1. Tạo 2 bảng:  
   account: id, username, full\_name, phone, email, created\_date  
   address: id, address, created\_date  
   và tạo thêm khoá phụ để liên kết 2 bảng này ở dạng quan hệ 1 – 1

**CREATE** **TABLE** **ACCOUNT** (

id **INT** **PRIMARY** **KEY** **AUTO\_INCREMENT**,

username **VARCHAR** (50),

full\_name **VARCHAR** (50),

phone **INT** ,

email **VARCHAR** (50),

created\_date **DATETIME**

);

**CREATE** **TABLE** address (

id **INT** **PRIMARY** **KEY** **AUTO\_INCREMENT**,

address **VARCHAR** (50),

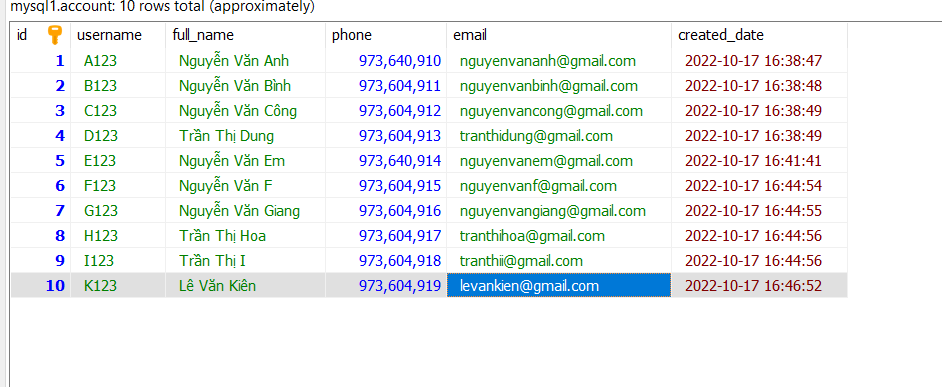
created\_date **DATETIME**,

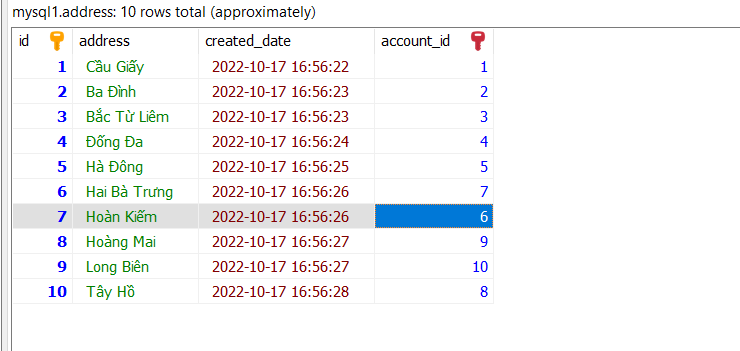
account\_id **INT** **UNIQUE**,

**FOREIGN** **KEY** (account\_id) **REFERENCES** **account**(id)

);

1. Thêm mỗi bảng 10 bản ghi

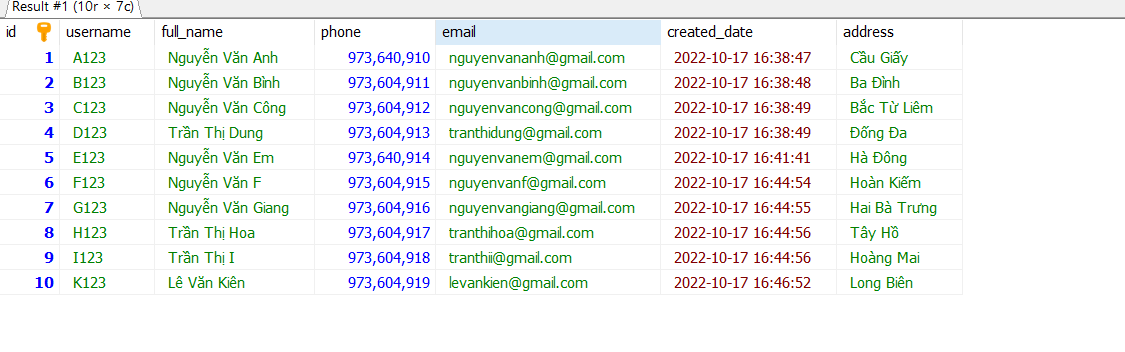




1. Viết câu lệnh truy vấn dữ liệu gồm những cột: id, username, full\_name, phone, email, created\_date, address

**SELECT** **account**.id, username, full\_name, phone, email, **account**.created\_date, address **FROM**

**account** **JOIN** address **ON** **account**.id = address.account\_id



1. Sau khi làm xong câu trên, xoá 2 bảng đó đi, tạo lại 2 bảng như câu 1, rồi tạo thêm khoá phụ để liên kết 2 bảng này ở dạng quan hệ 1 – N

**CREATE** **TABLE** **ACCOUNT** (

id **INT** **PRIMARY** **KEY** **AUTO\_INCREMENT**,

username **VARCHAR** (50),

full\_name **VARCHAR** (50),

phone **INT** ,

email **VARCHAR** (50),

created\_date **DATETIME**

);

**CREATE** **TABLE** address (

id **INT** **PRIMARY** **KEY** **AUTO\_INCREMENT**,

address **VARCHAR** (50),

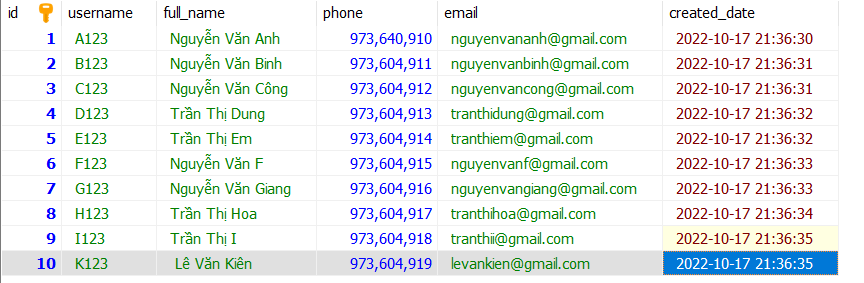
created\_date **DATETIME**,

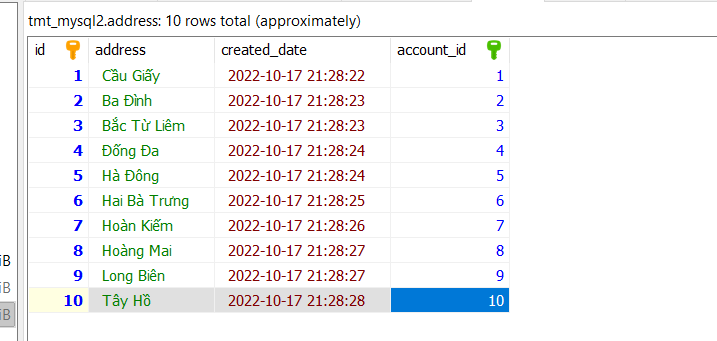
account\_id **INT** ,

**FOREIGN** **KEY** (account\_id) **REFERENCES** **account**(id)

);

1. . Yêu cầu giống câu 2

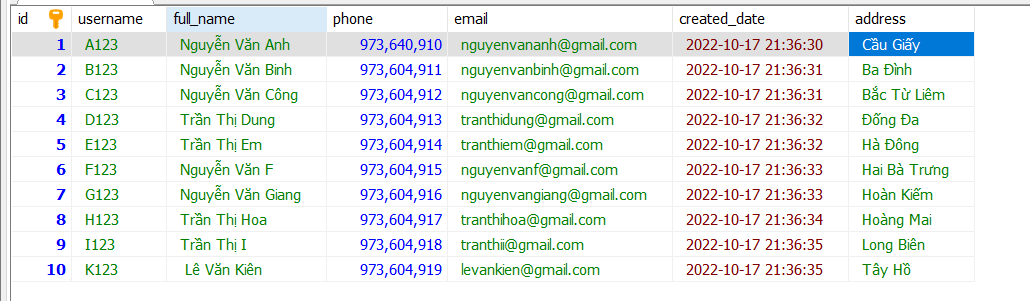




6 Yêu cầu giống câu 3

**SELECT** **account**.id, username, full\_name, phone, email, **account**.created\_date, address **FROM**

**account** **JOIN** address **ON** **account**.id = address.account\_id



7 Sau khi làm xong câu trên, xoá 2 bảng đó đi, tạo lại 2 bảng như câu 1, rồi tạo thêm khoá phụ để liên kết 2 bảng này ở dạng quan hệ N – N

**CREATE** **TABLE** **ACCOUNT** (

id **INT** **PRIMARY** **KEY** **AUTO\_INCREMENT**,

username **VARCHAR** (50),

full\_name **VARCHAR** (50),

phone **INT** ,

email **VARCHAR** (50),

created\_date **DATETIME**

);

**CREATE** **TABLE** address (

id **INT** **PRIMARY** **KEY** **AUTO\_INCREMENT**,

address **VARCHAR** (50),

created\_date **DATETIME**

);

**CREATE** **TABLE** account\_address(

account\_id **INT** ,

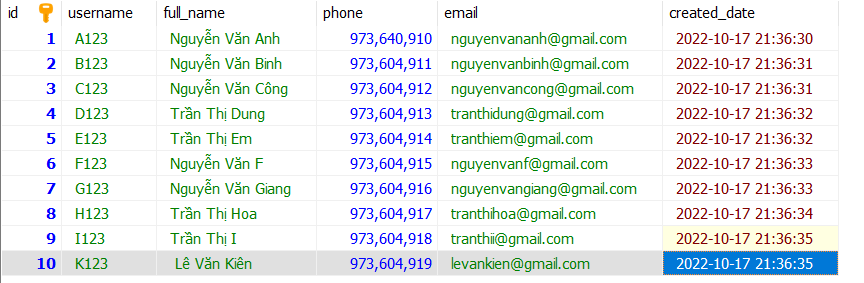
adress\_id **INT** ,

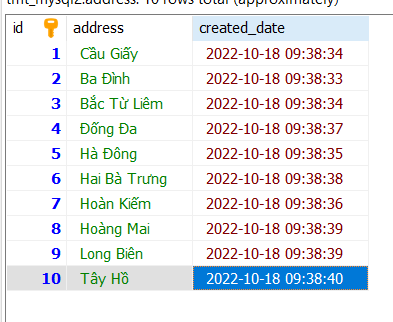
**FOREIGN** **KEY** (account\_id) **REFERENCES** **account**(id),

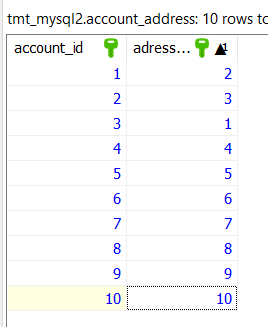
**FOREIGN** **KEY** (adress\_id) **REFERENCES** address(id)

);

8. Yêu cầu giống câu 2





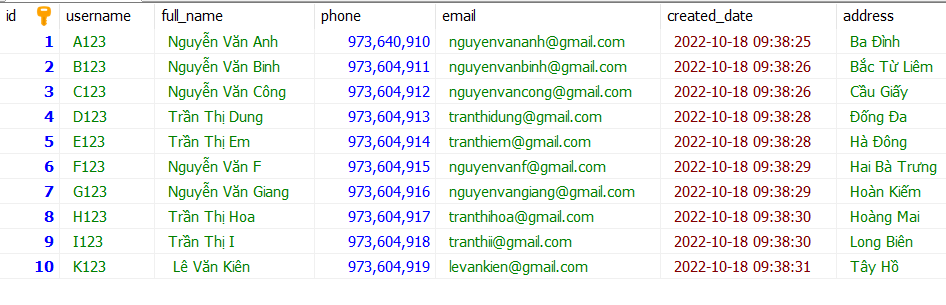


9. Giống yêu cầu 3

**SELECT** **account**.id, username, full\_name, phone, email, **account**.created\_date, address **FROM**

**account** **JOIN** account\_address **ON** **account**.id = account\_address.account\_id

**JOIN** address **ON** account\_address.adress\_id = address.id



10. Sau khi làm xong câu trên, xoá 2 bảng đó đi, tạo lại 2 bảng như câu 1 nhưng với điều kiện sử dùng CASCADE loại DELETE, UPDATE

**CREATE** **TABLE** **ACCOUNT** (

id **INT** **PRIMARY** **KEY** **AUTO\_INCREMENT**,

username **VARCHAR** (50),

full\_name **VARCHAR** (50),

phone **INT** ,

email **VARCHAR** (50),

created\_date **DATETIME**

);

**CREATE** **TABLE** address (

id **INT** **PRIMARY** **KEY** **AUTO\_INCREMENT**,

address **VARCHAR** (50),

created\_date **DATETIME**,

account\_id **INT** **UNIQUE**,

**FOREIGN** **KEY** (account\_id) **REFERENCES** **account**(id) **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE**

);

11 Viết câu lệnh sử dụng CASCADE để xoá dữ liệu bảng cha, và bảng con cũng sẽ bị xoá dữ liệu theo

**DELETE** **FROM** **account** **WHERE** username = 'A123'