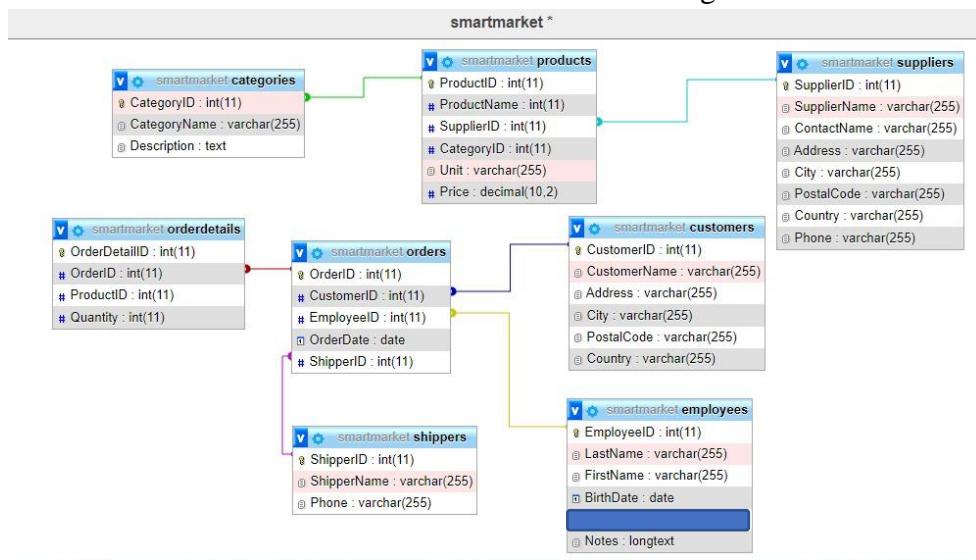


8. Cara lain untuk merelasikan tabel adalah dengan syntax SQL

```
ALTER TABLE `categories` ADD FOREIGN KEY (`CategoryID`) REFERENCES `products`(`ProductID`) ON DELETE RESTRICT ON UPDATE CASCADE;
```

f. Hasil dan Pembahasan

Tabel – tabel dalam database smartmarket terelasi dengan benar.



g. Kesimpulan

Mahasiswa berhasil merelasikan tabel-tabel dengan pengaturan jenis foreign key yang tepat berdasarkan studi kasus smartmarket.

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak seusai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 5

Pokok Bahasan	: DDL (Create Statement)
Acara Praktikum/Pertemuan	: Minggu 3/1
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa memahami konsep alter tabel
2. Mahasiswa mampu mengimplementasikan alter tabel

b. Indikator

Kemampuan mahasiswa dalam memahami dan mengimplementasikan alter tabel menggunakan tools phpMyAdmin

c. Dasar Teori

SQL Create Statement

Terdapat dua jenis SQL Create Statement diantaranya adalah create database dan create table.

Pernyataan CREATE DATABASE digunakan untuk membuat database SQL baru. Sedangkan pernyataan CREATE TABLE digunakan untuk membuat tabel baru dalam database.

Berikut adalah aturan syntax CREATE DATABASE :

```
CREATE DATABASE databasename;
```

Contoh pernyataan SQL untuk membuat database yang disebut "testDB" adalah dengan

menggunakan syntax sebagai berikut :

```
CREATE DATABASE testDB;
```

Berikut adalah aturan syntax pernyataan CREATE TABLE digunakan untuk membuat tabel baru dalam database :

```
CREATE TABLE table_name (
    column1 datatype,
    column2 datatype,
    column3 datatype,
    ...
);
```

Keterangan :

- Parameter kolom menentukan nama kolom tabel.
- Parameter tipe data menentukan tipe data yang dapat disimpan oleh kolom (ex. varchar, integer, tanggal, dll.).
- Tips: Untuk gambaran umum tentang tipe data yang tersedia, buka Referensi Tipe Data lengkap.

Buat Tabel Menggunakan Tabel Lain (Create Table Using Another Table)

Dalam membuat tabel menggunakan tabel lain terdapat beberapa pernyataan yang perlu diperhatikan diantaranya adalah :

1. Salinan tabel yang ada dapat dibuat menggunakan CREATE TABLE.
2. Penamaan kolom pada tabel baru harus sesuai dengan penamaan kolom yang ada pada tabel lama. Kolom pada tabel baru dapat dibuat beberapa atau semua sesuai kebutuhan.
3. Jika Anda membuat tabel baru menggunakan tabel yang sudah ada, maka tabel baru akan diisi dengan nilai yang ada dari tabel lama.
4. Tabel baru yang dibuat sifatnya tidak akan menghapus tabel lama yang sudah ada.

Berikut adalah aturan syntax untuk membuat tabel menggunakan tabel yang lain :

```
CREATE TABLE new_table_name AS
    SELECT column1, column2, ...
    FROM existing_table_name
    WHERE ....;
```

Berikut adalah contoh pembuatan tabel baru menggunakan tabel lain yaitu tabel “TestTable” dengan kolom CustomerName dan Country yang menggunakan tabel “Customers” :

Syntax :

```
MySQL returned an empty result set (i.e. zero rows). (Query took 0.0340 seconds.)  
CREATE TABLE TestTable AS SELECT CustomerName, Country FROM customers;
```

Tampilan :

CustomerName	Country
Alfreds Futterkiste	Germany
Ana Trujillo Emparedados y helados	Mexico
Antonio Moreno Taquería	Mexico
Around the Horn	UK
Berglunds snabbköp	Sweden
Blauer See Delikatessen	Germany
Blondel père et fils	France
Bólido Comidas preparadas	Spain
Bon app	France
Bottom-Dollar Marketse	Canada
B's Beverages	UK
Cactus Comidas para llevar	Argentina
Centro comercial Moctezuma	Mexico
Chop-suey Chinese	Switzerland
Comércio Mineiro	Brazil
Consolidated Holdings	UK
Drachenblut Delikatessend	Germany
Du monde entier 67	France
Eastern Connection	UK
Ernst Handel	Austria

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

1. Buat tabel dengan nama "Persons" yang berisi 5 (lima) kolom diantaranya adalah : PersonID, LastName, FirstName, Address, dan City dengan ketentuan sebagai berikut :
 - Kolom PersonID bertipe int dan akan menampung bilangan bulat.
 - Kolom LastName, FirstName, Address, dan City bertipe varchar dan akan menampung karakter, dan panjang maksimum untuk bidang ini adalah 255 karakter.
 - Tabel "Orang" yang kosong sekarang akan terlihat seperti ini:

PersonID	LastName	FirstName	Address	City

2. Buat tabel “BirthDateEmployees” untuk mengetahui BirthDate dari Employees dengan ketentuan beberapa kolom diantaranya adalah : EmployeeID, FirstName, LastName, dan BirthDate.

EmployeeID	FirstName	LastName	BirthDate
1	Nancy	Davolio	1968-08-12
2	Andrew	Fuller	1952-02-19
3	Janet	Leverling	1963-08-30
4	Margaret	Peacock	1958-09-19
5	Steven	Buchanan	1955-03-04
6	Michael	Suyama	1963-02-07
7	Robert	King	1960-05-29
8	Laura	Callahan	1958-01-09
9	Anne	Dodsworth	1969-07-02
10	Adam	West	1928-09-19

f. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)
2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf

Kumpulkan pada <http://iti.polije.ac.id/elearning/>

g. Kesimpulan

Mahasiswa berhasil membuat jenis – jenis CREATE Statement sesuai ketentuan.

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 6

Pokok Bahasan	: DDL (Alter Table Statement)
Acara Praktikum/Pertemuan	: Minggu 3/2
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

3. Mahasiswa memahami konsep alter tabel
4. Mahasiswa mampu mengimplementasikan alter tabel

b. Indikator

Kemampuan mahasiswa dalam memahami dan mengimplementasikan alter tabel menggunakan tools phpMyAdmin

c. Dasar Teori

ALTER TABLE Statement merupakan pernyataan yang digunakan untuk menambah, menghapus, atau memodifikasi kolom dalam tabel yang sudah ada. Pernyataan ALTER TABLE juga digunakan untuk menambah dan menghapus berbagai batasan pada tabel yang ada.

Berikut beberapa contoh syntax yang ada pada ALTER TABLE Statement adalah sebagai berikut :

ALTER TABLE - ADD Column digunakan untuk mengubah tabel dengan menambahkan kolom baru

1. Untuk menambah sebuah kolom didalam sebuah tabel gunakan syntax berikut :

```
ALTER TABLE table_name  
ADD column_name datatype;
```

Contoh syntax SQL untuk menambahkan kolom “Email” di tabel “Customers”

```
ALTER TABLE Customers  
ADD Email varchar(255);
```

2. Alternatif lain dapat langsung melalui tools di PhpMyAdmin dengan tahapan sebagai berikut :

- Klik tabel yang ingin ditambahkan attributnya lalu klik Structure.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	CustomerID	int(11)			No	None			Change Drop More
2	CustomerName	varchar(255)	utf8mb4_general_ci		Yes	NULL			Change Drop More
3	Address	varchar(255)	utf8mb4_general_ci		Yes	NULL			Change Drop More
4	City	varchar(255)	utf8mb4_general_ci		Yes	NULL			Change Drop More
5	PostalCode	varchar(255)	utf8mb4_general_ci		Yes	NULL			Change Drop More
6	Country	varchar(255)	utf8mb4_general_ci		No	None			Change Drop More

- Pada bagian bawah terdapat Add untuk menentukan banyaknya kolom (column(s)) yang ingin ditambah. Sedangkan column untuk menentukan peletakkan attribute tersebut. Misal akan ditambahkan kolom email di tabel Customer maka isikan 1 pada kolom setelah attribute Country lalu klik Go

Add column(s) Go

Indexes

Action	Keyname	Type
Edit	Rename	Drop
	PRIMARY	BTREE

after Country

- at beginning of table
- after CustomerID
- after CustomerName
- after Address
- after City
- after PostalCode
- after Country

- Isikan nama attribute "Email" dengan type data "VARCHAR 255" lalu klik Save

VARCHAR None

Structure

Online transaction Preview SQL Save

- d. Berikut hasil attribute yang berhasil ditambah

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	CustomerID	int(11)		No	None				Change Drop More
2	CustomerName	varchar(255)	utf8mb4_general_ci	Yes	NULL				Change Drop More
3	Address	varchar(255)	utf8mb4_general_ci	Yes	NULL				Change Drop More
4	City	varchar(255)	utf8mb4_general_ci	Yes	NULL				Change Drop More
5	PostalCode	varchar(255)	utf8mb4_general_ci	Yes	NULL				Change Drop More
6	Country	varchar(255)	utf8mb4_general_ci	No	None				Change Drop More
7	Email	varchar(255)	utf8mb4_general_ci	No	None				Change Drop More

1. Penggunaan ALTER TABLE – ADD juga dapat diimplementasikan pada penambahan primary key pada satu kolom dalam tabel tersebut dengan syntax SQL sebagai berikut :

```
ALTER TABLE table_name
ADD PRIMARY KEY (column_name);
```

ALTER TABLE - DROP COLUMN digunakan untuk merubah tabel dengan menghapus kolom yang ada didalamnya

1. Untuk menghapus kolom dalam tabel, gunakan sintaks berikut (perhatikan bahwa beberapa sistem database tidak mengizinkan penghapusan kolom) :

```
ALTER TABLE table_name
DROP COLUMN column_name;
```

2. Contoh syntax SQL untuk menghapus kolom "Email" dari tabel "Pelanggan" adalah sebagai berikut :

```
ALTER TABLE Customers
DROP COLUMN Email;
```

Maka kolom Email akan terhapus dari tabel “Customers”.

ALTER TABLE - MODIFY COLUMN digunakan untuk merubah tabel dengan merubah kolom yang dingin dirubah dalam tabel tersebut

1. Untuk mengubah tipe data kolom dalam tabel, dapat menggunakan sintaks berikut :

```
ALTER TABLE table_name
MODIFY COLUMN column_name datatype;
```

d. Alat dan Bahan

5. Client Server: Xampp
6. Text Editor: PhpMyAdmin
7. Kertas A4 / Folio
8. Pulpen

e. Prosedur Kerja

Lihat Structure tabel Employees

1	EmployeeID	int(11)	No	None
2	LastName	varchar(255) utf8mb4_general_ci	Yes	NULL
3	FirstName	varchar(255) utf8mb4_general_ci	Yes	NULL
4	BirthDate	date	Yes	NULL
5	photo	varchar(255) utf8mb4_general_ci	No	None
6	Notes	longtext utf8mb4_general_ci	No	None

1. Tambahkan attribute foto pada tabel employees yang diletakkan setelah BirthDate dengan type data varchar(255).
2. Tambahkan Primary Key pada PersonID di tabel Persons
3. Rubah type data pada kolom DateOfBirth menjadi year di tabel Persons

f. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)
2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdfKumpulkan pada

<http://iti.polije.ac.id/elearning/>

g. Kesimpulan

Mahasiswa berhasil mengimplementasikan jenis – jenis alter tabel sesuai ketentuan.

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 7

Pokok Bahasan	: DDL (Rename Table Statement)
Acara Praktikum/Pertemuan	: Minggu 3/3
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa memahami konsep rename pada tabel
2. Mahasiswa mampu mengimplementasikan penggunaan rename pada tabel

b. Indikator

Kemampuan mahasiswa dalam memahami dan mengimplementasikan penggunaan rename pada tabel menggunakan tools phpMyAdmin

c. Dasar Teori

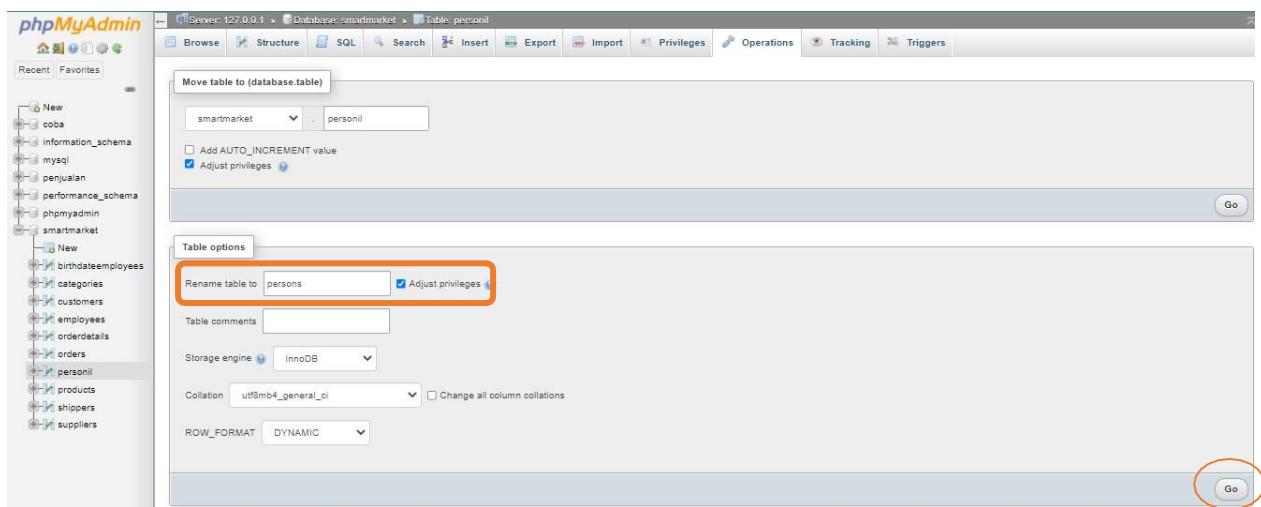
RENAME TABLE Statement merupakan pernyataan yang digunakan untuk merubah sebuah nama tabel yang terdapat pada sebuah database. Perlu diingat RENAME ini tidak bisa digunakan untuk merubah sebuah field pada table. Untuk merubah sebuah field pada database harus menggunakan ALTER yang sudah dijelaskan pada materi sebelumnya.

RENAME TABLE berguna ketika kita mengalami typo saat membuat atau menulis nama tabel. Dengan RENAME TABLE kita dapat merubah nama tabel tanpa harus menghapus tabel tersebut.

Berikut syntax dari RENAME TABLE :

```
RENAME TABLE table_name_old TO table_name_new;
```

Jika menggunakan tools pada PhpMyAdmin anda bisa menggunakan menubar Operations. Misal terdapat tabel dengan nama “personil” akan dirubah menjadi “persons” klik tabel yang akan di rubah Namanya (personil), lalu klik Operations selanjutnya anda bisa mengganti nama tabel pada field Rename table to menjadi “persons” lalu klik GO



d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

1. Implementasikan penggunaan RENAME TABLE pada salah satu tabel dari database smartmarket
2. Rubah nama tabel “customers” menjadi “pelanggan” dengan menggunakan syntax SQL lalu screenshot bukti hasil RENAME TABLE.
3. Rubah nama tabel “pelanggan” menjadi “Customers” dengan menggunakan tools pada PhpMyAdmin lalu screenshot bukti hasil RENAME TABLE

f. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)
2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf

Kumpulkan pada <http://jti.polije.ac.id/elearning/>

g. Kesimpulan

Mahasiswa berhasil mengimplementasikan penggunaan RENAME TABLE.

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 8

Pokok Bahasan	: DDL (Drop Table and Database Statement)
Acara Praktikum/Pertemuan	: Minggu 3/4
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

3. Mahasiswa memahami konsep drop table statement tabel
4. Mahasiswa mampu mengimplementasikan drop table statement

b. Indikator

Kemampuan mahasiswa dalam memahami dan mengimplementasikan drop table statement pada phpMyAdmin

c. Dasar Teori

Perintah Drop berfungsi untuk menghapus sebuah tabel maupun juga database, function, index, procedure, trigger, view.

Untuk menghapus database di MySQL, gunakan statement DROP DATABASE diikuti dengan nama database yang ingin dihapus. Sehingga ada baiknya, membackup terlebih dahulu database tersebut sebelum menghapusnya.

Berdasarkan penjelasan diatas, DROP terdiri dari 2 (dua) jenis diantaranya adalah :

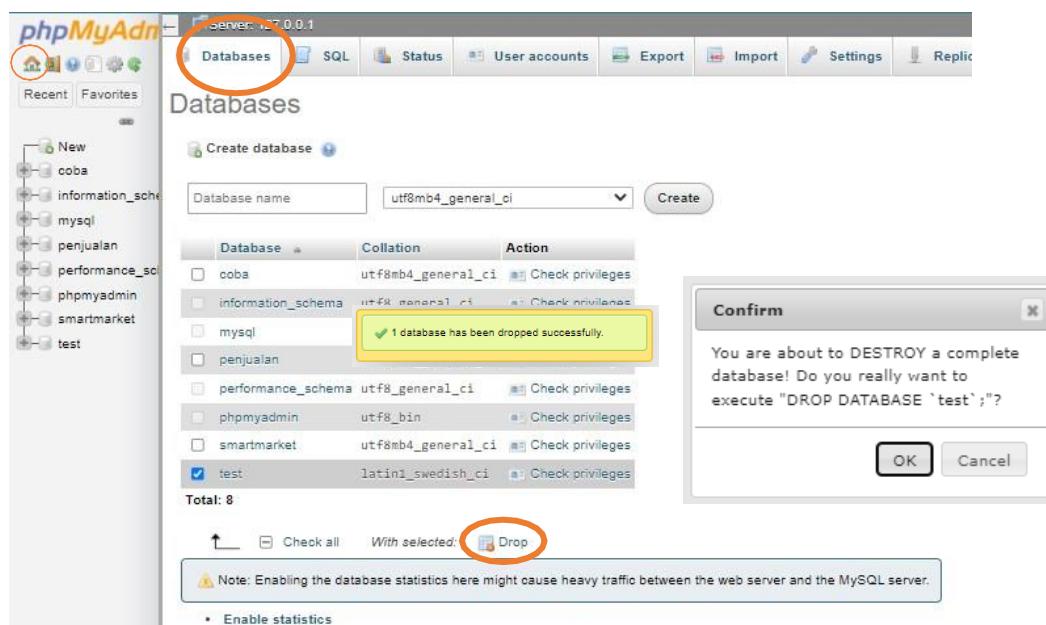
1. DROP DATABASE

Adakalanya database dihapus karena tidak diperlukan lagi. Penghapusan suatu database, mengakibatkan penghapusan semua tabel, data, dan relasi didalamnya.

Untuk menghapus database menggunakan query SQL adalah sebagai berikut :

DROP DATABASE *databasename*;

Untuk menghapus database dari tools PhpMyAdmin, klik home lalu klik databases selanjutnya centang database yang ingin di hapus lalu klik drop kemudian muncul notifikasi sebagai konfirmasi database akan dihapus lalu klik OK maka muncul notifikasi bahwa database berhasil di hapus.



2. DROP TABLE

Terdapat beberapa jenis hapus tabel diantaranya adalah :

a. The SQL DROP TABLE Statement

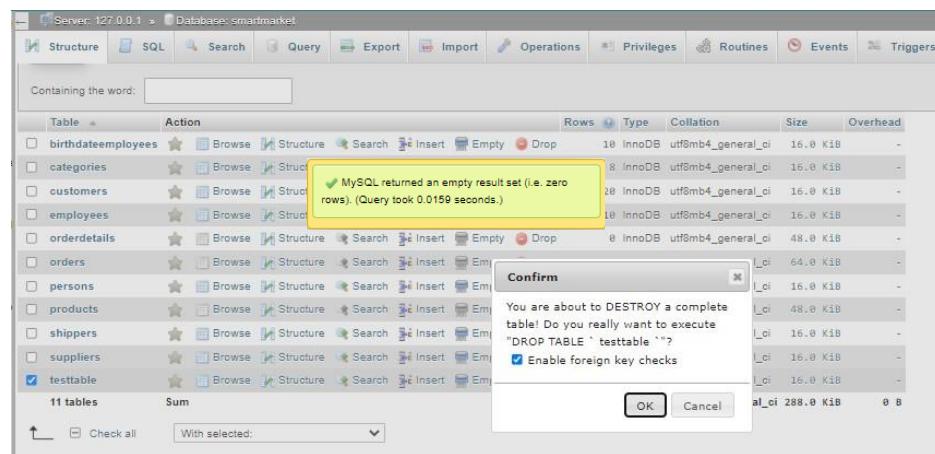
SQL DROP TABLE Statement merupakan pernyataan DROP TABLE yang digunakan untuk menjatuhkan/menghapus tabel yang ada dalam database. Selain DROP TABLE ada pula statement yang digunakan untuk menghapus data di dalam tabel, tetapi bukan tabel itu sendiri yang disebut sebagai SQL TRUNCATE TABLE.

Berikut merupakan syntax SQL untuk DROP TABLE :

```
DROP TABLE table_name;
```

Untuk menghapus tabel menggunakan tools PhpMyAdmin adalah sebagai berikut :

1. klik pada database yang diinginkan
2. centang tabel yang ingin dihapus
3. klik drop pada bagian tabel yang ingin dihapus hingga muncul notifikasi konfirmasi penghapusan tabel lalu klik OK hingga muncul notifikasi bahwa tabel telah berhasil dihapus.



a. The SQL TRUNCATE TABLE Statement

Berikut merupakan syntax SQL TRUNCATE TABLE :

```
TRUNCATE TABLE table_name;
```

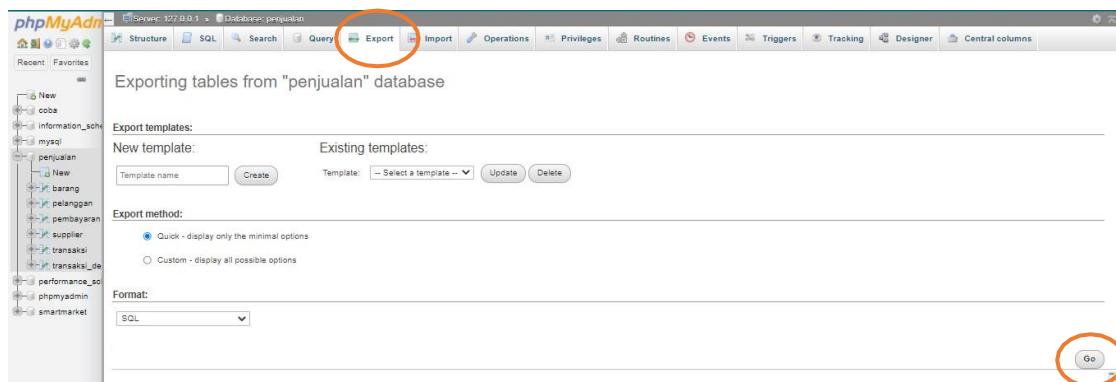
d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

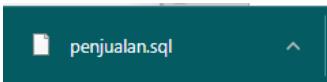
e. Prosedur Kerja

Menghapus Database

1. Buka database penjualan yang sudah dibuat pada pertemuan sebelumnya.
2. Backup database terlebih dahulu sebelum dihapus dengan cara klik export lalu Go



3. Database penjualan telah terdownload lalu simpan di folder yang diinginkan



4. Hapus database penjualan pada PhpMyAdmin
5. Screen shot bukti bahwa database telah terhapus
6. Atau dapat ketik syntax **SHOW DATABASES** untuk cek apakah database “penjualan” masih ada atau tidak. Jika tidak maka database “penjualan” berhasil terhapus.

Menghapus Tabel

1. Buka database smartmarket lalu hapus tabel

```
DROP TABLE persons;
```

```
DROP TABLE BirthDateEmployees;
```

2. Perlihatkan bukti bahwa tabel “persons” dan tabel “birthdateemployees” telah terhapus dengan syntax **SHOW TABLES**
3. Screenshot bukti bahwa tabel berhasil dihapus.

f. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)
2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf

Kumpulkan pada <http://jti.polije.ac.id/elearning/>

g. Kesimpulan

Mahasiswa berhasil menampilkan select tabel sesuai ketentuan.

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak seusai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 9

Pokok Bahasan	: DML (Insert INTO Statement)
Acara Praktikum/Pertemuan	: Minggu 4/1
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa memahami konsep INSERT INTO Statement pada tabel
2. Mahasiswa mampu mengimplementasikan penggunaan INSERT INTO Statement

b. Indikator

Kemampuan mahasiswa dalam memahami dan mengimplementasikan penggunaan INSERT INTO Statement menggunakan tools phpMyAdmin

c. Dasar Teori

INSERT INTO merupakan statement yang digunakan untuk menambah record baru pada suatu tabel

Dimungkinkan untuk menulis pernyataan INSERT INTO dengan dua cara :

1. Tentukan nama kolom dan nilai yang akan disisipkan :

```
INSERT INTO table_name (column1, column2, column3, ...)
VALUES (value1, value2, value3, ...);
```

2. Jika Anda menambahkan nilai untuk semua kolom tabel, Anda tidak perlu menentukan nama kolom dalam kueri SQL. Namun, pastikan urutan nilai dalam urutan yang sama dengan kolom dalam tabel. Di sini, sintaks INSERT INTO adalah sebagai berikut:

```
INSERT INTO table_name  
VALUES (value1, value2, value3, ...);
```

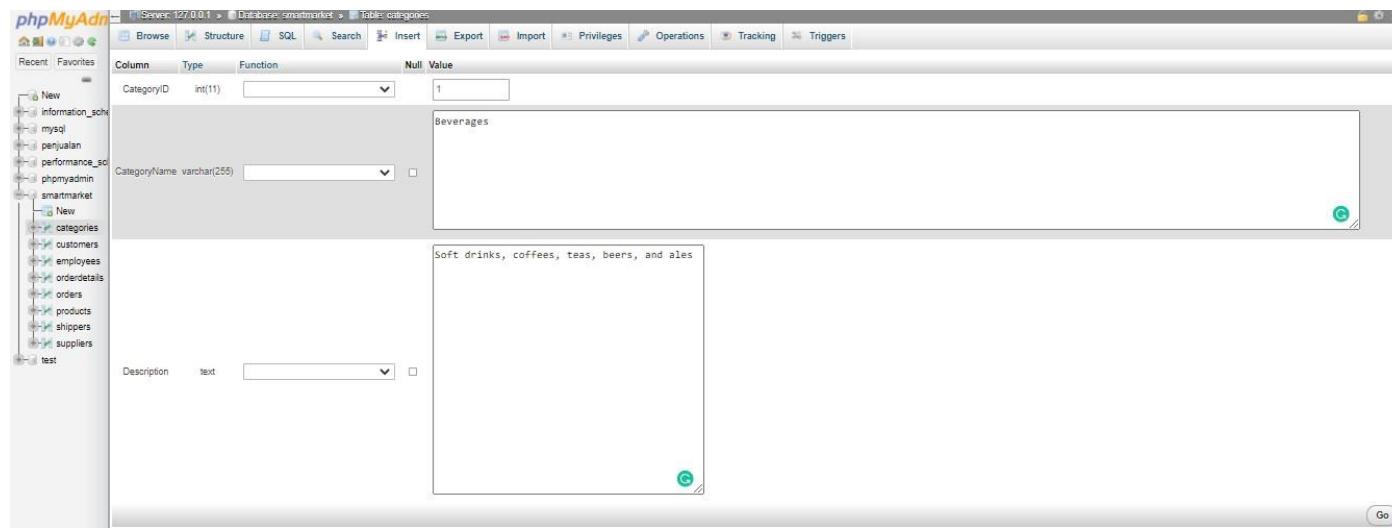
Pernyataan SQL berikut menyisipkan catatan baru di tabel "Customers":

```
INSERT INTO Customers (CustomerName, ContactName, Address, City, PostalCode, Country)  
VALUES ('Cardinal', 'Tom B. Erichsen', 'Skagen 21', 'Stavanger', '4006', 'Norway');
```

Terdapat dua cara Insert tabel menggunakan tools PhpMyAdmin yaitu dapat melalui tools insert atau dapat menggunakan query SQL.

Cara Insert tabel menggunakan tools phpMyAdmin adalah sebagai berikut :

1. Klik tabel yang ingin ditambahkan record baru lalu klik insert pada menu bar atas. Contoh disini akan menambahkan record baru pada tabel categories. Isikan record baru pada tabel categories dalam attribute yang ada-klik Go di pojok kanan bawah.



Continue insertion with rows

2. Continue insertion with ... rows digunakan untuk menentukan banyaknya record yang ingin ditambah.

3. Setelah klik Go akan muncul tampilan seperti dibawah dan itu menadakan bahwa record baru sudah tertambah jadi tidak perlu klik go lagi dan langsung klik pada tabel categories untuk memastikan bahwa record baru telah berhasil di tambah (insert)

The screenshot shows the MySQL Workbench interface with the following details:

- Server: 127.0.0.1 > Database: smartmarket > Table: categories
- Toolbar: Browse, Structure, SQL, Search, Insert, Export, Import, Privileges, Operations, Tracking, Triggers.
- Message Bar: 1 row inserted.
- SQL Editor: INSERT INTO `categories` (`CategoryID`, `CategoryName`, `Description`) VALUES ('1', 'Beverages', 'Soft drinks, coffees, teas, beers, and ales');
- Buttons: [Edit inline] [Edit] [Create PHP code]
- Text Area: Run SQL query/queries on table smartmarket.categories: 1 INSERT INTO `categories` (`CategoryID`, `CategoryName`, `Description`) VALUES ('1', 'Beverages', 'Soft drinks, coffees, teas, beers, and ales');
- Table View: CategoryID, CategoryName, Description. One row is displayed: CategoryID: 1, CategoryName: Beverages, Description: Soft drinks, coffees, teas, beers, and ales.
- Action Buttons: Edit, Copy, Delete, Check all, With selected: Edit, Copy, Delete, Export.

Cara insert tabel menggunakan query SQL adalah sebagai berikut :

- Untuk menambahkan record tabel melalui query SQL bisa langsung ketik syntax di menu bar SQL lalu klik Go

```
4 (3, 'Leverling', 'Janet', '1963-8-30', 'Jane has a BS degree in chemistry from Boston College. She has also completed a certificate program in food retailing management. Janet was hired as a sales associate and was promoted to sales representative.'),  
5 (4, 'Peacock', 'Margaret', '1958-9-19', 'Margaret holds a BA in English literature from Concordia College and an MA from the American Institute of Culinary Arts. She was temporarily assigned to the London office before returning to her permanent post in Seattle.'),  
6 (5, 'Buchanan', 'Steven', '1955-3-4', 'Steven Buchanan graduated from St. Andrews University, Scotland, with a BSC degree. Upon joining the company as a sales representative, he spent 6 months in an orientation program at the Seattle office and then returned to his permanent post in London, where he was promoted to sales manager. Mr. Buchanan has completed the courses "Successful Telemarketing" and "International Sales Management". He is fluent in French.'),  
7 (6, 'Suyama', 'Michael', '1963-2-7', 'Michael is a graduate of Sussex University (MA, economics) and the University of California at Los Angeles (MBA, marketing). He has also taken the courses "Multi-Cultural Selling" and "Time Management for the Sales Professional". He is fluent in Japanese and can read and write French, Portuguese, and Spanish.'),  
8 (7, 'King', 'Robert', '1960-5-29', 'Robert King served in the Peace Corps and traveled extensively before completing his degree in English at the University of Michigan and then joining the company. After completing a course entitled "Selling in Europe", he was transferred to the London office.'),  
9 (8, 'Callahan', 'Laura', '1958-1-9', 'Laura received a BA in psychology from the University of Washington. She has also completed a course in business French. She reads and writes French.'),  
10 (9, 'Dodsworth', 'Anne', '1969-7-2', 'Anne has a BA degree in English from St. Lawrence College. She is fluent in French and German.'),  
11 (10, 'West', 'Adam', '1928-9-19', 'An old chum.')
```

EmployeeID
LastName
FirstName
BirthDate
Photo
Notes

SELECT * | SELECT | INSERT | UPDATE | DELETE | Clear | Format | Get auto-saved query | << | Go | Delimiter : | Show this query here again | Retain query box | Rollback when finished | Enable foreign key checks

2. Data berhasil tersimpan dapat dibuktikan dengan hasil dibawah

The screenshot shows the MySQL Workbench interface with the following details:

- Server: 127.0.0.1
- Database: smartmarket
- Table: employees
- Toolbar buttons: Browse, Structure, SQL, Search, Insert, Export, Import, Privileges, Operations, Tracking, Triggers.
- Message bar: "Show query box" and "10 rows inserted. (Query took 0.0176 seconds.)"
- Query results pane:

```
INSERT INTO Employees (EmployeeID, LastName, FirstName, BirthDate, Notes) VALUES (1, 'Davolio', 'Nancy', '1968-8-12', 'Education includes a BA in psychology from Colorado State University. She also completed ("The Art of the Cold Call"). Nancy is a member of Toastmasters International'), (2, 'Fuller', 'Andrew', '1952-2-19', 'Andrew received his BTS commercial and a Ph.D. in international marketing from the University of Dallas. He is fluent in French and Italian and reads German. He joined the company as a sales representative, was promoted to sales manager and was then named vice president of sales. Andrew is a member of the Sales Management Roundtable, the Seattle Chamber of Commerce, and the Pacific Rim Importers Association.'), (3, 'Leverling', 'Janet', '1963-8-30', 'Janet has a BS degree in chemistry from Boston College. She has also completed a certificate program in food retailing management. Janet was hired as a sales associate and was promoted to sales representative.'), (4, 'Peacock[...]
```
- Bottom left: "[Edit]" button.

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

Isi data di tabel dari database yang sudah dibuat pada pertemuan sebelumnya dengan ketentuan sebagai berikut :

1. Tabel Customers

CustomerID	CustomerName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Avda. de la Constitución 2222	México D.F.	5021	Mexico
3	Antonio Moreno Taquería	Mataderos 2312	México D.F.	5023	Mexico
4	Around the Horn	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Berguvsvägen 8	Luleå	S-958 22	Sweden

6	Blauer See Delikatessen	Forsterstr. 57	Mannheim	68306	Germany
7	Blondel père et fils	24, place Kléber	Strasbourg	67000	France
8	Bólido Comidas preparadas	C/ Araquil, 67	Madrid	28023	Spain
9	Bon app'	12, rue des Bouchers	Marseille	13008	France
10	Bottom-Dollar Marketse	23 Tsawassen Blvd.	Tsawassen	T2F 8M4	Canada
11	B's Beverages	Fauntleroy Circus	London	EC2 5NT	UK
12	Cactus Comidas para llevar	Cerrito 333	Buenos Aires	1010	Argentina
13	Centro comercial Moctezuma	Sierras de Granada 9993	México D.F.	5022	Mexico
14	Chop-suey Chinese	Hauptstr. 29	Bern	3012	Switzerland
15	Comércio Mineiro	Av. dos Lusíadas, 23	São Paulo	05432-043	Brazil
16	Consolidated Holdings	Berkeley Gardens 12 Brewery	London	WX1 6LT	UK
17	Drachenblut Delikatessend	Walserweg 21	Aachen	52066	Germany
18	Du monde entier	67, rue des Cinquante Otages	Nantes	44000	France
19	Eastern Connection	35 King George	London	WX3 6FW	UK
20	Ernst Handel	Kirchgasse 6	Graz	8010	Austria

2. Tabel Categories

CategoryID	CategoryName	Description
1	Beverages	Soft drinks, coffees, teas, beers, and ales
2	Condiments	Sweet and savory sauces, relishes, spreads, and seasonings
3	Confections	Desserts, candies, and sweet breads
4	Dairy Products	Cheeses

5	Grains/Cereals	Breads, crackers, pasta, and cereal
6	Meat/Poultry	Prepared meats
7	Produce	Dried fruit and bean curd
8	Seafood	Seaweed and fish

3. Tabel Employees

EmployeeID	LastName	FirstName	BirthDate	Notes
1	Davolio	Nancy	12/8/1968	Education includes a BA in psychology from Colorado State University. She also completed (The Art of the Cold Call). Nancy is a member of 'Toastmasters International'.
2	Fuller	Andrew	2/19/1952	Andrew received his BTS commercial and a Ph.D. in international marketing from the University of Dallas. He is fluent in French and Italian and reads German. He joined the company as a sales representative, was promoted to sales manager and was then named vice president of sales. Andrew is a member of the Sales Management Roundtable, the Seattle Chamber of Commerce, and the Pacific Rim Importers Association.
3	Leverling	Janet	8/30/1963	Janet has a BS degree in chemistry from Boston College). She has also completed a certificate program in food retailing management. Janet was hired as a sales associate and was promoted to sales representative.
4	Peacock	Margaret	9/19/1958	Margaret holds a BA in English literature from Concordia College and an MA from the American Institute of Culinary Arts. She was temporarily assigned to the London office before returning to her permanent post in Seattle.
5	Buchanan	Steven	3/4/1955	Steven Buchanan graduated from St. Andrews University, Scotland, with a BSC degree. Upon joining the company as a sales representative, he spent 6 months in an orientation

				program at the Seattle office and then returned to his permanent post in London, where he was promoted to sales manager. Mr. Buchanan has completed the courses 'Successful Telemarketing' and 'International Sales Management'. He is fluent in French.
6	Suyama	Michael	7/2/1963	Michael is a graduate of Sussex University (MA, economics) and the University of California at Los Angeles (MBA, marketing). He has also taken the courses 'Multi-Cultural Selling' and 'Time Management for the Sales Professional'. He is fluent in Japanese and can read and write French, Portuguese, and Spanish.
7	King	Robert	5/29/1960	Robert King served in the Peace Corps and traveled extensively before completing his degree in English at the University of Michigan and then joining the company. After completing a course entitled 'Selling in Europe', he was transferred to the London office.
8	Callahan	Laura	1/9/1958	Laura received a BA in psychology from the University of Washington. She has also completed a course in business French. She reads and writes French.
9	Dodsworth	Anne	7/2/1969	Anne has a BA degree in English from St. Lawrence College. She is fluent in French and German.
10	West	Adam	9/19/1928	An old chum.

4. Tabel Shippers

ShipperID	ShipperName	Phone
1	Speedy Express	(503) 555-9831
2	United Package	(503) 555-3199
3	Federal Shipping	(503) 555-9931

5. Tabel Suppliers

SupplierID	SupplierName	ContactName	Address	City	PostalCode	Country	Phone
1	Exotic Liquid	Charlotte Cooper	49 Gilbert St.	Londona	EC1 4SD	UK	(171) 555-2222
2	New Orleans Cajun Delights	Shelley Burke	P.O. Box 78934	New Orleans	70117	USA	(100) 555-4822
3	Grandma Kelly's Homestead	Regina Murphy	707 Oxford Rd.	Ann Arbor	48104	USA	(313) 555-5735
4	Tokyo Traders	Yoshi Nagase	9-8 Sekimai Musashino-shi	Tokyo	100	Japan	(03) 3555-5011
5	Cooperativa de Quesos 'Las Cabras'	Antonio del Valle Saavedra	Calle del Rosal 4	Oviedo	33007	Spain	(98) 598 76 54
6	Mayumi's	Mayumi Ohno	92 Setsuko Chuo-ku	Osaka	545	Japan	(06) 431-7877
7	Pavlova, Ltd.	Ian Devling	74 Rose St. Moonie Ponds	Melbourne	3058	Australia	(03) 444-2343
8	Specialty Biscuits, Ltd.	Peter Wilson	29 King's Way	Manchester	M14 GSD	UK	(161) 555-4448
9	PB Knäckebröd AB	Lars Peterson	Kaloadagatan 13	Göteborg	S-345 67	Sweden	031-987 65 43
10	Refrescos Americanas LTDA	Carlos Diaz	Av. das Americanas 12.890	São Paulo	5442	Brazil	(11) 555 4640
11	Heli Süßwaren GmbH & Co. KG	Petra Winkler	Tiergartenstraße 5	Berlin	10785	Germany	(010) 9984510

12	Plutzer Lebensmittelgroßmärkte AG	Martin Bein	Bogenallee 51	Frankfurt	60439	Germany	(069) 992755
13	Nord-Ost-Fisch Handelsgesellschaft mbH	Sven Petersen	Frahmredder 112a	Cuxhaven	27478	Germany	(04721) 8713
14	Formaggi Fortini s.r.l.	Elio Rossi	Viale Dante, 75	Ravenna	48100	Italy	(0544) 60323
15	Norske Meierier	Beate Vileid	Hatlevegen 5	Sandvika	1320	Norway	(0)2-953010
16	Bigfoot Breweries	Cheryl Saylor	3400 - 8th Avenue Suite 210	Bend	97101	USA	(503) 555-9931
17	Svensk Sjöföda AB	Michael Björn	Brovallavägen 231	Stockholm	S-123 45	Sweden	08-123 45 67
18	Aux joyeux ecclésiastiques	Guylène Nodier	203, Rue des Francs- Bourgeois	Paris	75004	France	(1) 03.83.00.68
19	New England Seafood Cannery	Robb Merchant	Order Processing Dept. 2100 Paul Revere Blvd.	Boston	2134	USA	(617) 555-3267
20	Leka Trading	Chandra Leka	471 Serangoon Loop, Suite #402	Singapore	512	Singapore	555-8787

21	Lyngbysild	Niels Petersen	Lyngbysild Fiskebakken 10	Lyngby	2800	Denmark	43844108
22	Zaanse Snoepfabriek	Dirk Luchte	Verkoop Rijnweg 22	Zaandam	9999 ZZ	Netherlands	(12345) 1212
23	Karkki Oy	Anne Heikkonen	Valtakatu 12	Lappeenranta	53120	Finland	(953) 10956
24	G'day, Mate	Wendy Mackenzie	170 Prince Edward Parade Hunter's Hill	Sydney	2042	Australia	(02) 555-5914
25	Ma Maison	Jean-Guy Lauzon	2960 Rue St. Laurent	Montréal	H1J 1C3	Canada	(514) 555-9022
26	Pasta Buttini s.r.l.	Giovanni Giudici	Via dei Gelsomini, 153	Salerno	84100	Italy	(089) 6547665
27	Escargots Nouveaux	Marie Delamare	22, rue H. Voiron	Montceau	71300	France	85.57.00.07
28	Gai pâturage	Eliane Noz	Bat. B 3, rue des Alpes	Annecy	74000	France	38.76.98.06
29	Forêts d'éables	Chantal Goulet	148 rue Chasseur	Ste- Hyacinthe	J2S 7S8	Canada	(514) 555-2955

f. Hasil dan Pembahasan

Record baru pada masing – masing tabel telah berhasil ditambah

Tabel Shippers

ShipperID	ShipperName	Phone
1	Speedy Express	(503) 555-9831
2	United Package	(503) 555-3199
3	Federal Shipping	(503) 555-9931

Tabel Customers

CustomerID	CustomerName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Avda. de la Constitución 2222	México D.F.	5021	Mexico
3	Antonio Moreno Taquería	Mataderos 2312	México D.F.	5023	Mexico
4	Around the Horn	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Berguvsvägen 8	Luleå	S-958 22	Sweden
6	Blauer See Delikatessen	Forsterstr. 57	Mannheim	68306	Germany
7	Blondel père et fils	24, place Kléber	Strasbourg	67000	France
8	Bólido Comidas preparadas	C/ Araquil, 67	Madrid	28023	Spain
9	Bon app	12, rue des Bouchers	Marseille	13008	France
10	Bottom-Dollar Marketse	23 Tsawassen Blvd.	Tsawassen	T2F8M4	Canada
11	B s Beverages	Fauntleroy Circus	London	EC2 5NT	UK
12	Cactus Comidas para llevar	Cerrito 333	Buenos Aires	1010	Argentina
13	Centro comercial Moctezuma	Sierras de Granada 9993	México D.F.	5022	Mexico
14	Chop-suey Chinese	Hauptstr. 29	Bern	3012	Switzerland
15	Comércio Mineiro	Av. dos Lusíadas, 23	São Paulo	05432-043	Brazil
16	Consolidated Holdings	Berkeley Gardens 12 Brewery	London	WX1 6LT	UK
17	Drachenblut Delikatessend	Walserweg 21	Aachen	52066	Germany
18	Du monde entier	67 rue des Cinquante Otages	Nantes	44000	France
19	Eastern Connection	35 King George	London	WX3 6FW	UK
20	Ernst Handel	Kirchgasse 6	Graz	8010	Austria

Tabel Suppliers

SupplierID	SupplierName	ContactName	Address	City	PostalCode	Country	Phone
1	Exotic Liquid	Charlotte Cooper	49 Gilbert St.	Londona	EC1 4SD	UK	(171) 555-2222
2	New Orleans Cajun Delights	Shelley Burke	P.O. Box 78934	New Orleans	70117	USA	(100) 555-4822
3	Grandma Kelly's Homestead	Regina Murphy	707 Oxford Rd.	Ann Arbor	48104	USA	(313) 555-5735
4	Tokyo Traders	Yoshi Nagase	9-8 Sekimai Musashino-shi	Tokyo	100	Japan	(03) 3555-5011
5	Cooperativa de Quesos 'Las Cabras'	Antonio del Valle Saavedra	Calle del Rosal 4	Oviedo	33007	Spain	(98) 598 76 54
6	Mayumi's	Mayumi Ohno	92 Setsuko Chuo-ku	Osaka	545	Japan	(06) 431-7877
7	Pavlova, Ltd.	Ian Devling	74 Rose St. Moonie Ponds	Melbourne	3058	Australia	(03) 444-2343
8	Specialty Biscuits, Ltd.	Peter Wilson	29 King's Way	Manchester	M14 GSD	UK	(161) 555-4448
9	PB Knäckebröd AB	Lars Peterson	Kaloadagatan 13	Göteborg	S-345 67	Sweden	031-987 65 43
10	Refrescos Americanas LTDA	Carlos Diaz	Av. das Americanas 12.890	São Paulo	5442	Brazil	(11) 555 4640

Tabel Employees

EmployeeID	LastName	FirstName	BirthDate	Photo	Notes
1	Davolio	Nancy	1968-08-12	NULL	Education includes a BA in psychology from Colorad...
2	Fuller	Andrew	1952-02-19	NULL	Andrew received his BTS commercial and a Ph.D. in ...
3	Leverling	Janet	1963-08-30	NULL	Janet has a BS degree in chemistry from Boston Col...
4	Peacock	Margaret	1958-09-19	NULL	Margaret holds a BA in English literature from Con...
5	Buchanan	Steven	1955-03-04	NULL	Steven Buchanan graduated from St. Andrews Univers...
6	Suyama	Michael	1963-02-07	NULL	Michael is a graduate of Sussex University (MA, ec...
7	King	Robert	1960-05-29	NULL	Robert King served in the Peace Corps and traveled...
8	Callahan	Laura	1958-01-09	NULL	Laura received a BA in psychology from the Univers...
9	Dodsworth	Anne	1969-07-02	NULL	Anne has a BA degree in English from St. Lawrence ...
10	West	Adam	1928-09-19	NULL	An old chum.

1. Dokumentasi tugas berupa laporan (Tugas)

2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulandan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E123456_BI.pdfKumpulkan pada

<http://jti.polije.ac.id/elearning/>

g. Kesimpulan

Mahasiswa berhasil menambah record baru di masing – masing tabel

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4

1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan			V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan		V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat	V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V		
	Total Skor*	Skor x 25		

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 10

Pokok Bahasan	: DML (Select Statement)
Acara Praktikum/Pertemuan	: Minggu 4/2
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa memahami konsep select pada tabel
2. Mahasiswa mampu mengimplementasikan penggunaan select tabel

b. Indikator

Kemampuan mahasiswa dalam memahami dan mengimplementasikan penggunaan select antar tabel menggunakan tools phpMyAdmin

c. Dasar Teori

SELECT merupakan statement yang digunakan untuk mengambil data dari database.

Bentuk umum untuk pemanggilan SELECT adalah menggunakan syntax :

```
SELECT column1, column2, ...
FROM tableName;
```

Contoh implementasi SELECT pada tabel Customers yang menampilkan atribut CustomerName dan City adalah sebagai berikut :

```
SELECT CustomerName, City FROM Customers;
```

Di sini, kolom1, kolom2, ... adalah nama bidang tabel yang ingin Anda pilih datanya.

Jika Anda ingin memilih semua bidang yang tersedia dalam tabel, gunakan Syntax berikut:

```
SELECT * FROM tableName;
```

Contoh implementasi SELECT untuk menampilkan semua atribut pada tabel Customers adalah sebagai berikut :

```
SELECT * FROM Customers
```

Selain itu ada tatement SELECT DISTINCT

Pernyataan SELECT DISTINCT digunakan untuk menampilkan nilai yang berbeda dengan

```
SELECT DISTINCT column1, column2, ...
FROM table_name;
```

syntax sebagai berikut :

Contoh implementasi SELECT DISTINCT pada tabel customers di atribut Country dimana atribut tersebut akan ditampilkan dengan nilai yang berbeda adalah sebagai berikut :

```
SELECT DISTINCT Country FROM Customers;
```

Pada statemen SELECT juga terdapat clausa WHERE yang digunakan jika ingin membatasi data yang ditampilkan. Clusa Ini digunakan untuk mengekstrak hanya catatan yang memenuhi kondisi tertentu. Berikut aturan syntax dengan penggunaan clausa WHERE :

```
SELECT column1, column2, ...
FROM table_name
WHERE condition;
```

Pada tools PhpMyAdmin ada beberapa cara agar record tabel dapat ditampilkan diantaranya melalui query SQL seperti yang sudah dijelaskan sebelumnya atau dapat menggunakan tools PhpMyAdmin. Untuk menampilkan seluruh record tabel menggunakan tools PhpMyAdmin adalah sebagai berikut :

1. Buka Xampp-PhpMyAdmin lalu klik database yang diinginkan.
2. Klik tabel yang ingin ditampilkan record datanya. Salah satu contoh hasil record seluruh tabel Customers adalah sebagai berikut :

CustomerID	CustomerName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Avda. de la Constitución 2222	México D.F.	5021	Mexico
3	Antonio Moreno Taquería	Mataderos 2312	México D.F.	5023	Mexico
4	Around the Horn	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Berguvsvägen 8	Luleå	S-958 22	Sweden
6	Blauer See Delikatessen	Forsterstr. 57	Mannheim	68306	Germany
7	Blondel père et fils	24, place Kléber	Strasbourg	67000	France
8	Bólido Comidas preparadas	C/ Araquil, 67	Madrid	28023	Spain
9	Bon app	12, rue des Bouchers	Marseille	13008	France
10	Bottom-Dollar Marketse	23 Tsawassen Blvd.	Tsawassen	T2F8M4	Canada
11	B's Beverages	Fauntleroy Circus	London	EC2 5NT	UK
12	Cactus Comidas para llevar	Cerito 333	Buenos Aires	1010	Argentina
13	Centro comercial Moctezuma	Sierras de Granada 9993	México D.F.	5022	Mexico
14	Chop-suey Chinese	Hauptstr. 29	Bern	3012	Switzerland
15	Comercio Mineiro	Av. dos Lusíadas, 23	São Paulo	05432-043	Brazil

d. Alat dan Bahan

1. Client Server : Xampp
2. Text Editor : PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

1. Tampilkan seluruh record dari semua tabel, dengan menggunakan syntax dan tools dengan hasil dan tampilan pada PhpMyAdmin
2. Tampilkan Customer Name dan City pada tabel Customers

CustomerName	City
Alfreds Futterkiste	Berlin
Ana Trujillo Emparedados y helados	México D.F.
Antonio Moreno Taquería	México D.F.
Around the Horn	London
Berglunds snabbköp	Luleå
Blauer See Delikatessen	Mannheim
Blondel père et fils	Strasbourg
Bólido Comidas preparadas	Madrid
Bon app	Marseille
Bottom-Dollar Marketse	Tsawassen
B s Beverages	London
Cactus Comidas para llevar	Buenos Aires
Centro comercial Moctezuma	México D.F.
Chop-suey Chinese	Bern
Comércio Mineiro	São Paulo
Consolidated Holdings	London
Drachenblut Delikatessend	Aachen
Du monde entier 67	Nantes
Eastern Connection	London
Ernst Handel	Graz

3. Tampilkan kolom CustomerID, CustomerName, City, PostalCode, dan County yang berasal dari Germany

CustomerID	CustomerName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Obere Str. 57	Berlin	12209	Germany
6	Blauer See Delikatessen	Forsterstr. 57	Mannheim	68306	Germany
17	Drachenblut Delikatessend	Walserweg 21	Aachen	52066	Germany

4. Tampilkan semua atribut Customer ID, Customer Name, dan City berdasarkan Customer ID = 10

CustomerID	CustomerName	City
10	Bottom-Dollar Marketse	Tsawassen

f. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)
2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf Kumpulkan pada

<http://jti.polije.ac.id/elearning/>

g. Kesimpulan

Mahasiswa berhasil menampilkan select tabel sesuai ketentuan.

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak seusai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 11

Pokok Bahasan	: DML (Update Statement)
Acara Praktikum/Pertemuan	: Minggu 4/3
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa memahami konsep Update Statement pada tabel
2. Mahasiswa mampu mengimplementasikan Update Statement pada tabel

b. Indikator

Kemampuan mahasiswa dalam memahami dan mengimplementasikan Update Statement pada tabel menggunakan tools phpMyAdmin

c. Dasar Teori

Update Statement merupakan pernyataan yang digunakan untuk mengubah data/catatan yang ada dalam tabel.

Berikut adalah syntax untuk Update Statement :

```
UPDATE table_name  
SET column1 = value1, column2 = value2, ...  
WHERE condition;
```

Catatan : Berhati-hatilah saat memperbarui catatan dalam tabel! Perhatikan klausa WHERE dalam pernyataan UPDATE. Klausa WHERE menentukan record mana yang harus diperbarui. Jika Anda menghilangkan klausa WHERE, semua catatan dalam tabel akan diperbarui. Update Statement juga dapat di implementasikan menggunakan tools PhpMyAdmin dengan

cara :

1. klik tabel - pilih record data yang ingin di update lalu klik edit/update. Contoh disini adalah mengubah “Hansen” dengan “Frans” di attribute LastName pada tabel “persons”.

+ Options					
	PersonID	LastName	FirstName	Address	City
<input type="checkbox"/>	1	Hansen	Ola	Timoteivn 10	Sandnes
<input type="checkbox"/>	2	Svendson	Tove	Borgvn 23	Sandnes
<input type="checkbox"/>	3	Pettersen	Kari	Storgt 20	Stavanger

 With selected:

2. Ganti Hanson dengan Frans. Dengan menggunakan tools kita dapat dengan mudah dan leluasa untuk Update data yang ada pada tabel.

Column	Type	Function	Null	Value
PersonID	int(11)		1	
LastName	varchar(255)			Frans
FirstName	varchar(255)			Ola
Address	varchar(255)			Timoteivn 10
City	varchar(255)			Sandnes

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

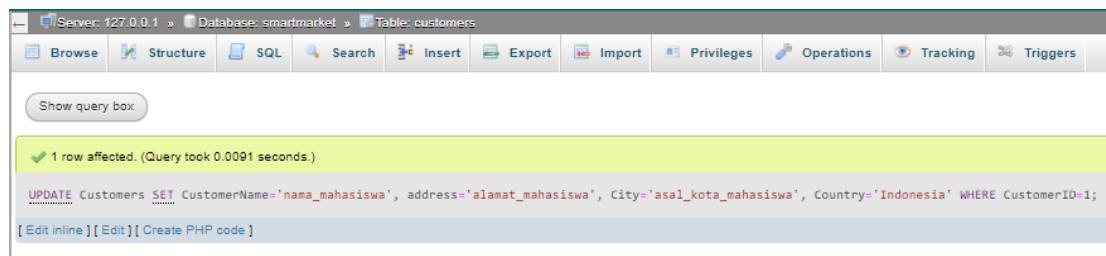
e. Prosedur Kerja

1. Pada workshop kali ini kita akan merubah (update) data dari tabel yang sudah kita buat pada pertemuan sebelumnya di database smartmarket.
2. Rubah isi dari CustomerName, address, city, dan country pada tabel Customers menjadi nama dan asal kalian masing-masing dengan menggunakan syntax dibawah :

```
UPDATE Customers
SET CustomerName='nama_mahasiswa', address='alamat_mahasiswa', City='asal_kota_mahasiswa', Country='Indonesia'
WHERE CustomerID=1;
```

3. Screenshot syntax dan hasil perubahan data.

Contoh :



The screenshot shows the PhpMyAdmin interface for a database named 'smartmarket' with a table named 'customers'. The query executed is:

```
UPDATE Customers SET CustomerName='nama_mahasiswa', address='alamat_mahasiswa', City='asal_kota_mahasiswa', Country='Indonesia' WHERE CustomerID=1;
```

The result message indicates "1 row affected. (Query took 0.0091 seconds.)". Below the query, there are links for "Edit inline", "Edit", and "Create PHP code".

CustomerID	CustomerName	Address	City	PostalCode	Country
1	Deependra Tsaqif Alfarizky	Jl. Letjen Sutoyo	Jember	12209	Germany
2	Ana Trujillo Emparedados y helados	Avda. de la Constitución 2222	México D.F.	5021	Mexico
3	Antonio Moreno Taquería	Mataderos 2312	México D.F.	5023	Mexico
4	Around the Horn	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Berguvsvägen 8	Luleå	S-958 22	Sweden

4. Rubah CustomerName, Address, City, dan Country minimal 5 data dengan menggunakan tools PhpMyAdmin.
 5. Screenshot setiap proses perbaikan datanya dan hasil dari perubahan datanya.

f. Hasil dan Pembahasan

- Dokumentasi tugas berupa laporan (Tugas)
- Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf

Kumpulkan pada <http://jti.polje.ac.id/elearning/>

g. Kesimpulan

Mahasiswa berhasil menampilkan select tabel sesuai ketentuan.

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 12

Pokok Bahasan	: DML (Delete Statement)
Acara Praktikum/Pertemuan	: Minggu 4/4
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa memahami konsep relasi antar tabel
2. Mahasiswa mampu membuat relasi antar tabel

b. Indikator

Kemampuan mahasiswa dalam memahami dan mengimplementasikan relasi antar tabel menggunakan tools phpMyAdmin

c. Dasar Teori

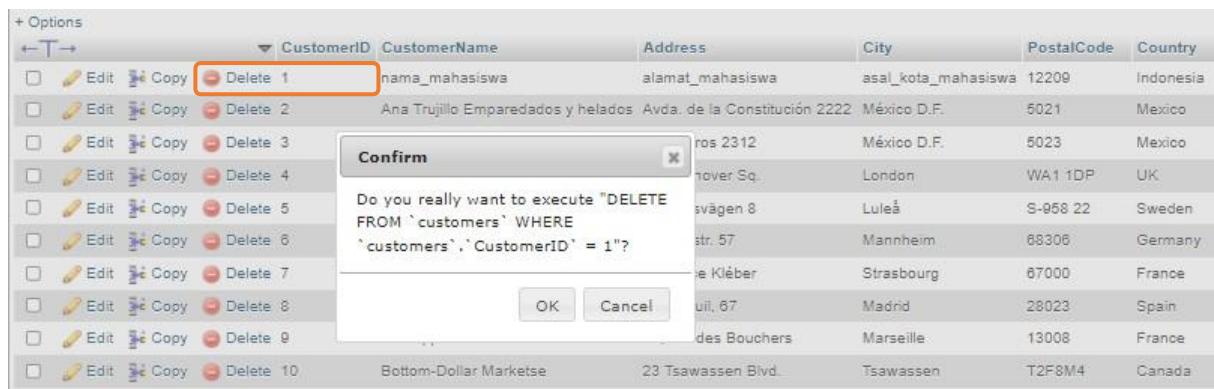
The Delete Statement merupakan pernyataan yang digunakan untuk menghapus record yang ada dalam tabel.

Berikut adalah syntax untuk menghapus record tabel :

```
DELETE FROM table_name WHERE condition;
```

Catatan: Hati-hati saat menghapus record dalam tabel! Perhatikan klausa WHERE dalam pernyataan DELETE. Klausa WHERE menentukan record mana yang harus dihapus. Jika Anda menghilangkan klausa WHERE, semua record dalam tabel akan dihapus!

Selain menggunakan syntax, menghapus record tabel dapat menggunakan tools PhpMyAdmin dengan pilih record mana yang akan dihapus lalu klik delete hingga muncul notifikasi konfirmasi apakah record tersebut benar – benar akan dihapus lalu klik OK.



A screenshot of the PhpMyAdmin interface showing a table named 'customers'. The table has columns: CustomerID, CustomerName, Address, City, PostalCode, and Country. A row for 'CustomerID 1' is selected, and a delete button is highlighted with an orange border. A confirmation dialog box is overlaid on the page, asking 'Do you really want to execute "DELETE FROM `customers` WHERE `customers`.`CustomerID` = 1"?' with 'OK' and 'Cancel' buttons.

	CustomerID	CustomerName	Address	City	PostalCode	Country
<input type="checkbox"/>	Edit	Copy	Delete 1	nama_mahasiswa	alamat_mahasiswa	asal_kota_mahasiswa
<input type="checkbox"/>	Edit	Copy	Delete 2	Ana Trujillo Emparedados y helados	Avda. de la Constitución 2222	México D.F.
<input type="checkbox"/>	Edit	Copy	Delete 3	Antonio Moreno Pizzería	Calle Madero 80	México D.F.
<input type="checkbox"/>	Edit	Copy	Delete 4	Dr. Harris	4321 St	London
<input type="checkbox"/>	Edit	Copy	Delete 5	Eduardo Holtzapfel	Obrajes 57	Luleå
<input type="checkbox"/>	Edit	Copy	Delete 6	Felicity Kész	Reuterstraße 57	Mannheim
<input type="checkbox"/>	Edit	Copy	Delete 7	Frédéric Cestac	54, rue Kléber	Strasbourg
<input type="checkbox"/>	Edit	Copy	Delete 8	Gloria M. Watanabe	Champs Elysees 67	Madrid
<input type="checkbox"/>	Edit	Copy	Delete 9	Hilary Foster	23 Tswassen Blvd.	Marseille
<input type="checkbox"/>	Edit	Copy	Delete 10	Horst D. Müller	Des Bouchers	Tsawassen
<input type="checkbox"/>	Edit	Copy	Delete 11	Julieta C. Serrano	Bottom-Dollar Marketse	Tsawassen
<input type="checkbox"/>	Edit	Copy	Delete 12	Klaus O. Greger	23 Tswassen Blvd.	Tsawassen
<input type="checkbox"/>	Edit	Copy	Delete 13	Marie-France Dubois	23 Tswassen Blvd.	Tsawassen
<input type="checkbox"/>	Edit	Copy	Delete 14	Robert Schmitz	23 Tswassen Blvd.	Tsawassen
<input type="checkbox"/>	Edit	Copy	Delete 15	Thomas Baumann	23 Tswassen Blvd.	Tsawassen
<input type="checkbox"/>	Edit	Copy	Delete 16	Vito Cervantes	23 Tswassen Blvd.	Tsawassen
<input type="checkbox"/>	Edit	Copy	Delete 17	Winston Z. Chang	23 Tswassen Blvd.	Tsawassen
<input type="checkbox"/>	Edit	Copy	Delete 18	Yousaf Ali	23 Tswassen Blvd.	Tsawassen
<input type="checkbox"/>	Edit	Copy	Delete 19	Zoë Wierwille	23 Tswassen Blvd.	Tsawassen
<input type="checkbox"/>	Edit	Copy	Delete 20	Yves Bozonnet	23 Tswassen Blvd.	Tsawassen

Notifikasi dibawah menandakan bahwa record data pada tabel telah berhasil dihapus.

1 row affected. (Query took 0.0085 seconds.)

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

1. Tampilkan tabel Customers sebelum dilakukan penghapusan data.
2. Hapus record data Country yang berasal dari “UK” pada tabel Customers menggunakan syntax dibawah :


```
DELETE FROM Customers WHERE Country='UK';
```
3. Screenshot bukti bahwa data berhasil terhapus dan screenshoot hasil data terakhir. Jika record data dengan country tidak ada yang UK maka data dipastikan berhasil terhapus.



A screenshot of the PhpMyAdmin interface showing the results of a DELETE query. The query was `DELETE FROM Customers WHERE Country='UK';`. The result shows 4 rows affected. The SQL history also contains the same query.

4. Tampilkan kembali tabel Customers setelah dilakukan penghapusan data.

g. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)
2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulandan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf

Kumpulkan pada <http://iti.polije.ac.id/elearning/>

g. Kesimpulan

Mahasiswa berhasil menampilkan select tabel sesuai ketentuan.

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak seusai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 13

Pokok Bahasan	: Tabel Joins
Acara Praktikum/Pertemuan	: Minggu 5/1
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa memahami konsep Joins dalam tabel
2. Mahasiswa mampu mengetahui jenis – jenis Join dalam tabel

b. Indikator

Kemampuan mahasiswa dalam memahami konsep joins dan mengetahui jenis – jenis join dalam tabel.

c. Dasar Teori

Penjelasan Join

Join merupakan cara yang digunakan untuk menggabungkan data yang diambil dari tabel – tabel melalui sebuah kolom yang menghubungkan tabel – tabel tersebut. Sehingga untuk menggabungkan baris dari dua atau lebih tabel, berdasarkan kolom terkait di antara tabel – tabel tersebut dapat menggunakan klausula JOIN.

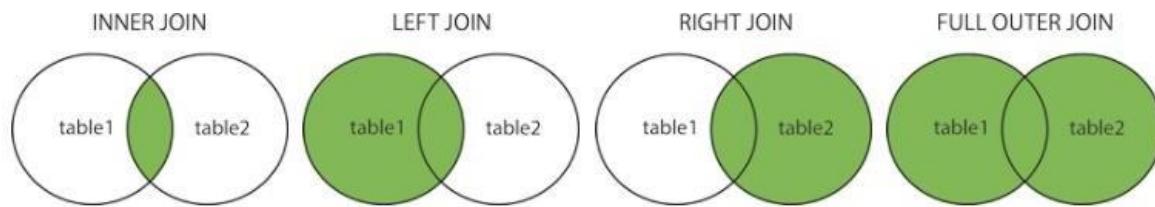
Peran Join

SQL tidak hanya menyediakan mekanisme query dan operasi modifikasi database saja, tetapi SQL juga menyediakan mekanisme untuk menggabungkan (join) relasi-relasi. Saat data yang dibutuhkan berasal lebih dari satu table, maka kondisi join dibutuhkan. Join memperbolehkan kita untuk mengambil data dari beberapa tabel melalui satu query. Data yang terbatas atau bahkan terlalu banyak tersimpan/didapatkan sehingga tabelnya menjadi kurang baik dapat hanya menggunakan sebuah tabel. JOIN merupakan bagian dari **relational database** karena perannya yang menghubungkan satu tabel dengan tabel yang lain.

Jenis – Jenis Join Secara Umum

JOIN memiliki beberapa tipe yang berbeda diantaranya adalah sebagai berikut :

1. (INNER) JOIN : Mengembalikan records yang memiliki nilai yang cocok di kedua tabel
2. LEFT (OUTER) JOIN : Mengembalikan semua record dari tabel kiri, dan record yang cocok dari tabel kanan
3. RIGHT (OUTER) JOIN : Mengembalikan semua record dari tabel kanan, dan record yang cocok dari tabel kiri
4. FULL(OUTER) JOIN : Mengembalikan semua records ketika ada kecocokan di tabel kiri atau kanan



Selain jenis – jenis join diatas, terdapat beberapa implementasi Join lainnya diantaranya adalah:

1. Left [Outer] Join without Intersection
2. Right [Outer] Join without Intersection
3. Full [Outer] Join without Intersection
4. Join Menggunakan Operator “,” (koma)
5. SELF JOIN
6. CROSS JOIN
7. NATURAL JOIN

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyadmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

Resume Jenis – Jenis join yang dapat diterapkan untuk menghubungkan dua tabel atau lebih selain (INNER) JOIN, LEFT (OUTER) JOIN, RIGHT (OUTER) JOIN, dan FULL(OUTER) JOIN

f. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)
2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf

Kumpulkan pada <http://jti.polije.ac.id/elearning/>

g. Kesimpulan

Mahasiswa berhasil menampilkan select tabel sesuai ketentuan.

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 14

Pokok Bahasan	: Join Tabel (“,”, Cross Join & Inner Join)
Acara Praktikum/Pertemuan	: Minggu 5/2
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa memahami konsep Cross Join dan Inner Join tanpa kondisi
2. Mahasiswa mampu mengimplementasikan konsep Cross Join dan Inner Join tanpa kondisi

b. Indikator

Kemampuan mahasiswa dalam memahami dan mengimplementasikan konsep Cross Join dan Inner Join.

c. Dasar Teori

Pada perintah “,”, CROSS JOIN dan [INNER] JOIN kita dapat mengimplementasikan dua jenis perintah yaitu tanpa menggunakan kondisi dan dengan menggunakan kondisi. Perintah “,”, CROSS JOIN dan [INNER] JOIN yang tanpa menggunakan kondisi akan menghasilkan suatu himpunan Cartesian product yang akan mengkombinasikan semua baris di tabel pertama dengan semua baris di tabel kedua. Jika anda menggunakan join menggunakan perintah tersebut, maka jika tabel pertama terdiri dari 5 baris dan tabel kedua terdiri dari 3 baris maka akan menghasilkan baris sebanyak 3. Sedangkan perintah dengan menggunakan kondisi akan menampilkan record sesuai kondisi yang diminta. Misalkan record data ID pada foreignkey ditabel A akan menampilkan record data ID pada primarykey ditabel B.

Berikut adalah contoh implementasi “,”, CROSS JOIN dan [INNER] JOIN tanpa kondisi:

```
SELECT ProductID, ProductName, CategoryID  
FROM Products
```

25 Rows

```
SELECT CategoryID, CategoryName  
FROM Categories
```

8 Rows

The screenshot shows two tables from a MySQL database:

- Products Table:** Contains 25 rows of product information. The columns are ProductID, ProductName, and CategoryID. The data includes items like Chais (CategoryID 1), Chang (CategoryID 1), Aniseed Syrup (CategoryID 2), etc.
- Categories Table:** Contains 8 rows of category information. The columns are CategoryID and CategoryName. The data includes Beverages, Condiments, Confections, Dairy Products, Grains/Cereals, Meat/Poultry, Produce, and Seafood.

Contoh 1 :

```
SELECT p.ProductID, p.ProductName, p.CategoryID,
       c.categoryName, c.Description
  FROM products p, categories c;
```

Perhatikan alias tabel :
Tabel products -> p; Tabel categoris -> c

Contoh 2 :

```
SELECT p.ProductID, p.ProductName, p.CategoryID,
       c.categoryName, c.Description
  FROM products p CROSS JOIN categories c;
```

Contoh 3 :

```
SELECT p.ProductID, p.ProductName, p.CategoryID,
       c.categoryName, c.Description
  FROM products p INNER JOIN categories c;
```

Hasil eksekusi

Showing rows 0 - 24 (200 total, Query took 0.0038 seconds.)

```
SELECT p.ProductID, p.ProductName, p.CategoryID, c.categoryName, c.Description
FROM products p CROSS JOIN categories c;
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

ProductID	ProductName	CategoryID	categoryName	Description
1	Chais	1	Beverages	Soft drinks, coffees, teas, beers, and ales
1	Chais	1	Condiments	Sweet and savory sauces, relishes, spreads, and seasonings
1	Chais	1	Confections	Desserts, candies, and sweet breads
1	Chais	1	Dairy Products	Cheeses
1	Chais	1	Grains/Cereals	Breads, crackers, pasta, and cereal
1	Chais	1	Meat/Poultry	Prepared meats
1	Chais	1	Produce	Dried fruit and bean curd
1	Chais	1	Seafood	Seaweed and fish
2	Chang	1	Beverages	Soft drinks, coffees, teas, beers, and ales
2	Chang	1	Condiments	Sweet and savory sauces, relishes, spreads, and seasonings
2	Chang	1	Confections	Desserts, candies, and sweet breads
2	Chang	1	Dairy Products	Cheeses
2	Chang	1	Grains/Cereals	Breads, crackers, pasta, and cereal
2	Chang	1	Meat/Poultry	Prepared meats
2	Chang	1	Produce	Dried fruit and bean curd
2	Chang	1	Seafood	Seaweed and fish
3	Aniseed Syrup	2	Beverages	Soft drinks, coffees, teas, beers, and ales
3	Aniseed Syrup	2	Condiments	Sweet and savory sauces, relishes, spreads, and seasonings
3	Aniseed Syrup	2	Confections	Desserts, candies, and sweet breads
3	Aniseed Syrup	2	Dairy Products	Cheeses

Showing rows 0 - 24 (200 total, Query took 0.0038 seconds.)

```
SELECT p.ProductID, p.ProductName, p.CategoryID, c.categoryName, c.Description
FROM products p CROSS JOIN categories c;
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

ProductID	ProductName	CategoryID	categoryName	Description
1	Chais	1	Beverages	Soft drinks, coffees, teas, beers, and ales
1	Chais	1	Condiments	Sweet and savory sauces, relishes, spreads, and seasonings
1	Chais	1	Confections	Desserts, candies, and sweet breads
1	Chais	1	Dairy Products	Cheeses
1	Chais	1	Grains/Cereals	Breads, crackers, pasta, and cereal
1	Chais	1	Meat/Poultry	Prepared meats
1	Chais	1	Produce	Dried fruit and bean curd
1	Chais	1	Seafood	Seaweed and fish
2	Chang	1	Beverages	Soft drinks, coffees, teas, beers, and ales
2	Chang	1	Condiments	Sweet and savory sauces, relishes, spreads, and seasonings
2	Chang	1	Confections	Desserts, candies, and sweet breads
2	Chang	1	Dairy Products	Cheeses
2	Chang	1	Grains/Cereals	Breads, crackers, pasta, and cereal
2	Chang	1	Meat/Poultry	Prepared meats
2	Chang	1	Produce	Dried fruit and bean curd
2	Chang	1	Seafood	Seaweed and fish
3	Aniseed Syrup	2	Beverages	Soft drinks, coffees, teas, beers, and ales
3	Aniseed Syrup	2	Condiments	Sweet and savory sauces, relishes, spreads, and seasonings
3	Aniseed Syrup	2	Confections	Desserts, candies, and sweet breads
3	Aniseed Syrup	2	Dairy Products	Cheeses

2**1****3**

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

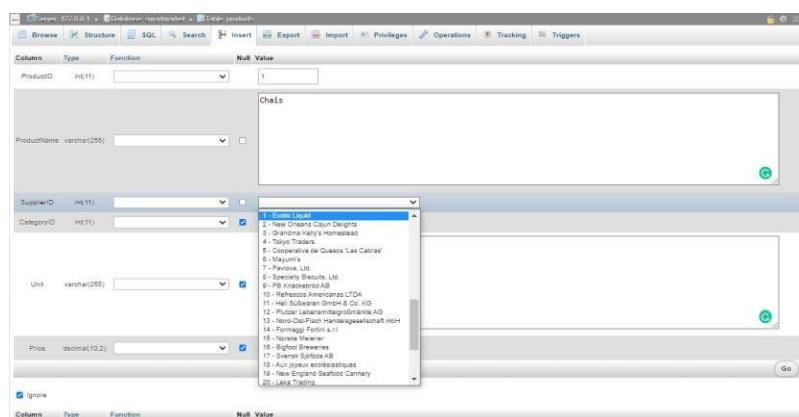
1. Tambahkan record data pada tabel “products” dengan data sebagai berikut :

ProductID	ProductName	SupplierID	CategoryID	Unit	Price
1	Chais	1	1	10 boxes x 20 bags	18
2	Chang	1	1	24 - 12 oz bottles	19
3	Aniseed Syrup	1	2	12 - 550 ml bottles	10
4	Chef Anton's Cajun Seasoning	2	2	48 - 6 oz jars	22
5	Chef Anton's Gumbo Mix	2	2	36 boxes	21.35
6	Grandma's Boysenberry Spread	3	2	12 - 8 oz jars	25
7	Uncle Bob's Organic Dried Pears	3	7	12 - 1 lb pkgs.	30
8	Northwoods Cranberry Sauce	3	2	12 - 12 oz jars	40
9	Mishi Kobe Niku	4	6	18 - 500 g pkgs.	97
10	Ikura	4	8	12 - 200 ml jars	31
11	Queso Cabrales	5	4	1 kg pkg.	21
12	Queso Manchego La Pastora	5	4	10 - 500 g pkgs.	38
13	Konbu	6	8	2 kg box	6
14	Tofu	6	7	40 - 100 g pkgs.	23.25
15	Genen Shouyu	6	2	24 - 250 ml bottles	15.5
16	Pavlova	7	3	32 - 500 g boxes	17.45
17	Alice Mutton	7	6	20 - 1 kg tins	39
18	Carnarvon Tigers	7	8	16 kg pkg.	62.5
19	Teatime Chocolate Biscuits	8	3	10 boxes x 12 pieces	9.2
20	Sir Rodney's Marmalade	8	3	30 gift boxes	81
21	Sir Rodney's Scones	8	3	24 pkgs. x 4 pieces	10
22	Gustaf's Knäckebröd	9	5	24 - 500 g pkgs.	21
23	Tunnbröd	9	5	12 - 250 g pkgs.	9
24	Guaraná Fantástica	10	1	12 - 355 ml cans	4.5
25	NuNuCa Nuß-Nougat-Creme	11	3	20 - 450 g glasses	14
26	Gumbär Gummibärchen	11	3	100 - 250 g bags	31.23
27	Schoggi Schokolade	11	3	100 - 100 g pieces	43.9
28	Rössle Sauerkraut	12	7	25 - 825 g cans	45.6
29	Thüringer Rostbratwurst	12	6	50 bags x 30 sausgs.	123.79
30	Nord-Ost Matjeshering	13	8	10 - 200 g glasses	25.89

31	Gorgonzola Telino	14	4	12 - 100 g pkgs	12.5
32	Mascarpone Fabioli	14	4	24 - 200 g pkgs.	32
33	Geitost	15	4	500 g	2.5
34	Sasquatch Ale	16	1	24 - 12 oz bottles	14
35	Steeleye Stout	16	1	24 - 12 oz bottles	18
36	Inlagd Sill	17	8	24 - 250 g jars	19
37	Gravad lax	17	8	12 - 500 g pkgs.	26
38	Côte de Blaye	18	1	12 - 75 cl bottles	263.5
39	Chartreuse verte	18	1	750 cc per bottle	18
40	Boston Crab Meat	19	8	24 - 4 oz tins	18.4
41	Jack's New England Clam Chowder	19	8	12 - 12 oz cans	9.65
42	Singaporean Hokkien Fried Mee	20	5	32 - 1 kg pkgs.	14
43	Ipoh Coffee	20	1	16 - 500 g tins	46
44	Gula Malacca	20	2	20 - 2 kg bags	19.45
45	Røgede sild	21	8	1k pkg.	9.5
46	Spegesild	21	8	4 - 450 g glasses	12
47	Zaanse koeken	22	3	10 - 4 oz boxes	9.5
48	Chocolade	22	3	10 pkgs.	12.75
49	Maxilaku	23	3	24 - 50 g pkgs.	20
50	Valkoinen suklaa	23	3	12 - 100 g bars	16.25
51	Manjimup Dried Apples	24	7	50 - 300 g pkgs.	53
52	Filo Mix	24	5	16 - 2 kg boxes	7
53	Perth Pasties	24	6	48 pieces	32.8
54	Tourtière	25	6	16 pies	7.45
55	Pâté chinois	25	6	24 boxes x 2 pies	24
56	Gnocchi di nonna Alice	26	5	24 - 250 g pkgs.	38
57	Ravioli Angelo	26	5	24 - 250 g pkgs.	19.5
58	Escargots de Bourgogne	27	8	24 pieces	13.25
59	Raclette Courdavault	28	4	5 kg pkg.	55
60	Camembert Pierrot	28	4	15 - 300 g rounds	34
61	Sirop d'éable	29	2	24 - 500 ml bottles	28.5
62	Tarte au sucre	29	3	48 pies	49.3
63	Vegie-spread	7	2	15 - 625 g jars	43.9
64	Wimmers gute Semmelknödel	12	5	20 bags x 4 pieces	33.25
65	Louisiana Fiery Hot Pepper Sauce	2	2	32 - 8 oz bottles	21.05
66	Louisiana Hot Spiced Okra	2	2	24 - 8 oz jars	17
67	Laughing Lumberjack Lager	16	1	24 - 12 oz bottles	14
68	Scottish Longbreads	8	3	10 boxes x 8 pieces	12.5
69	Gudbrandsdalsost	15	4	10 kg pkg.	36
70	Outback Lager	7	1	24 - 355 ml bottles	15
71	Fløtemysost	15	4	10 - 500 g pkgs.	21.5

72	Mozzarella di Giovanni	14	4	24 - 200 g pkgs.	34.8
73	Röd Kaviar	17	8	24 - 150 g jars	15
74	Longlife Tofu	4	7	5 kg pkg.	10
75	Rhönbräu Klosterbier	12	1	24 - 0.5 l bottles	7.75
76	Lakkalikööri	23	1	500 ml	18
77	Original Frankfurter grüne Soße	12	2	12 boxes	13

2. Tabel products berelasi dengan tabel suppliers dan tabel categories. Jika kita ingin menambahkan record data pada tabel products menggunakan tools PhpMyAdmin, maka Foreign Key pada tabel products akan terhubung dari primary key dari tabel suppliers dan tabel categories. Berikut adalah contoh tabel products dengan record categoryID didapatkan dari record pada tabel categories yang telah di inputkan sebelumnya.



3. Implementasikan perintah “”, CROSS JOIN dan [INNER] JOIN tanpa kondisi yang sudah dijelaskan dalam dasar teori.

ProductID	ProductName	CategoryID	CategoryID	categoryName	Description
1	Chais	1	1	Beverages	Soft drinks, coffees, teas, beers, and ales
2	Chang	1	1	Beverages	Soft drinks, coffees, teas, beers, and ales
24	Guaraná Fantástica	1	1	Beverages	Soft drinks, coffees, teas, beers, and ales
3	Aniseed Syrup	2	2	Condiments	Sweet and savory sauces, relishes, spreads, and se...
4	Chef Anton's Cajun Seasoning	2	2	Condiments	Sweet and savory sauces, relishes, spreads, and se...
5	Chef Anton's Gumbo Mix	2	2	Condiments	Sweet and savory sauces, relishes, spreads, and se...
6	Grandma's Boysenberry Spread	2	2	Condiments	Sweet and savory sauces, relishes, spreads, and se...
8	Northwoods Cranberry Sauce	2	2	Condiments	Sweet and savory sauces, relishes, spreads, and se...
15	Genen Shouyu	2	2	Condiments	Sweet and savory sauces, relishes, spreads, and se...
16	Pavlova	3	3	Confections	Desserts, candies, and sweet breads
19	Teatime Chocolate Biscuits	3	3	Confections	Desserts, candies, and sweet breads
20	Sir Rodneys Marmalade	3	3	Confections	Desserts, candies, and sweet breads
21	Sir Rodneys Scones	3	3	Confections	Desserts, candies, and sweet breads
25	NuNuCa Nuß-Nougat-Creme	3	3	Confections	Desserts, candies, and sweet breads
11	Queso Cabrales	4	4	Dairy Products	Cheeses
12	Queso Manchego La Pastora	4	4	Dairy Products	Cheeses
22	Gustaf's Knäckebrot	5	5	Grains/Cereals	Breads, crackers, pasta, and cereal
23	Tunnbröd	5	5	Grains/Cereals	Breads, crackers, pasta, and cereal
9	Mishi Kobe Niku	6	6	Meat/Poultry	Prepared meats
17	Alice Mutton	6	6	Meat/Poultry	Prepared meats
7	Uncle Bob's Organic Dried Pears	7	7	Produce	Dried fruit and bean curd
14	Tofu	7	7	Produce	Dried fruit and bean curd
10	Ikura	8	8	Seafood	Seaweed and fish
13	Konbu	8	8	Seafood	Seaweed and fish
18	Carnarvon Tigers	8	8	Seafood	Seaweed and fish

4. Implementasikan “,”, CROSS JOIN dan [INNER] JOIN dengan kondisi dengan hasil sebagai berikut :

ProductID	ProductName	CategoryID	CategoryID	categoryName	Description
1	Chais	1	1	Beverages	Soft drinks, coffees, teas, beers, and ales
2	Chang	1	1	Beverages	Soft drinks, coffees, teas, beers, and ales
24	Guaraná Fantástica	1	1	Beverages	Soft drinks, coffees, teas, beers, and ales
3	Aniseed Syrup	2	2	Condiments	Sweet and savory sauces, relishes, spreads, and se...
4	Chef Anton's Cajun Seasoning	2	2	Condiments	Sweet and savory sauces, relishes, spreads, and se...
5	Chef Anton's Gumbo Mix	2	2	Condiments	Sweet and savory sauces, relishes, spreads, and se...
6	Grandmas Boysenberry Spread	2	2	Condiments	Sweet and savory sauces, relishes, spreads, and se...
8	Northwoods Cranberry Sauce	2	2	Condiments	Sweet and savory sauces, relishes, spreads, and se...
15	Genen Shouyu	2	2	Condiments	Sweet and savory sauces, relishes, spreads, and se...
16	Pavlova	3	3	Confections	Desserts, candies, and sweet breads
19	Teatime Chocolate Biscuits	3	3	Confections	Desserts, candies, and sweet breads
20	Sir Rodney's Marmalade	3	3	Confections	Desserts, candies, and sweet breads
21	Sir Rodney's Scones	3	3	Confections	Desserts, candies, and sweet breads
25	NuNuCa Nuß-Nougat-Creme	3	3	Confections	Desserts, candies, and sweet breads
11	Queso Cabrales	4	4	Dairy Products	Cheeses
12	Queso Manchego La Pastora	4	4	Dairy Products	Cheeses
22	Gustafs Knäckebröd	5	5	Grains/Cereals	Breads, crackers, pasta, and cereal
23	Tunnbröd	5	5	Grains/Cereals	Breads, crackers, pasta, and cereal
9	Mishi Kobe Niku	6	6	Meat/Poultry	Prepared meats
17	Alice Mutton	6	6	Meat/Poultry	Prepared meats
7	Uncle Bob's Organic Dried Pears	7	7	Produce	Dried fruit and bean curd
14	Tofu	7	7	Produce	Dried fruit and bean curd
10	Ikura	8	8	Seafood	Seaweed and fish
13	Konbu	8	8	Seafood	Seaweed and fish
18	Camarvon Tigers	8	8	Seafood	Seaweed and fish

25 rows

Field yang sama

5. Berikan penjelasan secara rinci dari proses hingga hasil percobaan tersebut.

g. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)
2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf

Kumpulkan pada <http://jti.polje.ac.id/elearning/>

i. Kesimpulan

Mahasiswa berhasil menampilkan select tabel sesuai ketentuan.

j. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 15

Pokok Bahasan	: Join Tabel (Natural Join)
Acara Praktikum/Pertemuan	: Minggu 5/3
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa memahami konsep Natural Join
2. Mahasiswa mampu mengimplementasikan konsep Natural Join dalam tabel

b. Indikator

Kemampuan mahasiswa dalam memahami dan mengimplementasikan konsep Natural Join dalam tabel.

c. Dasar Teori

NATURAL JOIN digunakan untuk menjoinkan 2 tabel atau lebih yang mempunyai 1 atau banyak kolom dengan nama yang sama dalam semua tabelnya. Jika tabel 1 mempunyai 2 kolom yang mempunyai nama field yang sama dengan di tabel 2, maka NATURAL JOIN akan melakukan kondisi relasi berdasarkan kedua field tersebut. Dengan NATURAL JOIN, anda tidak perlu mendefinisikan kondisi relasi.

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

1. Implementasikan perintah NATURAL JOIN dengan syntax sebagai berikut :

```
SELECT p.ProductID, p.ProductName, p.CategoryID,
       c.CategoryID, c.categoryName, c.Description
FROM products p NATURAL JOIN categories c;
```

2. Tampilkan hasil implementasi dari NATURAL JOIN
3. Amati tampilan dari NATURAL JOIN
4. Buat laporan dengan memberikan penjelasan dari implementasi NATURAL JOIN

h. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)
2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf

Kumpulkan pada <http://jti.polije.ac.id/elearning/>

k. Kesimpulan

Mahasiswa berhasil menampilkan select tabel sesuai ketentuan.

l. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak seusai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 16

Pokok Bahasan	: Join Tabel (CLAUSA USING)
Acara Praktikum/Pertemuan	: Minggu 5/4
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa memahami konsep Join Tabel dengan Clause Using()
2. Mahasiswa mampu mengimplementasikan konsep Join Tabel dengan Clause Using() dalam tabel

b. Indikator

Kemampuan mahasiswa dalam memahami dan mengimplementasikan konsep Join Tabel dengan Clause Using() dalam tabel.

c. Dasar Teori

Clause USING() digunakan untuk menyingkat kondisi relasi yang nama kolomnya identik.

Dengan menggunakan USING() pada JOIN, maka NATURAL JOIN sebenarnya sama dengan JOIN yang menggunakan USING untuk semua kolom yang identik pada kedua tabelnya.

Berikut adalah contoh penggunaan perintah JOIN yang dirubah dengan menggunakan Clause USING() :

```
SELECT p.ProductID, p.ProductName, p.CategoryID,
       c.CategoryID, c.categoryName, c.Description
  FROM products p JOIN categories c
 WHERE p.CategoryID = c.CategoryID;
```

Menjadi :

```
SELECT p.ProductID, p.ProductName, p.CategoryID,
       c.CategoryID, c.categoryName, c.Description
  FROM products p JOIN categories c
 USING(CategoryID);
```

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

Implementasikan contoh penggunaan Clusa Using() berdasarkan dasar teori lalu amati serta berikan penjelasan.

f. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)
2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf Kumpulkan pada <http://iti.polije.ac.id/elearning/>

g. Kesimpulan

Mahasiswa berhasil menampilkan select tabel sesuai ketentuan.

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak seusai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 17

Pokok Bahasan	: Join Tabel (Self Join)
Acara Praktikum/Pertemuan	: Minggu 5/5
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa memahami konsep Self Join
2. Mahasiswa mampu mengimplementasikan konsep Self Join dalam tabel

b. Indikator

Kemampuan mahasiswa dalam memahami dan mengimplementasikan konsep Self Join dalam tabel.

c. Dasar Teori

Self Join adalah suatu kondisi dimana suatu tabel berelasi tabel itu sendiri.. Self Join disebut sebagai tipe join lebih dari join pada umumnya yang merupakan strategi tabel untuk mereferensikan dirinya sendiri dan ketika tabel mereferensikan dirinya, Self Join dapat menggunakan jenis join mana saja yang diperlukan berdasarkan studi kasus.

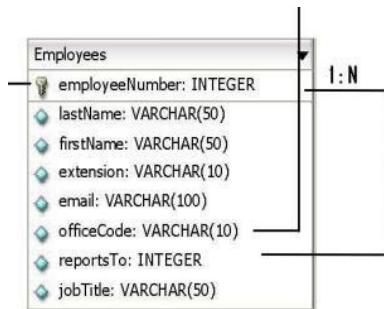
Berikut adalah syntax dari Self Join :

```
SELECT column_name(s)
FROM table1 T1, table1 T2
WHERE condition;
```

Dimana ti dan t2 adalah alias tabel yang berbeda untuk tabel yang sama

Berikut ini terdapat contoh kasus dimana dalam self join, penggunaan alias tabel pasti digunakan dengan kondisi sebagai berikut :

- Tampilan data pegawai lengkap dengan data atasannya. Atasannya berasal dari tabel yang sama.
- Perhatikan pada skema relasi pada gambar, pada table Employees, ada field reportsTo yang berelasi dengan field CustomerNumber pada tabel yang sama.



- Bisa kita anggap bahwa kolom reportsTo merupakan kolom yang menyatakan siapa yang menjadi atasan seorang pegawai
- Terdapat dua opsi penggunaan self join adalah dapat menggunakan koma dan where, atau dapat menggunakan self join dengan klausa Join On.
- Berikut adalah contoh syntaxnya :

SOLUSI 1 (MENGGUNAKAN KOMA DAN WHERE) :

```

SELECT peg.EmployeeNumber NoPegawai,
       concat_ws(' ',peg.firstName,peg.lastName) NamaPegawai,
       peg.reportsTo,
       ats.EmployeeNumber NoAtasan,
       concat_ws(' ',ats.firstName,ats.lastName) NamaAtasan
  FROM employees peg, employees ats
 WHERE peg.reportsTo = ats.employeeNumber
  
```

SOLUSI 2 (JOIN ON) :

```

SELECT peg.EmployeeNumber NoPegawai,
       concat_ws(' ',peg.firstName,peg.lastName) NamaPegawai,
       peg.reportsTo,
       ats.EmployeeNumber NoAtasan,
       concat_ws(' ',ats.firstName,ats.lastName) NamaAtasan
  FROM employees peg JOIN employees ats
 ON peg.reportsTo = ats.employeeNumber
  
```

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

- Berikut merupakan tabel “Customers”

CustomerID	CustomerName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Avda. de la Constitución 2222	México D.F.	5021	Mexico
3	Antonio Moreno Taquería	Mataderos 2312	México D.F.	5023	Mexico
5	Berglunds snabbköp	Berguvsvägen 8	Luleå	S-958 22	Sweden
6	Blauer See Delikatessen	Forsterstr. 57	Mannheim	68306	Germany
7	Blondel père et fils	24, place Kléber	Strasbourg	67000	France
8	Bólido Comidas preparadas	C/ Araquil, 67	Madrid	28023	Spain
9	Bon app	12, rue des Bouchers	Marseille	13008	France
10	Bottom-Dollar Marketse	23 Tsawassen Blvd.	Tsawassen	T2F8M4	Canada
12	Cactus Comidas para llevar	Cerrito 333	Buenos Aires	1010	Argentina
13	Centro comercial Moctezuma	Sierras de Granada 9993	México D.F.	5022	Mexico
14	Chop-suey Chinese	Hauptstr. 29	Bern	3012	Switzerland
15	Comércio Mineiro	Av. dos Lusíadas, 23	São Paulo	05432-043	Brazil
17	Drachenblut Delikatessend	Walserweg 21	Aachen	52066	Germany
18	Du monde entier	67 rue des Cinquante Otages	Nantes	44000	France
20	Ernst Handel	Kirchgasse 6	Graz	8010	Austria

- Kalian dapat mendambahkan record baru pada tabel Customers yang telah di upload d elearning
- Lakukan self join tabel dengan masukkan syntax berikut :

```
SELECT A.CustomerName AS CustomerName1, B.CustomerName AS CustomerName2, A.City
FROM Customers A, Customers B
WHERE A.CustomerID <> B.CustomerID
AND A.City = B.City
ORDER BY A.City;
```

- Tampilkan hasil dari self join tabel yang telah dilakukan, amati dan jelaskan maksud dari perintah tersebut!

f. Hasil dan Pembahasan

- Dokumentasi tugas berupa laporan (Tugas)
- Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf Kumpulkan pada

<http://iti.polije.ac.id/elearning/>

g. Kesimpulan

Mahasiswa berhasil menampilkan select tabel sesuai ketentuan.

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 18

Pokok Bahasan	: Fungsi Agregat (Fungsi Dasar di Agregat)
Acara Praktikum/Pertemuan	: Minggu 6/1
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami pengertian Fungsi Agregat
2. Mahasiswa mampu memahami penggunaan dasar pada jenis – jenis fungsi agregat.
3. Mahasiswa mampu mengimplementasi penggunaan query Min Max, Count, Average, dan Sum.

b. Indikator

Kemampuan mahasiswa dalam memahami dan mengimplementasikan penggunaan dasar pada jenis – jenis fungsi aregat yaitu penggunaan query Min Max, Count, Average, dan Sum.

c. Dasar Teori

Fungsi Agregat (*aggregate*) merupakan fungsi yang menerima koleksi nilai dan mengembalikan nilai tunggal sebagai hasilnya. Terdapat 5 (lima) jenis fungsi agregat yang terdefinisi sesuai dengan standar ISO, diantaranya adalah :

Fungsi	Deskripsi
COUNT	Mengembalikan jumlah (banyaknya atau kemunculannya) nilai di suatu kolom
SUM	Mengembalikan jumlah (total atau <i>sum</i>) nilai di suatu kolom
AVG	Mengembalikan rata-rata (<i>average</i>) nilai di suatu kolom
MIN	Mengembalikan nilai terkecil (<i>minimal</i>) di suatu kolom
MAX	Mengembalikan nilai terbesar (<i>maximal</i>) di suatu kolom

Fungsi agregat bekerja dengan cara menggabungkan beberapa elemen yang berbeda.

Berikut adalah contoh syntax dari lima jenis fungsi agregat :

- Fungsi COUNT

```
SELECT COUNT(column_name)
FROM table_name
WHERE condition;
```

Fungsi COUNT() mengembalikan jumlah baris yang cocok dengan kriteria yang ditentukan.

- Fungsi SUM

```
SELECT SUM(column_name)
FROM table_name
WHERE condition;
```

Fungsi SUM() mengembalikan jumlah total kolom numerik.

- Fungsi AVG

```
SELECT AVG(column_name)
FROM table_name
WHERE condition;
```

Fungsi AVG() mengembalikan nilai rata-rata kolom numerik.

- Fungsi MIN

```
SELECT MIN(column_name)
FROM table_name
WHERE condition;
```

Fungsi MIN() mengembalikan nilai terkecil dari kolom yang dipilih.

- Fungsi MAX

```
SELECT MAX(column_name)
FROM table_name
WHERE condition;
```

Fungsi MAX() mengembalikan nilai terbesar dari kolom yang dipilih.

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio

4. Pulpén

e. Prosedur Kerja

Untuk mengimplementasikan fungsi agregat (count, sum, avg, min, dan max) kita akan menggunakan tabel relasi yaitu tabel **Products**. Dimana tabel **Products** berelasi dengan tabel **Suppliers** dan **Categories**. Dua tabel tersebut berelasi dengan tabel **Products** karena memiliki field yang sama yaitu **SupplierID** dari tabel **Products** dan **Suppliers**, serta **CategoryID** dari tabel **Categories** dan **Products**. Terkadang tabel – tabel yang berelasi tersebut dibutuhkan klausa JOIN untuk mengimplementasikan fungsi agregat.

Lakukan implementasi dari fungsi agregat dengan ketentuan sebagai berikut :

1. Pastikan record tabel Products, Suppliers, dan Categories telah terisi.
2. Lakukan beberapa prosedur kerja untuk mengimplementasikan fungsi agregat yang terdiri dari :
 - MIN

Tampilkan nama supplier (SupplierName) dan nama panggilan dari supplier (ContactName) yang berasal dari tabel **Suppliers** dengan unit paling sedikit pada tabel **Products** dengan query sebagai berikut :

```
SELECT s.SupplierName as 'Nama Supplier', s.ContactName as 'Nama Panggilan', s.SupplierID as 'ID',
       min(p.unit) as 'Minimal Unit'
  FROM Suppliers s join Products p
 WHERE s.SupplierID = p.SupplierID
```

Tampilkan hasilnya dan buat kesimpulan!

- MAX

Tampilkan nama supplier (SupplierName) dan nama panggilan dari supplier (ContactName) yang berasal dari tabel **Suppliers** dengan unit terbesar pada tabel **Products**, dimana SupplierID tidak sama dengan 1 dengan query sebagai berikut :

```
SELECT s.SupplierName as 'Nama Supplier', s.ContactName as 'Nama Panggilan', s.SupplierID as 'ID',
       max(p.unit) as 'Unit Terbanyak'
  FROM Suppliers s join Products p
 WHERE s.SupplierID = p.SupplierID
       AND s.SupplierID <> 1;
```

Tampilkan hasilnya dan buat kesimpulan!

- COUNT

Hitung jumlah Customers berdasarkan CustomerID pada tabel Products

```
SELECT COUNT(ProductID)
  FROM Products;
```

Tampilkan hasilnya dan buat kesimpulan!

- **SUM**

Tampilkan total transaksi baik dari total unit (**Unit**) maupun total uang (**Price**) dari product keseluruhan pada tabel **Products**

```
SELECT SUM(Unit) as 'Total Unit', SUM(Price) as 'Total Uang dari Product Keseluruhan'
FROM Products;
```

Tampilkan hasilnya dan buat kesimpulan!

- **AVG**

Kali ini kita akan menggunakan tabel **OrderDetails** dimana tabel tersebut berelasi dengan tabel **Orders** dan **Products**. Yang akan kita gunakan disini hanyalah tabel **Orders** untuk menghitung rata – rata pesanan. Masukkan record data baru pada tabel **OrderDetails**.

OrderDetailID	OrderID	ProductID	Quantity
1	10248	11	12
2	10248	42	10
3	10248	72	5
4	10249	14	9
5	10249	51	40
6	10250	41	10
7	10250	51	35
8	10250	65	15
9	10251	22	6
10	10251	57	15
11	10251	65	20
12	10252	20	40
13	10252	33	25
14	10252	60	40
15	10253	31	20
16	10253	39	42
17	10253	49	40
18	10254	24	15
19	10254	55	21
20	10254	74	21
21	10255	2	20
22	10255	16	35
23	10255	36	25
24	10255	59	30
25	10256	53	15
26	10256	77	12

27	10257	27	25
28	10257	39	6
29	10257	77	15
30	10258	2	50
31	10258	5	65
32	10258	32	6
33	10259	21	10
34	10259	37	1
35	10260	41	16
36	10260	57	50
37	10260	62	15
38	10260	70	21
39	10261	21	20
40	10261	35	20
41	10262	5	12
42	10262	7	15
43	10262	56	2
44	10263	16	60
45	10263	24	28
46	10263	30	60
47	10263	74	36
48	10264	2	35
49	10264	41	25
50	10265	17	30
51	10265	70	20
52	10266	12	12
53	10267	40	50

54	10267	59	70
55	10267	76	15
56	10268	29	10
57	10268	72	4
58	10269	33	60
59	10269	72	20
60	10270	36	30
61	10270	43	25
62	10271	33	24
63	10272	20	6
64	10272	31	40
65	10272	72	24
66	10273	10	24
67	10273	31	15
68	10273	33	20
69	10273	40	60
70	10273	76	33
71	10274	71	20
72	10274	72	7

73	10275	24	12
74	10275	59	6
75	10276	10	15
76	10276	13	10
77	10277	28	20
78	10277	62	12
79	10278	44	16
80	10278	59	15
81	10278	63	8
82	10278	73	25
83	10279	17	15
84	10280	24	12
85	10280	55	20
86	10280	75	30
87	10281	19	1
88	10281	24	6
89	10281	35	4
90	10282	30	6

Pada implementasi AVG (Averege) ini kita akan menghitung nilai rata – rata dari hasil transaksi barang yang di order (**Quantity**) dengan menampilkan jumlah pesanan (OrderID) pada tabel **Orders** dengan syntax sebagai berikut :

```
SELECT count(o.OrderID) as 'Jumlah Pesanan', AVG(q.Quantity) as 'Rata - Rata Pesanan'
FROM Orders o join OrderDetails q
ON o.OrderID = q.OrderID
```

Tampilkan seluruh record pada tabel OrderDetails, lalu tampilkan hasil dari implementasi AVG dan buat kesimpulan!

f. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)
2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdfKumpulkan

pada <http://jti.polije.ac.id/elearning/>

g. Kesimpulan

Mahasiswa berhasil menampilkan select tabel sesuai ketentuan.

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 19

Pokok Bahasan	: Fungsi Agregat (Like dan Wildcards)
Acara Praktikum/Pertemuan	: Minggu 6/2
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

1. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami fungsi agregat
2. Mahasiswa mampu memahami Like dan Wildcard sebagai fungsi agregat
3. Mahasiswa mampu membedakan Like dan Wildcards sebagai fungsi agregat
4. Mahasiswa mampu mengimplementasikan fungsi agregat Like dan Wildcards

2. Indikator

Kemampuan mahasiswa dalam memahami, membedakan dan mengimplementasikan Like dan Wildcards sebagai fungsi Agregat

3. Dasar Teori

Dalam fungsi agregat yang melibatkan Like dan Wildcard, terdapat dua hal yang seperti sama dari keduanya karena Wildcard melibatkan operator Like didalamnya tapi keduanya memiliki fungsi yang berbeda. Like biasa disebut sebagai operator, sedangkan Wildcard biasa disebut sebagai karakter.

Operator LIKE digunakan dalam klausa WHERE untuk mencari pola tertentu pada sebuah kolom. Sedangkan karakter Wildcards digunakan untuk mencari/menggantikan satu atau lebih karakter dalam sebuah string (*field*). Fitur Wildcards diimplementasikan seperti operator LIKE dan terletak di klausa WHERE.

Ada dua wildcard yang sering digunakan bersama dengan operator LIKE :

1. Tanda persen (%) mewakili nol, satu, atau beberapa karakter
2. Tanda garis bawah (_) mewakili satu karakter tunggal

Tanda persen dan garis bawah juga dapat digunakan dalam kombinasi!

Berikut adalah syntax dari LIKE :

```
SELECT column1, column2, ...
FROM table_name
WHERE columnN LIKE pattern;
```

Tip: Anda juga dapat menggabungkan sejumlah kondisi menggunakan operator AND atau OR. Berikut ini merupakan tabel yang menunjukkan symbol apa saja yang ada pada Wildcards dalam SQL Server :

Simbol	Diskripsi	Contoh
%	Mewakili nol atau lebih karakter	bl% finds bl, black, blue, and blob
_	Mewakili karakter tunggal	h_t finds hot, hat, and hit
[]	Mewakili Represents any single character within the brackets	h[oa]t finds hot and hat, but not hit
^	Represents any character not in the brackets	h[^oa]t finds hit, but not hot and hat
-	Represents any single character within the specified range	c[a-b]t finds cat and cbt

Semua wildcard juga dapat digunakan dalam kombinasi!

Berikut adalah beberapa contoh yang menunjukkan operator LIKE yang berbeda dengan wildcard '%' dan '_':

Operator LIKE	Diskripsi
WHERE CustomerName LIKE 'a%'	Menemukan nilai apapun yang dimulai dengan "a"
WHERE CustomerName LIKE '%a'	Menemukan nilai apapun yang diakhiri dengan "a"
WHERE CustomerName LIKE '%or%'	Menemukan nilai apapun yang memiliki "or" di posisi apapun
WHERE CustomerName LIKE '_r%'	menemukan nilai apapun yang memiliki "r" diposisi kedua
WHERE CustomerName LIKE 'a_%'	Menemukan nilai apapun yang diawali dengan "a" dan panjangnya minimal dua karakter
WHERE CustomerName LIKE 'a_.%'	Menemukan nilai apapun yang diawali dengan "a" dan panjangnya minimal panjangnya tiga karakter
WHERE ContactName LIKE 'a%o'	Menemukan nilai apapun yang diawali dengan "a" dan diakhiri dengan "o"

4. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

5. Prosedur Kerja

Operator LIKE

1. Lakukan implementasi sederhana pada penggunaan operator LIKE yang menampilkan semua record dari nama pelanggan (CustomerName) yang berawalan huruf "a" dengan ketikkan syntax sebagai berikut :

```
SELECT * FROM Customers  
WHERE CustomerName LIKE 'a%';
```

Tampilkan hasilnya dan buat kesimpulan!

Tampilkan semua nama pelanggan (CustomerName) yang memiliki unsur huruf “or” diposisi manapun dengan syntax sebagai berikut :

```
SELECT * FROM Customers  
WHERE CustomerName LIKE '%or%';
```

Tampilkan hasilnya dan tunjukkan kebenarannya!

2. Tampilkan Nama Produk (ProductName) dan unit pada tabel **Products** dengan nama kategori (CategoryName) pada tabel **Categories** yang awalnya “condi_”. Karena atribut CategoryName tidak berada dalam satu tabel, maka perlu dilakukan JOIN antara tabel **Products** dengan tabel **Categories**. Gunakan syntax sebagai berikut :

```
SELECT p.ProductName, p.unit, c.CategoryName  
FROM Products p join Categories c  
ON p.CategoryID = c.CategoryID  
WHERE c.CategoryName like 'condi_%';
```

Tampilkan hasinya dan buat kesimpulan!

Wildcards

1. Tampilkan implementasi sederhana dari penggunaan Wildcards dengan syntax sebagai berikut :

```
SELECT * FROM Customers  
WHERE City LIKE '[a-c]%';
```

Tampilkan hasilnya dan berikan tanda untuk menunjukkan kondisi berdasarkan perintah dari syntax diatas!

2. Buatlah syntax yang mengimplementasikan penggunaan Wildcards, tampilkan hasilnya dan berikan penjelasan!

f. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)
2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf

Kumpulkan pada <http://jti.polije.ac.id/elearning/>

i. Kesimpulan

Mahasiswa berhasil menampilkan select tabel sesuai ketentuan.

j. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 20

Pokok Bahasan	: Fungsi Agregat (IN dan Aliases)
Acara Praktikum/Pertemuan	: Minggu 6/3
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami fungsi agregat
2. Mahasiswa mampu memahami IN dan Aliases sebagai fungsi agregat
3. Mahasiswa mampu mengimplementasikan fungsi agregat IN dan Aliases

b. Indikator

Kemampuan mahasiswa dalam memahami dan mengimplementasikan IN dan Aliases sebagai bagian dari fungsi agregat.

c. Dasar Teori

1. IN

Operator IN merupakan salah satu operator yang bisa digunakan pada perintah SELECT.. Bentuk singkatan untuk beberapa kondisi OR menggunakan Operator IN. Perintah IN memungkinkan untuk mencari beberapa nilai dalam (IN) menggunakan klausa WHERE. Operator IN ini berguna melakukan pencocokan dengan salah satu yang ada pada suatu daftar nilai. Berikut syntax dari operator IN adalah sebagai berikut :

```
SELECT column_name(s)
FROM table_name
WHERE column_name IN (value1, value2, ...);
```

Atau dapat juga menggunakan syntax :

```
SELECT column_name(s)
FROM table_name
WHERE column_name IN (SELECT STATEMENT);
```

Contoh penggunaan dari operator IN adalah dengan menampilkan semua data Customers yang lokasinya di Germany dan USA.

Showing rows 0 - 23 (24 total, Query took 0.0023 seconds.)

SELECT * FROM `customers` WHERE country IN ('Germany', 'USA');

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 | Filter rows: Search this table: Sort by key: None

+ Options

	CustomerID	CustomerName	Address	City	PostalCode	Country	Email
<input type="checkbox"/>	Edit Copy Delete 1	Alfreds Futterkiste	Obere Str. 57	Berlin	12209	Germany	
<input type="checkbox"/>	Edit Copy Delete 8	Blauer See Delikatessen	Forsterstr. 57	Mannheim	68306	Germany	
<input type="checkbox"/>	Edit Copy Delete 17	Drachenblut Delikatessend	Walserweg 21	Aachen	52066	Germany	
<input type="checkbox"/>	Edit Copy Delete 25	Frankenversand	Berliner Platz 43	München	80805	Germany	
<input type="checkbox"/>	Edit Copy Delete 32	Great Lakes Food Market	2732 Baker Blvd.	Eugene	97403	USA	
<input type="checkbox"/>	Edit Copy Delete 36	Hungry Coyote Import Store	City Center Plaza 516 Main St.	Elgin	97827	USA	
<input type="checkbox"/>	Edit Copy Delete 39	Königlich Essen	Maubelstr. 90	Brandenburg	14776	Germany	
<input type="checkbox"/>	Edit Copy Delete 43	Lazy K Kountry Store	12 Orchestra Terrace	Walla Walla	99362	USA	
<input type="checkbox"/>	Edit Copy Delete 44	Lehmanna Marktstand	Magazinweg 7	Frankfurt a.M.	60528	Germany	
<input type="checkbox"/>	Edit Copy Delete 45	Let's Stop N Shop	87 Polk St. Suite 5	San Francisco	94117	USA	
<input type="checkbox"/>	Edit Copy Delete 48	Lonesome Pine Restaurant	89 Chiaroscuro Rd.	Portland	97219	USA	
<input type="checkbox"/>	Edit Copy Delete 52	Morgenstern Gesundkost	Heerstr. 22	Leipzig	4179	Germany	
<input type="checkbox"/>	Edit Copy Delete 55	Old World Delicatessen	2743 Bering St.	Anchorage	99508	USA	
<input type="checkbox"/>	Edit Copy Delete 56	Ottilies Käseladen	Mehrheimerstr. 389	Köln	50739	Germany	
<input type="checkbox"/>	Edit Copy Delete 63	QUICK-Stop	Taucherstraße 10	Cunewalde	1307	Germany	
<input type="checkbox"/>	Edit Copy Delete 65	Rattlesnake Canyon Grocery	2817 Milton Dr.	Albuquerque	87110	USA	
<input type="checkbox"/>	Edit Copy Delete 71	Save-a-lot Markets	187 Suffolk	Ln. Boise	83720	USA	
<input type="checkbox"/>	Edit Copy Delete 75	Split Rail Beer & Ale	P.O. Box 555	Lander	82520	USA	
<input type="checkbox"/>	Edit Copy Delete 77	The Big Cheese	89 Jefferson Way Suite 2	Portland	97201	USA	
<input type="checkbox"/>	Edit Copy Delete 78	The Cracker Box	55 Grizzly Peak Rd.	Butte	59801	USA	
<input type="checkbox"/>	Edit Copy Delete 79	Toms Spezialitäten	Luisenstr. 48	Münster	44087	Germany	
<input type="checkbox"/>	Edit Copy Delete 82	Trail's Head Gourmet Provisioners	722 DaVinci Blvd.	Kirkland	98034	USA	
<input type="checkbox"/>	Edit Copy Delete 86	Die Wandermde Kuh	Adenauerallee 900	Stuttgart	70563	Germany	
<input type="checkbox"/>	Edit Copy Delete 89	White Clover Markets	305 - 14th Ave. S. Suite 3B	Seattle	98128	USA	

Contoh penggunaan operator IN dengan SELECT adalah dengan menampilkan semua record Customers yang berasal dari negara (country) yang sama dengan Suppliers.

2. Aliases

Aliases digunakan untuk memberi tabel, atau kolom dalam tabel, nama sementara. Dibuat dengan kata kunci AS dan sering digunakan untuk membuat nama kolom agar lebih mudah dibaca. AS hanya ada selama query tersebut dijalankan. AS tidak tersimpan secara permanen didalam sebuah tampilan tabel. Berikut syntax dari penggunaan Aliases adalah sebagai berikut :

- Aliases Kolom

```
SELECT column_name AS alias_name
FROM table_name;
```

- Aliases Tabel

```
SELECT column_name(s)
FROM table_name AS alias_name;
```

Terdapat beberapa kondisi contoh implementasi penggunaan AS yang berbeda – beda, yaitu :

- Simple AS dengan menampilkan CustomerID dengan ID, dan CustomerName dengan Pelanggan

```
SELECT CustomerID AS ID, CustomerName AS Pelanggan  
FROM Customers;
```

- Nama aliases yang berisi spasi membutuhkan kutip ganda atau kurung siku dengan menampilkan CustomerID alias No Urut dan CustomerName alias Nama Pelanggan

```
SELECT CustomerID AS "Nomor Urut", CustomerName AS [Nama Pelanggan]  
FROM Customers;
```

- Nama alias yang menggabungkan beberapa kolom menjadi satu dengan menampilkan CustomerName alias Nama Pelanggan dan Address alias alamat yang didalamnya terdapat (Address, PostalCode, City, dan Country)

```
SELECT CustomerName AS [Nama Pelanggan],  
       Address + ', ' + PostalCode + ' ' + City + ', ' + Country AS Alamat  
  FROM Customers;
```

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

Operator IN

1. Tampilkan semua record tabel Customers dan semua record tabel Suppliers
2. Tampilkan semua record Customers yang berasal dari country yang sama dengan suppliers.
3. Carilah country mana yang tidak terdapat pada tabel Customers dengan Tabel Suppliers?
4. Temukan syntaxnya!

Penggunaan Aliases

Tampilkan implementasi penggunaan aliases berdasarkan dasar teori dan tampilkan hasilnya dari setiap implementasi tersebut. Amati dan buat kesimpulan!

h. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)
2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf

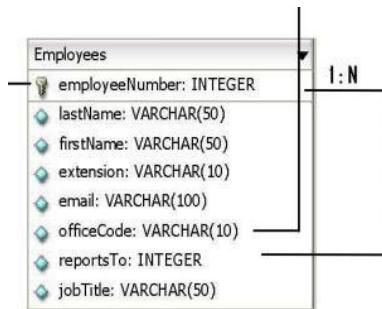
Kumpulkan pada <http://iti.polije.ac.id/elearning/>

k. Kesimpulan

Mahasiswa berhasil menampilkan select tabel sesuai ketentuan.

l. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak seusai	V			
	Total Skor*	Skor x 25			



- Bisa kita anggap bahwa kolom reportsTo merupakan kolom yang menyatakan siapa yang menjadi atasan seorang pegawai
- Terdapat dua opsi penggunaan self join adalah dapat menggunakan koma dan where, atau dapat menggunakan self join dengan klausa Join On.
- Berikut adalah contoh syntaxnya :

SOLUSI 1 (MENGGUNAKAN KOMA DAN WHERE) :

```

SELECT peg.EmployeeNumber NoPegawai,
       concat_ws(' ',peg.firstName,peg.lastName) NamaPegawai,
       peg.reportsTo,
       ats.EmployeeNumber NoAtasan,
       concat_ws(' ',ats.firstName,ats.lastName) NamaAtasan
  FROM employees peg, employees ats
 WHERE peg.reportsTo = ats.employeeNumber
  
```

SOLUSI 2 (JOIN ON) :

```

SELECT peg.EmployeeNumber NoPegawai,
       concat_ws(' ',peg.firstName,peg.lastName) NamaPegawai,
       peg.reportsTo,
       ats.EmployeeNumber NoAtasan,
       concat_ws(' ',ats.firstName,ats.lastName) NamaAtasan
  FROM employees peg JOIN employees ats
 ON peg.reportsTo = ats.employeeNumber
  
```

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

- Berikut merupakan tabel “Customers”

CustomerID	CustomerName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Avda. de la Constitución 2222	México D.F.	5021	Mexico
3	Antonio Moreno Taquería	Mataderos 2312	México D.F.	5023	Mexico
5	Berglunds snabbköp	Berguvsvägen 8	Luleå	S-958 22	Sweden
6	Blauer See Delikatessen	Forsterstr. 57	Mannheim	68306	Germany
7	Blondel père et fils	24, place Kléber	Strasbourg	67000	France
8	Bólido Comidas preparadas	C/ Araquil, 67	Madrid	28023	Spain
9	Bon app	12, rue des Bouchers	Marseille	13008	France
10	Bottom-Dollar Marketse	23 Tsawassen Blvd.	Tsawassen	T2F8M4	Canada
12	Cactus Comidas para llevar	Cerrito 333	Buenos Aires	1010	Argentina
13	Centro comercial Moctezuma	Sierras de Granada 9993	México D.F.	5022	Mexico
14	Chop-suey Chinese	Hauptstr. 29	Bern	3012	Switzerland
15	Comércio Mineiro	Av. dos Lusíadas, 23	São Paulo	05432-043	Brazil
17	Drachenblut Delikatessend	Walserweg 21	Aachen	52066	Germany
18	Du monde entier	67 rue des Cinquante Otages	Nantes	44000	France
20	Ernst Handel	Kirchgasse 6	Graz	8010	Austria

- Kalian dapat mendambahkan record baru pada tabel Customers yang telah di upload d elearning
- Lakukan self join tabel dengan masukkan syntax berikut :

```
SELECT A.CustomerName AS CustomerName1, B.CustomerName AS CustomerName2, A.City
FROM Customers A, Customers B
WHERE A.CustomerID <> B.CustomerID
AND A.City = B.City
ORDER BY A.City;
```

- Tampilkan hasil dari self join tabel yang telah dilakukan, amati dan jelaskan maksud dari perintah tersebut!

f. Hasil dan Pembahasan

- Dokumentasi tugas berupa laporan (Tugas)
- Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf Kumpulkan pada

<http://iti.polije.ac.id/elearning/>

g. Kesimpulan

Mahasiswa berhasil menampilkan select tabel sesuai ketentuan.

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 22

Pokok Bahasan	: Implementasi Operator
Acara Praktikum/Pertemuan	: Minggu 7/1
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami Operator dan jenis – jenisnya pada MySQL.
2. Mahasiswa mampu mengimplementasi penggunaan Operator UNION dan EXIST dalam MySQL.

b. Indikator

Kemampuan mahasiswa dalam memahami dan mengimplementasikan jenis-jenis operator pada MySQL khususnya UNION dan EXIST.

c. Dasar Teori

Operator merupakan kata atau karakter khusus yang digunakan SQL untuk melakukan operasi perbandingan dan aritmatika. Operator biasanya menggunakan clausa WHERE dalam mengimplementasikan operasi tersebut. Selain itu, operator juga digunakan untuk menentukan kondisi dalam pernyataan SQL dan berfungsi sebagai konjungsi untuk beberapa kondisi dalam sebuah pernyataan.

Operator memiliki beberapa jenis diantaranya adalah sebagai berikut :

1. Operator Aritmatika

Operator aritmatika digunakan untuk operasi matematika pada data numerik, seperti penambahan atau pengurangan. Berikut adalah contoh penggunaan operator aritmatika :

Operator	Deskripsi	Contoh
+	Penambahan	<code>SELECT 40 + 10;</code>
-	Pengurangan	<code>SELECT 30 - 10;</code>
*	Perkalian	<code>SELECT 70 * 0;</code>
/	Pembagian	<code>SELECT 40 / 20;</code>
%	Modulo	<code>SELECT 30 % 20;</code>

2. Operator Bitwise

Operator bitwise melakukan manipulasi bit antara dua ekspresi tipe data integer. Operator bitwise mengubah bilangan bulat menjadi bit biner dan kemudian melakukan operasi AND (& simbol), OR (|, ^) atau NOT (~) pada setiap bit individu, sebelum akhirnya mengubah hasil biner kembali menjadi bilangan bulat. Berikut adalah contoh penggunaan operator bitwise :

Operator	Deskripsi
&	Bitwise AND
	Bitwise OR
^	Bitwise exclusive OR
&=	Bitwise AND Assignment
=	Bitwise OR Assignment
^=	Bitwise exclusive OR Assignment
~	Bitwise NOT

3. Operator Perbandingan (Comparison Operator)

Operator perbandingan digunakan untuk membandingkan dua nilai dan menguji apakah keduanya sama.

Operator	Deskripsi	Contoh
=	Sama Dengan	<code>SELECT * FROM Produk WHERE Harga= 18;</code>
>	Lebih Besar Dari	<code>SELECT * FROM Produk WHERE Harga > 40;</code>
<	Lebih Kecil Dari	<code>SELECT * FROM Produk WHERE Harga < 30;</code>
>=	Lebih Besar atau Sama Dengan	<code>SELECT * FROM Produk WHERE Harga >= 25;</code>
<=	Lebih Kecil atau Sama Dengan	<code>SELECT * FROM Produk WHERE Harga <= 18;</code>
<>	Tidak Sama Dengan	<code>SELECT * FROM Produk WHERE Harga <> 17;</code>

4. Operator Logika

Operator logika adalah operator yang mengembalikan nilai true atau false, seperti operator AND, yang mengembalikan nilai true ketika kedua ekspresi terpenuhi.

Operator	Deskripsi	Contoh
ALL	TRUE jika semua nilai subkueri memenuhi kondisi	<code>SELECT * FROM Product WHERE Price > ALL (SELECT Price FROM Product WHERE Price > 500);</code>
AND	TRUE jika semua ketentuan yang dipisahkan oleh AND is TRUE	<code>SELECT * FROM Customers WHERE City = "London" AND Country = "UK";</code>
ANY	TRUE jika salah satu nilai subkueri memenuhi kondisi	<code>SELECT * FROM Product WHERE Price > ANY (SELECT Price FROM Product WHERE Price > 70);</code>

BETWEEN	TRUE jika operand berada dalam jangkauan perbandingan	<code>SELECT * FROM Product WHERE Price BETWEEN 40 AND 60;</code>
EXISTS	TRUE jika subkueri mengembalikan satu atau lebih rekaman	<code>SELECT * FROM Product WHERE EXISTS (SELECT Price FROM Product WHERE Price > 30);</code>
IN	TRUE jika operan sama dengan salah satu daftar ekspresi	<code>SELECT * FROM Customers WHERE City IN ('Paris','London');</code>
LIKE	TRUE jika operan cocok dengan pola	<code>SELECT * FROM Customers WHERE City LIKE 'u%';</code>
NOT	Menampilkan record jika kondisi NOT TRUE	<code>SELECT * FROM Customers WHERE City NOT LIKE 'L%';</code>
OR	TRUE jika salah satu ketentuan yang dipisahkan oleh OR adalah TRUE	<code>SELECT * FROM Customers WHERE City = "London" OR Country = "UK";</code>
SOME	TRUE jika salah satu nilai subkueri memenuhi kondisi	<code>SELECT * FROM Products WHERE Price > SOME (SELECT Price FROM Produk WHERE Price > 10);</code>

5. Operator UNION

Operator UNION disebut juga sebagai Operator Gabungan merupakan operator yang digunakan untuk menggabungkan kumpulan hasil dari dua atau lebih pernyataan SELECT.

Beberapa aturan yang perlu diperhatikan ketika menggunakan Operator UNION diantaranya adalah sebagai berikut :

- Setiap pernyataan SELECT dalam UNION harus memiliki jumlah kolom yang sama
- Kolom juga harus memiliki tipe data yang serupa
- Kolom di setiap pernyataan SELECT juga harus dalam urutan yang sama

Berikut adalah macam – macam penggunaan operator UNION dengan syntax sebagai berikut:

a. UNION

```
SELECT column_name(s) FROM table1
UNION
SELECT column_name(s) FROM table2;
```

Note: Operator UNION memilih hanya nilai yang berbeda secara default. Untuk memungkinkan nilai-nilai duplikat, gunakan kata kunci ALL dengan UNION.

b. UNION All

```
SELECT column_name(s) FROM table1
UNION ALL
SELECT column_name(s) FROM table2;
```

Catatan: Nama kolom dalam kumpulan hasil biasanya sama dengan nama kolom dalam pernyataan SELECT pertama.

c. UNION dengan WHERE

UNION dengan WHERE pada contoh kasus menampilkan City dan Country dengan mengembalikan kota Germany (hanya nilai yang berbeda) dari tabel Customers dan Suppliers :

```
SELECT City, Country FROM Customers
WHERE Country='Germany'
UNION
SELECT City, Country FROM Suppliers
WHERE Country='Germany'
ORDER BY City;
```

d. UNION All dengan WHERE

UNION All dengan WHERE pada contoh kasus menampilkan City dan Country dengan mengembalikan kota Germany (dengan nilai yang terduplikasi juga) dari tabel Customers dan Suppliers :

```
SELECT City, Country FROM Customers
WHERE Country='Germany'
UNION ALL
SELECT City, Country FROM Suppliers
WHERE Country='Germany'
ORDER BY City;
```

e. Another UNION

Another UNION pada contoh kasus menampilkan Customers yang ditampilkan sebagai kolom Type, ContactName, City, Country, dari tabel “Customers” yang digabungkan dengan Supplier dimasukkan dalam kolom Type, ContactName, City, dan Country dari tabel Suppliers.

```
SELECT 'Customer' AS Type, ContactName, City, Country
FROM Customers
UNION
SELECT 'Supplier', ContactName, City, Country
FROM Suppliers;
```

6. Operator EXISTS

EXISTS merupakan sebuah operator yang digunakan untuk menguji keberadaan record apapun dalam subquery. Operator EXISTS mengembalikan nilai TRUE jika subquery mengembalikan satu atau lebih record.

Berikut merupakan syntax penggunaan operator EXISTS :

```
SELECT column_name(s)
FROM table_name
WHERE EXISTS
(SELECT column_name FROM table_name WHERE condition);
```

f. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

g. Prosedur Kerja

UNION

1. Tampilkan Country dari tabel Customers dan tabel Suppliers
2. Tampilkan seluruh record Country (baik yang duplikasi data) dari tabel Customers dan tabel Suppliers
3. Lakukan implementasi kasus UNION dengan WHERE berdasarkan dasar teori diatas, tampilkan hasilnya dan buat penjelasannya!
4. Lakukan implementasi kasus UNION ALL dengan WHERE berdasarkan dasar teori diatas, tampilkan hasilnya dan buat penjelasannya!
5. Lakukan implementasi kasus Another UNION berdasarkan dasar teori diatas, tampilkan hasilnya dan buat penjelasannya!

EXISTS

1. Tampilkan seluruh record tabel Products dan tabel Suppliers
2. Buatlah pernyataan SQL untuk mengembalikan TRUE dan mencantumkan pemasok (Suppliers) dengan harga produk (Products) kurang dari 20!
3. Tampilkan hasilnya dan buat penjelasan!

g. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)
2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf

Kumpulkan pada <http://jti.polije.ac.id/elearning/>

g. Kesimpulan

Mahasiswa berhasil mengetahui jenis-jenis operator

Mahasiswa berhasil mengimplementasikan operator lanjutan dari UNION dan EXISTS

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 23

Pokok Bahasan	: Implementasi Statement
Acara Praktikum/Pertemuan	: Minggu 7/2
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

1. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami seluruh Statement yang ada pada MySQL
2. Mahasiswa mampu mengimplementasikan Group By dan CASE Statement

2. Indikator

1. Kemampuan mahasiswa dalam memahami seluruh statement yang ada pada MySQL
2. Kemampuan mahasiswa dalam memahami dan mengimplementasikan Group By dan CASE Statement

3. Dasar Teori

Statement merupakan perintah-perintah baik dasar maupun pengembangan yang dimiliki oleh Structure Query Language (SQL). Ada beberapa statement yang digunakan untuk implementasi SQL diantaranya adalah :

1. Select

Merupakan perintah dasar SQL yang digunakan untuk memilih data dari database.

```
SELECT column1, column2, ...
FROM table_name;
```

```
SELECT * FROM table_name;
```

2. Select Distinct

Pernyataan SELECT DISTINCT digunakan hanya untuk mengembalikan nilai yang berbeda. Dalam tabel sering kali berisi kolom dengan banyak nilai duplikat, dan terkadang kita hanya ingin membuat daftar nilai yang berbeda. Berikut syntax penggunaan SELECT DISTINCT :

```
SELECT DISTINCT column1, column2, ...
FROM table_name;
```

3. Insert Into

INSERT INTO Statement digunakan untuk menyisipkan record baru ke dalam tabel. Ada dua cara penggunaan syntax SQL dalam implementasi INSERT INTO Statement diantaranya adalah :

1. Tentukan nama kolom dan nilai yang akan disisipkan:

```
INSERT INTO table_name (column1, column2, column3, ...)
VALUES (value1, value2, value3, ...);
```

2. Jika ingin menambahkan nilai untuk semua kolom tabel, maka tidak perlu menentukan nama kolom dalam kueri SQL. cukup pastikan urutan nilai dalam urutan yang sama dengan kolom dalam tabel.

```
INSERT INTO table_name
VALUES (value1, value2, value3, ...);
```

4. Update

Pernyataan UPDATE digunakan untuk mengubah catatan yang ada dalam tabel. Berikut adalah syntax SQL dalam implementasi UPDATE Statement :

```
UPDATE table_name
SET column1 = value1, column2 = value2, ...
WHERE condition;
```

Catatan: Berhati-hatilah saat memperbarui catatan dalam tabel! Perhatikan klausa WHERE dalam pernyataan UPDATE. Klausa WHERE menentukan record mana yang harus diperbarui. Jika Anda menghilangkan klausa WHERE, semua catatan dalam tabel akan diperbarui!

5. Delete

Pernyataan DELETE digunakan untuk menghapus record yang ada dalam tabel. Berikut adalah syntax SQL dalam implementasi DELETE Statement :

```
DELETE FROM table_name WHERE condition;
```

Catatan: Hati-hati saat menghapus record dalam tabel! Perhatikan klausa WHERE dalam pernyataan DELETE. Klausa WHERE menentukan record mana yang harus dihapus. Jika Anda menghilangkan klausa WHERE, semua record dalam tabel akan dihapus!

6. Group By

Group By Statement digunakan untuk mengelompokkan baris yang memiliki nilai yang sama ke dalam baris ringkasan, seperti "temukan jumlah pelanggan di setiap negara". Pernyataan GROUP BY sering digunakan dengan fungsi agregat (COUNT(), MAX(), MIN(), SUM(), AVG()) untuk mengelompokkan kumpulan hasil menurut satu atau beberapa kolom. Berikut adalah syntax penggunaan GROUP BY Statement:

```
SELECT column_name(s)
FROM table_name
WHERE condition
GROUP BY column_name(s)
ORDER BY column_name(s);
```

Selain itu penggunaan statement GROUP BY bisa diimplementasikan menggunakan JOIN dengan contoh menampilkan record dari setiap pengirim yang mencantumkan jumlah pesanan yang dikirim dengan kode sebagai berikut :

```
SELECT Shippers.ShipperName, COUNT(Orders.OrderID) AS NumberOfOrders FROM Orders
LEFT JOIN Shippers ON Orders.ShipperID = Shippers.ShipperID
GROUP BY ShipperName;
```

7. Insert Into Select

```
INSERT INTO table2
SELECT * FROM table1
WHERE condition;
```

```
INSERT INTO table2 (column1, column2, column3, ...)
SELECT column1, column2, column3, ...
FROM table1
WHERE condition;
```

8. Case

Statement CASE melewati kondisi dan mengembalikan nilai ketika kondisi pertama terpenuhi (seperti pernyataan if-then-else). Jadi, setelah suatu kondisi dinyatakan benar, maka akan berhenti membaca dan mengembalikan hasilnya. Jika tidak ada kondisi yang benar, maka kondisi tersebut akan mengembalikan nilai dalam klausa ELSE.

Jika tidak ada bagian ELSE dan tidak ada kondisi yang benar, maka kondisi tersebut akan mengembalikan NULL.

Berikut adalah syntax dari statement CASE :

```

CASE
    WHEN condition1 THEN result1
    WHEN condition2 THEN result2
    WHEN conditionN THEN resultN
    ELSE result
END;

```

3. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

4. Prosedur Kerja

1. Tampilkan seluruh record data pada tabel suppliers, products, categories.

Ketentuan	Tampilan																																		
Tampilkan daftar total Pemasok (Suppliers) di setiap negara (Country)	<table border="1"> <thead> <tr> <th>Negara</th><th>Total Pemasok</th></tr> </thead> <tbody> <tr><td>Australia</td><td>2</td></tr> <tr><td>Brazil</td><td>1</td></tr> <tr><td>Canada</td><td>2</td></tr> <tr><td>Denmark</td><td>1</td></tr> <tr><td>Finland</td><td>1</td></tr> <tr><td>France</td><td>3</td></tr> <tr><td>Germany</td><td>3</td></tr> <tr><td>Italy</td><td>2</td></tr> <tr><td>Japan</td><td>2</td></tr> <tr><td>Netherlands</td><td>1</td></tr> <tr><td>Norway</td><td>1</td></tr> <tr><td>Singapore</td><td>1</td></tr> <tr><td>Spain</td><td>1</td></tr> <tr><td>Sweden</td><td>2</td></tr> <tr><td>UK</td><td>2</td></tr> <tr><td>USA</td><td>4</td></tr> </tbody> </table>	Negara	Total Pemasok	Australia	2	Brazil	1	Canada	2	Denmark	1	Finland	1	France	3	Germany	3	Italy	2	Japan	2	Netherlands	1	Norway	1	Singapore	1	Spain	1	Sweden	2	UK	2	USA	4
Negara	Total Pemasok																																		
Australia	2																																		
Brazil	1																																		
Canada	2																																		
Denmark	1																																		
Finland	1																																		
France	3																																		
Germany	3																																		
Italy	2																																		
Japan	2																																		
Netherlands	1																																		
Norway	1																																		
Singapore	1																																		
Spain	1																																		
Sweden	2																																		
UK	2																																		
USA	4																																		
Tampilkan daftar total Pelanggan (Customers) di setiap negara (Country) yang diurutkan dari rendah ke tinggi	<table border="1"> <thead> <tr> <th>Country</th><th>Total Suppliers</th></tr> </thead> <tbody> <tr><td>Norway</td><td>1</td></tr> <tr><td>Singapore</td><td>1</td></tr> <tr><td>Denmark</td><td>1</td></tr> <tr><td>Netherlands</td><td>1</td></tr> <tr><td>Spain</td><td>1</td></tr> <tr><td>Finland</td><td>1</td></tr> <tr><td>Brazil</td><td>1</td></tr> <tr><td>Italy</td><td>2</td></tr> <tr><td>UK</td><td>2</td></tr> <tr><td>Japan</td><td>2</td></tr> <tr><td>Australia</td><td>2</td></tr> <tr><td>Canada</td><td>2</td></tr> <tr><td>Sweden</td><td>2</td></tr> <tr><td>France</td><td>3</td></tr> <tr><td>Germany</td><td>3</td></tr> <tr><td>USA</td><td>4</td></tr> </tbody> </table>	Country	Total Suppliers	Norway	1	Singapore	1	Denmark	1	Netherlands	1	Spain	1	Finland	1	Brazil	1	Italy	2	UK	2	Japan	2	Australia	2	Canada	2	Sweden	2	France	3	Germany	3	USA	4
Country	Total Suppliers																																		
Norway	1																																		
Singapore	1																																		
Denmark	1																																		
Netherlands	1																																		
Spain	1																																		
Finland	1																																		
Brazil	1																																		
Italy	2																																		
UK	2																																		
Japan	2																																		
Australia	2																																		
Canada	2																																		
Sweden	2																																		
France	3																																		
Germany	3																																		
USA	4																																		

Tampilkan jumlah products yang ada di CategoryName :

CategoryName	NumberofProducts
Dairy Products	2
Grains/Cereals	2
Meat/Poultry	2
Produce	2
Seafood	3
Beverages	3
Confections	5
Condiments	6

2. Tampilkan Produk, harga, dan keterangan dengan ketentuan sebagai berikut :

- (1). Harga > 25.00 “Harga Produk diatas 25.00; (2). Harga = 25.00 “Harga Produk = 25.00”; (3). Harga <25.00 “Harga Produk dibawah 25.00

Nama Produk	Harga	Keterangan
Chais	18.00	Harga Produk dibawah 25.00
Chang	19.00	Harga Produk dibawah 25.00
Aniseed Syrup	10.00	Harga Produk dibawah 25.00
Chef Anton's Cajun Seasoning	22.00	Harga Produk dibawah 25.00
Chef Anton's Gumbo Mix	21.00	Harga Produk dibawah 25.00
Grandmas Boysenberry Spread	25.00	Harga Produk = 25.00
Uncle Bobs Organic Dried Pears	30.00	Harga Produk diatas 25.00
Northwoods Cranberry Sauce	40.00	Harga Produk diatas 25.00
Mishi Kobe Niku	97.00	Harga Produk diatas 25.00
Ikura	31.00	Harga Produk diatas 25.00
Queso Cabrales	21.00	Harga Produk dibawah 25.00
Queso Manchego La Pastora	38.00	Harga Produk diatas 25.00
Konbu	6.00	Harga Produk dibawah 25.00
Tofu	23.25	Harga Produk dibawah 25.00
Genen Shouyu	15.50	Harga Produk dibawah 25.00
Pavlova	17.45	Harga Produk dibawah 25.00
Alice Mutton	39.00	Harga Produk diatas 25.00
Camarón Tigers	62.50	Harga Produk diatas 25.00
Teatime Chocolate Biscuits	9.20	Harga Produk dibawah 25.00
Sir Rodney's Marmalade	81.00	Harga Produk diatas 25.00
Sir Rodney's Scones	10.00	Harga Produk dibawah 25.00
Gustafs Knäckebrot	21.00	Harga Produk dibawah 25.00
Tunnbröd	9.00	Harga Produk dibawah 25.00
Guaraná Fantástica	4.50	Harga Produk dibawah 25.00
NuNuCa Nuñ-Nougat-Creme	14.00	Harga Produk dibawah 25.00

f. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)
2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf

Kumpulkan pada <http://iti.polije.ac.id/elearning/>

i. Kesimpulan

Mahasiswa berhasil mengetahui jenis-jenis statement yang ada pada database

Mahasiswa berhasil menampilkan implementasi statement pada database sesuai ketentuan

j. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat	V			
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 24

Pokok Bahasan	: Implementasi Calusa
Acara Praktikum/Pertemuan	: Minggu 7/3
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami penjelasan dari Clause
2. Mahasiswa mampu memahami jenis – jenis Clause
3. Mahasiswa mampu mengimplementasikan Clause

b. Indikator

Kemampuan mahasiswa dalam memahami dan menggunakan having sebagai bagian implementasi Clause.

c. Dasar Teori

Dalam kamus Bahasa Indonesia klausa merupakan gabungan kata yang terdiri dari subjek dan predikat. Selain itu klausa juga biasanya dilengkapi dengan menggunakan objek, pelengkap, dan keterangan. Dalam MySQL juga terdapat klausa dan jenis – jenis klausa yang ada pada MySQL adalah sebagai berikut :

1. Where

Klausa WHERE digunakan untuk memfilter record. Klausa ini digunakan untuk mengekstrak catatan yang hanya memenuhi kondisi tertentu. Berikut adalah syntax dari klausa WHERE :

```
SELECT column1, column2, ...
FROM table_name
WHERE condition;
```

2. Select Top

Klausa SELECT TOP digunakan untuk menentukan jumlah record yang akan dikembalikan. Klausa SELECT TOP berguna pada tabel besar dengan ribuan record. Mengembalikan sejumlah besar catatan dapat memengaruhi kinerja. Tidak semua sistem

database mendukung Klausula SELECT TOP. Biasanya klausula ini dapat berjalan pada SQL namun tidak berjalan pada MySQL. Ada beberapa klausula yang dapat mendukung Klausula SELECT TOP pada database lainnya, diantaranya adalah :

1. MySQL menggunakan LIMIT Clause untuk memilih sejumlah record yang terbatas
2. Oracle menggunakan FETCH FIRST n ROWS ONLY dan ROWNUM.

Berikut ini merupakan syntaxnya :

SQL Server / MS Access Syntax:

```
SELECT TOP number|percent column_name(s)
FROM table_name
WHERE condition;
```

MySQL Syntax:

```
SELECT column_name(s)
FROM table_name
WHERE condition
LIMIT number;
```

Oracle Syntax:

```
SELECT column_name(s)
FROM table_name
ORDER BY column_name(s)
FETCH FIRST number ROWS ONLY;
```

3. Having

Klausula Having digunakan untuk mengelompokan nilai/data berdasarkan kondisi tertentu dengan fungsi agregat yang tidak bisa menggunakan klausula WHERE. Misal terdapat record data pada tabel Products dengan field ProductID, ProductName, dan Price dengan ProductID 1 s/d 10 maka syntax dan hasilnya adalah sebagai berikut :

```
SELECT ProductID, ProductName, Price FROM `products` WHERE ProductID <=10;
```

ProductID	ProductName	Price
1	Chais	18.00
2	Chang	19.00
3	Aniseed Syrup	10.00
4	Chef Anton's Cajun Seasoning	22.00
5	Chef Anton's Gumbo Mix	21.00
6	Grandmas Boysenberry Spread	25.00
7	Uncle Bobs Organic Dried Pears	30.00
8	Northwoods Cranberry Sauce	40.00
9	Mishi Kobe Niku	97.00
10	Ikura	31.00

Dari 10 data tersebut kita akan menggunakan klausula HAVING untuk menampilkan Price yang kurang dari 20.00 dan mengelompokkan data berdasarkan ProductName. Berikut

```
SELECT ProductID, ProductName, Price FROM `products`  
WHERE ProductID <=10  
GROUP BY ProductName  
HAVING Price<20.00;
```

ProductID	ProductName	Price
3	Aniseed Syrup	10.00
1	Chais	18.00
2	Chang	19.00

syntax dan hasilnya :

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

1. Buatlah rangkuman dari Operator, Statement, dan Clause
2. Implementasikan penggunaan LIMIT Clause pada database ‘smartmarket’
3. Implementasikan penggunaan HAVING Clause dengan penggunaan syntax berikut :

```
SELECT ProductID, ProductName, Price  
SELECT ProductID, ProductName, Price FROM `products`  
WHERE ProductID <=10  
GROUP BY ProductName  
HAVING Price<20.00;
```

Tampilkan semua syntax diatas dan berikan penjelasan dari perbedaan hasilnya. Sebutkan perbedaan antara klausa Where dan klausa Having!

h. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)
2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulandan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf

Kumpulkan pada <http://iti.polije.ac.id/elearning/>

k. Kesimpulan

Mahasiswa berhasil mengimplementasikan jenis-jenis klausula sesuai ketentuan.

l. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak seusai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 25

Pokok Bahasan	: Mencari Studi Kasus
Acara Praktikum/Pertemuan	: Minggu 9/1
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami contoh kasus dari implementasi database
2. Mahasiswa mampu menerapkan contoh kasus dari implementasi database

b. Indikator

Keberhasilan mahasiswa dalam memahami contoh kasus dan mengimplementasikannya kedalam database

c. Dasar Teori

Setiap hari kita selalu berurusan dengan data. Ketika kita mendengarkan lagu favorit dan memutar playlist di smartphone kita, kita sedang berurusan dengan basis data. Playlist tersebut merupakan basis data. Ketika kita mengunggah foto ke media sosial misal Facebook dari galeri foto, kita juga sedang berurusan dengan basis data. Galeri foto tersebut merupakan basis data. Saat kita menelusuri marketplace seperti tokopedia, dan menambahkan barang yang akan kita beli ke keranjang belanja, dapat dikatakan bahwa kita sedang berurusan dengan basis data. Basis data sangat dibutuhkan dalam kehidupan kita sehari - hari. Untuk mendapatkan basis data yang sesuai dengan kebutuhan maka kita perlu untuk merancang basis data tersebut.

Perancangan merupakan suatu hal yang sangat penting dalam pembuatan basis data. Permasalahan yang dihadapi pada waktu perancangan yaitu bagaimana basis data yang akan dibangun ini dapat memenuhi kebutuhan saat ini dan masa yang akan datang. Untuk itu diperlukan perancangan basis data baik secara fisik maupun secara konseptualnya. Perancangan konseptual akan menunjukkan entity dan relasinya berdasarkan proses yang diinginkan oleh organisasinya. Untuk menentukan entity dan relasinya perlu dilakukan analisis data tentang informasi yang ada dalam spesifikasi di masa yang akan datang. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Data adalah deskripsi tentang benda, kejadian, aktivitas dan transaksi, yang tidak mempunyai makna atau

tidak berpengaruh secara langsung kepada pemakai. Data adalah sesuatu yang belum mempunyai arti bagi penerimanya dan masih memerlukan adanya suatu pengolahan. Data adalah sekumpulan fakta yang diambil dari beberapa kejadian yang memiliki arti penting yang dapat berbentuk sebuah file yang dapat disimpan. Data merupakan bentuk yang masih mentah yang belum bercerita banyak, sehingga perlu diolah lebih lanjut menjadi suatu model untuk dihasilkan informasi. Meskipun data itu bersifat penting, namun data masih belum bisa dijadikan sesuatu yang bermanfaat untuk dijadikan sebuah keputusan. Supaya menjadi sebuah informasi yang bermanfaat, data-data yang ada kemudian dikumpulkan lalu diolah sedemikian rupa hingga menghasilkan suatu informasi. Elemen data adalah suatu data terkecil yang tidak dapat dipecah lagi menjadi unit data yang lain. Istilah lain untuk elemen data adalah field, kolom, item dan atribut.

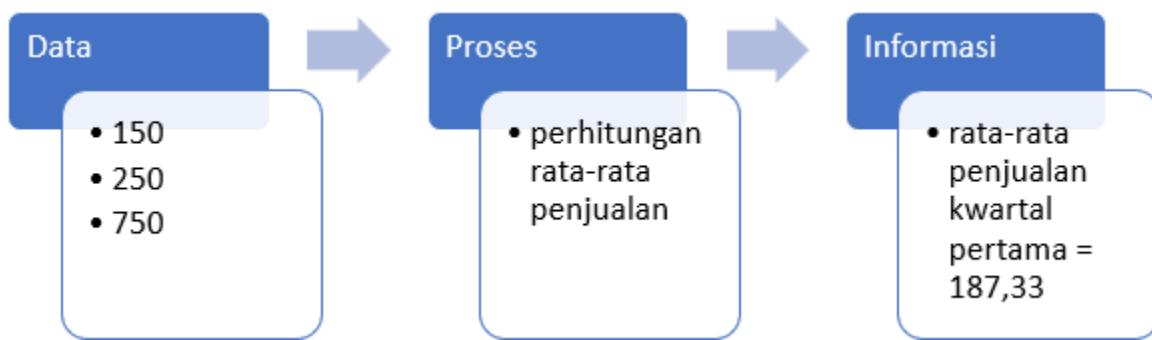
Contoh elemen data pada data Pegawai:

- nama pegawai
- alamat
- kota tempat tinggal
- atribut lain yang berkaitan dengan pegawai

Informasi adalah kumpulan data yang sudah melalui proses pengolahan sehingga dapat menjadi sesuatu yang bermanfaat bagi seseorang yang kemudian digunakan untuk pengambilan suatu keputusan atau tindakan. Informasi merupakan hasil pengolahan dari sebuah model, formasi, organisasi, ataupun suatu perubahan bentuk dari data yang memiliki nilai tertentu, dan bisa digunakan untuk menambah pengetahuan bagi yang menerimanya.

Data vs Informasi :

- Data = bahan mentah informasi
- Informasi mempunyai kandungan ‘makna’ , data tidak
- Data bagi seseorang bisa jadi merupakan informasi bagi orang lain
- Data bisa dianggap sebagai obyek dan informasi adalah suatu subyek yang bermanfaat bagi penerimanya.



Gambar. Alur Data

A. Tujuan Perancangan Database

- Untuk memenuhi informasi yang berisikan kebutuhan-kebutuhan user secara khusus dan aplikasi – aplikasinya.
- Memudahkan pengertian struktur informasi.
- Mendukung kebutuhan-kebutuhan pemrosesan dan beberapa obyek penampilan (response time, processing time, dan storeage space).

B. Proses Perancangan Database

Proses perancangan database terdiri dari 6 tahap:

- Tahap 1, Pengumpulan data dan analisis
- Tahap 2, Perancangan database secara konseptual
- Tahap 3, Pemilihan DBMS
- Tahap 4, Perancangan database secara logika (data model mapping)
- Tahap 5, Perancangan database secara fisik
- Tahap 6, Implementasi Sistem database

Enam tahapan diatas tadi tidak harus diproses berurutan. Pada tahap ke 1 merupakan kumpulan informasi yang berhubungan dengan penggunaan database. Tahap 6 merupakan implementasi database-nya. Tahap 1 dan 6 kadang-kadang bukan merupakan bagian dari perancangan database. Sedangkan yang merupakan inti dari proses perancangan database adalah pada tahap 2, 4, 5.

- **Tahap 1 – Pengumpulan data dan Analisa**

Merupakan suatu tahap dimana kita melakukan proses indentifikasi dan analisa kebutuhan-kebutuhan data dan ini disebut pengumpulan data dan analisa. Untuk menentukan kebutuhan-kebutuhan suatu sistem *database*, kita harus mengenal terlebih dahulu bagian-bagian lain dari sistem informasi yang akan berinteraksi dengan sistem *database*, termasuk para *user* yang ada dan para *user* yang baru beserta aplikasi-aplikasinya. Kebutuhan-kebutuhan dari para *user* dan aplikasi-aplikasi inilah yang kemudian dikumpulkan dan dianalisa. Berikut ini adalah aktifitas-aktifitas pengumpulan data dan analisa:

- a) Menentukan kelompok pemakai dan bidang-bidang aplikasinya
- b) Peninjauan dokumentasi yang ada
- c) Analisa lingkungan operasi dan pemrosesan data
- d) Daftar pertanyaan dan wawancara

- **Tahap 2, Perancangan *database* secara konseptual**

Pada tahap ini akan dihasilkan *conceptual schema* untuk *database* yang tergantung pada sebuah DBMS yang spesifik. Sering menggunakan sebuah *high-level data model* seperti ER/EER model selama tahap ini. Dalam *conceptual schema*, kita harus merinci aplikasi-aplikasi *database* yang diketahui dan transaksi-transaksi yang mungkin. Tahap perancangan *database* secara konseptual mempunyai 2 aktifitas pararel:

- a) Perancangan skema konseptual

Menguji kebutuhan-kebutuhan data dari suatu *database* yang merupakan hasil dari tahap 1 dan menghasilkan sebuah *conceptual database schema* pada DBMS-*independent* model data tingkat tinggi seperti EER (*Enhanced Entity Relationship*) model. Untuk menghasilkan skema tersebut dapat dihasilkan dengan penggabungan bermacam-macam kebutuhan user dan secara langsung membuat skema database atau dengan merancang skema-skema yang terpisah dari kebutuhan tiap-tiap user dan kemudian menggabungkan skema-skema tersebut. Model data yang digunakan pada perancangan skema konseptual adalah DBMS-*independent* dan langkah selanjutnya adalah memilih DBMS untuk melakukan rancangan tersebut.

- b) Perancangan transaksi

Menguji aplikasi-aplikasi *database* dimana kebutuhan-kebutuhannya telah dianalisa pada fase 1, dan menghasilkan perincian transaksi-transaksi ini. Kegunaan tahap ini yang

diproses secara paralel bersama tahap perancangan skema konseptual adalah untuk merancang karakteristik dari transaksi-transaksi *database* yang telah diketahui pada suatu DBMS-*independent*. Transaksi-transaksi ini akan digunakan untuk memproses dan memanipulasi *database* suatu saat dimana *database* tersebut dilaksanakan.

- **Tahap 3, Pemilihan DBMS**

Pemilihan *database* ditentukan oleh beberapa faktor diantaranya faktor teknik, ekonomi, dan politik organisasi. Contoh faktor teknik: Keberadaan DBMS dalam menjalankan tugasnya seperti jenis-jenis DBMS (*relational*, *network*, *hierarchical*, dan lain-lain), struktur penyimpanan, dan jalur akses yang mendukung DBMS, pemakai, dan lain-lain. Faktor-faktor ekonomi dan organisasi yang mempengaruhi satu sama lain dalam pemilihan DBMS :

- a) Struktur data

Jika data yang disimpan dalam *database* mengikuti struktur hirarki, maka suatu jenis hirarki dari DBMS harus dipikirkan.

- b) Personal yang telah terbiasa dengan suatu system

Jika staf *programmer* dalam suatu organisasi sudah terbiasa dengan suatu DBMS, maka hal ini dapat mengurangi biaya latihan dan waktu belajar.

- c) Tersedianya layanan penjual

Keberadaan fasilitas pelayanan penjual sangat dibutuhkan untuk membantu memecahkan beberapa masalah sistem.

- **Tahap 4, Perancangan *database* secara logika (*data model mapping*)**

Tahap selanjutnya adalah membuat sebuah skema konseptual dan skema eksternal pada model data dari DBMS yang terpilih. Tahap ini dilakukan oleh pemetaan skema konseptual dan skema eksternal yang dihasilkan pada tahap 2. Pada tahap ini, skema konseptual ditransformasikan dari model data tingkat tinggi yang digunakan pada tahap 2 ke dalam model data dari model data dari DBMS yang dipilih pada tahap 3. Pemetaan tersebut dapat diproses dalam 2 tingkat:

- a) Pemetaan *system-independent*

Pemetaan ke dalam model data DBMS dengan tidak mempertimbangkan karakteristik atau hal-hal yang khusus yang berlaku pada implementasi DBMS dari model data tersebut.

- b) Penyesuaian skema ke DBMS yang spesifik

Mengatur skema yang dihasilkan pada langkah 1 untuk disesuaikan pada implementasi yang khusus di masa yang akan datang dari suatu model data yang digunakan pada DBMS yang dipilih. Hasil dari tahap ini memakai perintah-perintah DDL (*Data Definition Language*) dalam bahasa DBMS yang dipilih yang menentukan tingkat skema konseptual dan eksternal dari sistem *database*. Tetapi 10 dalam beberapa hal, perintah-perintah DDL memasukkan parameter-parameter rancangan fisik sehingga DDL yang lengkap harus menunggu sampai tahap perancangan *database* secara fisik telah lengkap. Tahap ini dapat dimulai setelah pemilihan sebuah implementasi model data sambil menunggu DBMS yang spesifik yang akan dipilih. Contoh: jika memutuskan untuk menggunakan beberapa *relational DBMS* tetapi belum memutuskan suatu relasi yang utama. Rancangan dari skema eksternal untuk aplikasi-aplikasi yang spesifik seringkali sudah selesai selama proses ini.

- **Tahap 5, Perancangan *database* secara fisik**

Perancangan *database* secara fisik merupakan proses pemilihan struktur-struktur penyimpanan dan jalur-jalur akses pada *file-file database* untuk mencapai penampilan yang terbaik pada bermacam-macam aplikasi. Selama fase ini, dirancang spesifikasi-spesifikasi untuk *database* yang disimpan yang berhubungan dengan struktur-struktur penyimpanan fisik, penempatan record dan jalur akses. Berhubungan dengan *internal schema* (pada istilah 3 level arsitektur DBMS). Beberapa petunjuk dalam pemilihan perancangan *database* secara fisik :

- a) *Response time*

Waktu yang telah berlalu dari suatu transaksi *database* yang diajukan untuk menjalankan suatu tanggapan. Pengaruh utama pada *response time* adalah di bawah pengawasan DBMS yaitu : waktu akses *database* untuk data item yang ditunjuk oleh suatu transaksi. *Response time* juga dipengaruhi oleh beberapa faktor yang tidak berada di bawah pengawasan DBMS, seperti penjadwalan sistem operasi atau penundaan komunikasi.

- b) *Space utility*

Jumlah ruang penyimpanan yang digunakan oleh *file-file database* dan struktur-struktur jalur akses.

c) *Transaction throughput*

Rata-rata jumlah transaksi yang dapat diproses per menit oleh sistem *database*, dan merupakan parameter kritis dari sistem transaksi (misal : digunakan pada pemesanan tempat di pesawat, bank, dll). Hasil dari fase ini adalah penentuan awal dari struktur penyimpanan dan jalur akses untuk *file-file database*.

- **Tahap 6, Implementasi Sistem *database***

Setelah perancangan secara logika dan secara fisik lengkap, kita dapat melaksanakan sistem *database*. Perintah-perintah dalam DDL dan SDL(*Storage Definition Language*) dari DBMS yang dipilih, dihimpun dan digunakan untuk membuat skema *database* dan *file-file database* (yang kosong). Sekarang *database* tersebut dimuat (disatukan) dengan datanya. Jika data harus dirubah dari sistem komputer sebelumnya, perubahan-perubahan yang rutin mungkin diperlukan untuk format ulang datanya yang kemudian dimasukkan ke *database* yang baru. Transaksi-transaksi *database* sekarang harus dilaksanakan oleh para programmer aplikasi. Spesifikasi secara konseptual diuji dan dihubungkan dengan kode program dengan perintah-perintah dari *embedded DML* yang telah ditulis dan diuji. Suatu saat transaksi-transaksi tersebut telah siap dan data telah dimasukkan ke dalam *database*, maka tahap perancangan dan implementasi telah selesai, dan kemudian tahap operasional dari sistem *database* dimulai.

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

1. Buat kelompok dengan anggota kelompok yang terdiri dari 3 mahasiswa
2. Cari studi kasus dan implementasikan kedalam database dengan ketentuan struktur memiliki tabel minimal 6 (enam) dengan jumlah minimal relasi yang terbentuk adalah 3 (tiga).
3. Masing-masing anggota harus memahami rancangan yang dibuat

4. Buat laporan yang berisikan :
 1. Pengumpulan data dan analisis
 2. Perancangan database secara konseptual (ER Diagram)

g. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)
2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf

Kumpulkan pada <http://iti.polije.ac.id/elearning/>

g. Kesimpulan

Mahasiswa mampu menguasai studi kasus dengan memahami rancangan yang dibuat.

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Kekompakan tim baik, kreatifitas ppt/laporan baik dan ketepatan dalam menjelaskan dari tugas ditunjang dengan bukti referensi				V
2	Kekompakan tim baik, kreatifitas ppt/laporan kurang dan ketepatan dalam menjelaskan dari tugas ditunjang dengan bukti referensi			V	
3	Kekompakan tim baik, kreatifitas ppt/laporan kurang dan ketepatan dalam menjelaskan dari tugas ditunjang dengan bukti referensi kurang		V		
4	Kelompok tidak kompak, kreatifitas ppt tidak baik, ketidaktepatan dalam menjelaskan tugas	V			
		Total Skor*		Skor x 25	

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 26

Pokok Bahasan : Presentasi Studi Kasus

Acara Praktikum/Pertemuan : Minggu 9/2

Tempat : Daring/Luring Politeknik Negeri Jember

Alokasi Waktu : 100 menit

1. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu mempresentasikan hasil diskusi dengan kelompok berupa pengumpulan data dan analisis serta perancangan database secara konseptual
2. Mahasiswa mampu bekerjasama dalam tim

2. Indikator

Kemampuan mahasiswa dalam mempresentasikan studi kasus dan bekerjasama dalam tim

3. Dasar Teori

-

4. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

5. Prosedur Kerja

1. Perwakilan kelompok mengambil lotrean / menggunakan lotrean online pada <https://id.rakko.tools/>
2. Masing - masing kelompok mempresentasikan dengan maksimal waktu 10 menit

3. Kelompok yang akan maju berikutnya berkewajiban untuk memberikan pertanyaan kepada kelompok yang presentasi minimal 1 pertanyaan

f. Hasil dan Pembahasan

Dosen akan memberikan komentar kepada presenter

i. Kesimpulan

Mahasiswa mampu mempresentasikan studi kasus dan bekerjasama dalam tim

j. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Kekompakan tim baik, kreatifitas ppt/laporan baik dan ketepatan dalam menjelaskan dari tugas ditunjang dengan bukti referensi				V
2	Kekompakan tim baik, kreatifitas ppt/laporan kurang dan ketepatan dalam menjelaskan dari tugas ditunjang dengan bukti referensi			V	
3	Kekompakan tim baik, kreatifitas ppt/laporan kurang dan ketepatan dalam menjelaskan dari tugas ditunjang dengan bukti referensi kurang		V		
4	Kelompok tidak kompak, kreatifitas ppt tidak baik, ketidaktepatan dalam menjelaskan tugas	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 27

Pokok Bahasan	: Struktur Tabel
Acara Praktikum/Pertemuan	: Minggu 10/1
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu membuat struktur tabel dari implementasi studi kasus
2. Mahasiswa mampu memahami struktur tabel yang dibuat berdasarkan studi kasus dari implementasi database
3. Mahasiswa mampu mengimplementasikan DDL ke dalam studi kasus
4. Mahasiswa mampu bekerjasama dengan tim

b. Indikator

1. Keberhasilan mahasiswa dalam membuat dan memahami struktur tabel dari studi kasus dan mengimplementasikannya kedalam database
2. Keberhasilan mahasiswa dalam mengimplementasikan DDL ke dalam studi kasus

c. Dasar Teori

Database atau basis data merupakan kumpulan data yang tersimpan secara sistematis di dalam computer dan dapat diolah atau dimanipulasi menggunakan perangkat lunak (program aplikasi) untuk menghasilkan informasi. Pendefinisian basis data terdiri dari tipe data, struktur data, dan batasan-batasan data yang akan disimpan. Basis data merupakan aspek yang sangat penting dalam sistem informasi dimana basis data merupakan gudang penyimpanan data yang akan diolah lebih lanjut. Basis data menjadi penting karena dapat mengorganisasi data, menghindari duplikasi data, hubungan antar data yang tidak jelas juga update yang rumit.

Berikut adalah jenis-jenis dari basis data yang dibagi menjadi 2 (dua) yaitu :

1. Basis data flat-file

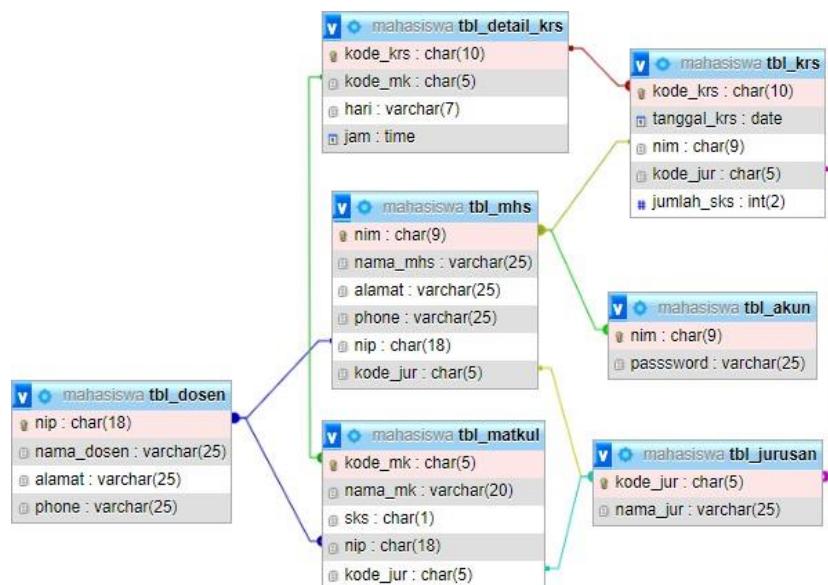
Basis data flat-file ideal untuk data yang berukuran kecil dan dapat dirubah dengan mudah.

Pada dasarnya, mereka tersusun dari sekumpulan string dalam satu atau lebih file yang dapat diurai untuk mendapatkan informasi yang disimpan. Basis data flat-file baik digunakan untuk menyimpan daftar atau data yang sederhana dan dalam jumlah kecil.

Basis data flat-file akan menjadi sangat rumit apabila digunakan untuk menyimpan data dengan struktur kompleks walaupun dimungkinkan pula untuk menyimpan data semacam itu. Beberapa kendala dalam menggunakan basis data jenis ini diantaranya adalah rentan pada korupsi data karena tidak adanya penguncian yang melekat ketika data digunakan atau dimodifikasi serta adanya duplikasi data yang mungkin sulit dihindari. Salah satu tipe basis data flat-file adalah file CSV yang menggunakan pemisah koma untuk setiap nilainya.

2. Basis data relasional

Basis data ini mempunya struktur yang lebih logis terkait cara penyimpanan. Kata “relasional” berasal dari kenyataan bahwa tabel-tabel yang berada di basis data dapat dihubungkan satu dengan yang lainnya. Basis data relasional menggunakan sekumpulan tabel dua dimensi yang masing-masing tabel tersusun atas baris (tupel) dan kolom (atribut). Untuk membuat hubungan antara dua atau lebih tabel, gunakan key (atribut kunci) yaitu **primary key** di salah satu tabel dan **foreign key** di tabel yang lain. Saat ini, basis data relational menjadi pilihan karena keunggulannya. Beberapa kelemahan yang mungkin dirasakan untuk basis data jenis ini adalah implementasi yang sulit untuk data dalam jumlah besar dengan tingkat kompleksitasnya yang tinggi dan proses pencarian informasi yang lebih lambat karena perlu menghubungkan tabel-tabel terlebih dahulu apabila datanya tersebar di beberapa tabel.



Gambar 10.1 Contoh Struktur Database

Terlihat pada Gambar 10.1 merupakan struktur database mahasiswa untuk mengakses krs dengan melibatkan data mahasiswa, dosen sebagai dosen wali, mata kuliah, jurusan, krs, detail krs, dan akun untuk user id. Dalam database tersebut ditunjukan bahwa struktur database tersebut memiliki form master yaitu : tbl_mhs, tbl_dosen, tbl_matkul, tbl_krs, dan tbl_jurusan. Selain itu juga terdapat tbl_akun untuk akses user id dan tbl_detail_krs untuk detail dari krs. Dapat dilihat dari gambar diatas, tabel satu dengan yang lainnya saling terhubung. Dalam struktur database primary key sangat berperan penting sebagai kunci utama agar tabel dapat saling terhubung. Field penghubung harus memiliki foreign key atau index disesuaikan dengan kebutuhan dari relasi antar tabel.

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

1. Buatlah database, tabel dan relasi dari hasil implementasi ER Diagram sesuai studi kasus masing-masing kelompok.
2. Jika terdapat tabel yang belum normal, maka lakukan normalisasi
3. Lakukan dokumentasi setiap langkah dalam menyelesaikan studi kasus pembuatan database dan tabel dengan perintah CREATE
4. Termasuk dalam penggunaan query ALTER dan DROP jika diperlukan.

g. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)

2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf

Kumpulkan pada <http://iti.polije.ac.id/elearning/>

g. Kesimpulan

1. Mahasiswa mampu menguasai studi kasus dengan memahami rancangan yang dibuat.
2. Mahasiswa mampu membuat struktur database pada studi kasus masing-masing kelompok
3. Implementasi hasil ER Diagram pada struktur tabel dan implementasi DDL pada database berdasarkan studi kasus

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Kekompakan tim baik, kreatifitas ppt/laporan baik dan ketepatan dalam menjelaskan dari tugas ditunjang dengan bukti referensi				V
2	Kekompakan tim baik, kreatifitas ppt/laporan kurang dan ketepatan dalam menjelaskan dari tugas ditunjang dengan bukti referensi			V	
3	Kekompakan tim baik, kreatifitas ppt/laporan kurang dan ketepatan dalam menjelaskan dari tugas ditunjang dengan bukti referensi kurang	V			
4	Kelompok tidak kompak, kreatifitas ppt tidak baik, ketidaktepatan dalam menjelaskan tugas	V			
		Total Skor*		Skor x 25	

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 28

Pokok Bahasan	: Memasukan data ke dalam tabel
Acara Praktikum/Pertemuan	: Minggu 10/2
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

1. Capaian Pembelajaran Mata Kuliah (CPMK)

Mahasiswa mampu mengimplementasikan DML kedalam studi kasus

2. Indikator

Kemampuan mahasiswa dalam mengimplementasikan DML kedalam studi kasus

3. Dasar Teori

Dapat dilihat pada Minggu ke 4 workshop Acara ke 9 sampai dengan Acara ke 12.

4. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

5. Prosedur Kerja

1. Isikan record data dimasing-masing tabel pada studi kasus dengan statement INSERT
2. Gunakan statement SELECT, UPDATE, DELETE jika diperlukan
3. Lakukan dokumentasi di setiap langkah pengisian record data pada studi kasus.

f. Hasil dan Pembahasan

Dokumentasi berupa laporan

i. Kesimpulan

Mahasiswa mampu mempresentasikan studi kasus dan bekerjasama dalam tim

j. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 29

Pokok Bahasan	: Studi Kasus : Seleksi antar tabel (Join dua Tabel)
Acara Praktikum/Pertemuan	: Minggu 11/1
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

Mahasiswa mampu mengimplementasikan join tabel pada studi kasus

b. Indikator

Keberhasilan mahasiswa dalam mengimplementasikan join tabel pada studi kasus

c. Dasar Teori

Dapat dilihat pada Minggu ke-5 Workshop Acara 14 sampai dengan Acara 18

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

1. Implementasikan join tabel dengan relasi 2 tabel dari studi kasus
2. Implementasikan Join dengan Kondisi dan Join tanpa Kondisi pada studi kasus
3. Implementasikan Natural Join pada studi kasus
4. Implementasikan Join dengan clausa yang lain pada studi kasus
5. Lakukan dokumentasi disetiap implementasi join pada studi kasus

f. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)

2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf

Kumpulkan pada <http://iti.polije.ac.id/elearning/>

g. Kesimpulan

Mahasiswa mampu mengimplementasikan relasi dua tabel pada studi kasus.

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 30

Pokok Bahasan	: Seleksi antar tabel (Join Tiga Tabel)
Acara Praktikum/Pertemuan	: Minggu 11/2
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

Mahasiswa mampu mengimplementasikan DML kedalam studi kasus

b. Indikator

Kemampuan mahasiswa dalam mengimplementasikan DML kedalam studi kasus

c. Dasar Teori

JOIN 3 Table Dengan Klausula Where

Mengembalikan baris dimana setidaknya ada satu buah key yang cocok diantara ketiga table

```
SELECT table-1.nama_field, table-1. nama_field, table-2.nama_field,  table-3.nama_field
FROM table-1, table-2, table-3
WHERE table-1.nama_field(PK) = table-2.nama_field(FK)
AND  table-2.nama_field(PK) = table-3.nama_field(FK)
```

Contoh:

Terdapat tiga tabel yaitu tabel karyawan, tabel Bagian, tabel Gaji dengan masing-masing record data sebagai berikut :

1. Tabel Karyawan

← ↑ →	▼	idkrywn	namakrywn	alamat
<input type="checkbox"/>   		KY001	Ucok	pasar senen
<input type="checkbox"/>   		KY002	Suleman	mumbulsari
<input type="checkbox"/>   		KY003	A. Bidin	Kencong

2. Tabel Bagian

	<input type="checkbox"/> Edit <input type="button" value="Copy"/> <input type="button" value="Delete"/>	idbagian	namabagian
<input type="checkbox"/>	<input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	KBG01	Instalasi
<input type="checkbox"/>	<input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	KBG02	Gudang
<input type="checkbox"/>	<input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	KBG03	Marketing

3. Tabel Gaji

	<input type="checkbox"/> Edit <input type="button" value="Copy"/> <input type="button" value="Delete"/>	idkrywn	idbagian	totalgaji
<input type="checkbox"/>	<input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	KY001	KBG01	1000000
<input type="checkbox"/>	<input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	KY002	KBG02	21000
<input type="checkbox"/>	<input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	KY003	KBG03	34000

Dengan menampilkan record data sebagai berikut :

namakrywn	alamat	namabagian	totalgaji
Ucok	pasar senen	Instalasi	1000000
Suleman	mumbulsari	Gudang	21000
A. Bidin	Kencong	Marketing	34000

Maka penulisan syntaxnya adalah sebagai berikut :

```
SELECT karyawan.namakrywn, karyawan.alamat, bagian.namabagian, gaji.totalgaji
FROM karyawan, bagian, gaji
WHERE karyawan.idkrywn=gaji.idkrywn
AND bagian.idbagian=gaji.idbagian;
```

Join 3 Tabel dengan Klausa Join

Mengembalikan baris dimana setidaknya ada satu buah key yang cocok diantara ketiga table dengan klausa *join*.

```
SELECT table-1.nama_field, table-1.nama_field, table-2.nama_field, table-3.nama_field
FROM table-1
JOIN table-2
ON table-1.nama_field(PK) = table-2.nama_field(FK)
JOIN table-3
ON table-2.nama_field(PK) = table-3.nama_field(FK)
```

Contoh :

Untuk menampilkan record data tersebut :

namakrywn	alamat	namabagian	totalgaji
Ucok	pasar senen	Instalasi	1000000
Suleman	mumbulsari	Gudang	21000
A. Bidin	Kencong	Marketing	34000

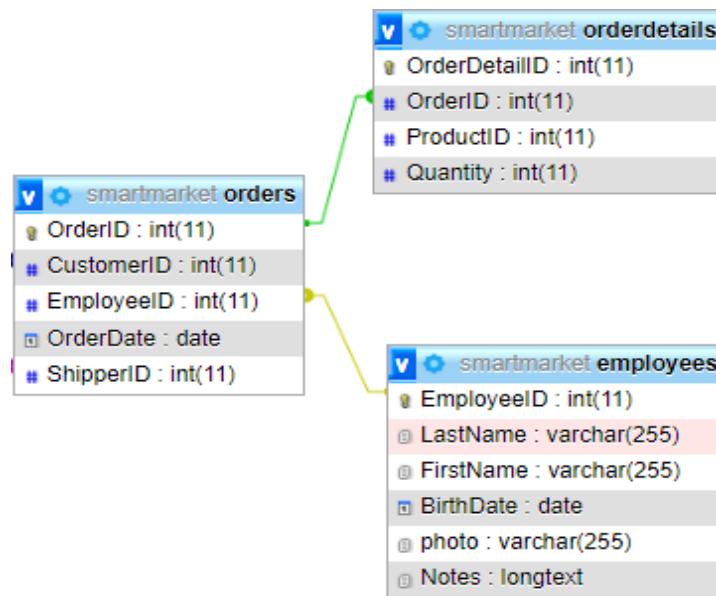
Maka syntax dengan klausa join adalah sebagai berikut :

```
SELECT karyawan.namakrywn, karyawan.alamat, bagian.namabagian, gaji.totalgaji
FROM karyawan JOIN gaji
ON karyawan.idkrywn=gaji.idkrywn
JOIN bagian
ON bagian.idbagian=gaji.idbagian;
```

Join Tiga Tabel Dengan Kondisi

Berikut ini adalah contoh join tiga tabel dengan kondisi yang menampilkan EmployeID dari tabel **employees**, Name Employee (FirstName) dari tabel **employees**, shiperID dari tabel **orders**, dan quantity orders (quantity) dari tabel **orderdetails** yang quantity ordersnya terbanyak yang dimiliki oleh name employee.

Berikut ini adalah gambar relasi tabelnya :



Pada gambar diatas menunjukkan bahwa terdapat tiga tabel yang berelasi yaitu tabel orders, tabel orderdetails, dan tabel employees.

Dimana ketika kita ingin menampilkan rangking dari nama employee dengan quantity order terbanyak hingga terkecil adalah seperti pada tabel berikut :

EmployeeID	FirstName	COUNT(orderdetails.quantity)	ShipperID
4	Margaret	24	2
3	Janet	14	1
8	Laura	12	3
6	Michael	10	1
9	Anne	8	3
5	Steven	8	3
1	Nancy	7	1
2	Andrew	7	1

Maka gunakan join tiga tabel dengan syntax sebagai berikut :

```
SELECT employees.EmployeeID, employees.FirstName, COUNT(orderdetails.quantity), orders.ShipperID
FROM orders
JOIN OrderDetails
ON orders.OrderID=orderdetails.OrderID
JOIN employees
ON orders.EmployeeID=employees.EmployeeID
GROUP BY employees.FirstName
ORDER BY COUNT(orderdetails.quantity) DESC;
```

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

1. Implementasikan join tiga tabel pada studi kasus kalian
2. Implementasikan join tiga tabel dengan kondisi pada studi kasus kalian
3. Lakukan dokumentasi di setiap implementasi join tiga tabel pada studi kasus.

f. Hasil dan Pembahasan

Dokumentasi berupa laporan

g. Kesimpulan

Mahasiswa mampu mengimplementasikan join tiga tabel pada studi kasus

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 31

Pokok Bahasan	: Studi Kasus : Seleksi antar tabel (Fungsi Agregat)
Acara Praktikum/Pertemuan	: Minggu 11/3
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

Mahasiswa mampu mengimplementasikan fungsi agregasi pada studi kasus

b. Indikator

Keberhasilan mahasiswa dalam mengimplementasikan fungsi agregasi pada studi kasus

c. Dasar Teori

Dapat dilihat pada Minggu ke-6 Workshop Acara 19 sampai dengan Acara 21

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

1. Implementasikan COUNT, SUM, AVG, MIN, dan MAX pada studi kasus
2. Implementasikan LIKE dan WILDCARDS pada studi kasus
3. Implementasikan IN pada studi kasus
4. Lakukan dokumentasi disetiap implementasi fungsi aggregate pada studi kasus

f. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)

2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf

Kumpulkan pada <http://iti.polije.ac.id/elearning/>

g. Kesimpulan

Mahasiswa mampu mengimplementasikan fungsi aggregate pada studi kasus.

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 32

Pokok Bahasan	: Studi Kasus : Seleksi antar tabel (Operator)
Acara Praktikum/Pertemuan	: Minggu 11/4
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

Mahasiswa mampu mengimplementasikan Operator pada studi kasus

b. Indikator

Keberhasilan mahasiswa dalam mengimplementasikan Operator pada studi kasus

c. Dasar Teori

Dapat dilihat pada Minggu ke-7 Workshop Acara 22

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

1. Implementasikan salah satu penggunaan Operator Aritmatika pada studi kasus
2. Implementasikan salah satu penggunaan Operarot Bitwise pada studi kasus
3. Implementasikan salah satu penggunaan Operator Perbandingan studi kasus
4. Implementasikan salah satu penggunaan Operator Operator Logika
5. Implementasikan penggunaan Operator UNION dengan kondisi
6. Implementasikan penggunaan Operator EXISTS
7. Lakukan dokumentasi disetiap implementasi operator pada studi kasus

f. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)
2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulandan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E123456_BI.pdf

Kumpulkan pada <http://jti.polije.ac.id/elearning/>

g. Kesimpulan

Mahasiswa mampu mengimplementasikan relasi dua tabel pada studi kasus.

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 32

Pokok Bahasan	: Studi Kasus : Seleksi antar tabel (Case, Limit, Having)
Acara Praktikum/Pertemuan	: Minggu 11/4
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu mengimplementasikan Statement Case pada studi kasus
2. Mahasiswa mampu mengimplementasikan clausa Limit dan Having pada studi kasus

b. Indikator

Keberhasilan mahasiswa dalam mengimplementasikan Statement Case, Klausula Limit, dan Klausula Having pada studi kasus

c. Dasar Teori

Dapat dilihat pada Minggu ke-7 Workshop Acara 23 dan Acara 24.

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

1. Implementasikan salah satu penggunaan Statement Case pada studi kasus
2. Implementasikan salah satu penggunaan Klausula Limit pada studi kasus
3. Implementasikan salah satu penggunaan Klausula Having pada studi kasus
4. Lakukan dokumentasi disetiap implementasi operator pada studi kasus

f. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)
2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulandan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E123456_BI.pdf

Kumpulkan pada <http://jti.polije.ac.id/elearning/>

g. Kesimpulan

Mahasiswa mampu mengimplementasikan relasi dua tabel pada studi kasus.

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
		Total Skor*		Skor x 25	

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 33

Pokok Bahasan	: Store Procedures
Acara Praktikum/Pertemuan	: Minggu 12/1
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami penggunaan Store Procedure
2. Mahasiswa mampu mengimplementasikan penggunaan Store Procedures pada studi kasus di PhpMyAdmin

b. Indikator

Kemampuan mahasiswa dalam memahami dan mengimplementasikan penggunaan Store Procedures di PhpMyadmin berdasarkan studi kasus

c. Dasar Teori

Kondisi dimana kondisi query SQL ditulis berulang kali maka dapat disimpan sebagai Store Procedures agar dapat dipanggil kapan saja ketika ingin menjalankannya. Sehingga Store Procedures dapat diartikan sebagai program (subroutine) yang tersimpan dalam database untuk mempermudah pemanggilan query SQL yang digunakan secara berulang kali. Tujuan dari Store Procedures adalah agar pemanggilan dapat dilakukan dengan cepat karena tersimpan di server, programnya juga dapat dipanggil berulang kali (reusable), serta meningkatkan keamanan. Berikut ini merupakan tiga komponen utama yang ada pada Store Procedures, diantaranya adalah :

1. Nama Store Procedure
2. Parameter yang digunakan untuk menangani penggunaan parameter di dalam store procedures. Ada tiga jenis parameter dalam store procedure diantaranya adalah IN, OUT, dan INOUT. Parameter dapat diimplementasikan dengan single parameter maupun multi parameter yang bisa menggunakan IN atau OUT. Parameter pada store procedure juga dapat mengkombinasikan kedua parameter.

3. Tubuh (Body) yang berisi statement SQL yang akan dieksekusi.

Berikut adalah syntax dari penggunaan Store Procedures

```
CREATE PROCEDURE procedure_name
AS
sql_statement
GO;
```

Untuk Menjalankan Store Procedure gunakan syntax sebagai berikut :

```
EXEC procedure_name;
```

Contoh Implementasi :

CustomerID	CustomerName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Avda. de la Constitución 2222	México D.F.	5021	Mexico
3	Antonio Moreno Taquería	Mataderos 2312	México D.F.	5023	Mexico
5	Berglunds snabbköp	Berguvsvägen 8	Luleå	S-958 22	Sweden
6	Blauer See Delikatessen	Forsterstr. 57	Mannheim	68306	Germany
7	Blondel père et fils	24, place Kléber	Strasbourg	67000	France
8	Bólido Comidas preparadas	C/ Araquil, 67	Madrid	28023	Spain
9	Bon app	12, rue des Bouchers	Marseille	13008	France
10	Bottom-Dollar Marketse	23 Tsawassen Blvd.	Tsawassen	T2F8M4	Canada

Tabel diatas merupakan contoh tabel Customers. Tabel tersebut sering dipanggil untuk dieksekusi dalam query SQL sebagai seleksi antar tabel. Maka tabel Customers ini perlu dibuatkan Store Procedure dengan syntax sebagai berikut :

```
CREATE PROCEDURE SelectAllCustomers
AS
SELECT * FROM Customers
GO;
```

Store Procedure “SelectAllCustomer” dapat dipanggil dengan menggunakan syntax

sebagai berikut :

```
EXEC SelectAllCustomers;
```

Store Procedure dengan satu parameter

Contoh penggunaan Store Procedure dengan satu parameter dapat menggunakan syntax sebagai berikut :

```
CREATE PROCEDURE SelectAllCustomers @City nvarchar(30)
AS
SELECT * FROM Customers WHERE City = @City
GO;
```

Store Procedure dengan satu parameter tersebut dapat dipanggil dengan syntax sebagai berikut :

```
EXEC SelectAllCustomers @City = 'London';
```

Store Procedure dengan multi parameter

Dalam Store Procedure dengan multi parameter dapat disiapkan beberapa parameter yang sangat mudah. Cukup daftarkan setiap parameter dan tipe data yang dipisahkan dengan koma Berikut ini adalah syntax dari penggunaan Store Prosedur yang memilih pelanggan dari kota tertentu dengan Kodepos tertentu dari tabel "Pelanggan" :

```
CREATE PROCEDURE SelectAllCustomers @City nvarchar(30), @PostalCode nvarchar(10)
AS
SELECT * FROM Customers WHERE City = @City AND PostalCode = @PostalCode
GO;
```

Berikut adalah syntax untuk memanggil store procedures dengan multi parameter :

```
EXEC SelectAllCustomers @City = 'London', @PostalCode = 'WA1 1DP';
```

Variabel pad Store Procedures

Bagian ini digunakan untuk memanfaatkan implementasi variabel pada suatu store procedures. Pertama, lakukan deklarasi variabel beserta tipe data untuk menyimpan nilainya. Pada kasus ini, nilai yang akan disimpan bertipe mata uang, sehingga tipe data yang digunakan yaitu double. Berikut syntaxnya :

```
DELIMITER //
CREATE PROCEDURE getSumPrices()
BEGIN
DECLARE sumPrices DOUBLE DEFAULT 0.0;

SELECT SUM(Price) INTO sumPrices
FROM products;

SELECT sumPrices;
END //

sumPrices
684.4
```

Selanjutnya variabel digunakan untuk menyimpan tipe data varchar yang diperoleh dari nama lengkap customer pada tabel customer. Parameter customer_id digunakan sebagai input untuk menemukan nama pelanggan. Perhatikan kode dibawah ini.

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

1. Pelajari Store Procedure dari dasar teori diatas
 2. Implementasikan Store Procedures, Store Procedure satu parameter, dan Store Procedures multi parameter pada masing-masing studi kasus
- 3. Implementasikan store procedures parameter IN**

Pada bagian ini implementasikan parameter IN pada stored procedures. Pertama, penerapan store procedures tanpa parameter IN, namun sebenarnya menggunakan parameter IN secara default. Perhatikan syntax program di bawah ini :

```
DELIMITER //
CREATE PROCEDURE OrderdetailsView()
BEGIN
SELECT * FROM orderdetails;
END//
CALL OrderdetailsView();
```

OrderDetailID	OrderID	ProductID	Quantity
1	10248	11	12
2	10248	42	10
3	10248	72	5
4	10249	14	9
5	10249	51	40
6	10250	41	10
7	10250	51	35
8	10250	65	15
9	10251	22	6
10	10251	57	15

Dst...

Selanjutnya menggunakan parameter IN sebagai input variabel pada stored procedure.

Perhatikan syntax di bawah ini.

```
DELIMITER //
CREATE PROCEDURE getOrderdetailsViewLimit(IN lim INT)
BEGIN
SELECT OrderDetailID, ProductID, Quantity
FROM orderdetails
WHERE Quantity <= lim;
END //
CALL getOrderdetailsViewLimit(5);
```

OrderDetailID	ProductID	Quantity
3	72	5
34	37	1
43	56	2
57	72	4
87	19	1
89	35	4

4. Implementasikan Store Procedures Parameter OUT

Parameter ini digunakan untuk mengembalikan (return value) pada suatu stored procedures. Parameter OUT tidak memiliki nilai default. Perhatikan syntax di bawah ini :

```
DELIMITER //
CREATE PROCEDURE countCustomer(
OUT jumlah_cust INT)
BEGIN
SELECT COUNT(CustomerID) INTO jumlah_cust
FROM customers;
END //
```

Syntax diatas dibuat untuk mengetahui jumlah customer dan bisa dipanggil kapanpun untuk mengetahui total dari penambahan customerID pada tabel customers dengan menggunakan syntax berikut :

```
CALL countCustomer(@jumlah_cust);
SELECT @jumlah_cust;
```

Berikut adalah hasilnya :

```
@jumlah_cust
84
```

Maka dari hasil diatas diketahui bahwa total dari customerID pada tabel Customers sebanyak 84.

5. Implementasikan Store Procedure Parameter INOUT

Mode INOUT ini pada dasarnya merupakan kombinasi dari mode IN dan OUT.

Berikut adalah syntax dari penggunaan store procedure parameter INOUT :

```
DELIMITER //
CREATE PROCEDURE getINOUTProduct(
    INOUT jumlah INT)
BEGIN
    SELECT COUNT(CategoryID) INTO jumlah
    FROM products
    WHERE CategoryID = jumlah;
END //
```

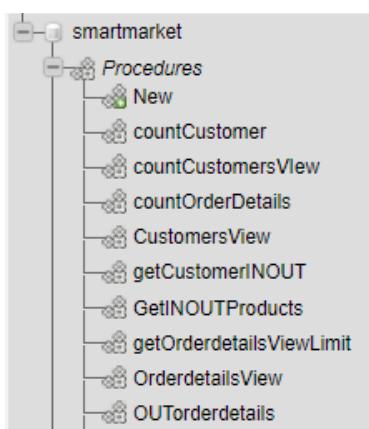
Syntax diatas dibuat untuk mengetahui berapa total categoryID tertentu pada tabel Products. Store Procedure dibuat agar dapat dipanggil setiap saat ketika ingin mengetahui total dari CategoryID tertentu pada tabel Product yang mengalami penambahan transaksi dengan menggunakan syntax sebagai berikut :

```
SET @total_CategoryID3 = 3;
CALL getINOUTProducts(@total_CategoryID3);
SELECT @total_CategoryID3;
```

Hasil dari total CategoryID 3 pada tabel products adalah sebanyak 5.

```
@total_CategoryID3
5
```

- Untuk mengakses Store Procedure yang ada pada PhpMyadmin dapat klik Procedures yang ada dalam database anda. Disini kita contohkan pada database smartmarket :



- Lakukan dokumentasi disetiap implementasi Store Procedure

f. Hasil dan Pembahasan

Dokumentasi tugas berupa laporan (Tugas)

Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulandan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf

Kumpulkan pada <http://jti.polje.ac.id/elearning/>

g. Kesimpulan

Laporan mahasiswa hasil dokumentasi dari implementasi store procedure

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 34

Pokok Bahasan	: Store Functions
Acara Praktikum/Pertemuan	: Minggu 12/2
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami Store Functions
2. Mahasiswa mampu mengimplementasikan Store Functions pada studi kasus
3. Mahasiswa mampu mengetahui perbedaan Store Procedures dengan Store Functions

b. Indikator

1. Kemampuan mahasiswa dalam memahami dan mengimplementasikan Store Function pada studi kasus
2. Kemampuan mahasiswa dalam membedakan Store Procedures dan Store Functions

c. Dasar Teori

Stored function merupakan suatu program (subroutine) yang tersimpan pada database untuk mempermudah pemanggilan query SQL yang dipakai secara berulang kali dan memiliki nilai kembalian (return value). Kelebihannya dapat membuat fungsi yang tidak tersedia. Store function terdiri dari nama, parameter, nilai return dan tubuh (body).

```
CREATE FUNCTION nama_function → Nama Function
-----
(param1, param2, paramN) → Parameter
-----
RETURN <tipe_data> → Return Value
-----
CHARACTERISTICS → karakteristik function
-----
BEGIN
{Syntaks SQL} → Body
END
```

Perhatikan format penulisan di atas. Terdapat bagian characteristic pada suatu stored function yang bernilai (pilih salah satu) DETERMINISTIC, NO SQL, or READS SQL DATA. Klausus characteristic digunakan oleh function untuk menentukan operasi apa yang diperlukan kepada server database. Apabila tidak ditentukan secara spesifik, maka gunakan DETERMINISTIC.

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

Pada bagian ini akan menampilkan sebuah stored function untuk menampilkan tanggal hari ini. Hasil return value yang dikembalikan berupa format tanggal yang sudah ditentukan. Perhatikan Syntax di bawah ini :

```
DELIMITER //
CREATE FUNCTION getDateToday()
RETURNS CHAR(30)
DETERMINISTIC
BEGIN
RETURN DATE_FORMAT(NOW(), '%D %m %Y');
END//
SELECT getDateToday();
```

getDateToday()

5th 04 2022

1. Buatlah fungsi untuk mencetak bilangan ganjil berdasarkan nilai yang dimasukkan. Nilai yang dicetak mulai dari 1 hingga nilai input yang diberikan.
2. Implementasikan Store Function kedalam studi kasus kalian.
3. Dokumentasikan setiap implementasi dari store functions yang dilakukan serta berikan penjelasan

f. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)

2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf

Kumpulkan pada <http://iti.polije.ac.id/elearning/>

g. Kesimpulan

Mahasiswa berhasil menampilkan select tabel sesuai ketentuan.

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 35

Pokok Bahasan	: Index dan View
Acara Praktikum/Pertemuan	: Minggu 13/1
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa memahami definisi Indeks dan View pada suatu database.
2. Mahasiswa mampu mempraktikkan penggunaan Index dan View pada database
3. Mahasiswa mampu mempraktikkan penggunaan View pada database

b. Indikator

Kemampuan mahasiswa dalam mendefinisikan dan mempraktikkan index dan view pada database

c. Dasar Teori**Index**

Index Indeks digunakan untuk menemukan baris dengan value dari kolom yang spesifik. Tanpa menggunakan indeks, maka DBMS harus memulai pembacaan dari baris pertama dan dilanjutkan hingga baris akhir untuk menemukan nilai yang relevant. Jika suatu tabel memiliki indeks, maka DBMS dapat dengan cepat menemukan value yang dicari baik itu berada ditangah, maupun tanpa harus melihat semua isi data. Indeks pada MySQL seperti PRIMARY KEY, UNIQUE, INDEXT, dan FULLTEXT tersimpan di B-Trees. PRIMARY KEY disebut sebagai clustered indeks karena wajibkan urutan baris pada tabel. PRIMARY KEY bersifat spesial karena indeks ini tersimpan bersama data pada tabel yang sama. Format penulisan sebagai berikut :

```
CREATE INDEX index_name  
ON table_name (column1, column2, ...);
```

Syntax CREATE UNIQUE INDEX dimana duplikasi nilai tidak diperbolehkan :

```
CREATE UNIQUE INDEX index_name  
ON table_name (column1, column2, ...);
```

Catatan: Memperbarui tabel dengan indeks membutuhkan waktu lebih lama daripada memperbarui tabel tanpa (karena indeks juga memerlukan pembaruan). Jadi, hanya buat indeks pada kolom yang akan sering dicari.

Statement DROP INDEX untuk menghapus indeks dalam tabel :

```
ALTER TABLE table_name
DROP INDEX index_name;
```

View

Views adalah tabel virtual yang tidak menyimpan data sendiri, tetapi dapat menampilkan data yang tersimpan di tabel lain. Views hanya berisi SQL Query saja dan dapat menampilkan dari satu atau banyak tabel. Jika query yang digunakan secara berkelanjutan, maka lebih baik disimpan dalam bentuk view. Format penulisannya adalah sebagai berikut :

```
CREATE VIEW view_name AS
SELECT column1, column2, ...
FROM table_name
WHERE condition;
```

Catatan: Tampilan selalu menampilkan data terkini! Database engine akan membuat ulang tampilan, setiap kali pengguna menanyakannya.

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

1. INDEX

Pada implementasi index kali ini kita akan gunakan untuk menganalisis pencarian pada suatu query, seberapa cepatkah ketika sebelum dan sesudah menerapkan indeks pada suatu kolom. Query berikut dibuat untuk menampilkan seluruh atribut pada tabel ordertedails yang OrderIDnya adalah 10250.

```
SELECT * FROM `orderdetails`
WHERE OrderID='10250';
```

Kemudian periksa performa MySQL ketika mengeksekusi query tersebut dengan menambahkan klause EXPLAIN sebelum klausa SELECT seperti berikut ini :

```
EXPLAIN SELECT * FROM `orderdetails`
WHERE OrderID = '10250';
```

Maka hasilnya adalah sebagai berikut :

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	orderdetails	ref	OrderID	OrderID	5	const	3	

Selanjutnya buat indeks pada kolom OrderID dengan menggunakan syntax berikut :

```
EXPLAIN SELECT * FROM `orderdetails`
WHERE OrderID = '10250';

CREATE INDEX OrderIDIndexView ON orderdetails(OrderID);
```

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	orderdetails	ref	OrderID,IndexOrderID,OrderIDIndex	OrderID	5	const	3	

Rows pada EXPLAIN SELECT sebelum dibuatkan indeks dengan setelah dibuatkan indeks menghasilkan nilai yang sama yaitu 3. Itu dikarenakan record data di kolom OrderID pada tabel OrderDetails masih sangat sedikit. Perbedaan dari performa pemindaian (scan) dapat dirasakan perbedaannya ketika record data pada tabel tersebut akan semakin banyak.

Tugas :

Buatlah percobaan pada tabel OrderDetails yang kalian miliki sesuai prosedur diatas. Lakukan pengamatan pada performa pemindaian dan lihat apakah ada perbedaan performa pada row dari sebelum dibuatkan INDEX dengan setelahnya.

2. VIEW

Buat tampilan yang memperlihatkan semua pelanggan dari Brasil menggunakan syntax berikut :

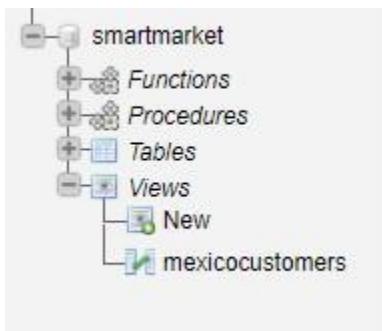
Untuk menampilkan seluruh record data pada VIEW yang sudah dibuat dapat menggunakan query sebagai berikut :

```
CREATE VIEW MexicoCustomers AS
SELECT CustomerName, Address
FROM Customers
WHERE Country = 'Mexico';
```

```
SELECT * FROM MexicoCustomers;
```

CustomerName	Address
Ana Trujillo Emparedados y helados	Avda. de la Constitución 2222
Antonio Moreno Taquería	Mataderos 2312
Centro comercial Moctezuma	Sierras de Granada 9993
Pericles Comidas clásicas	Calle Dr. Jorge Cash 321
Tortuga Restaurante	Avda. Azteca 123

Untuk dapat melihat VIEWS yang sudah dibuat dapat dilihat pada gambar berikut :



Tugas :

1. Implementasikan VIEWS pada contoh diatas
2. Buatlah VIEWS yang menampilkan OrderDetailID dari tabel orderdetails, ProductName dari tabel products, Quantity dari tabel orderdetails yang ProductNamanya adalah “Chang”.

OrderDetailID	ProductName	Quantity
21	Chang	20
30	Chang	50
48	Chang	35

f. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)

2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdfKumpulkan

pada <http://iti.polije.ac.id/elearning/>

g. Kesimpulan

Mahasiswa dapat memahami dan mempraktikkan penggunaan Indeks dan Views pada database

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 35

Pokok Bahasan	: Studi Kasus Index dan View
Acara Praktikum/Pertemuan	: Minggu 13/2
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu mengimplementasikan Index pada studi kasus
2. Mahasiswa mampu mengimplementasikan Views pada studi kasus

b. Indikator

Kemampuan mahasiswa dalam mengimplementasikan Indeks dan Views pada studi kasus

c. Dasar Teori

-

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

1. Buatlah 3 (tiga) Indeks berdasarkan studi kasus masing-masing
2. Buatlah View satu tabel berdasarkan studi kasus masing-masing
3. Buatlah View dengan Join 2 tabel berdasarkan studi kasus masing-masing
4. Buatlah View dengan join 3 tabel atau lebih berdasarkan studi kasus masing-masing
5. Dokumentasikan setiap Indeks dan View yang sudah dibuat

f. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)
2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf

Kumpulkan pada <http://jti.polije.ac.id/elearning/>

g. Kesimpulan

Mahasiswa berhasil menampilkan select tabel sesuai ketentuan.

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 36

Pokok Bahasan	: Trigger
Acara Praktikum/Pertemuan	: Minggu 14 / 1 dan 2
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami penggunaan SQL *Trigger*.
2. Mahasiswa mampu mengimplementasikan penggunaan SQL *Trigger* pada database.

b. Indikator

1. Mahasiswa memahami penggunaan SQL *Trigger*.
2. Mahasiswa mengimplementasikan penggunaan SQL *Trigger* pada database.

c. Dasar Teori

Trigger didefinisikan sebagai himpunan kode (procedural) yang dieksekusi secara otomatis sebagai respon atas suatu kejadian berkaitan dengan tabel basis data. Kejadian (event) yang dapat membangkitkan trigger umumnya berupa pernyataan INSERT, UPDATE, dan DELETE.

Berdasarkan ruang lingkupnya, trigger diklasifikasikan menjadi dua jenis: row trigger dan statement trigger. Trigger baris (row) mendefinisikan aksi untuk setiap baris tabel; trigger pernyataan hanya berlaku untuk setiap pernyataan INSERT, UPADATE, atau DELETE. Dari sisi perilaku (behavior) eksekusi, trigger dapat dibedakan menjadi beberapa jenis; namun umumnya ada dua jenis: trigger BEFORE dan AFTER.

Sesuai penamaannya, jenis-jenis ini merepresentasikan waktu eksekusi trigger, misalnya sebelum ataukah sesudah pernyataan-pernyataan yang berkorespondensi. Trigger juga dipandang sebagai bentuk spesifik dari stored procedure (terkait pendefinisan body). Bagaimanapun, trigger akan dipanggil (secara otomatis) ketika event terjadi, sedangkan stored procedure harus dipanggil secara eksplisit.

Berikut cara pembuatan Trigger :

```

DELIMITER $$

CREATE TRIGGER nama_trigger
{BEFORE | AFTER} {INSERT | UPDATE| DELETE }
    ON nama_table
    FOR EACH ROW
BEGIN
    KODE SQL
END$$
DELIMITER ;

```

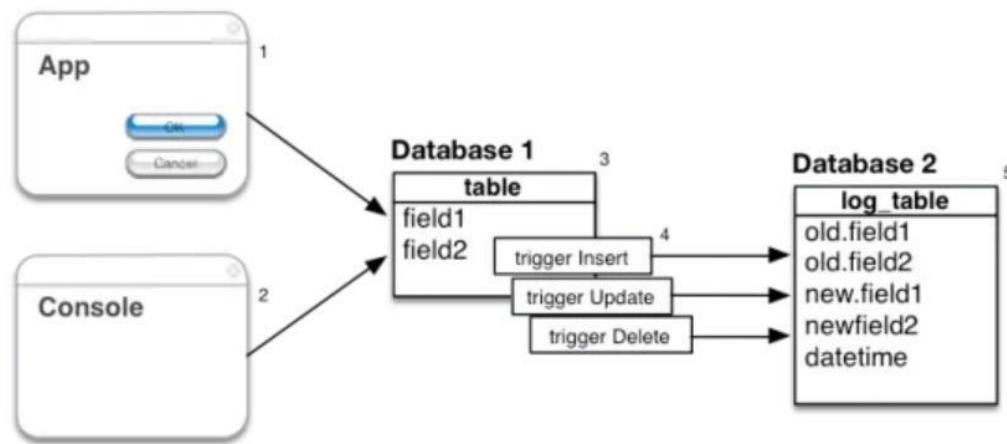
- Untuk memulai menggunakan TRIGGER, gunakan **CREATE TRIGGER** dilanjutkan **nama TRIGGER** yang ingin dibuat.
- {BEFORE | AFTER}** adalah waktu TRIGGER akan dijalankan, apakah sebelum atau sesudah database dimodifikasi oleh perintah DML.
- {INSERT | UPDATE | DELETE}** adalah perintah DML yang mengaktifkan TRIGGER
Lebih detail untuk waktu TRIGGER dijelaskan sebagai berikut :

Waktu TRIGGER	Keterangan TRIGGER
BEFORE INSERT	TRIGGER dijalankan sebelum record dimasukkan ke database
AFTER INSERT	TRIGGER dijalankan sesudah record dimasukkan ke database
BEFORE UPDATE	TRIGGER dijalankan sebelum record dirubah di database
AFTER UPDATE	TRIGGER dijalankan sesudah record dirubah database
BEFORE DELETE	TRIGGER dijalankan sebelum record dihapus di database
AFTER DELETE	TRIGGER dijalankan sesudah record dihapus di database

- ON** mendefinisikan table yang mengaktifkan TRIGGER
- BEGIN END** adalah pernyataan yang membungkus kode TRIGGER
Pastikan diawal gunakan **DELIMITER \$\$** dan diakhir dikembalikan ke **DELIMITER ;** seperti dalam membuat Stored Procedure

Referensi “OLD” dan “NEW”

Karena trigger digunakan pada saat terjadi perubahan row data, maka kita perlu referensi ke row sebelum dan sesudah perubahan. Untuk ini ada dua alias yang berfungsi untuk hal tersebut yaitu OLD dan NEW. Sesuai namanya, OLD digunakan untuk referensi sebelum perubahan dan NEW untuk referensi setelah perubahan.



Biasanya dalam relasional database, trigger dapat ditemui ketika melakukan perintah eksekusi table. Hal itu menjadi lebih bermanfaat trigger menjadikan penulisan pemrograman yang sederhana dan dapat menjaga informasi agar tetap konsisten dalam database. Selain itu, terdapat beberapa fungsi trigger yang akan dijelaskan dibawah ini.

1. Integritas Data

Dengan adanya trigger, dapat mempertahankan integritas table yang terdapat dalam database. Sebab, dapat melakukan operasi-operasi yang berkaitan dengan pengolahan database, seperti INSERT, UPDATE dan DELETE

2. Mencegah Error

Trigger dalam MySQL dapat mencegah terjadinya error dalam pengoperasian data. Jika terjadi error dalam pendefinisian trigger, error tersebut tidak mengganggu trigger yang sedang berjalan.

3. Membuat Tugas Kerja menjadi Terjadwal

Ketika trigger telah dijalankan, maka dapat menggunakan berbagai bahasa pemrograman tanpa harus bingung bagaimana cara mengaksesnya.

4. Mencegah proses transaksi yang tidak sah

Dalam praktiknya, biasa trigger digunakan untuk melakukan proses transaksi. Dapat menyimpan record transaksi tersebut ke table lain (history) tanpa harus takut jika data tersebut di-update atau delete. Semua perubahan yang terjadi juga dapat diacak berdasarkan waktu pembuatannya.

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

1. Buatlah satu tabel audit dengan nama “DeleteOrders” yang berisi rows yang dihapus dari tabel Orders dengan tambahan dua filed diantaranya adalah date_delete dan username dengan syntax sebagai berikut ;

```
CREATE TABLE DeleteOrders LIKE orders;

ALTER TABLE OrdersDelete ADD (
    `update_delete` DATETIME,
    `username` VARCHAR (25)
);
```

2. Buat trigger yang akan melakukan populasi data yang dihapus dari tabel orders ke DeleteOrders dengan menggunakan syntax berikut :

```
DELIMITER |
CREATE TRIGGER DeleteOrders AFTER DELETE
    ON orders FOR EACH ROW
BEGIN
    INSERT INTO DeleteOrders(
        OrderId,
        CustomerID,
        EmployeeID,
        OrderDate,
        ShipperID,
        update_delete,
        username
    )
    VALUES (
        OLD.OrderID,
        OLD.CustomerID,
        OLD.EmployeeID,
        OLD.orderdate,
        OLD.shipperID,
        SYSDATE(),
        CURRENT_USER
    );
END;
```

3. Setelah membuat trigger, selanjutnya kita akan membuat pengujian dengan cara : hapus 3 (tiga) rows data dari tabel Orders dan amati apa yang terjadi pada tabel DeleteOrders dengan menjalankan syntax berikut :

```
DELETE FROM orders LIMIT 3;
```

```
SELECT * FROM deleteOrders;
```

4. Catatan: Jika data tidak berhasil di delete, maka rubah relasi tabel yang semula **RESTRICT** menjadi **CASCADE** pada constraint properties di bagian **ON DELETE**. Lihat perubahan data setelah di delete sebagai berikut :

The screenshot shows a MySQL database interface. At the top, a green bar indicates "3 rows affected. (Query took 0.0176 seconds.)". Below it, the SQL command `DELETE FROM orders LIMIT 3;` is shown. Underneath, there are two sections: one for `orders` and one for `deleteOrders`. The `orders` section shows "Showing rows 0 - 2 (3 total, Query took 0.0003 seconds.)" and the `deleteOrders` section shows the same message. Both sections have a "SELECT * FROM deleteOrders;" command below them. At the bottom, there is a table view of the `deleteOrders` data, which contains three rows of information.

	OrderID	CustomerID	EmployeeID	OrderDate	ShipperID	update_delete	username
<input type="checkbox"/>	10248	90	5	1996-07-04	3	2022-04-06 15:09:06	root@localhost
<input type="checkbox"/>	10249	81	6	1996-07-05	1	2022-04-06 15:09:06	root@localhost
<input type="checkbox"/>	10250	34	4	1996-07-08	2	2022-04-06 15:09:06	root@localhost

5. Silahkan dibuat laporannya beserta penjelasan.

g. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)
2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdfKumpulkan

pada <http://iti.polije.ac.id/elearning/>

g. Kesimpulan

Mahasiswa mampu memahami dan mengimplementasikan penggunaan TRIGGER

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 37

Pokok Bahasan	: Trigger ON Insert and Update
Acara Praktikum/Pertemuan	: Minggu 14/3
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami Trigger pada saat insert dan update
2. Mahasiswa mampu membuat Trigger pada saat insert dan update

b. Indikator

Kemampuan mahasiswa dalam memahami dan mengimplementasikan trigger pada saat insert dan update

c. Dasar Teori

-

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

1. Buatlah tiga tabel diantaranya :

Tabel Pertama :

```
CREATE TABLE beli(
    `id_beli` INT(11) NOT NULL AUTO_INCREMENT,
    `kd_barang` VARCHAR(5) DEFAULT NULL,
    `nama_barang` VARCHAR(20) DEFAULT NULL,
    `jumlah` INT(11) DEFAULT NULL,
    PRIMARY KEY (`id_beli`)
);
```

Tabel Kedua :

```
CREATE TABLE jual(
    `id_jual` INT(11) NOT NULL AUTO_INCREMENT,
    `kd_pelanggan` VARCHAR(10) NOT NULL,
    `kd_barang` VARCHAR(20) DEFAULT NULL,
    `nama_barang` VARCHAR(20) DEFAULT NULL,
    PRIMARY KEY(`id_jual`)
);
```

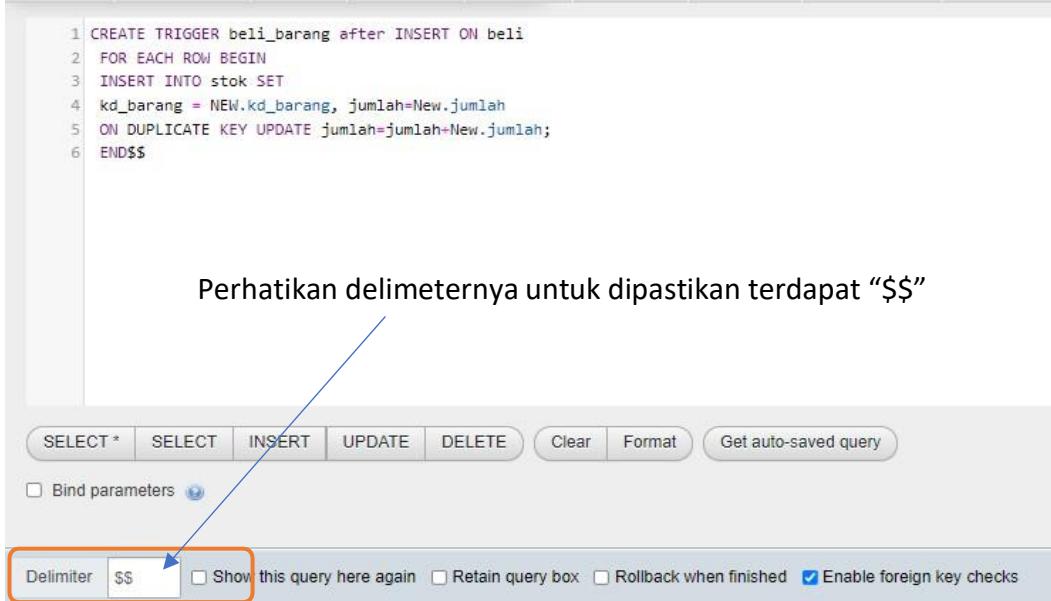
Tabel Ketiga :

```
CREATE TABLE stok(
    `kd_barang` VARCHAR(5) NOT NULL,
    `jumlah` INT(11) NOT NULL,
    PRIMARY KEY (`kd_barang`)
);
```

2. Terdapat Skenario sebagai berikut :

- Jika terdapat transaksi pada tabel **beli**, maka akan mempengaruhi transaksi pada tabel **stok**. Ex : Kita **membeli** barang ke distributor, maka stok barang akan **bertambah**.
- Jika terdapat transaksi **penjualan** pada tabel **jual**, maka stok barang pada tabel **stok** akan **berkurang**

Berikut adalah syntaxnya :



```
1 CREATE TRIGGER beli_barang after INSERT ON beli
2 FOR EACH ROW BEGIN
3     INSERT INTO stok SET
4     kd_barang = NEW.kd_barang, jumlah=New.jumlah
5     ON DUPLICATE KEY UPDATE jumlah=jumlah+New.jumlah;
6 END$$
```

Perhatikan delimeternya untuk dipastikan terdapat “\$\$”

Below the code editor:

- Buttons: SELECT *, SELECT, INSERT, UPDATE, DELETE, Clear, Format, Get auto-saved query
- Checkboxes: Bind parameters, Show this query here again, Retain query box, Rollback when finished, Enable foreign key checks
- Text input field: Delimiter with value \$\$, with a blue arrow pointing to it.

Keterangan : Trigger diberi nama **beli_barang**, dan trigger akan bekerja setelah memasukkan data kedalam table **beli**. Dan secara otomatis akan menambahkan stok barang

kedalam table stok. Karena data pada table beli masih nol, jika kita memasukkan data ke table beli, maka tidak akan ada penjumlahan yang terjadi.

3. Sekarang cobalah masukkan sebuah data ke dalam table beli. Dan lakukan uji coba insert pada tabel tersebut kemudian lihat pada tabel stok dan amati apa yang terjadi :

Contoh lakukan penambahan record data pada tabel beli :

id_beli	kd_barang	nama_barang	jumlah
1	B001	GUCCI	10
2	B002	NIKE	10

Cek pada tabel stok, jika sukses maka record data

kd_barang	jumlah
B001	10
B002	10

4. Lakukan 10 transaksi untuk pembelian dan 5 transaksi untuk penjualan. Screen shot hasilnya dan berikan penjelasan
5. Dokumentasikan setiap implementasi percobaan diatas

g. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)
2. Dokumentasi tugas berupa file pdf.

i. Kesimpulan

Mahasiswa berhasil menampilkan select tabel sesuai ketentuan.

j. Rubrik Penilaian

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdf

Kumpulkan pada <http://iti.polje.ac.id/elearning/>

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 38

Pokok Bahasan	: Studi Kasus Trigger
Acara Praktikum/Pertemuan	: Minggu 15/1 dan 2
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu membuat trigger dari studi kasus masin – masing kelompok

b. Indikator

Kemampuan mahasiswa dalam membuat trigger dari studi kasus masing-masing kelompok

c. Dasar Teori

Minggu 14 Acara 38 dan Acara 39

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

1. Buatlah dua tabel untuk audit yang record datanya berasal dari salah satu tabel pada masing-masing studi kasus kalian.
Lihat pada implementasi trigger pada minggu ke-14 pertemuan ke-1.
2. Buatlah dua Trigger yang akan melakukan populasi data yang dipus dari tabel yang sudah dibuat pada prosedur ke-1.
3. Buatlah pengujinya dengan cara hapus masing-masing tabael sebanyak 5 rows data dari tabel kalian dan tampilkan hasilnya

f. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)
2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdfKumpulkan

pada <http://jti.polije.ac.id/elearning/>

g. Kesimpulan

Mahasiswa mampu membuat trigger di studi kasus masing-masing kelompok

h. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			

BKPM WORKSHOP MANAJEMEN BASIS DATA: ACARA 39

Pokok Bahasan	: Studi Kasus Trigger
Acara Praktikum/Pertemuan	: Minggu 15/3 dan 4
Tempat	: Daring/Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu membuat trigger dari studi kasus masing – masing kelompok

b. Indikator

Kemampuan mahasiswa dalam membuat trigger dari studi kasus masing-masing kelompok

c. Dasar Teori

Minggu 14 Acara 38 dan Acara 39

d. Alat dan Bahan

1. Client Server: Xampp
2. Text Editor: PhpMyAdmin
3. Kertas A4 / Folio
4. Pulpen

e. Prosedur Kerja

1. Lakukan implementasi trigger pada saat Insert di studi kasus kalian masing-masing
2. Lakukan implementasi trigger pada saat Update di studi kasus kalian masing-masing

g. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (Tugas)

2. Dokumentasi tugas berupa file pdf.

Format Laporan

- Kertas A4
- Format *.pdf
- Struktur Laporan: Cover, Pendahuluan, Hasil Praktik, Kesimpulan dan Daftar Pustaka
- Berikan Identitas Diri: NIM, Nama, Golongan, Tugas Minggu Ke-X
- Penamaan File: ACARA-X_GOL_NIM_NAMA.pdf
**Contoh: **ACARA-1_A_E1234356_BI.pdfKumpulkan

pada <http://jti.polije.ac.id/elearning/>

i. Kesimpulan

Mahasiswa mampu membuat trigger di studi kasus masing-masing kelompok

j. Rubrik Penilaian

No	Indikator	Nilai			
		1	2	3	4
1	Ketepatan waktu dan ketepatan Langkah kerja dalam mengerjakan				V
2	Ketepatan waktu dan Langkah kerja kurang tepat dalam mengerjakan			V	
3	Keterlambatan waktu dan Langkah kerja yang dilakukan kurang tepat		V		
4	Keterlambatan waktu pengumpulan dan Langkah kerja tidak selesai	V			
	Total Skor*	Skor x 25			