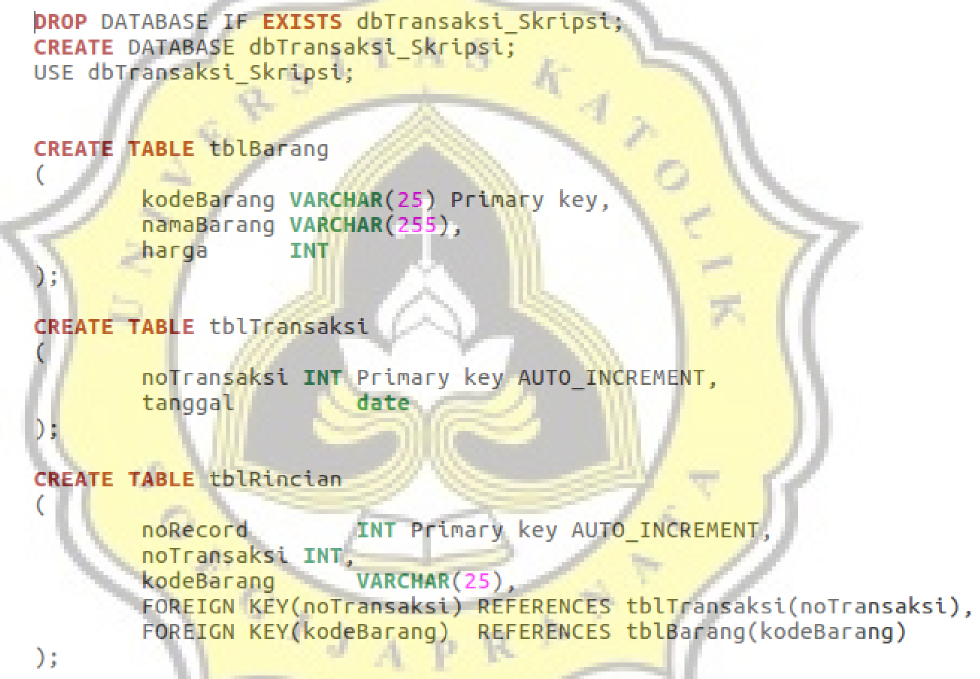
CHAPTER 3
RESEARCH METHODOLOGY

3.1 Creating Database and Tables

First step of this Project is creating Database and Tables. They are created using SQL Query. First, create and give name the database. Then, use the database. After that, create three tables and their attributes.



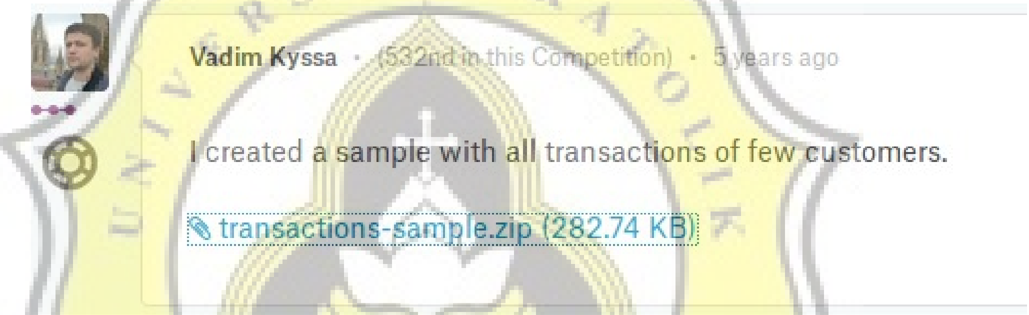
```
DROP DATABASE IF EXISTS dbTransaksi_Skripsi;
CREATE DATABASE dbTransaksi_Skripsi;
USE dbTransaksi_Skripsi;


CREATE TABLE tblBarang
(
        kodeBarang VARCHAR(25) Primary key,
        namaBarang VARCHAR(255),
        harga      INT
);

CREATE TABLE tblTransaksi
(
        noTransaksi INT Primary key AUTO_INCREMENT,
        tanggal         date
);

CREATE TABLE tblRincian
(
        noRecord        INT Primary key AUTO_INCREMENT,
        noTransaksi INT,
        kodeBarang      VARCHAR(25),
        FOREIGN KEY(noTransaksi) REFERENCES tblTransaksi(noTransaksi),
        FOREIGN KEY(kodeBarang)  REFERENCES tblBarang(kodeBarang)
);
```

Illustration 3.1: SQL - Create Database and Create Tables

As seen on illustration above, database is named “dbTransaksi_Skripsi” and there were three tables that were used in this project. First table is **tblBarang**. This table saved three data of an item. There are *kodeBarang*, *namaBarang*, and *harga* (itemCode, itemName, and itemPrice). TblBarang saved data item records. Next, there was tblTransaksi which was saved *noTransaksi* (Transaction ID) and *tanggal* (date of transaction). Last table, tblRincian was used for recording detailed transaction. This table also connected two tables before, tblBarang and tblTransaksi. Last, this file is saved into SQL file. With database and tables have created, this process is done.

3.2 Collecting Data

Next, Collecting data process. Data of this project is taken from web https://www.kaggle.com/c/acquire-valued-shoppers-challenge/discussion/7659 , this site provided 1000 sample sales transaction records that can be downloaded easily.



Vadim Kyssa · (532nd in this Competition) · 5 years ago

I created a sample with all transactions of few customers.

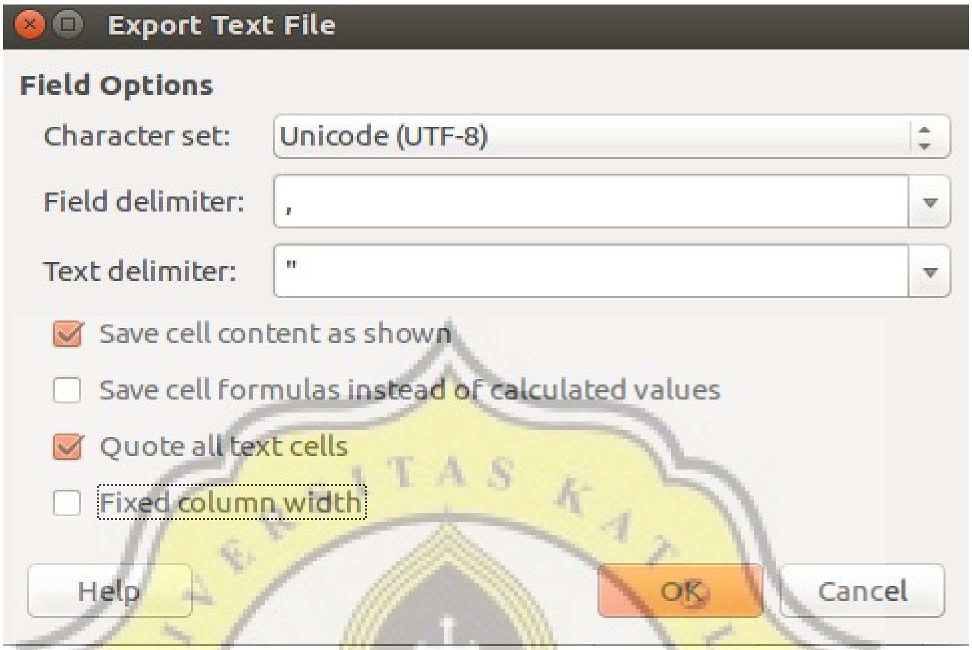
transactions-sample.zip (282.74 KB)

Illustration 3.2: Sample Data

Download transactions-sample.zip and extract it to get transactions-sample.csv file. Then, file is opened and modified according table’s attributes above. After that, save new edited detailed transactions into new table and save into spreadsheet files (ODS or XLXS file). After that, create random data items for data item and data transaction. So, there are three sample data that can be processed.

3.3 Processing Data

First, the sample data is prepared into CSV file. Before that, make sure that three sample data have already matched with table structure. After that, All sample data record which is formatted in csv file, followed this same format.



Export Text File

Field Options

Character set: Unicode (UTF-8)

Field delimiter: ,

Text delimiter: "

☑ Save cell content as shown

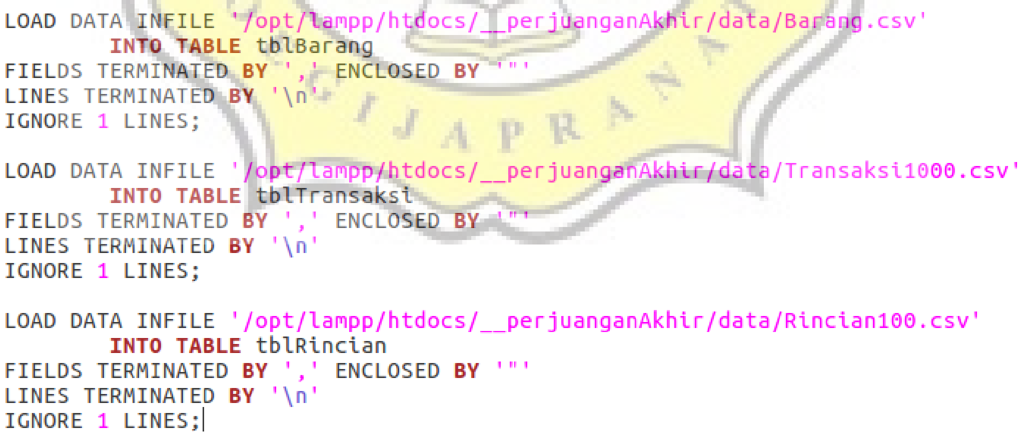
□ Save cell formulas instead of calculated values

☑ Quote all text cells

□ Fixed column width

Illustration 3.3: Format File CSV

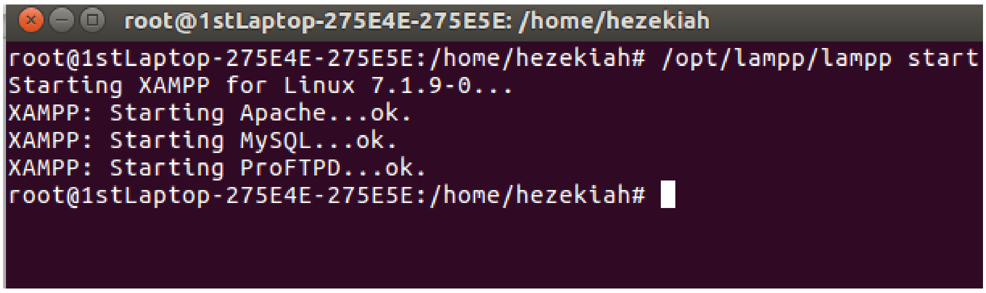
After documents are saved into CSV files, add SQL file a query to inserting data from .csv file. Sample data which is taken from web, is inserted into tblRincian. Illustration below told how to create code to import data from CSV files to database server.



```
LOAD DATA INFILE '/opt/lampp/htdocs/__perjuanganAkhir/data/Barang.csv'
        INTO TABLE tblBarang
FIELDS TERMINATED BY ',' ENCLOSED BY '"'
LINES TERMINATED BY '\n'
IGNORE 1 LINES;

LOAD DATA INFILE '/opt/lampp/htdocs/__perjuanganAkhir/data/Transaksi1000.csv'
        INTO TABLE tblTransaksi
FIELDS TERMINATED BY ',' ENCLOSED BY '"'
LINES TERMINATED BY '\n'
IGNORE 1 LINES;

LOAD DATA INFILE '/opt/lampp/htdocs/__perjuanganAkhir/data/Rincian100.csv'
        INTO TABLE tblRincian
FIELDS TERMINATED BY ',' ENCLOSED BY '"'
LINES TERMINATED BY '\n'
IGNORE 1 LINES;
```

Illustration 3.4: SQL - Import Data from CSV File

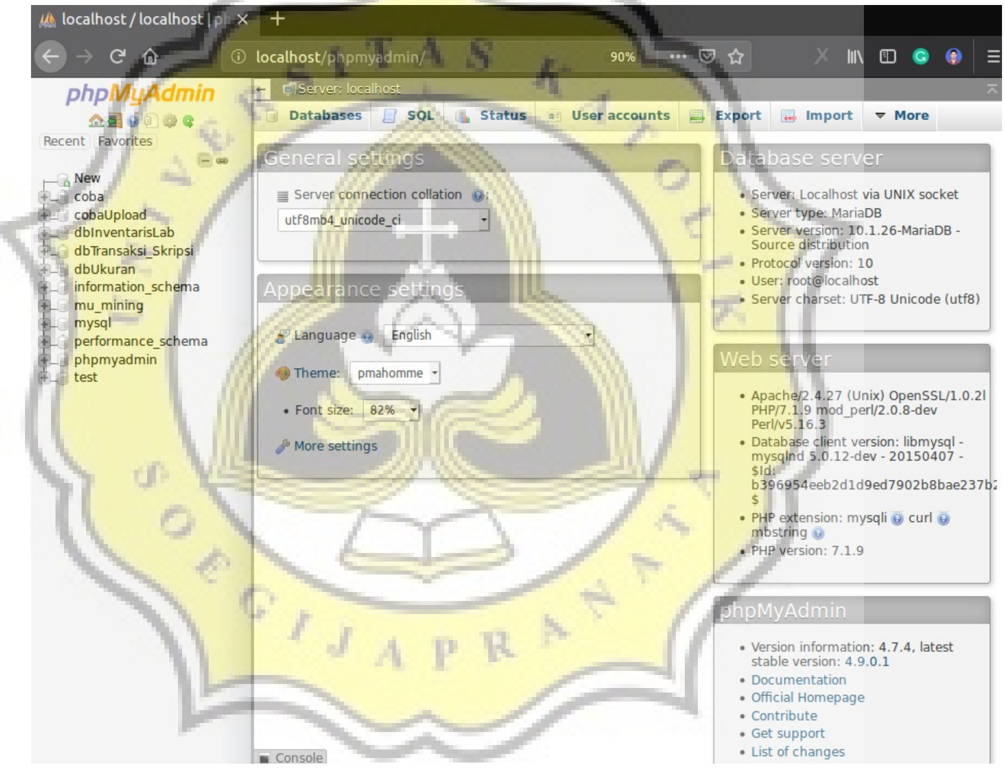
Last, Import SQL file to database server. As this project is still hosted in localhost, so this project used MySql as database server. Lampp is turned on first to activate Apache and MySql. This activation can using Lampp control panel or Ubuntu terminal.



```
root@1stLaptop-275E4E-275E5E: /home/hezekiah
root@1stLaptop-275E4E-275E5E:/home/hezekiah# /opt/lampp/lampp start
Starting XAMPP for Linux 7.1.9-0...
XAMPP: Starting Apache...ok.
XAMPP: Starting MySQL...ok.
XAMPP: Starting ProFTPD...ok.
root@1stLaptop-275E4E-275E5E:/home/hezekiah# 
```

Illustration 3.5: Lampp activation

After Lampp has started, open the browser and type **localhost/phpmyadmin/** .



- New
- coba
- cobaUpload
- dbInventarisLab
- dbTransaksi_Skripsi
- dbUkuran
- information_schema
- mu_mining
- mysql
- performance_schema
- phpmyadmin
- test

General settings

Server connection collation: utf8mb4_unicode_ci

Appearance settings

Language: English

Theme: pmahomme

- Font size: 82%

More settings

Database server

- Server: Localhost via UNIX socket
- Server type: MariaDB
- Server version: 10.1.26-MariaDB - Source distribution
- Protocol version: 10
- User: root@localhost
- Server charset: UTF-8 Unicode (utf8)

Web server

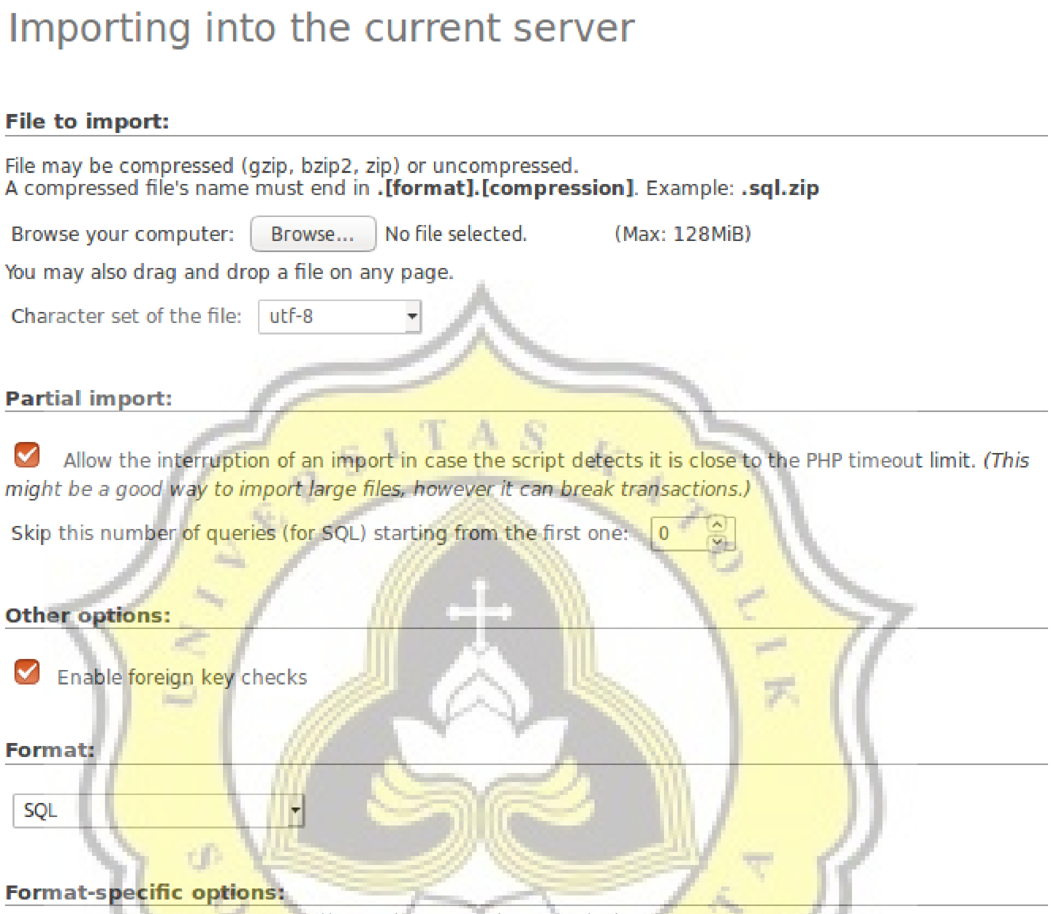
- Apache/2.4.27 (Unix) OpenSSL/1.0.2l PHP/7.1.9 mod_perl/2.0.8-dev Perl/v5.16.3
- Database client version: libmysql - mysqlnd 5.0.12-dev - 20150407 - $Id: b396954eeb2d1d9ed7902b8bae237b2 $
- PHP extension: mysqli curl mbstring
- PHP version: 7.1.9

phpMyAdmin

- Version information: 4.7.4, latest stable version: 4.9.0.1
- Documentation
- Official Homepage
- Contribute
- Get support
- List of changes

Illustration 3.6: PhpMyAdmin - Homepage

Then click on menu Import on top bar of the page. In this menu, Sql data could be imported into MySql.



Importing into the current server

File to import:

File may be compressed (gzip, bzip2, zip) or uncompressed.
A compressed file's name must end in **.[format].[compression]**. Example: **.sql.zip**

Browse your computer: No file selected. (Max: 128MiB)

You may also drag and drop a file on any page.

Character set of the file: utf-8

Partial import:

☑ Allow the interruption of an import in case the script detects it is close to the PHP timeout limit. *(This might be a good way to import large files, however it can break transactions.)*

Skip this number of queries (for SQL) starting from the first one: 0

Other options:

☑ Enable foreign key checks

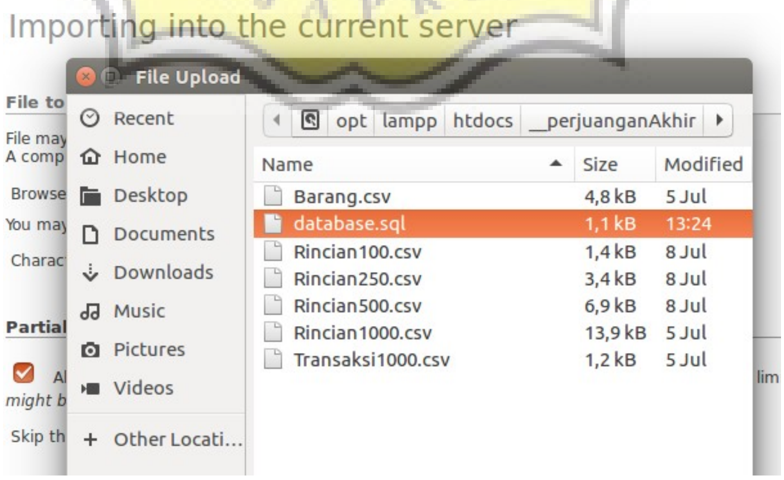
Format:

SQL

Format-specific options:

Illustration 3.7: PhpMyAdmin - Import

Then, click **Browse…**



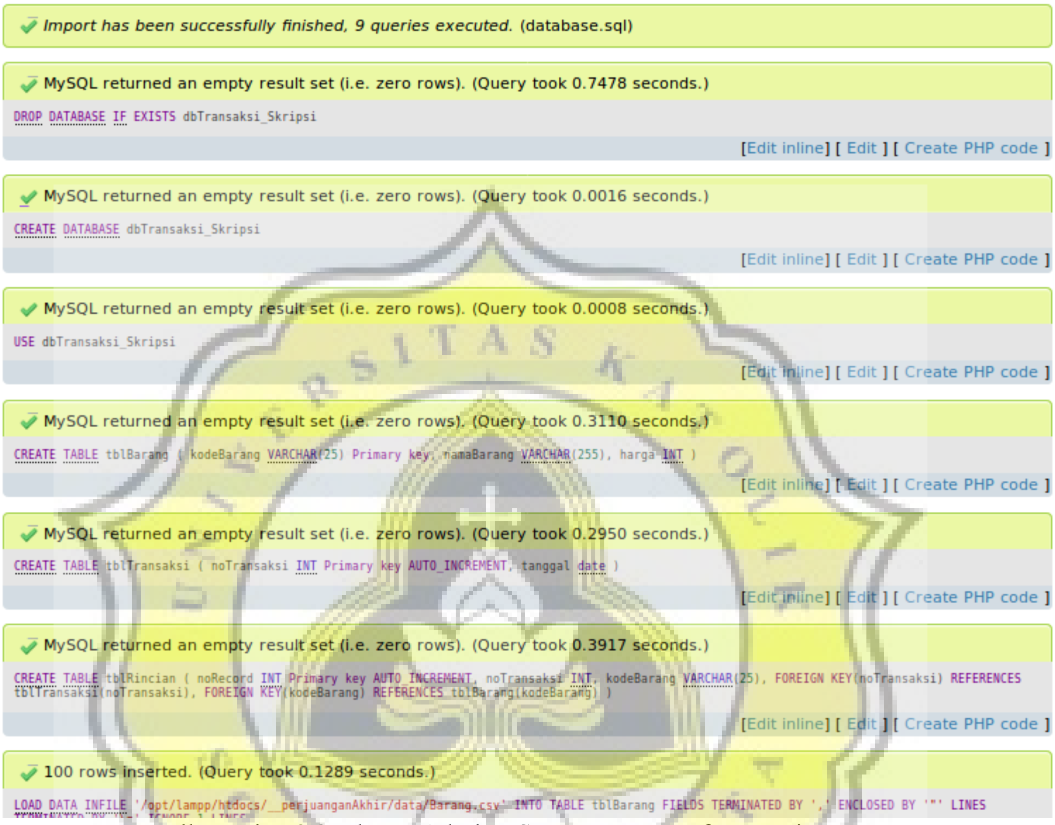
Importing into the current server

File Upload

Name	Size	Modified
Barang.csv	4,8 kB	5 Jul
database.sql	1,1 kB	13:24
Rincian100.csv	1,4 kB	8 Jul
Rincian250.csv	3,4 kB	8 Jul
Rincian500.csv	6,9 kB	8 Jul
Rincian1000.csv	13,9 kB	5 Jul
Transaksi1000.csv	1,2 kB	5 Jul

Illustration 3.8: PhpMyAdmin - Search SQL File

Locate and click Sql file and click GO on bottom of this page. After that, server processed SQL file. If importing process is correct, the result is like this :



```
Import has been successfully finished, 9 queries executed. (database.sql)

MySQL returned an empty result set (i.e. zero rows). (Query took 0.7478 seconds.)
DROP DATABASE IF EXISTS dbTransaksi_Skripsi
[Edit inline] [ Edit ] [ Create PHP code ]

MySQL returned an empty result set (i.e. zero rows). (Query took 0.0016 seconds.)
CREATE DATABASE dbTransaksi_Skripsi
[Edit inline] [ Edit ] [ Create PHP code ]

MySQL returned an empty result set (i.e. zero rows). (Query took 0.0008 seconds.)
USE dbTransaksi_Skripsi
[Edit inline] [ Edit ] [ Create PHP code ]

MySQL returned an empty result set (i.e. zero rows). (Query took 0.3110 seconds.)
CREATE TABLE tblBarang ( kodeBarang VARCHAR(25) Primary key, namaBarang VARCHAR(255), harga INT )
[Edit inline] [ Edit ] [ Create PHP code ]

MySQL returned an empty result set (i.e. zero rows). (Query took 0.2950 seconds.)
CREATE TABLE tblTransaksi ( noTransaksi INT Primary key AUTO_INCREMENT, tanggal date )
[Edit inline] [ Edit ] [ Create PHP code ]

MySQL returned an empty result set (i.e. zero rows). (Query took 0.3917 seconds.)
CREATE TABLE tblRincian ( noRecord INT Primary key AUTO_INCREMENT, noTransaksi INT, kodeBarang VARCHAR(25), FOREIGN KEY(noTransaksi) REFERENCES
tblTransaksi(noTransaksi), FOREIGN KEY(kodeBarang) REFERENCES tblBarang(kodeBarang) )
[Edit inline] [ Edit ] [ Create PHP code ]

100 rows inserted. (Query took 0.1289 seconds.)
LOAD DATA INFILE '/opt/lampp/htdocs/__perjuanganAkhir/data/Barang.csv' INTO TABLE tblBarang FIELDS TERMINATED BY ',' ENCLOSED BY '"' LINES
```

Illustration 3.9: PhpMyAdmin - Success Report of Importing Data

With this report, Collecting Data process is finished. Data is successfully inputted to MySql and can be processed according algorithm process.

3.4 Building a System

Database and tables have been created, data have also been inserted, then next step is building a System. System is built from many codes from one or more languages. System language that used in this project are PHP and HTML. Building system is started from its preparation. Preparation is started from connecting database to system which can be seen in browser. After that, next preparation is showing data tables into browser. This process send a query to database and saved in the variable. Later, the variable is printed each row. So, the table from database can be seen or shown in browser. That is the preparation of building a system.

Then, system is started to built. The system is built step by step according to FP-Growth algorithm. Each step required data modification and modification through PHP codes. After each step is finished, data that has been modified or result from PHP codes is showed to the browser. System is continued until it is finished, until Frequent Pattern Sets that are used for item set recommendation that most purchased together are found. For system analysis, system timer and used memory counter are added. Main system's process is done.

Last, create 100 data, 250 data, and 500 data, and repeat same steps from inserting Sql until last.

3.5 System Analysis

When every system is done processing data, system timer and used memory counter are recorded. After system is done processing all data, System timer record and used memory record are saved into table and presented as a chart. After that, from those two chart / table, can be analyzed the performance of the system

3.6 Writing

After system simulation have done, and system performance has been analyzed, write a Project report as a research documentation.