

*Lập trình toàn năng*

*Training Assignments*

|  |  |
| --- | --- |
| **Program Code** | **DTN2503** |
| **Issue/Revision** |  |
| **Effective date** |  |
| **Author** | **Huu Nguyen Van** |

**Bài tập cuối học phần 3**

Để quản lý nhân viên, người ta xây dựng một cơ sở dữ liệu có sơ đồ quan hệ sau (trường có dấu gạch chân là PrimaryKey)

Country(country\_id, country\_name)

Location(location\_id, street\_address, postal\_code, country\_id)

Employee(employee\_id, full\_name, email, location\_id)

**Question 1:**

Tạo table với các ràng buộc và kiểu dữ liệu

Thêm ít nhất 3 bản ghi vào table

**Trả lời:**

CREATE TABLE Country (

country\_id INT PRIMARY KEY,

country\_name VARCHAR(100) NOT NULL

);

CREATE TABLE Location (

location\_id INT PRIMARY KEY,

street\_address VARCHAR(200),

postal\_code VARCHAR(20),

country\_id INT,

FOREIGN KEY (country\_id) REFERENCES Country(country\_id)

);

CREATE TABLE Employee (

employee\_id INT PRIMARY KEY,

full\_name VARCHAR(100),

email VARCHAR(100) UNIQUE,

location\_id INT,

FOREIGN KEY (location\_id) REFERENCES Location(location\_id)

ON DELETE SET NULL

);

INSERT INTO Country VALUES (1, 'Vietnam'), (2, 'Japan'), (3, 'USA');

INSERT INTO Location VALUES

(100, '1 Nguyen Trai', '70000', 1),

(200, '5 Shibuya', '15000', 2),

(300, '1600 Amphitheatre', '94043', 3);

INSERT INTO Employee VALUES

(1, 'Nguyen Nam', 'nn03@gmail.com', 100),

(2, 'Yamada Taro', 'yt01@gmail.com', 200),

(3, 'John Smith', 'js02@gmail.com', 300);

**Question 2:** Viết lệnh để

a) Lấy tất cả các nhân viên thuộc Việt nam

**Trả lời:**

SELECT e.\*

FROM Employee e

JOIN Location l ON e.location\_id = l.location\_id

JOIN Country c ON l.country\_id = c.country\_id

WHERE LOWER(c.country\_name) LIKE 'viet%nam';

b) Lấy ra tên quốc gia của employee có email là "[nn03@gmail.com](mailto:nn03@gmail.com)"

**Trả lời:**

SELECT c.country\_name

FROM Employee e

JOIN Location l ON e.location\_id = l.location\_id

JOIN Country c ON l.country\_id = c.country\_id

WHERE e.email LIKE 'nn03@gmail.com';

c) Thống kê mỗi country, mỗi location có bao nhiêu employee đang

làm việc.

**Trả lời**:

SELECT

c.country\_name,

l.location\_id,

COUNT(e.employee\_id) AS employee\_count

FROM Country c

JOIN Location l ON c.country\_id = l.country\_id

LEFT JOIN Employee e ON e.location\_id = l.location\_id

GROUP BY c.country\_name, l.location\_id;

**Question 3:**

Tạo trigger cho table Employee chỉ cho phép insert mỗi quốc gia có tối đa 10 employee

**Trả lời:**  
DELIMITER //

CREATE TRIGGER trg\_limit\_employee\_per\_country

BEFORE INSERT ON Employee

FOR EACH ROW

BEGIN

DECLARE emp\_count INT;

SELECT COUNT(e.employee\_id)

INTO emp\_count

FROM Employee e

JOIN Location l ON e.location\_id = l.location\_id

WHERE l.country\_id = (

SELECT country\_id FROM Location WHERE location\_id = NEW.location\_id

);

IF emp\_count >= 10 THEN

SIGNAL SQLSTATE '45000'

SET MESSAGE\_TEXT = 'Each country can have at most 10 employees.';

END IF;

END;

//

DELIMITER ;

**Question 4:**

Hãy cấu hình table sao cho khi xóa 1 location nào đó thì tất cả employee ở location đó sẽ có location\_id = null

**Trả lời:**

-- Bước 1: Kiểm tra constraint hiện tại (nếu có)

SELECT CONSTRAINT\_NAME

FROM information\_schema.KEY\_COLUMN\_USAGE

WHERE TABLE\_NAME = 'Employee'

AND COLUMN\_NAME = 'location\_id'

AND CONSTRAINT\_SCHEMA = DATABASE();

-- Bước 2: Xóa foreign key cũ nếu đã tồn tại

ALTER TABLE Employee

DROP FOREIGN KEY employee\_ibfk\_1;

-- Bước 3: Thêm lại foreign key với ON DELETE SET NULL

ALTER TABLE Employee

ADD CONSTRAINT fk\_emp\_location

FOREIGN KEY (location\_id) REFERENCES Location(location\_id)

ON DELETE SET NULL;