Nama   : Fadel Majid Muhammad
Nim   : 181402030
Matkul   : Data Warehouse & Bisnis Intelligence

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).

```
MariaDB [dwbi1]> tee Tugas1_DWBI.txt
Logging to file 'Tugas1_DWBI.txt'
MariaDB [dwbi1]> create table Customer (
    -> custNo char(11) not null,
    -> custName varchar(100) not null,
    -> address varchar(100) not null,
    -> Internal varchar(2) not null,
    -> Contact varchar(100) not null,
    -> Phone varchar(14) not null,
    -> City varchar(10) not null,
    -> state varchar(10) not null,
    -> zip varchar(10) not null,
    -> CONSTRAINT CustomerPK PRIMARY KEY(custNo));
ERROR 2006 (HY000): MySQL server has gone away
No connection. Trying to reconnect...
Connection id:    51
Current database: dwbi1

Query OK, 0 rows affected, 1 warning (0.037 sec)
```

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).

```
MariaDB [dwbi1]> create table Facility (
    -> facno char(11) not null,
    -> facname varchar(100) not null,
    -> CONSTRAINT FacilityPK PRIMARY KEY(facno));
Query OK, 0 rows affected, 1 warning (0.021 sec)
```

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 (nots null).

```
MariaDB [dwbi1]> create table Location (
    -> locno char(20) not null,
    -> facno char(20) not null,
    -> locname varchar(100) not null,
    -> CONSTRAINT LocationPK PRIMARY KEY(locno));
Query OK, 0 rows affected, 1 warning (0.021 sec)
```

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.

<div align="center">Jawab</div>

Pada ketiga table tersebut terdapat dua table yang saling berhubungan yaitu table *Location*

dan *Facility* dimana table *Location* merupakan parent dari table *Facility* sementara table

customer tidak berhubungan langsung dengan kedua table tersebut. Table *Location* menjadi

parent table dan table *Facility* juga table *customer* merupakan child table dan foreign key

pada ketiga table diatas terdapat pada table *Location* dengan nama atribut *facno*.

5. Extend your CREATE TABLE statement from problem (3) with referential integrity

   constraints.

```
MariaDB [dwbi1]> alter table Location add CONSTRAINT LocationFK FOREIGN KEY (facno) REFERENCES Facility(facno
) on DELETE CASCADE on UPDATE CASCADE;
Query OK, 0 rows affected (0.080 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

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.

<div align="center">Jawab</div>

Menurut saya, foreign key pada table location boleh-boleh saja null (tidak diisi) dikarenakan foreign key tidak memiliki fungsi untuk mengidentifikasi record yang terdapat dalam table. Berbeda halnya dengan primary key nilai(value) yang digunakan tidak boleh NULL(kosong), record yang dibuat harus terisi nilai. Jika nilai dalam record bersifat NULL maka tidak bisa mengidentifikasi Nilai dalam tabel tersebut.

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.

```
MariaDB [dwbi1]> alter table Facility add UNIQUE (facno);
Query OK, 0 rows affected (0.025 sec)
Records: 0  Duplicates: 0  Warnings: 0
```