

**LAPORAN**  
**TUGAS 1**  
**DATA WAREHOUSE DAN BISNIS INTELLIGENCE**



Oleh:  
Ian Ariessa Sitorus  
(181402093)

**TEKNOLOGI INFORMASI**  
**FAKULTAS ILMU KOMPUTER - TEKNOLOGI INFORMASI**  
**UNIVERSITAS SUMATERA UTARA**  
**MEDAN**  
**2021**

## Module 3 Problems

The problems use the *Customer*, *Facility*, and *Location* tables of the intercollegiate athletic database. The *Customer* table contains clients who initiate event requests. The *Facility* table contains available facilities. The *Location* table contains several locations inside facilities. The primary keys of the tables are *CustNo* for *Customer*, *FacNo* for *Facility*, and *LocNo* for *Location*.

### Customer

custno	custname	address	Internal	contact	phone	city	state	zip
C100	Football	Box 352200	Y	Marv Manager	6857100	Boulder	CO	80309
C101	Men's Basketball	Box 352400	Y	Sally	5431700	Boulder	CO	80309
C103	Baseball	Box 352020	Y	Bill Baseball	5431234	Boulder	CO	80309
C104	Women's Softball	Box 351200	Y	Sue Softball	5434321	Boulder	CO	80309
C105	High School	123	N	Coach Bob	4441234	Louisville	CO	80027

### Facility

facn	facname
F100	Football stadium
F101	Basketball arena
F102	Baseball field
F103	Recreation room

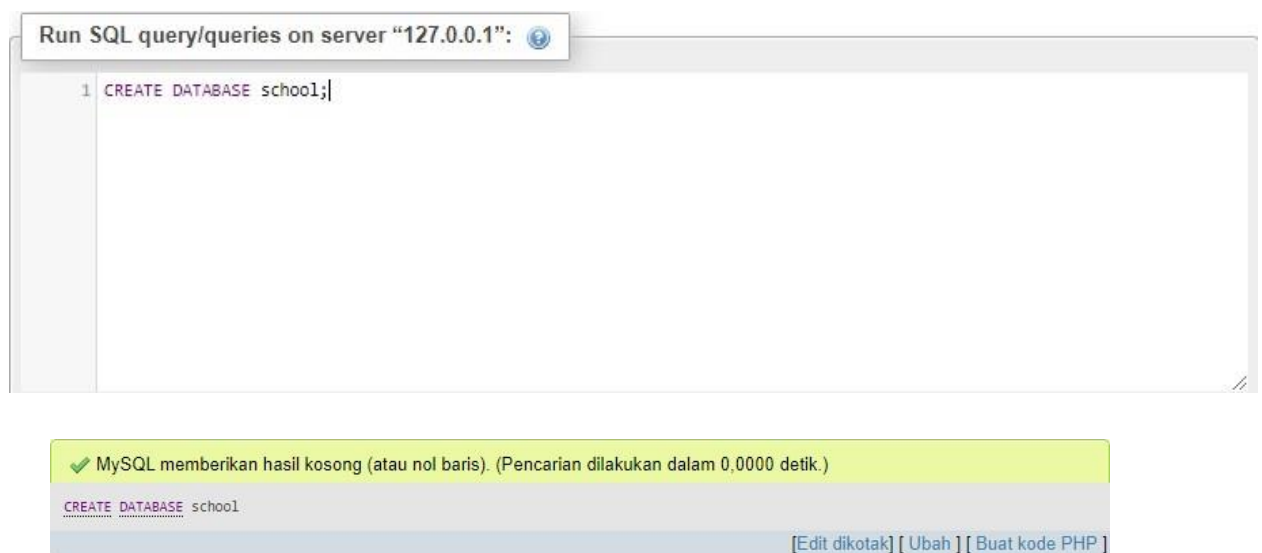
### Location

locno	facno	locname
L100	F100	Locker room
L101	F100	Plaza
L102	F100	Vehicle gate
L103	F101	Locker room
L104	F100	Ticket Booth
L105	F101	Gate
L106	F100	Pedestrian

1. Write a CREATE TABLE statement for the *Customer* table. Choose data types appropriate for the DBMS used in your course. All columns are required (not null).
2. Write a CREATE TABLE statement for the *Facility* table. Choose data types appropriate for the DBMS used in your course. All columns are required (not null).
3. Write a CREATE TABLE statement for the *Location* table. Choose data types appropriate for the DBMS used in your course. *LocName* column is required (not null).
4. Identify the foreign key(s) and 1-M relationship(s) among the *Customer*, *Facility*, and *Location* tables. For each relationship, identify the parent table and the child table.
5. Extend your CREATE TABLE statement from problem (3) with referential integrity constraints.
6. From examination of the sample data and your common understanding of scheduling and operation of events, are null values allowed for the foreign key in the *Location* table? Why or why not? Extend the CREATE TABLE statement in problem (5) to enforce the null value restrictions if any.
7. Extend your CREATE TABLE statement for the *Facility* table (problem 2) with a unique constraint for *FacName*. Use an external named constraint clause for the unique constraint.

## Penyelesaian

Untuk menyelesaikan tugas ini, saya akan menggunakan MySQL sebagai sistem manajemen database. Tahap pertama yang saya lakukan adalah *create Database* sesuai dengan arahan. Disini saya menggunakan aplikasi *phpmyadmin* untuk membantu saya dalam pengolahan database. Saya membuat penamaan *database* dengan nama *school*. Berikut *query* udaklam tahapan *create database* :



Setelah kita mengetik *query*, maka akan muncul notifikasi seperti di atas yang menunjukkan bahwa *database school* telah berhasil dibuat dan masih dalam keadaan kosong.

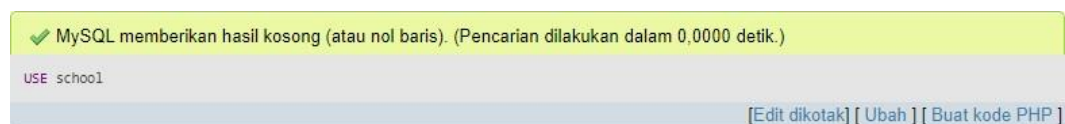
Kemudian kita mengetik *query* seperti berikut agar kita dapat menggunakan *database school* yang baru saja kita *create*.



Run SQL query/queries on server "127.0.0.1":

```
1 USE school;
```

Setelah itu, akan muncul notifikasi dibawah ini yang menunjukkan bahwa *query* yang kita gunakan berjalan dengan baik.



✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0000 detik.)

```
USE school
```

[Edit dikotak] [Ubah] [Buat kode PHP]

Setelah itu, kita akan menambahkan *table Customer*, *Facility*, dan *Location* pada *database school*.

Pertama kita akan menambahkan *table Customer* dengan ketentuan berikut :



Run SQL query/queries on server "127.0.0.1":

```
1 CREATE TABLE Customer
2 (CustNo varchar(8) not null,
3  CustName varchar(30) not null,
4  Address varchar(50) not null,
5  Internal char(1) not null,
6  Contact varchar(35) not null,
7  Phone varchar(11) not null,
8  City varchar(30) not null,
9  State varchar(2) not null,
10 Zip varchar(10) not null,
11 CONSTRAINT PK_CUSTOMER PRIMARY KEY(CustNo));
```

Kedua kita akan menambahkan *table Facility* dengan ketentuan berikut :

```
Jalankan perintah SQL pada basis data school:

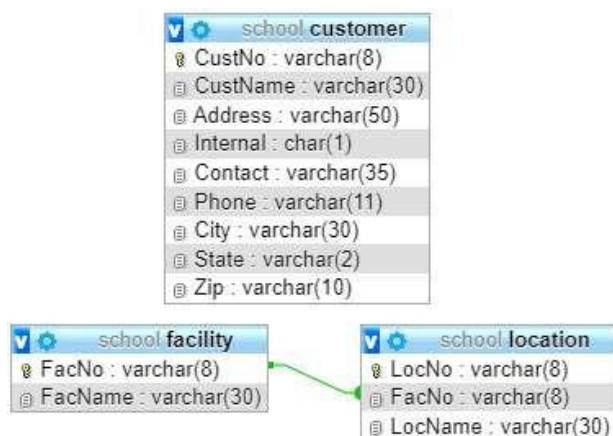
1 CREATE TABLE Facility
2 (FacNo varchar(8) not null,
3 FacName varchar(30) not null,
4 CONSTRAINT PK_FACILITY PRIMARY KEY(FacNo));
```

Ketiga kita akan menambahkan *table Location* dengan ketentuan berikut :


```
Jalankan perintah SQL pada basis data school:

1 CREATE TABLE Location
2 (LocNo varchar(8) not null,
3 FacNo varchar(8),
4 LocName varchar(30) not null,
5 CONSTRAINT PK_LOCATION PRIMARY KEY(LocNo));
```

Setelah itu, kita dapat melihat hubungan 1-M (*one to many*), yaitu pada table *Facility* (*FacNo-PK*) dengan *table Location* (*FacNo-FK*).



Kemudian kita menambahkan *query* untuk menambahkan *Constraints Referential Integrity* pada kolom *FacNo* pada pembuatan *table Location*.

Jalankan perintah SQL pada basis data school: 


```
1 CREATE TABLE Location
2 (LocNo VARCHAR(8) not null,
3  FacNo VARCHAR(8),
4  LocName VARCHAR(30) not null,
5  CONSTRAINT PK_LOCATION PRIMARY KEY(LocNo),
6  CONSTRAINT FK_FACNO FOREIGN KEY(FacNo)
7  REFERENCES FACILITY (FacNo));
```

Setelah saya analisis, pada kolom *FacNo* tidak boleh ada data nilai yang kosong (*null*). Karena pada *table Location* saling berkaitan dengan *table Facility* dan dalam pembuatan *table Location* telah ditambahkan *referential integrity* untuk menjaga konsistensi data pada *table Facility* melalui kolom *FacNo*. Sehingga jika nilai pada kolom *FacNo* dalam pembuatan *table Location*.

Jalankan perintah SQL pada basis data school: 

```
1 CREATE TABLE Location
2 (LocNo varchar(8) not null,
3  FacNo varchar(8) not null,
4  LocName varchar(30) not null,
5  CONSTRAINT PK_LOCATION PRIMARY KEY(LocNo),
6  CONSTRAINT FK_FACNO FOREIGN KEY(FacNo)
7  REFERENCES FACILITY (FacNo));
```

Terakhir, kita menambahkan *query* untuk melakukan batasan *constraints unique* di kolom *FacName* pada pembuatan *table Facility*.

Jalankan perintah SQL pada basis data school: 

```
1 CREATE TABLE Facility
2 (FacNo varchar(8) not null,
3  FacName varchar(30) not null,
4  CONSTRAINT PK_FACILITY PRIMARY KEY (FacNo),
5  CONSTRAINT Unique_FacName UNIQUE(FacName));
```