

PENGELOLAAN BASIS DATA

03. Database Backup & Recovery

10. Trigger

Doni Abdul Fatah
Universitas Trunojoyo Madura

Pokok Bahasan (Revisi)

01. Database Design Review

02. Database User Account Management

03. Database Backup & Recovery

04. Database Replication

05. Database Optimization

06. SQL Query Select

07. SQL Query Where / Groups

08. SQL Query Join

09. Stored Procedure / Function

10. Trigger

11. SQL DDL

12. Optimising DB Structure

13. Project

14. UAS

Penjelasan Pokok Bahasan (Revisi)

Database Design Review

- Relational Model
- ERD
- Conceptual Database Design
- Logical Database Design
- Physical Database Design

Database User Account Management

- Grant
- Revoke
- Privelege
- Adding User
- Limiting User Resource
- Tugas Besar 1

Database Backup & Recovery

- Backup
- Restore
- Check
- Repair
- Table Maintenance

Database Replication

- Overview
- Setup Replication
- Sql command (slave)

Penjelasan Pokok Bahasan (Revisi)

Database Optimization	SQL Query Select	SQL Query Where / Groups	SQL Query Join
<ul style="list-style-type: none">• Definisi• Design limitation / tradeoff• Benchmarking• Explain• Query performance	<ul style="list-style-type: none">• Select, where• Insert, update, delete• Join• Union• Truncate• replace	<ul style="list-style-type: none">• Select, where• Insert, update, delete• Join• Union• Truncate• replace	<ul style="list-style-type: none">• Select, where• Insert, update, delete• Join• Union• Truncate• replace

Penjelasan Pokok Bahasan (Revisi)

Stored Procedure / Function	Trigger	SQL DDL	Optimising DB Structure
<ul style="list-style-type: none">• Definisi• Parameter• Body• Local variable• Set• Flow control• Call stored procedure• Error message	<ul style="list-style-type: none">• Konsep Trigger• Create trigger• Trigger as Integrity constraints• Trigger and catalog	<ul style="list-style-type: none">• Definisi• Creating tables• Data types• Temporary table• Copying tables• Naming tables and columns• Table options• Integrity constraints• Primary keys• Alternate keys• Foreign keys	<ul style="list-style-type: none">• Konsep Design choices• Indexes• Multiple column indexes• Tuning server parameter• How mySQL uses memory

01. Pengelolaan BasisData

- 1) Trigger
- 2) Database Backup & Recovery
- 3) Projek Akhir
- 4) Referensi

Trigger

Trigger

- Kumpulan kode yang terdiri dari procedural dan declarative statements
- Tersimpan dalam katalog dan
- Diaktifkan oleh server database jika operasi yang spesifik dieksekusi dalam database dan jika ada kondisi yang ditentukan

Trigger

- BEFORE INSERT – activated before data is inserted into the table.
- AFTER INSERT – activated after data is inserted into the table.
- BEFORE UPDATE – activated before data in the table is updated.
- AFTER UPDATE – activated after data in the table is updated.
- BEFORE DELETE – activated before data is removed from the table.
- AFTER DELETE – activated after data is removed from the table.

Mengakses Nilai Baru dan Lama

- Dalam trigger untuk mengakses data lama dan data baru, data lama dapat direference dengan record OLD dan data baru dapat di reference dengan Record NEW.

OPERASI	NEW (READ/WRITE)	OLD (READ)
INSERT	✓	
UPDATE	✓	✓
DELETE		✓

- Untuk mengacu ke sebuah field dapat ditulis dengan NEW.namafiled atau OLD.namafiled.

Trigger (Contd-2)

```
<create trigger statement> ::=  
CREATE TRIGGER <trigger name>  
<trigger moment>  
<trigger event>  
[ <trigger condition> ]  
<trigger action>  
  
<trigger moment> ::=  
BEFORE | AFTER | INSTEAD OF  
  
<trigger event> ::=  
{ INSERT | DELETE | UPDATE [ OF <column list> ] }  
{ ON | OF | FROM | INTO } <table specification>  
[ REFERENCING { OLD | NEW | OLD_TABLE | NEW_TABLE }  
AS <variable> ] FOR EACH { ROW | STATEMENT }  
  
<trigger condition> ::= ( WHEN <condition> )  
<trigger action> ::= <begin-end BLOCK>
```

Contoh Trigger

Contoh 1: Buatlah tabel CHANGES!

```
CREATE TABLE CHANGES
  (USER CHAR(30) NOT NULL,
   CHA_TIME TIMESTAMP NOT NULL,
   CHA_PLAYERNO SMALLINT NOT NULL,
   CHA_TYPE CHAR(1) NOT NULL,
   CHA_PLAYERNO_NEW INTEGER,
   PRIMARY KEY (USER, CHA_TIME,
   CHA_PLAYERNO, CHA_TYPE));
```

Contoh Trigger Lanjutan

Contoh 2: Buatlah trigger yang akan meng-update tabel CHANGES secara otomatis setiap kali ada penambahan baris baru di dalam tabel PLAYERS!

```
DELIMITER $$
```

```
CREATE TRIGGER INSERT_PLAYERS
AFTER INSERT ON PLAYERS FOR EACH ROW
BEGIN
INSERT INTO CHANGES
(USER, CHA_TIME, CHA_PLAYERNO, CHA_TYPE, CHA_PLAYERNO_NEW)
VALUES (USER, CURDATE(), NEW.PLAYERNO, "I", NULL);

END$$
DELIMITER ;
```

Contoh Trigger Lanjutan

Contoh 4: Buatlah trigger yang meng-update tabel CHANGES setiap kali ada baris di dalam tabel PLAYERS yang dihapus!

```
DELIMITER $$

CREATE TRIGGER DELETE_PLAYER
AFTER DELETE ON PLAYERS FOR EACH ROW
BEGIN
    CALL INSERT_CHANGE (OLD.PLAYERNO, 'D', NULL);

END$$
DELIMITER ;
```

Contoh Trigger Lanjutan

Contoh 5: Buatlah trigger yang meng-update tabel CHANGES setiap kali ada baris di dalam tabel PLAYERS yang di-update!

```
DELIMITER $$
```

```
CREATE TRIGGER UPDATE_PLAYER
AFTER UPDATE ON PLAYERS FOR EACH ROW
BEGIN
CALL INSERT_CHANGES (NEW.PLAYERNO, 'U', OLD.PLAYERNO);

END$$
DELIMITER ;
```

Contoh Trigger Lanjutan

Contoh 6 : Buat trigger untuk menyimpan history division, jika division berubah, maka division lama harus disimpan ke tabel history division.

```
DELIMITER $$  
DROP TRIGGER IF EXISTS  
coba_update_teams$$  
CREATE TRIGGER coba_update_teams  
AFTER UPDATE ON teams FOR EACH ROW  
BEGIN  
INSERT INTO history_div_teams VALUES  
(NOW(), old.teamno, old.division,  
USER());  
END$$  
DELIMITER ;
```

Contoh Penggunaan Trigger

```
UPDATE teams SET division =  
"thrid" WHERE teamno=1;
```

```
SELECT * FROM teams;  
SELECT * FROM history_div_teams;
```

teams (3x5)		history_div_teams (5x8)		
waktu	teamno	playerno	division	oleh
2019-04-11 20:04:49	1	(NULL)	first	root@localhost
2019-04-11 20:05:09	1	(NULL)	first	root@localhost
2019-04-11 20:05:24	1	(NULL)	second	root@localhost
2019-04-11 20:26:20	1	(NULL)	thrid	root@localhost
2019-04-11 20:26:26	1	(NULL)	coba	root@localhost
2019-04-11 20:26:30	1	(NULL)	lagi	root@localhost
2019-04-11 20:31:14	1	(NULL)	sekali	root@localhost
2019-04-11 20:33:12	1	(NULL)	firste	root@localhost

Untuk melihat semua alamat yang pernah digunakan oleh teams yang bernomor =1

```
(SELECT NOW() waktu, teamno, division  
FROM teams WHERE teamno=1)  
UNION  
(SELECT waktu, teamno, division FROM  
history_div_teams WHERE teamno=1)  
ORDER BY waktu DESC;
```

2019-04-11 20:26:20	1	thrid
2019-04-11 20:05:24	1	second
2019-04-11 20:05:09	1	first
2019-04-11 20:04:49	1	first

Contoh Trigger Lanjutan

Trigger yang pertama mempunyai kekurangan yaitu ketika ada perubahan di tabel teams walaupun tdk mengubah kolom division, maka statement INSERT di tabel history akan dijalankan.

```
DELIMITER $$  
DROP TRIGGER if EXISTS  
coba_update_teams$$  
CREATE TRIGGER coba_update_teams  
AFTER UPDATE ON teams FOR EACH ROW  
BEGIN  
if old.division <> new.division  
then  
INSERT INTO history_div_teams  
VALUES (NOW(), old.teamno,  
old.division, USER());  
END if;  
END$$  
DELIMITER ;
```

Contoh Penggunaan Trigger

```
UPDATE teams SET division =  
"sekali" WHERE teamno=1;  
  
UPDATE teams SET division =  
"firste" WHERE teamno=1;  
  
UPDATE teams SET division =  
"first" WHERE teamno=1;  
  
SELECT * FROM teams;  
SELECT * from history_div_teams;
```

Untuk melihat semua alamat yang pernah digunakan oleh teams yang bernomor =1

```
(SELECT NOW() waktu, teamno,  
division FROM teams WHERE  
teamno=1)  
UNION  
(SELECT waktu, teamno, division  
FROM history_div_teams WHERE  
teamno=1)  
ORDER BY waktu DESC;
```

Contoh Trigger Lanjutan

Contoh 7 : Buat trigger akan dieksekusi ketika ada perubahan teamno di tabel teams yang akan melakukan update ke tabel history_div_teams untuk menyesuaikan Teamno agar relasi tidak terlepas.

```
DELIMITER $$  
DROP TRIGGER if EXISTS coba_update_teams$$  
CREATE TRIGGER coba_update_teams  
AFTER UPDATE ON teams FOR EACH ROW  
BEGIN  
if old.division <> new.division then  
INSERT INTO history_div_teams VALUES (NOW(),  
old.teamno, old.division, USER());  
END if;  
if old.teamno <> new.teamno then  
UPDATE history_div_teams SET  
teamno=new.teamno WHERE teamno=old.teamno;  
END if;  
END$$  
DELIMITER ;
```

Contoh Penggunaan Trigger

```
ALTER table history_div_teams  
ADD COLUMN playerno  
SMALLINT(6) AFTER teamno;
```

```
SELECT * from history_div_teams;
```

Contoh Trigger Lanjutan

Contoh 7 : Buat trigger akan dieksekusi ketika ada perubahan teamno di tabel teams yang akan melakukan update ke tabel history_div_teams untuk menyesuaikan Teamno agar relasi tidak terlepas.

```
DELIMITER $$  
DROP TRIGGER if EXISTS coba_update_teams$$  
CREATE TRIGGER coba_update_teams  
AFTER UPDATE ON teams FOR EACH ROW  
BEGIN  
if old.division <> new.division then  
INSERT INTO history_div_teams VALUES (NOW(),  
old.teamno, old.division, USER());  
END if;  
if old.playerno <> new.playerno then  
UPDATE history_div_teams SET  
playerno=new.playerno WHERE  
playerno=old.playerno;  
END if;  
END$$  
DELIMITER ;
```

Contoh Penggunaan Trigger

```
UPDATE teams SET teamno  
= 111 WHERE teamno=1;
```

```
UPDATE teams SET  
playerno =111 WHERE  
playerno=6;
```

03. Database Backup & Recovery

Backup

Restore

Check

Repair

Backup & Restore

- ❑ **Backup** → Menyalin data dalam database ke file eksternal (isi data dalam bentuk query sql)
- ❑ **Restore** → Menyalin data dari file eksternal (dengan mengeksekusi query sql) ke dalam database.
- ❑ **Alasannya :**
 - Data sewaktu-waktu bisa hilang atau rusak
- ❑ **Sehingga :**
 - Perlu media untuk mem-backup data -> Jaga-jaga
 - Perlu media untuk me-restore data -> Memulihkan data

Penyebab Kesalahan Database:

- Kesalahan User
- Kerusakan Statemen
- Kesalahan Proses
- Kesalahan Jaringan
- Kerusakan Database
- Kerusakan Media (Disk)
- Bencana Alam

Tools

❑ Mysqldump atau mysqlhotcopy

- Sebuah program atau aplikasi internal yang ikut disertakan ketika kita menginstall MySQL.
- Anda akan menemukan file **mysqldump.exe** di dalam folder **bin** pada folder instalasi MySQL.

Local Disk (C:) > xampp > mysql > bin			
Name	Date modified	Type	Size
mysql_upgrade	04/11/2016 19.01	Application	3.297 KB
mysqldump	04/11/2016 19.01	Application	3.680 KB
mysqlimport	04/11/2016 19.01	Application	3.620 KB
mysqlshow	04/11/2016 19.01	Application	3.622 KB
mysql	04/11/2016 19.01	Application	3.683 KB
mysqlcheck	04/11/2016 19.01	Application	3.628 KB
mysqltest	04/11/2016 19.01	Application	4.019 KB
myisamlog	04/11/2016 19.01	Application	3.400 KB
myisampack	04/11/2016 19.01	Application	3.423 KB
aria_dump_log	04/11/2016 19.01	Application	3.362 KB
aria_pack	04/11/2016 19.01	Application	3.596 KB
myisam_ftdump	04/11/2016 19.01	Application	3.384 KB
myisamchk	04/11/2016 19.01	Application	3.519 KB
...

Backup & Restore

Sintaknya Backup :

```
mysqldump -u [userdb] -p[pass] [dbname] > [backupfile.sql]  
mysqlhotcopy db_name /path/to/some/dir
```

Sintaknya Restore :

```
mysql -u [uname] -p[pass] [db_to_restore] < [backupfile.sql]
```

□ Keterangan :

- **[userdb]** Nama user database Anda
- **[pass]** Password database Anda
- **[dbname]** Nama database Anda
- **[backupfile.sql]** Nama file untuk database yang Anda backup nantinya
- **[db_to_restore]** Nama database yang sudah Anda buat sebelum melakukan restore

Backup

Contoh 1 : Lakukan backup database beserta datanya pada database President atau tennis kemudian simpan file dengan nama “**backup_namadatabase.sql**”.

Sintaknya :

```
mysqldump -u root -p president > backup_president.sql
```

Contoh 2: Lakukan backup database terhadap struktur tabelnya tanpa data di dalamnya pada database President atau tennis kemudian simpan **namabackup.sql**

Sintaknya :

```
mysqldump -u root -p --no-data president >  
backup_president_nodata.sql
```

Backup

Contoh 3 : Lakukan backup data atau isi database pada database President atau tennis kemudian simpan **namabackup.sql**

Sintaknya :

```
mysqldump -u root -p --no-create-info president >  
backup_president_data.sql
```

Contoh 4: Lakukan backup trigger dan stored-procedure (function) database pada database President atau tennis kemudian simpan **namabackup.sql**

Sintaknya :

```
mysqldump -u root -p --routines --no-create-info --no-data -  
--no-create-db --skip-opt president >  
backup_president_trigger-procedure.sql;
```

Contoh 5: Lakukan backup pada semua database

Sintaknya :

```
mysqldump --all-databases > backup.sql
```

Backup

Contoh 6 : Lakukan backup tabel state pada database President simpan **NamaTabelBackup.sql**

Sintaknya :

```
mysqldump -u root -p --databases president --tables  
state > backup_tabelstate_president.sql
```

Contoh 7: Lakukan backup database President dan test simpan **Backup2database.sql**

Sintaknya :

```
mysqldump -u root -p --databases president test >  
backup_test_president.sql
```

Restore

Contoh 8 : Lakukan Restore data dari backup database pada database President atau tennis yang sudah dibuat sebelumnya.

Langkah Awal : Menyiapkan sebuah database untuk menampung tabel yang akan diinput. karena file hasil **mysqldump** memang tidak menyertakan pembuatan database.

Sintaknya :

```
CREATE DATABASE president_baru;
```

```
SHOW TABLES;
```

1	SHOW TABLES;
 TABLE NAMES (1x0)	
	Tables_in_president_baru

Restore

Langkah selanjutnya : Kita melakukan backup database “**backup_president.sql**” ke dalam database “**president_baru**”.

Sintaknya :

```
mysql -u root -p president_baru < backup_president.sql
```

```
SHOW TABLES;
```

1 SHOW TABLES;
TABLE_NAMES (1x7)

Tables_in_president_baru
admin_pr_vp
administration
election
pres_hobby
pres_marriage
president
state

```
SELECT * FROM state;
```

1 SELECT * FROM state;
state (3x50)

STATE_NAME	ADMIN_ENTERED	YEAR_ENTERED
Alabama	8	1.819
Alaska	43	1.959
Arizona	31	1.912
Arkansas	12	1.836
California	16	1.850
Colorado	22	1.876
Connecticut	(NULL)	1.776
Delaware	(NULL)	1.776
Florida	14	1.845
Georgia	(NULL)	1.776
Hawaii	43	1.959
Idaho	26	1.890
Illinois	8	1.818
Indiana	7	1.816
Iowa	15	1.846

Check & Repair

- ❑ Mengecek, memperbaiki, mengoptimalkan dan menganalisis tabel-tabel mysql.
- ❑ Pada proses pemeriksa ada 2 kondisi :
 - **OK** : Tidak terdapat masalah.
 - **warning atau error** : Jika terdapat masalah
- ❑ **Tools :**
 - **Mysqlcheck** merupakan paket mysql-client, digunakan untuk mengecek, memperbaiki, mengoptimalkan dan menganalisis tabel-tabel mysql.
 - Atau **dengan sintaks :**
 - Untuk pengecekan : **check table [nama tabel]** ;
 - Untuk perbaikan / *Repair Table* :
 - **repair table [nama_table];** → 1 tabel
 - **repair table [nama_table1] [nama_table2];** → lebih dari 1 tabel.

Check & Repair

- Contoh 9 : bagaimana caranya mengetahui kondisi dari suatu database tertentu.

```
mysqlcheck --all-databases -u root -p
```

```
C:\xampp\mysql\bin>mysqlcheck --all-databases -u root -p
Enter password:
db_fakultas.contoh_at          OK
db_fakultas.contoh_bin          OK
db_fakultas.contoh_cha          OK
db_fakultas.contoh_dec          OK
db_fakultas.contoh_float        OK
db_fakultas.contoh_int          OK
db_fakultas.contoh_text         OK
db_fakultas.test                OK
db_fakultas.tennis               OK
db_fakultas.tennis.changes       OK
db_fakultas.tennis.changes2      OK
db_fakultas.tennis.committee_members OK
db_fakultas.tennis.history_div_teams OK
db_fakultas.tennis.hitung         OK
db_fakultas.tennis.matches        OK
db_fakultas.tennis.penalties      OK
db_fakultas.tennis.players         OK
db_fakultas.tennis.players_x       OK
db_fakultas.tennis.players_z       OK
db_fakultas.tennis.recr_players    OK
db_fakultas.tennis.teams          OK
db_fakultas.test.contoh_blob       OK
db_fakultas.test.contoh_cha        OK
db_fakultas.test.contoh_date       OK
db_fakultas.test.contoh_text        OK
db_fakultas.test.hobi              OK
db_fakultas.test.jurusan           OK
```

Check & Repair

- **Contoh 10 :** bagaimana caranya mengetahui kondisi dari suatu tabel tertentu dalam suatu database.

```
mysqlcheck --database tennis -u root -p
```

```
C:\xampp\mysql\bin>mysqlcheck --databases president -u root
president.admin_pr_vp          OK
president.administration        OK
president.election              OK
president.pres_hobby             OK
president.pres_marriage         OK
president.president              OK
president.state                  OK
```

```
C:\xampp\mysql\bin>mysqlcheck --databases test -u root
test.contoh_blob                OK
test.contoh_cha                 OK
test.contoh_date                OK
test.contoh_text                OK
test.hobi                       OK
test.jurusan                     OK
```

```
C:\xampp\mysql\bin>mysqlcheck --databases tennis -u root
tennis.changes                  OK
tennis.changes2                 OK
tennis.committee_members         OK
tennis.history_div_teams        OK
tennis.hitung                   OK
tennis.matches                  OK
tennis.penalties                OK
tennis.players                  OK
tennis.players_x                 OK
tennis.players_z                 OK
tennis.recr_players              OK
tennis.teams                     OK
```

Check & Repair

- **Contoh 11 :** bagaimana caranya mengetahui kondisi dari suatu tabel tertentu dari 2 atau lebih database.

```
mysqlcheck -u root -p --databases tennis test
```

```
C:\xampp\mysql\bin>mysqlcheck -u root -p --databases tennis test
Enter password:
tennis.changes          OK
tennis.changes2         OK
tennis.committee_members OK
tennis.history_div_teams OK
tennis.hitung           OK
tennis.matches          OK
tennis.penalties        OK
tennis.players          OK
tennis.players_x        OK
tennis.players_z        OK
tennis.recr_players     OK
tennis.teams            OK
test.contoh_blob         OK
test.contoh_cha          OK
test.contoh_date         OK
test.contoh_text         OK
test.hobi                OK
test.jurusan             OK
```

Check & Repair

- Contoh 12 : bagaimana caranya mengetahui kondisi dari suatu tabel state pada databases president.

```
Show TABLES;
```

```
CHECK TABLES state;
```

```
MariaDB [president]> show tables;
+-----+
| Tables_in_president |
+-----+
| admin_pr_vp          |
| administration        |
| election              |
| pres_hobby            |
| pres_marriage         |
| president             |
| state                 |
+-----+
7 rows in set (0.00 sec)
```

```
MariaDB [president]> check tables state;
+-----+-----+-----+
| Table      | Op    | Msg_type | Msg_text |
+-----+-----+-----+
| president.state | check | status   | OK
+-----+-----+-----+
1 row in set (0.07 sec)
```

```
MariaDB [president]> check tables election;
```

```
+-----+-----+-----+
| Table      | Op    | Msg_type | Msg_text |
+-----+-----+-----+
| president.election | check | status   | OK
+-----+-----+-----+
1 row in set (0.08 sec)
```

```
CHECK TABLE jurusan, hobi;
```

```
MariaDB [test]> check table jurusan, hobi;
+-----+-----+-----+
| Table      | Op    | Msg_type | Msg_text |
+-----+-----+-----+
| test.jurusan | check | status   | OK
| test.hobi   | check | status   | OK
+-----+-----+-----+
2 rows in set (0.00 sec)
```

Repair

- Lakukan pengecekan dan repair pada Semua table di semua database.

Sintaknya :

```
mysqlcheck -u root -p --all-databases --repair
```

Lakukan repair pada table tertentu di semua database.

```
REPAIR TABLE election;
```

```
MariaDB [president]> REPAIR TABLE election;
+-----+-----+-----+-----+
| Table | Op    | Msg_type | Msg_text |
+-----+-----+-----+-----+
| president.election | repair | status   | OK      |
+-----+-----+-----+-----+
1 row in set (0.01 sec)
```

```
SHOW CREATE TABLE state;
```

```
ALTER TABLE election
ENGINE=MYISAM;
```

```
MariaDB [(none)]> use president;
Database changed
MariaDB [president]> SHOW CREATE TABLE state;
+-----+
| Table | Create Table
+-----+
| state | CREATE TABLE `state` (
  `STATE_NAME` char(15) NOT NULL,
  `ADMIN_ENTERED` smallint(6) DEFAULT NULL,
  `YEAR_ENTERED` smallint(6) NOT NULL,
  PRIMARY KEY (`STATE_NAME`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 |
+-----+
1 row in set (0.00 sec)
```

Repair

- Lakukan repair pada 2 table tertentu pada database tertentu.

Show TABLES;

```
MariaDB [(none)]> use test;
Database changed
MariaDB [test]> show tables;
+-----+
| Tables_in_test |
+-----+
| contoh_blob   |
| contoh_cha    |
| contoh_date   |
| contoh_text   |
| hobi          |
| jurusan       |
+-----+
6 rows in set (0.00 sec)
```

REPAIR TABLE jurusan, hobi;

```
MariaDB [test]> repair table jurusan, hobi;
+-----+-----+-----+-----+
| Table      | Op     | Msg_type | Msg_text |
+-----+-----+-----+-----+
| test.jurusan | repair | status   | OK      |
| test.hobi    | repair | status   | OK      |
+-----+-----+-----+-----+
2 rows in set (0.02 sec)
```

Repair

- Lakukan pengecekan dan repair atau perbaikan pada secara otomatis pada table tertentu.

```
mysqlcheck --auto-repair --databases test --tables jurusan  
hobi -u -root
```

```
C:\xampp\mysql\bin>mysqlcheck --auto-repair --databases test --tables jurusan hobi -u root  
test.jurusan                                     OK  
test.hobi                                         OK
```

- **Optimasi tabel**

Mysqlcheck dijalankan dengan **argumen -o**.

- **Menganalisis Tabel**

Analisis tabel dilakukan dengan **argumen -a**.

```
C:\xampp\mysql\bin>  
C:\xampp\mysql\bin>mysqlcheck -a --databases test --tables jurusan -u root -p  
Enter password:  
test.jurusan                                         Table is already up to date  
  
C:\xampp\mysql\bin>mysqlcheck -o --databases test --tables jurusan -u root -p  
Enter password:  
test.jurusan                                         OK
```

```
mysqlcheck -a --  
databases test --  
tables jurusan -u root  
-p
```

```
mysqlcheck -o --  
databases test --  
tables jurusan -u root  
-p
```

8) Proyek Akhir

Proyek Akhir

- Membuat aplikasi sederhana dengan fokus **Penerapan Database** ke Aplikasi untuk menyimpan transaksi
- **Tahapannya :**
 - Penentuan Studi Kasus
 - Perancangan Database beserta Relasi Tabelnya
 - Pada database terdapat beberapa SQL Langguage yang dilakukan diantaranya : CRUD, Transactions, Function, Stored Procedure & Trigger, System Catalog hingga hak akses.
 - Untuk Aplikasi boleh Web atau Desktop, fokus pada penerapan Database.
 - Pembuatan Laporan atau Dokumentasi.
- **Poin penilaian:** Aplikasi (Penerapan Database), Dokumentasi, Presentasi.

9) Kebutuhan Software

Kebutuhan Software

Browser

- Adobe flash
- Chrome
- Firefox

Localserver

- Xampp
- Laragon

Desain Tools

- Power Designer
- Sparx Enterprise Architect

Editor

- Notepad++
- Sublime Text

Database GUI

- PostgreSQL
- HeidiSQL
- SQLYog
- FlySpeed SQL

Database

- Mysql
- Oracle

10) Contact

Contact

- Bahan Kuliah : github.com/doniaft
- Email : doniaft@gmail.com
- WA/Telegram :
- Komting TIF 4D : Ali Mustofa : 081 231 202 253

II) Referensi

Referensi (I)

- ❑ SQL For MySQL Developers, Rick F. van der Lans, Addison Wesley, 2007
- ❑ MySQL Reference Manual, MySQL 2003
- ❑ Database Systems - A Practical Approach to Design, Implementation, and Management, Thomas Connoly and Carolyn Begg, Addison Wesley 1999
- ❑ Raghu Ramakhrisnan, Johannes Gehrke , “Database Management System” 6rd Edition, Mc Graw Hill, 2003 - 2006
- ❑ Arief, Rudiyanto M, Pemrograman Basis Data menggunakan Transact-SQL dengan Microsoft SQL Server 2000, Andi Yogyakarta.
- ❑ Rick van der Lans, Introduction to SQL, Mastering Relational Database Language 2nd Edition, Addison-Wesley, 2000.
- ❑ Chris Bates, Web Programming: Building Internet Applications, Third Edition, John Wiley & Sons Ltd, England, 2006.
- ❑ Sebesta, R.W., Programming the World Wide Web, Addison Wesley, 2002.
- ❑ Elliot White III, Jonathan Eisenhamer, PHP 5 in Practice, Sams, 2006
- ❑ Elmasri and Navathe, Fundamental of Database Systems 4th Edition, Addison-Wesley, 2004
- ❑ Silberschatz, Korth and Sudarshan, Database System Concepts, 5th Edition, Mc Graw Hill, International Edition, 2006.
- ❑ Connoly, Thomas and Begg, Carolyn: Database Sytems 4th edition, Prentice Hall, 2005

Referensi (2)

- Andrea Tar. 2012. PHP and MySQL 24-Hour Trainer
- Brett McLaughlin. 2012. PHP & MySQL- The Missing Manual. USA-Brett McLaughlin. USA-O'REILLY Media
- Brett McLaughlin. 2013. PHP & MySQL- The Missing Manual, 2nd Edition. USA-Brett McLaughlin. USA-O'REILLY Media
- Head First PHP & MySQL
- Kroenke, David. 2013. Database Processing 12th Edition
- Mysql Official. 2016. MySQL 5.7 Reference Manual-en
- PHP6 and MySQL Bible by Steve Suehring
- Rochkin Mark. 2013. Expert PHP and MySQL
- Ruehning, dkk. php_mysql_javascript__html5_all-in-one_for_dummies
- Sams.Sams.Teach.Yourself.PHP.MySQL.and.Apache.All-in-One.ISBN0672326205
- Solichin, Achmad. Pemrograman Web dengan PHP MySQL
- Tutorialpoints.com - mysql tutorial
- Valade, Janet. PHP & MySQL Web Development All-in-One Desk Reference For Dummies. CanadaWiley Publishing, Inc
- Widigdo, Anon Kuncoro. 2003. php dan mysql
- <https://www.duniaIlkom.com/tutorial-belajar-mysql-dan-index-artikel-mysql/#mysqllanjutan>