

2. Lengkapi tabel rentalfilm sesuai dengan Contoh ERD.

✓ MySQL memberikan hasil kosong

```
CREATE DATABASE rentalfilm;
```

Pertama, buat *database* dengan nama rentalfilm terlebih dahulu, lalu *double click* atau ketikkan “use database rentalfilm” untuk dapat menggunakan *database* tersebut untuk tugas praktikum ini.

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilaku

```
1 CREATE TABLE Category(  
2   ID int(10) NOT NULL,  
3   Name varchar(25),  
4   Last_Update TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
5   PRIMARY KEY (ID)  
6 );
```

Selanjutnya, membuat tabel *Category* dengan atribut ID, Name, dan Last_Update dengan ketentuan hanya ID yang bersifat Not Null dan Primary Key.

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilaku

```
1 CREATE TABLE Language(  
2   ID int(10) NOT NULL,  
3   Name varchar(20),  
4   Last_Update TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
5   PRIMARY KEY (ID)  
6 );
```

Setelah itu, saya membuat tabel *Language* dengan atribut ID, Name, dan Last_Update dengan ketentuan hanya ID yang bersifat Not Null dan Primary Key.

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilaku

```

1 CREATE TABLE Film(
2   ID int(10) NOT NULL,
3   LanguageID int(10) NOT NULL,
4   Title varchar(255),
5   Description varchar(255),
6   Release_Year int(4),
7   Rental_Duration int(10),
8   Rental_Rate numeric(19,0),
9   Length int(2),
10  Replacement_Cost numeric(19,0),
11  Rating int(10),
12  Last_Update TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
13  Special_Features varchar(255),
14  Full_Text varchar(255),
15  PRIMARY KEY (ID),
16  FOREIGN KEY (LanguageID) REFERENCES Language(ID)
17 );

```

Setelah itu, saya membuat tabel *Film* dengan atribut ID, LanguageID, Title, Description, Release_Year, Rental_Duration, Rental_Rate, Length, Replacement_Cost, Rating, Last_Update, Special_Features, Full_Text dengan ketentuan hanya ID dan LanguageID yang bersifat Not Null dan ID sebagai Primary Key sedangkan LanguageID sebagai Foreign Key untuk tabel Language.

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilaku

```

1 CREATE TABLE Film_Category(
2   FilmID int(10) NOT NULL,
3   CategoryID int(10) NOT NULL,
4   Last_Update TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
5   FOREIGN KEY (FilmID) REFERENCES film(ID),
6   FOREIGN KEY (CategoryID) REFERENCES category(ID)
7 );

```

Setelah itu, saya membuat tabel *Film_Category* sebagai tabel penampung dengan atribut FilmID, CategoryID, Last_Update dengan ketentuan hanya FilmID dan CategoryID yang bersifat Not Null dan FilmID sebagai Foreign Key untuk tabel Film sedangkan CategoryID sebagai Foreign Key untuk tabel Category.

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilaku

```
1 CREATE TABLE Actor(
2     ID int(10) NOT NULL,
3     First_Name varchar(255),
4     Last_Name varchar(255),
5     Last_Update TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
6     PRIMARY KEY (ID)
7 );
```

Setelah itu, saya membuat tabel *Actor* dengan atribut ID, First_Name, Last_Name dan Last_Update dengan ketentuan hanya ID yang bersifat Not Null dan ID sebagai Primary Key.

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilaku

```
1 CREATE TABLE Film_Actor(
2     FilmID int(10) NOT NULL,
3     ActorID int(10) NOT NULL,
4     Last_Update TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
5     FOREIGN KEY (FilmID) REFERENCES film(ID),
6     FOREIGN KEY (ActorID) REFERENCES actor(ID)
7 );
```

Setelah itu, saya membuat tabel *Film_Actor* sebagai tabel penampung dengan atribut FilmID, ActorID, Last_Update dengan ketentuan hanya FilmID dan ActorID yang bersifat Not Null dan FilmID sebagai Foreign Key untuk tabel Film sedangkan ActorID sebagai Foreign Key untuk tabel Actor.

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilaku

```
1 CREATE TABLE Inventory(
2     ID int(10) NOT NULL,
3     FilmID int(10),
4     Last_Update TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
5     FOREIGN KEY (ID) REFERENCES film(ID)
6 );
```

Setelah itu, saya membuat tabel *Inventory* dengan atribut ID, FilmID, dan Last_Update dengan ketentuan hanya ID dan FilmID yang bersifat Not Null dan ID sebagai Foreign Key untuk tabel Film.

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilaku

```
1 CREATE TABLE Country(
2     ID int(10) NOT NULL,
3     Country varchar(50),
4     Last_Update TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
5     PRIMARY KEY (ID)
6 );
```

Setelah itu, saya membuat tabel *Country* dengan atribut ID, Country, dan Last_Update dengan ketentuan hanya ID yang bersifat Not Null dan ID sebagai Primary Key.

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilaku

```
1 CREATE TABLE City(
2     ID int(10) NOT NULL,
3     CountryID int(10),
4     City varchar(50),
5     Last_Update TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
6     FOREIGN KEY (CountryID) REFERENCES country(ID),
7     PRIMARY KEY (ID)
8 );
```

Setelah itu, saya membuat tabel *City* dengan atribut ID, CountryID, City, dan Last_Update dengan ketentuan hanya ID dan CountryID yang bersifat Not Null dan ID sebagai Primary Key sedangkan CountryID sebagai Foreign Key untuk tabel Country.

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilaku

```
1 CREATE TABLE Address(
2     ID int(10) NOT NULL,
3     CityID int(10),
4     Address varchar(50),
5     Address2 varchar(50),
6     District int(20),
7     Postal_Code varchar(10),
8     Phone varchar(20),
9     Last_Update TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
10    PRIMARY KEY (ID),
11    FOREIGN KEY (CityID) REFERENCES city(ID)
12 );
```

Setelah itu, saya membuat tabel *Address* dengan atribut ID, CityID, Address, Address2, District, Postal_Code, Phone dan Last_Update dengan ketentuan hanya ID dan CityID yang bersifat Not Null dan ID sebagai Primary Key sedangkan CityID sebagai Foreign Key untuk tabel City.

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan)

```

1 CREATE TABLE Store(
2     ID int(10) NOT NULL,
3     AddressID int(10),
4     Last_Update TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
5     PRIMARY KEY (ID),
6     FOREIGN KEY (AddressID) REFERENCES address(ID)
7 );

```

Setelah itu, saya membuat tabel *Store* dengan atribut ID, AddressID dan Last_Update dengan ketentuan hanya ID dan AddressID yang bersifat Not Null dan ID sebagai Primary Key sedangkan AddressID sebagai Foreign Key untuk tabel Address.

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan)

```

1 CREATE TABLE Staff(
2     ID int(10) NOT NULL,
3     AddressID int(10) NOT NULL,
4     StoreID int(10) NOT NULL,
5     PaymentID int(10) NOT NULL,
6     First_Name varchar(255),
7     Last_Name varchar(255),
8     Email varchar(50),
9     Active char(1),
10    Username varchar(16),
11    Password varchar(40),
12    Last_Update TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
13    PictureURL varchar(80),
14    PRIMARY KEY (ID),
15    FOREIGN KEY (AddressID) REFERENCES address(ID),
16    FOREIGN KEY (StoreID) REFERENCES store(ID)
17 );

```

Setelah itu, saya membuat tabel *Staff* dengan atribut ID, AddressID, StoreID int(10), PaymentID, First_Name, Last_Name, Email, Active, Username, Password, Last_Update dan PictureURL dengan ketentuan hanya ID, AddressID, StoreID dan PaymentID yang bersifat Not Null dan ID sebagai Primary Key sedangkan AddressID sebagai Foreign Key untuk tabel Address.

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan)

```

1 CREATE TABLE Customer(
2     ID int(10) NOT NULL,
3     AddressID int(10) NOT NULL,
4     AddressColumn int(10) NOT NULL,
5     First_Name varchar(255),
6     Last_Name varchar(255),
7     Email varchar(50),
8     Active char(1),
9     Create_Date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
10    Last_Update TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
11    PRIMARY KEY (ID),
12    FOREIGN KEY (AddressID) REFERENCES address(ID)
13 );

```

Setelah itu, saya membuat tabel *Customer* dengan atribut ID, AddressID, AddressColumn, First_Name, Last_Name, Email, Active, Create_Date, dan Last_Update dengan ketentuan hanya ID, AddressID, dan AddressColumn yang bersifat Not Null dan ID sebagai Primary Key sedangkan AddressID sebagai Foreign Key untuk tabel Address.

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan)

```

1 CREATE TABLE Rental(
2     ID int(10) NOT NULL,
3     StaffID int(10) NOT NULL,
4     CustomerID int(10) NOT NULL,
5     InventoryID int(10) NOT NULL,
6     Rental_Date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
7     Return_Date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
8     Last_Update TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
9     PRIMARY KEY (ID),
10    FOREIGN KEY (StaffID) REFERENCES staff(ID),
11    FOREIGN KEY (CustomerID) REFERENCES customer(ID),
12    FOREIGN KEY (InventoryID) REFERENCES inventory(ID)
13 );

```

Setelah itu, saya membuat tabel *Rental* dengan atribut ID, StaffID, CustomerID, InventoryID, Rental_Date, Return_Date dan Last_Update dengan ketentuan hanya ID, StaffID, CustomerID dan InventoryID yang bersifat Not Null dan ID sebagai Primary Key sedangkan StaffID sebagai Foreign Key untuk tabel Staff, CustomerID sebagai Foreign Key untuk tabel Customer, InventoryID sebagai Foreign Key untuk tabel Inventory.

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan)

```

1 CREATE TABLE Payment(
2     ID int(10) NOT NULL,
3     RentalID int(10) NOT NULL,
4     CustomerID int(10) NOT NULL,
5     StaffID int(10) NOT NULL,
6     Amount numeric(19,0),
7     Payment_Date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
8     PRIMARY KEY (ID),
9     FOREIGN KEY (RentalID) REFERENCES rental(ID),
10    FOREIGN KEY (CustomerID) REFERENCES customer(ID),
11    FOREIGN KEY (StaffID) REFERENCES staff(ID)
12 );

```

Setelah itu, saya membuat tabel *Payment* dengan atribut ID, RentalID, CustomerID, StaffID, Amount dan Payment_Date dengan ketentuan hanya ID, RentalID, CustomerID dan StaffID yang bersifat Not Null dan ID sebagai Primary Key sedangkan RentalID sebagai Foreign Key untuk tabel Rental, CustomerID sebagai Foreign Key untuk tabel Customer, StaffID sebagai Foreign Key untuk tabel staff.

3. Lakukan latihan menghapus dan merubah atribut tabel.

Disini saya akan melakukan perubahan dengan melakukan merubah atribut di tabel Inventory dengan menambahkan not null di FilmID, address dengan menambahkan not null di CityID dan Store dengan menambahkan not null di AddressID.

✓ MySQL memberikan hasil kosong (atau nol baris).

```

1 ALTER TABLE inventory
2 MODIFY FilmID int(10) NOT NULL;

```

✓ MySQL memberikan hasil kosong (atau nol baris).

```

1 ALTER TABLE address
2 MODIFY CityID int(10) NOT NULL;

```

✓ MySQL memberikan hasil kosong (atau nol baris).

```

1 ALTER TABLE store
2 MODIFY AddressID int(10) NOT NULL;

```


Selanjutnya, saya akan coba menghapus salah satu atribut di tabel Actor yaitu Last_Name.

✓ MySQL memberikan hasil kosong

```
1 ALTER TABLE actor
2 DROP COLUMN Last_Name;
```

Lalu, saya akan menambahkan kembali kolom Last_Name setelah kolom First_Name di tabel actor.

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencar

```
1 ALTER TABLE actor
2 ADD Last_Name varchar(255) AFTER First_Name;
```