

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN TOÁN ỨNG DỤNG VÀ TIN HỌC



BÁO CÁO MÔN HỌC
CƠ SỞ DỮ LIỆU

ĐỀ TÀI:

TRÌNH BÀY NỘI DUNG THỰC HÀNH HÀNG TUẦN

Sinh viên thực hiện: *Khuất Văn Vương*

Mã số sinh viên: *20216906*

Giảng viên hướng dẫn: *Ths. Nguyễn Danh Tú*

Hà Nội, tháng 8 năm 2023

MỤC LỤC

Contents

TUẦN 1: THIẾT LẬP MÔI TRƯỜNG LÀM VIỆC MySQL	5
Bài 1: Thiết lập được hệ quản trị cơ sở dữ liệu trên máy tính cá nhân	5
Bài 2: Tạo được một cơ sở dữ liệu mẫu để làm việc.....	5
TUẦN 2: VIẾT CÂU LỆNH ĐẠI SỐ QUAN HỆ.....	7
Bài 1: Đưa ra tên và địa chỉ của tất cả các nhân viên làm việc cho đơn vị;.....	7
Bài 2: Với mỗi dự án có địa điểm tại Hà nội, hãy liệt kê mã số dự án, mã số của đơn vị kiêm soát, Tên, địa chỉ và ngày sinh của người quản lý đơn vị	7
Bài 3: Tìm tên của các nhân viên làm việc trên tất cả các dự án do đơn vị có mã số 5 kiêm soát	8
Bài 4: Tạo ra một danh sách các mã số dự án đối với các dự án có một nhân viên.....	9
Bài 5: Đưa ra tên của tất cả các nhân viên có nhiều hơn hoặc bằng 2 người phụ.....	10
Bài 6: Đưa ra các nhân viên không có người phụ thuộc	11
Bài 7: Đưa ra tên của những người quản lý có ít nhất là một người phụ thuộc	12
TUẦN 3: THỰC HÀNH TRUY VẤN SQL.....	14
Bài 1: SELECT, ORDER BY, WHERE, SELECT DISTINCT	14
Bài 2: AND, OR, IN.....	15
Bài 3: BETWEEN, LIKE, LIMIT	15
Bài 4: IS NULL, Table & Column Aliases	16
Bài 5: JOINS, INNER JOIN, LEFT JOIN, RIGHT JOIN, SELFJOIN,CROSS JOIN	18
Bài 6: GROUP BY, HAVING, ROLLUP.....	20
Bài 7: Subquery, Derived Tables, EXISTS.....	22
Bài 8: UNION, MINUS, INTERSECT	24
TUẦN 4.1: THỰC HÀNH TÌM HIỂU ĐƯỢC CẤU TRÚC.....	26
CỦA MỘT CƠ SỞ DỮ LIỆU	26
Bài 1: Xây dựng lược đồ (Diagram)	26
Bài 2: Truy vấn các bảng hệ thống trong cơ sở dữ liệu	27
TUẦN 4.2: THỰC HÀNH XÂY DỰNG ĐƯỢC CẤU TRÚC CỦA MỘT CƠ SỞ DỮ LIỆU.....	28
Bài 3: Thiết lập khung nhìn (Views).....	29
Bài 4: Thiết lập thủ tục và hàm (procedure & function)	29
TUẦN 5: THỰC HÀNH CÁC CÂU LỆNH THAO TÁC DỮ LIỆU (PHẦN MYSQL DATA MANIPULATION)	31
Bài 1: Thực hiện các câu lệnh cập nhật (insert,update, delete dữ liệu).....	31
Bài 2: Cập nhật dữ liệu từ 1 bảng excel qua sinh tự động SQL Script	31
Bài 3: Cập nhật dữ liệu từ một bảng khác.....	33

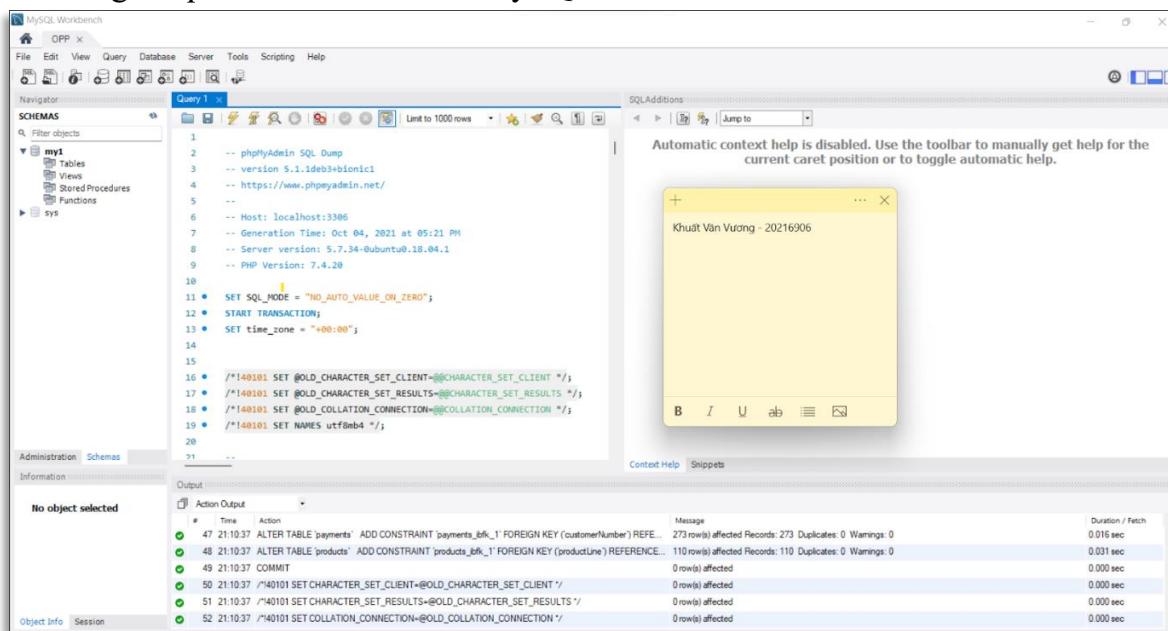
Bài 4: Cập nhật dữ liệu qua một procedure.....	33
TUẦN 6: THỰC HÀNH CÁC CÂU LỆNH SỬ DỤNG HÀM TRONG TRUY VẤN	35
Bài 1: Aggregate Functions, Math Functions, Comparison Functions	35
Bài 2: Control Flow Functions and Expressions	37
Bài 3: Window Functions	37
Bài 4: Date Functions, String Functions	38
Bài 5: Đánh chỉ mục dữ liệu một bảng	39
TUẦN 7: THỰC HÀNH THIẾT KẾ CƠ SỞ DỮ LIỆU LUU TRỮ ĐƠN HÀNG	40
Bài 1 + 2 : Xây dựng lược đồ R-E và E-R	40
Bài 3: Thiết kế trên hệ quản trị cơ sở dữ liệu	40
Bài 4: In diagram cơ sở dữ liệu.....	40
Bài 5: Viết câu lệnh thêm dữ liệu và bảng	41
Bài 6: Viết câu lệnh truy vấn Thay đổi cấu trúc CSDL	41
Bài 7: Thay đổi cấu trúc CSDL.....	42
TUẦN 8	43
Bài 1:.....	43
Bài 2:.....	44
Bài 3:.....	44
Bài 4:.....	44
Bài 5:.....	45
TUẦN 9: TRUY VẤN DỮ LIỆU TRÊN CƠ SỞ DỮ LIỆU OLTP	46
Bài 1: Truy vấn thông tin khách hàng phàn nàn	46
Bài 2: Truy vấn ra thông tin đơn hàng	46
Bài 3: Truy vấn nhân viên đã chăm sóc khách hàng đơn hàng này	47
Bài 4: Truy vấn thông tin sản phẩm bị phàn nàn	48
Bài 5: Kiểm tra kho hàng còn sản phẩm đó không	48
Bài 6: Đưa ra những dòng sản phẩm có cùng mức giá, chênh lệch giá nhỏ để tư vấn	49
Bài 7: Đưa ra những dòng xe có cùng một số đặc điểm với xe trước.....	50
Bài 8: Đưa ra những dòng xe có cùng một số đặc điểm với xe trước.....	50
Bài 9: Tìm 1 nhân viên đã có kinh nghiệm để tư vấn cho khách hàng	51
Bài 10: Hiển thị những khách hàng đã mua sản phẩm này để tiến hành khảo	52
Bài 11: Hiển thị top 5 khách hàng có tổng giá trị đơn hàng lớn nhất	53
Bài 12: Hiển thị top 5 sản phẩm có tỷ lệ doanh số cao nhất	54
Bài 13: Kiểm tra giao vận đã đúng thời gian yêu cầu hay chưa, hiển thị đơn	55
Bài 14: Đưa các sản phẩm không có mặt trong bất kỳ một đơn hàng nào.....	56
Bài 15: Đưa ra các sản phẩm có số lượng trong kho lớn hơn trung bình số	57
Bài 16: Thống kê tổng số lượng sản phẩm trong kho theo từng dòng sản.....	57

Bài 17: Thống kê ra mỗi sản phẩm được đặt hàng lần cuối vào thời gian nào và khách hàng đã đặt hàng.....	58
TUẦN 10: XÂY DỰNG DASHBOARD VÀ PHÂN TÍCH DỮ LIỆU TỪ CSDL OLTP	60
Bài 1: Thống kê doanh số của các nhân viên/quản lý theo phòng ban (cụ thể là	60
Bài 2: Thống kê doanh số theo các office	61
Bài 3: Thống kê doanh số theo nội địa tại các office	62
Bài 4: Thống kê top 5 những sản phẩm được khách hàng mua nhiều nhất theo khu.....	62
Bài 5: Xây dựng dashboard trên Excel/Power BI	63
TỔNG KẾT MÔN HỌC.....	64
Tài liệu tham khảo:	65

TUẦN 1: THIẾT LẬP MÔI TRƯỜNG LÀM VIỆC MySQL

Bài 1: Thiết lập được hệ quản trị cơ sở dữ liệu trên máy tính cá nhân

Em sử dụng hệ quản trị cơ sở dữ liệu MySQL.



Bài 2: Tạo được một cơ sở dữ liệu mẫu để làm việc

Mô tả cơ sở dữ liệu:

Cơ sở dữ liệu mẫu được dùng ở đây được download ở:

<https://www.mysqltutorial.org/mysql-sample-database.aspx/>

<https://drive.google.com/file/d/1TzqcgPRIyILZ4tdXJmWzZuPx1CxgvURj/view>

Cơ sở dữ liệu mẫu này có tên **classicmodels**. Bao gồm các bảng sau:

- Customers: Lưu trữ dữ liệu của khách hàng.
- Products: Lưu trữ kích cỡ các xe mẫu.
- ProductLines: Lưu trữ các danh mục dòng sản phẩm
- Orders: Lưu trữ các đơn bán hàng do khách hàng đặt
- OrderDetails: Lưu trữ các chi tiết của mỗi đơn hàng
- Payments: Lưu trữ các khoản thanh toán
- Employees: Lưu trữ tất cả thông tin của nhân viên
- Offices: Lưu trữ dữ liệu bán hàng

Input: Câu lệnh chọn bảng Customers.

Output: Bảng Customers từ cơ sở dữ liệu mẫu Classicmodels.

Mã nguồn:

USE classicmodels;

SELECT * FROM customers;

Giao diện kết quả:

The screenshot shows the MySQL Workbench interface with the following details:

- Server:** MySQL Server
- SCHEMAS:** classicmodels
- Table:** customers
- Query:** SELECT * FROM customers LIMIT 0, 1000
- Result Grid:** Displays the first 1000 rows of the customers table.
- Context Menu:** An open context menu is positioned over the row for customer ID 330, "Andrea Gruber". The menu includes options like "Copy Row", "Copy Column", "Copy Table", "Copy Structure", "Edit Row", "Edit Column", "Edit Structure", "Delete Row", and "Delete Column".
- Session Output:** Shows the SQL statements run and their execution details.

TUẦN 2: VIẾT CÂU LỆNH ĐẠI SỐ QUAN HỆ

Bài 1: Dưa ra tên và địa chỉ của tất cả các nhân viên làm việc cho đơn vị: Đại số quan hệ

$\Pi_{\text{TENNV}, \text{DIACHI}}(\text{NHANVIEN})$

Mã nguồn:

```
SELECT * FROM HONV, TENNV, DCHI  
FROM nhanvien
```

Giao diện kết quả:

HONV	TENNV	DCHI
Nguyen Diep	Anh	Cau Giay, Ha Noi
Tran Van	Anh	Hoang Mai, Ha Noi
Hoang Ngoc	Bich	Thuong Tin, Ha Noi
Nguyen Thi Kim	Dung	Thanh Xuan, Ha Noi
Pham Hong	Dung	Thuong Tin, Ha Noi
Vu Viet	Ha	Quan 2, HCM
Tran Ngoc	Ha	Tran Phu, Ha Noi
Do Manh	Hanh	Hoan Kiem, Ha Noi
Do Quoc	Hung	Ha Dong, Ha Noi
Lo Phuong	Lien	Thanh Xuan, Ha Noi
Nguyen Anh	Mei	Cau Giay, Ha Noi
Nguyen Hoang	Nam	Thuong Tin, Ha Noi
Tran Ho	Nguyen	Thuong Tin, Ha Noi
Trinh Ha	Phuong	Cau Giay, Ha Noi
Lo Manh	Tan	Thanh Xuan, Ha Noi
Tran Dieu	Thuy	Dong Da, Ha Noi

Bài 2: Với mỗi dự án có địa điểm tại Hà nội, hãy liệt kê mã số dự án, mã số của đơn vị kiểm soát, Tên, địa chỉ và ngày sinh của người quản lý đơn vị

Đại số quan hệ:

$\Pi_{\text{MADA}, \text{MADV}, \text{TENNV}, \text{DCHI}, \text{NS}}(\sigma_{\text{DCHI} = "HaNoi"}(\text{NHANVIEN} \bowtie \text{DONVI} \bowtie \text{DEAN}))$

Mã nguồn:

```
SELECT MADA, nhanvien.MADV, HONV, TENNV, DCHI, NS
FROM nhanvien, dean, donvi
WHERE DD_DA = 'Ha Noi' AND dean.MADV = donvi.MADV AND donvi.MANQ = nhanvien.MANV
```

Giao diện kết quả:

The screenshot shows the MySQL Workbench interface. In the top-left, the Navigator pane lists SCHEMAS (containing conpty), TABLES (containing dean, nhanvien, nv_dean, donvi, and others), and other database objects. A central Query Editor window contains the following SQL code:

```
1 select MADA, nhanvien.MADV, HONV, TENNV, DCHI, NS
2 from nhanvien, dean, donvi
3 where DD_DA='Ha Noi' and dean.MADV=donvi.MADV and donvi.MANQ=nhanvien.MANV
```

The Result Grid shows the output of the query:

MADA	MADV	HONV	TENNV	DCHI	NS
DA1	DV1	Nguyen Dang Anh	Cau Giay, Ha Noi	10/6/1999	
DA4	DV4	Nguyen Thi Kim Dung	Thanh Xuan, Ha Noi	10/9/1999	

Below the result grid, the Action Output pane displays the following log entries:

- # 68 16:27:45 selected HONV, TENNV, DCHI from nhanvien LIMIT 0, 1000 Message 20 row(s) returned Duration / Fetch 0.000 sec / 0.000 sec
- # 69 16:33:05 select MADA, nhanvien.MADV, HONV, TENNV, DCHI, NS from nhanvien, dean, donvi where DD_DA='Ha Noi' 2 row(s) returned Duration / Fetch 0.000 sec / 0.000 sec

Bài 3: Tìm tên của các nhân viên làm việc trên tất cả các dự án do đơn vị có mã số 5 kiểm soát**Đại số quan hệ:**

$$\prod_{TENNV} (\sigma_{MANV = "5"} (DONVI \bowtie NHANVIEN))$$

Mã nguồn:

```
SELECT HONV, TENNV
FROM nhanvien, nv_dean, dean
WHERE dean.MADV = '5' AND nhanvien.MADV = nv_dean.MADV
      AND dean.MADA = nv_dean.MADA
```

Giao diện kết quả:

The screenshot shows the MySQL Workbench interface. In the top-left, the Navigator pane displays the schema 'congty' with tables like 'dean', 'donvi', 'nv_dean', and 'nhanvien'. The top-right pane shows the SQL editor with the following query:

```
1 select HONV, TENNV
2 from nhanvien, nv_dean
3 where dean.MADV='DV5' and nhanvien.MANV=nv_dean.MANV and dean.MADA=nv_dean.MADA;
```

The bottom-right pane shows the results grid with the following data:

HONV	TENNV
Pham Hong	Dang
Nguyen Anh	Mai
Tran Dieu	Thuy

Bài 4: Tạo ra một danh sách các mã số dự án đối với các dự án có một nhân viên hoặc một người quản lý đơn vị kiểm soát dự án có tên là ‘Nam’

Đại số quan hệ:

$$\begin{aligned} A &= \prod_{TENNV, MADA} (\sigma_{MANV = MANV(MANQL(NHANVIEN \bowtie DONVI \bowtie DEAN))}) \\ B &= \prod_{TENNV, MADA} (NHANVIEN \bowtie DEAN) \end{aligned}$$

Mã nguồn:

```
SELECT MADA
FROM nhanvien, nv_dean
WHERE nhanvien.MANV = nv_dean.MANV
AND TENNV = "NAM"
```

Giao diện kết quả:

```

MySQL Workbench
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
    comify
        Tables
        dean
        donvi
        donvi_did
        nhavien
        nv_dean
        thanhan
        Views
        Stored Procedures
        Functions
        sys
        world
Administration Schemas Information
No object selected
Object Info Session
Result 1 x
Output
Action Output
# Time Action Message Duration / Fetch
1 12:11:31 SELECT MADA FROM nhanvien, nv_dean WHERE nhanvien.MANV=nv_dean.MANV AND TENNV="NAM" L... 1 rows(s) returned 0.000 sec / 0.000 sec

```

Bài 5: Đưa ra tên của tất cả các nhân viên có nhiều hơn hoặc bằng 2 người phụ thuộc

Đại số quan hệ:

$$\begin{aligned}
 A &= \sum_{MANV, COUNT} (TENNV) \geq 2 \\
 B &= A \bowtie_{MANV = MANV} (NHANVIEN) \\
 KQ &= \prod_{TENNV} (B)
 \end{aligned}$$

Mã nguồn:

```

SELECT thannhan.MANV, TENNV
FROM nhanvien, thannhan
WHERE nhanvien.MANV = thannhan.MANV
GROUP BY thannhan.MANV, TENNV
HAVING count(thannhan.MANV)>=2

```

Giao diện kết quả:

The screenshot shows the MySQL Workbench interface. In the top-left, the Navigator pane displays the schema 'congty' with its tables: dean, donvi, donvi_dd, nhanvien, nv_danh_sach, and thanhnhans. The 'nhanvien' table is selected. The top-right pane shows the SQL editor with the following query:

```

1 select thanhhan.MANV, TENNV
2   from nhanvien, thanhhan
3  where nhanvien.MANV=thanhhan.MANV
4 group by thanhhan.MANV, TENNV
5 having count(thanhhan.MANV)>=2;
    
```

The bottom-right pane shows the results in a grid:

MANV	TENNV
NV.001	Arh
NV.005	Dung
NV.014	Mai
NV.016	Trung

Bài 6: Đưa ra các nhân viên không có người phụ thuộc

Đại số quan hệ:

$$\Pi_{MANV}(NHANVIEN) - \Pi_{MANV}(THANNHAN)$$

Mã nguồn:

```

SELECT nhanvien.MANV, TENNV
FROM nhanvien
LEFT JOIN thanhhan ON nhanvien.MANV = thanhhan.MANV
GROUP BY nhanvien.MANV, TENNV
HAVING count(thanhhan.MANV)=0
    
```

Giao diện kết quả:

```

MySQL Workbench
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Schemas
congy
Tables
Columns
Indexes
Foreign Keys
Triggers
nv_danh
thanhhan
Columns
MANV
TENNV
NS
GT
QUANHE
Index
Foreign Keys
Administration Schemas
Information
Schema: congy
Result Grid Filter Rows Export Wrap Cell Content
MANV TENNV
NV.013 Phuong
Result Grid Form Editor
Field Types
Query Plan
Read Only Context Help Snippets
Object Info Session
Result 8
Output
Action Output
# Time Action Message Duration / Fetch
42 21:36:00 selected thanhhan.MANV, TENNV from nhanvien left join thanhhan on nhanvien.MANV=thanhhan.MANV group... 1 row(s) returned 0.000 sec / 0.000 sec
43 21:36:33 selected nhanvien.MANV, TENNV from nhanvien left join thanhhan on nhanvien.MANV=thanhhan.MANV grou... 1 row(s) returned 0.000 sec / 0.000 sec

```

Bài 7: Đưa ra tên của những người quản lý có ít nhất là một người phụ thuộc

Đại số quan hệ:

$$\begin{aligned}
 A &= \prod_{TENNV} (\sigma_{MANV = MANQL(NHANVIEN)}) \\
 KQ &= \prod_{TENNV} (A \bowtie THANHAN)
 \end{aligned}$$

Mã nguồn:

```

SELECT nhanvien.MANV, TENNV
FROM nhanvien, thanhhan, donvi
WHERE nhanvien.MANV = thanhhan.MANV AND donvi.MANQL = nhanvien.MANV
GROUP BY nhanvien.MANV, TENNV
HAVING count(thanhhan.MANV)>=1
    
```

Giao diện kết quả:

The screenshot shows the MySQL Workbench interface. In the top-left, the Navigator pane displays the schema structure of the 'congty' database, including tables like 'thannhan', 'nhanvien', and 'donvi'. The top-right pane shows the SQL Editor with the following query:

```
1 select nhanvien.MANV, TENNV
2   from nhanvien, thannhan, donvi
3  where nhanvien.MANV=thannhan.MANV and donvi.MANQ=nhanvien.MANV
4 group by nhanvien.MANV, TENNV
5 having count(thannhan.MANV)>1;
```

The bottom-left pane shows the Result Grid with the following data:

MANV	TENNV
NV.001	Anh
NV.002	Anh
NV.005	Dung
NV.004	Dang
NV.007	Ha

The bottom-right pane shows the Output panel with the following log entries:

Action	Time	Action	Message	Duration / Fetch
43	21:36:33	select nhanvien.MANV, TENNV from nhanvien left join thannhan on nhanvien.MANV=thannhan.MANV group by nhanvien.MANV, TENNV	1 row(s) returned	0.000 sec / 0.000 sec
44	21:39:26	select nhanvien.MANV, TENNV from nhanvien, thannhan, donvi where nhanvien.MANV=thannhan.MANV and	5 row(s) returned	0.000 sec / 0.000 sec

TUẦN 3: THỰC HÀNH TRUY VẤN SQL

Bài 1: SELECT, ORDER BY, WHERE, SELECT DISTINCT

- SELECT: Cho phép bạn chọn dữ liệu từ một hoặc nhiều bản.
- ORDER BY: Sắp xếp lại các kết quả tìm được.
- WHERE: Thêm điều kiện tìm kiếm cho các kết quả được trả về.
- SELECT DISTINCT: Loại bỏ các kết quả trùng lặp.

Input: Các câu lệnh sử dụng SELECT, ORDER BY, WHERE, SELECT DISTINCT.

Output: Từ cơ sở dữ liệu Classicmodels, chọn ra các kết quả không trùng lặp từ các trường state, city, country trong bảng Customers với điều kiện state không để trống và sắp xếp theo state.

Mã nguồn:

```
USE classicmodels;

SELECT DISTINCT state, city, country, FROM customers;

WHERE state IS NOT NULL

ORDER BY state
```

Giao diện kết quả:

The screenshot shows the MySQL Workbench interface. In the top-left pane, the Navigator displays the schema 'classicmodels' with tables like customers, employees, and offices. In the central pane, a SQL Editor window contains the following code:

```
1 USE classicmodels;
2 SELECT DISTINCT state, city, country FROM customers;
3 WHERE state IS NOT NULL
4 ORDER BY state
```

Below the SQL editor is a Results grid showing the output of the query. The data consists of three columns: state, city, and country. The results are as follows:

state	city	country
BC	Vancouver	Canada
CA	Toronto	USA
CA	Edmonton	USA
CA	Calgary	USA
CA	Montreal	USA
CA	Quebec	USA
CA	Saskatoon	USA
CA	Ottawa	USA
CA	Victoria	USA
CA	Saint John	USA
CA	St. John's	USA
CA	Calgary	UK
CA	Guelph	UK
CA	New Haven	UK
CA	Dartmouth	UK
CA	Montreal	UK
CA	Victoria	UK
CA	Saint John	UK
CA	Cork	UK
CA	Edmonton	UK
CA	Montreal	UK
CA	Victoria	UK
CA	Saint John	UK

Bài 2: AND, OR, IN

AND: cho phép tồn tại nhiều điều kiện trong câu lệnh.

OR: dùng để kết hợp nhiều điều kiện trong một câu lệnh.

IN: Khai báo một tập hợp các giá trị, nếu dữ liệu cần tìm trùng khớp với giá trị khai báo thì hợp lệ.

Input: câu lệnh có sử dụng AND, OR, IN

Output: Từ cơ sở dữ liệu Classicmodels, chọn ra các kết quả từ trường officeCode, city, phone, country, state có trong bảng Offices, với điều kiện Country là USA hoặc Japan và officeCode < 6.

Mã nguồn:

```
USE classicmodels;
SELECT officeCode, city, phone, country, state FROM offices,
WHERE country IN ('USA' OR 'JAPAN') AND officeCode < 6
```

Giao diện kết quả:

officeCode	city	phone	country	state
CA	Montgomery	+1 500 234 4782	USA	Alabama
US	Boston	+1 212 555 0222	USA	MA
CA	Seattle	+1 206 555 1234	USA	WA
CA	Tokyo	+81 33 224 3204	Japan	Chiyoda-ku
CA	London	+44 10 123 4567	UK	

Bài 3: BETWEEN, LIKE, LIMIT

BETWEEN: Kiểm tra xem một giá trị có nằm trong phạm vi không.

LIKE: truy xuất dữ liệu dựa theo quy tắc tìm kiếm.

LIMIT: hạn chế số lượng kết quả tìm được.

Input: các câu lệnh chứa BETWEEN, LIKE, LIMIT.

Output: Từ cơ sở dữ liệu classicmodels, chọn ra các kết quả từ trường customerNumber, customerName, creditLimit, postalCode trong bảng customers với điều kiện customerName và creditLimit trong khoảng từ 100000 đến 400000

Mã nguồn:

```
USE classicmodels;

SELECT customerNumber, customerName, creditLimit, postalCode FROM customers,
WHERE customerName LIKE '%a%' AND (creditLimit BETWEEN 100000.00 AND
400000.00)

ORDER BY customerNumber LIMIT 200
```

Giao diện kết quả:

customerNumber	customerName	creditLimit	postalCode
114	Australian Collector's	30394	30094
119	Le Rochele Offs.	118206.03	44000
124	Lundberg Events	109000.00	33020
141	Euro-e-Shipping Channel	27600.03	28034
142	Blue Chip	100000.00	33024
149	Dragon Supplies Ltd	20200.03	29993
151	Huise Machine Int'l	13930.03	45032
153	Diamond Import/Export	13930.03	33029
157	AV Stores, Inc.	13000.03	33237
160	Lebanese Perfumehouse	10000.03	33000
174			
175			
176			
177			
178			
179			
180			
181			
182			
183			
184			
185			
186			
187			
188			
189			
190			
191			
192			
193			
194			
195			
196			
197			
198			
199			
200			

Bài 4: IS NULL, Table & Column Aliases

IS NULL: Kiểm tra xem giá trị có phải NULL không

ALIAS: Dùng để định danh tên mới cho cột hoặc bảng.

Input: câu lệnh sử dụng IS NULL và ALIAS

Output: Từ cơ sở dữ liệu classicmodels, chọn ra các kết quả từ trường contactFirstName, contactLastName, state từ bảng customers với điều kiện state khác NULL.

Mã nguồn:

```
USE经典models;

SELECT contactFirstName, contactLastName, state
FROM customers
WHERE state IS NULL
```

```
USE经典models;

SELECT concat_ws(', ', contactLastName, contactFirstName) `contactFullName`
FROM customers
```

Giao diện kết quả:

contactId	contactName	state
1	Carmen	
2	Lorraine	
3	Peter	
4	Roland	
5	Doris	
6	Christina	
7	Bengt	
8	Helmer	
9	Hilary	

contactId	contactName	contactFullName
1	Carmen	Carmen
2	Lorraine	Lorraine
3	Peter	Peter
4	Roland	Roland
5	Doris	Doris
6	Christina	Christina
7	Bengt	Bengt
8	Helmer	Helmer
9	Hilary	Hilary

Bài 5: JOINS, INNER JOIN, LEFT JOIN, RIGHT JOIN, SELF JOIN,CROSS JOIN

INNER JOIN: Ghép hàng trong một bảng với mọi hàng trong bảng khác.

LEFT JOIN: Chọn dữ liệu từ bảng trái ghép với mọi hàng trong bảng phải.

RIGHT JOIN: Chọn dữ liệu từ bảng phải ghép với mọi hàng trong bảng trái.

SELF JOIN: Dùng để nối một bảng với chính nó.

Input: Các câu lệnh chứa mệnh đề JOIN

Mã nguồn:

```
USE classicmodels;  
SELECT customerNumber, customerName, phone FROM customers  
CROSS JOIN orderdetails
```

```
USE classicmodels;  
SELECT orderNumber, orderDate, requiredDate, shippedDate FROM orders  
INNER JOIN orderdetails USING (orderNumber)  
INNER JOIN products USING (productCode)  
ORDER BY orderNumber
```

```
USE classicmodels;  
SELECT customerNumber, customerName, phone FROM customers  
LEFT JOIN orders USING (customerNumber);
```

```
USE classicmodels;  
SELECT customerNumber, customerName, phone  
FROM customers  
RIGHT JOIN orders USING (customerNumber);
```

Giao diện kết quả:

MySQL Workbench

```

USE classmatebd;
SELECT customerNumber, customerName, phone
FROM customers
CROSS JOIN orderdetails;

```

Result Grid

customerNumber	customerName	phone
1	Alfreds Futterkiste	(069) 123-4567
2	Blumerhandel	(061) 123-4567
3	Caique Importador	(021) 123-4567
4	Edsbyn Kärrbräcka Begrif	(020) 123-4567
5	Faber-Landtmann	(021) 123-4567
6	Frans Gustafsson	(020) 123-4567
7	Hilbertssons AB	(020) 123-4567
8	Kärrbräcka Kärrs	(020) 123-4567
9	Luftfrachtspedition	(020) 123-4567
10	Megha International Trade	(020) 123-4567
11	Monika's Market	(020) 123-4567
12	North Mountain Brewing Company	(020) 123-4567
13	Northway Traders Ltd.	(020) 123-4567
14	Obione	(020) 123-4567
15	Oncore Marketing Inc.	(020) 123-4567
16	Paikka Oy	(020) 123-4567
17	Rheinhausen Versandhandel	(020) 123-4567
18	Spicy Times	(020) 123-4567
19	Tekirova Ticaret ve Sanayi A.S.	(020) 123-4567
20	Ullensvang Bergverk	(020) 123-4567
21	Valko Oy	(020) 123-4567
22	Wartburg-Schlachterei	(020) 123-4567
23	Wessels Discount Hypermart	(020) 123-4567
24	Wistow Furniture	(020) 123-4567
25	Zeyneli Gıda Sanayi ve Ticaret A.S.	(020) 123-4567

Action Output

```

Time: 0.000 sec
0 rows affected
Duration / Fetch: 0.000 sec / 0.000 sec

```

MySQL Workbench

```

USE classmatebd;
SELECT
    customers.customerNumber,
    customers.customerName,
    customers.phone,
    orderNumber,
    orderDate,
    requiredDate,
    shipVia,
    shipAddress
FROM
    customers
INNER JOIN
    orders
ON
    customers.customerNumber = orders.orderNumber
INNER JOIN
    orderdetails
ON
    orders.orderNumber = orderdetails.orderNumber
INNER JOIN
    products
ON
    orderdetails.productCode = products.productCode
ORDER BY
    orderNumber;

```

Result Grid

customerNumber	customerName	phone	orderNumber	orderDate	requiredDate	shipVia	shipAddress
1	Alfreds Futterkiste	(069) 123-4567	10000	2023-01-09	2023-01-10	1	Obere Str. 57, 8053 Zürich, Switzerland
2	Blumerhandel	(061) 123-4567	10001	2023-01-09	2023-01-10	1	Obere Str. 57, 8053 Zürich, Switzerland
3	Caique Importador	(021) 123-4567	10002	2023-01-09	2023-01-10	1	Obere Str. 57, 8053 Zürich, Switzerland
4	Edsbyn Kärrbräcka Begrif	(020) 123-4567	10003	2023-01-09	2023-01-10	1	Obere Str. 57, 8053 Zürich, Switzerland
5	Faber-Landtmann	(021) 123-4567	10004	2023-01-09	2023-01-10	1	Obere Str. 57, 8053 Zürich, Switzerland
6	Frans Gustafsson	(020) 123-4567	10005	2023-01-09	2023-01-10	1	Obere Str. 57, 8053 Zürich, Switzerland
7	Hilbertssons AB	(020) 123-4567	10006	2023-01-09	2023-01-10	1	Obere Str. 57, 8053 Zürich, Switzerland
8	Luftfrachtspedition	(020) 123-4567	10007	2023-01-09	2023-01-10	1	Obere Str. 57, 8053 Zürich, Switzerland
9	Monika's Market	(020) 123-4567	10008	2023-01-09	2023-01-10	1	Obere Str. 57, 8053 Zürich, Switzerland
10	Obione	(020) 123-4567	10009	2023-01-09	2023-01-10	1	Obere Str. 57, 8053 Zürich, Switzerland
11	Oncore Marketing Inc.	(020) 123-4567	10010	2023-01-09	2023-01-10	1	Obere Str. 57, 8053 Zürich, Switzerland
12	Paikka Oy	(020) 123-4567	10011	2023-01-09	2023-01-10	1	Obere Str. 57, 8053 Zürich, Switzerland
13	Rheinhausen Versandhandel	(020) 123-4567	10012	2023-01-09	2023-01-10	1	Obere Str. 57, 8053 Zürich, Switzerland
14	Spicy Times	(020) 123-4567	10013	2023-01-09	2023-01-10	1	Obere Str. 57, 8053 Zürich, Switzerland
15	Ullensvang Bergverk	(020) 123-4567	10014	2023-01-09	2023-01-10	1	Obere Str. 57, 8053 Zürich, Switzerland
16	Wartburg-Schlachterei	(020) 123-4567	10015	2023-01-09	2023-01-10	1	Obere Str. 57, 8053 Zürich, Switzerland
17	Wessels Discount Hypermart	(020) 123-4567	10016	2023-01-09	2023-01-10	1	Obere Str. 57, 8053 Zürich, Switzerland
18	Zeyneli Gıda Sanayi ve Ticaret A.S.	(020) 123-4567	10017	2023-01-09	2023-01-10	1	Obere Str. 57, 8053 Zürich, Switzerland

Action Output

```

Time: 0.000 sec
0 rows affected
Duration / Fetch: 0.000 sec / 0.000 sec

```

MySQL Workbench

```

USE classmatebd;
SELECT
    customers.customerNumber,
    customers.customerName,
    phone
FROM
    customers
LEFT JOIN
    orders
USING
    (customerNumber);

```

Result Grid

customerNumber	customerName	phone
1	Alfreds Futterkiste	(069) 123-4567
2	Blumerhandel	(061) 123-4567
3	Caique Importador	(021) 123-4567
4	Edsbyn Kärrbräcka Begrif	(020) 123-4567
5	Faber-Landtmann	(021) 123-4567
6	Frans Gustafsson	(020) 123-4567
7	Hilbertssons AB	(020) 123-4567
8	Luftfrachtspedition	(020) 123-4567
9	Monika's Market	(020) 123-4567
10	Obione	(020) 123-4567
11	Oncore Marketing Inc.	(020) 123-4567
12	Paikka Oy	(020) 123-4567
13	Rheinhausen Versandhandel	(020) 123-4567
14	Spicy Times	(020) 123-4567
15	Ullensvang Bergverk	(020) 123-4567
16	Wartburg-Schlachterei	(020) 123-4567
17	Wessels Discount Hypermart	(020) 123-4567
18	Zeyneli Gıda Sanayi ve Ticaret A.S.	(020) 123-4567

Action Output

```

Time: 0.000 sec
0 rows affected
Duration / Fetch: 0.000 sec / 0.000 sec

```

```

MySQL Workbench
File Edit View Query Database Server Tools Script Help
File File Editor Database Server Tools Script Help
SELECT customerNumber, customerName, phone
FROM customers
RIGHT JOIN orders USING (customerNumber);

```

customerNumber	customerName	phone
103	Asterias Graphics	46.32.2255
103	Asterias Graphics	46.32.2255
103	Asterias Graphics	46.32.2255
110	Excalibur Gifts	70.70.55.8808
112	Sigmar Gift Stores	70.70.55.8808
114	Digital Gear	70.70.55.8802
124	Australian Collectors Co.	03.9326.4555
124	Australian Collectors Co.	03.9326.4555
124	Australian Collectors Co.	03.9326.4555
124	Australian Collectors Co.	03.9326.4555
124	Australian Collectors Co.	03.9326.4555
129	Laithwaite Gifts	46.47.8555
130	Exotic Pets Imports	07.09.9553

Result 1 of 1

Object Output

- Time
- 1 17:45:45 USE classicmodels;
- 2 17:45:45 SELECT customerNumber, customerName, phone FROM customers LEFT JOIN orders USING (customerNumber) LIMIT 0, 1000
- 3 17:50:20 USE classicmodels
- 4 17:50:20 SELECT customerNumber, customerName, phone FROM customers RIGHT JOIN orders USING (customerNumber) LIMIT 0, 1000

Message Duration / Fetch
0 rows(s) returned 0.000 sec / 0.000 sec
0 rows(s) affected 0.000 sec
0 rows(s) returned 0.000 sec / 0.000 sec

```

MySQL Workbench
File Edit View Query Database Server Tools Script Help
File File Editor Database Server Tools Script Help
USE classicmodels;
SELECT city, c1.country, c1.salesRepEmployeeNumber
FROM customers c1
INNER JOIN customers c2 ON c1.city = c2.city
ORDER BY salesRepEmployeeNumber;

```

city	country	salesRepEmployeeNumber
Wiesbaden	Germany	0000
Frankfurt	Germany	0000
Hamburg	Germany	0000
Munich	Germany	0000
Paris	France	0000
Lyon	France	0000
Singapore	Singapore	0000
Malta	Portugal	0000
Singapore	Singapore	0000
Copenhagen	Denmark	0000
London	United Kingdom	0000
Paris	France	0000
Padua	Spain	0000

Result 1 of 1

Object Output

- Time
- 1 10:01:20 USE classicmodels
- 2 10:01:20 SELECT city, c1.country, c1.salesRepEmployeeNumber FROM customers c1 INNER JOIN customers c2 ON c1.city = c2.city ORDER BY salesRepEmployeeNumber

Message Duration / Fetch
0 rows(s) returned 0.000 sec / 0.000 sec
0 rows(s) affected 0.000 sec
0 rows(s) returned 0.000 sec / 0.000 sec

Bài 6: GROUP BY, HAVING, ROLLUP

- Group By: Hợp các câu lệnh để sắp xếp dữ liệu giống nhau thành một nhóm.
- Having: Chỉ định điều kiện lọc mà kết quả nhóm xuất hiện trong kết quả.

Mã nguồn:

```

USE classicmodels;

SELECT city, COUNT(customerName) AS total_customers FROM customers
GROUP BY city
ORDER BY city

```

```
USE classicmodels;

SELECT city, COUNT(customerName) AS total_customers FROM customers
GROUP BY city WITH ROLLUP
HAVING count(customerName) >=3
```

Giao diện kết quả:

city	total_customers
Auckland	3
Bogota	2
Brasilia	2
Cambridge	2
Kuala Lumpur	2
Lima	2
London	2
Mexico City	2
Nairobi	2
New York	2
Nicosia	2
NYC	2
Paris	3
Singapore	2
total	17

city	total_customers
Auckland	3
Bogota	2
Brasilia	2
Cambridge	2
Kuala Lumpur	2
Lima	2
London	2
Mexico City	2
Nairobi	2
New York	2
Nicosia	2
NYC	2
Paris	3
Singapore	2
total	17

Bài 7: Subquery, Derived Tables, EXISTS

- Subquery: truy vấn lồng vào bên trong mệnh đề Where.
- Derived Tables: Bảng được trả về từ câu lệnh Select.
- Exists: Kiểm tra sự tồn tại.

Mã nguồn:

```
USE classicmodels;

SELECT customerNumber, customerName, creditLimit,
       (CASE
            WHEN creditLimit < 10000 THEN 'Bronze'
            WHEN creditLimit BETWEEN 10000 AND 100000 THEN 'Silver'
            WHEN creditLimit > 100000 THEN 'Gold'
        END) CustomerGroup
FROM customers GROUP BY customerName
```

```
USE classicmodels;

SELECT customerNumber, checkNumber, amount FROM payments
WHERE EXISTS
      (SELECT * FROM payments WHERE customerNumber > 400 )
```

```
USE classicmodels;

SELECT customerNumber, checkNumber, amount FROM payments
WHERE amount = (SELECT MAX (amount) FROM payments)
```

Giao diện kết quả:

MySQL Workbench screenshot showing a query results grid. The query uses a CASE statement to categorize customers based on credit limit into 'Bronze', 'Silver', and 'Gold'. The results show customer numbers, names, and their assigned group.

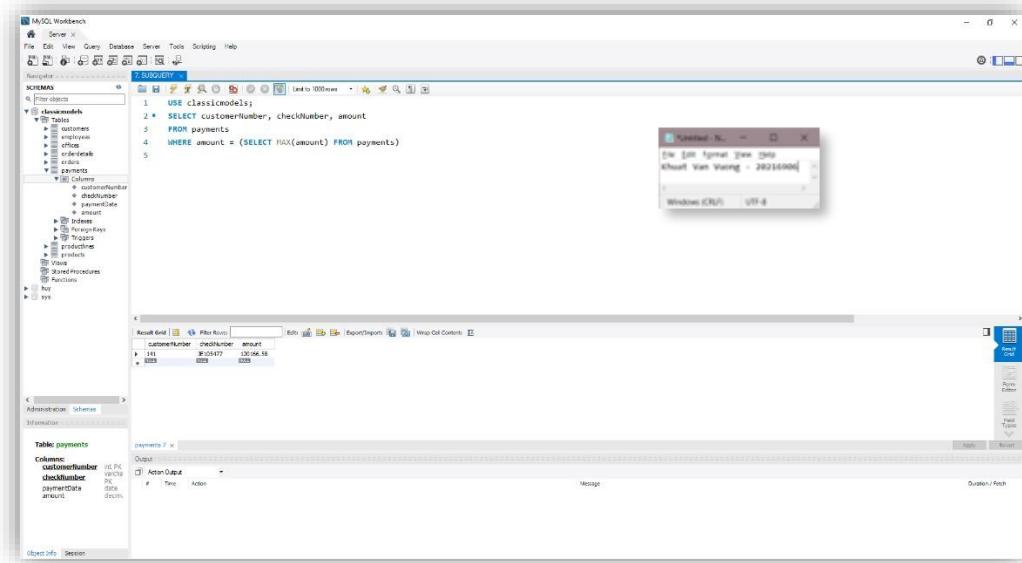
```

USE classicmodels;
SELECT customerNumber, customerName, creditLimit,
       (CASE
        WHEN creditLimit < 10000 THEN 'Bronze'
        WHEN creditLimit BETWEEN 10000 AND 100000 THEN 'Silver'
        WHEN creditLimit > 100000 THEN 'Gold'
       END ) CustomerGroup
FROM customers
GROUP BY customerName
  
```

MySQL Workbench screenshot showing a query results grid. The query uses a subquery in a WHERE EXISTS clause to find customers who have made payments above a certain amount. The results list customer numbers, check numbers, and amounts.

```

USE classicmodels;
SELECT customerNumber, checkNumber, amount
FROM payments
WHERE EXISTS ( SELECT * FROM payments WHERE customerNumber > 400 )
  
```



Bài 8: UNION, MINUS, INTERSECT

- UNION: kết hợp các kết quả của hai hoặc nhiều câu lệnh SELECT mà không trả về trùng lặp.
- MINUS: loại bỏ kết quả thu được đầu tiên khỏi tập kết quả thu được thứ hai.
- INTERSECT: Chỉ trả về các hàng chung được trả về bởi câu lệnh SELECT

Mã nguồn:

```

USE classicmodels;

SELECT DISTINCT customerName, contactFirstName FROM customers
INNER JOIN payments USING (customerNumber)

```

```

USE classicmodels;

SELECT customerName, contactFirstName FROM customers
LEFT JOIN orders USING (CustomerNumber)
WHERE status IS NOT NULL;

```

Giao diện kết quả:

```

USE classicmodels;
SELECT DISTINCT customerName, contactFirstName
FROM customers
INNER JOIN payments USING (customerNumber);
    
```

customerName	contactFirstName
Sign Off Stores	Dean
Austrian Collector's Co.	Peter
Uncle Bob's Books	George
Dante's Inferno	Jane
Han's Import	John
Her Of The Islands Ltd.	Sabu
Horizon Hobby	Albert
Hot Wheels Co.	Julie
Lam's Toys Co.	Paul
Par's Toy Channel	Thomas
Valve Hotel Supplies, Co.	Christina
David's Bazaar	John
Savvy & Savvy, Co.	Kathy
Dragon Source, Ltd.	Eric
Horizon Hobby	Jeff
Decor Classics Inc.	Karen
Technix Stores Inc.	Andy
Horizon Hobby	Vivian
Horizon Hobby Imports	Marie
Le Grand Bonheur, Co. HK	Mark

```

USE classicmodels;
SELECT customerName, contactFirstName
FROM customers
LEFT JOIN orders USING (customerNumber)
WHERE status IS NOT NULL;
    
```

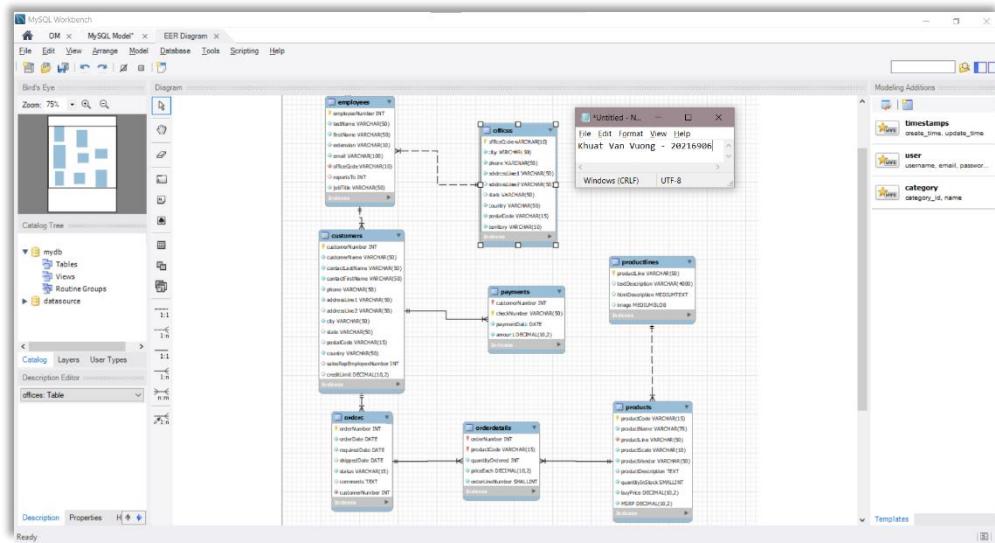
customerName	contactFirstName
Blue Fern Auto, Co.	Ronald
Blue Fern Auto, Co.	Richard
Blue Fern Auto, Co.	Patricia
Blue Fern Auto, Co.	George
Blue Fern Auto, Co.	Jeffrey
Blue Fern Auto, Co.	James
Land of Toys Inc.	Karen
Citrus Fresh Fruits	Albert
Hoppe Pet Distributors Inc.	Rose
Air Stores, Co.	Rachel
Wingfield Supplies, Co.	Sam
Wingfield Supplies, Co.	Christine
Mr. Gaff's Distributors Ltd.	Susan
La Grande Bonheur, Co.	Maria
Classic Legends Inc.	Maria
Kayee Bedding, Ltd.	Percie
Dragon Source, Ltd.	Eric
Bruce Database	Rosalie
Selling Collectibles	George
Horizon Hobby Imports, Co.	Peter

TUẦN 4.1: THỰC HÀNH TÌM HIỂU ĐƯỢC CẤU TRÚC CỦA MỘT CƠ SỞ DỮ LIỆU

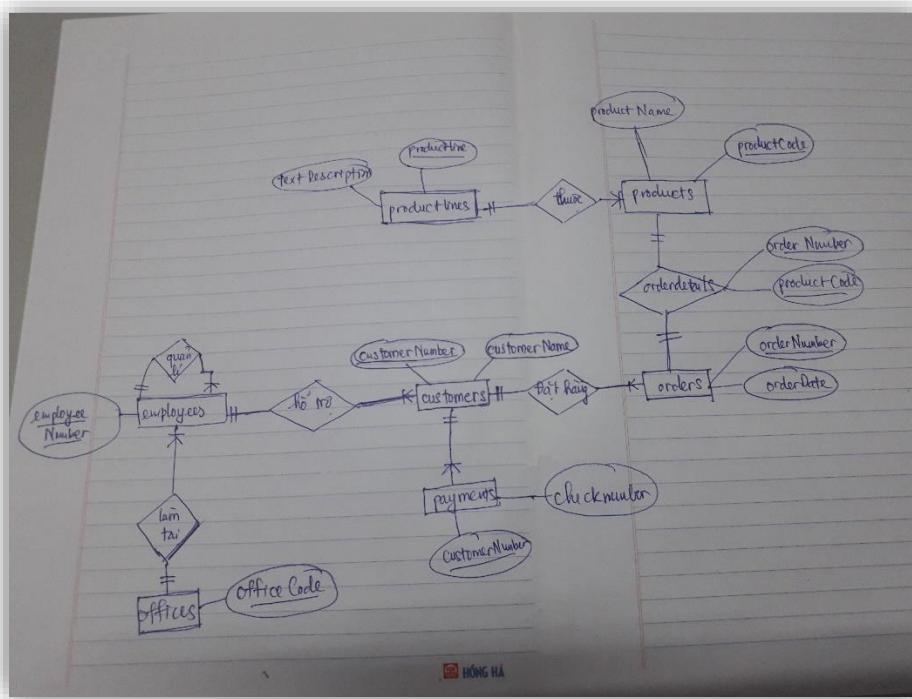
Bài 1: Xây dựng lược đồ (Diagram)

a) Sử dụng công cụ vẽ được lược đồ

Sử dụng công cụ có sẵn Reverse Engineer để vẽ lược đồ ER Diagram.



b) Đọc và phân tích lược đồ



Bài 2: Truy vấn các bảng hệ thống trong cơ sở dữ liệu

a) Tên các bảng

Mã nguồn:

```
USE classicmodels;

SELECT TABLE_NAME FROM information_schema.tables

WHERE table_schema = 'classicmodels' ORDER BY TABLE_NAME
```

Giao diện kết quả:

TABLE_NAME
Address
Customer
Employee
EmployeeDetail
Inventory
Language
ProductCategory
ProductLine
Product
Region

b) Tên các cột của một bảng và tính chất các cột đó

Mã nguồn:

```
SELECT * FROM information_schema.columns

WHERE table_schema = 'classicmodels'

ORDER BY TABLE_NAME, ordinal_position
```

Giao diện kết quả:

COLUMN_NAME	DATA_TYPE	CHARACTER_MAXIMUM_LENGTH	CHARACTER_OCTET_LENGTH	COLLATION	IS_NULLABLE	ORDINAL_POSITION	PRECISION	SCALE	CHARACTER_SET_NAME	COLUMN_TYPE	COLUMN_KEY	EXTRA	COLUMN_COMMENT
ADDRESS_ID	int	11	11	latin1_swedish_ci	NO	1	10	0	latin1	int(11)	PRI		
ADDRESS	varchar	45	45	latin1_swedish_ci	NO	2	45	0	latin1	varchar(45)			
CITY	varchar	30	30	latin1_swedish_ci	NO	3	30	0	latin1	varchar(30)			
STATE	varchar	2	2	latin1_swedish_ci	NO	4	2	0	latin1	varchar(2)			
POSTAL_CODE	varchar	10	10	latin1_swedish_ci	NO	5	10	0	latin1	varchar(10)			
COUNTRY_ID	char	2	2	latin1_swedish_ci	NO	6	2	0	latin1	char(2)			
ROW_SOURCE	varchar	10	10	latin1_swedish_ci	NO	7	10	0	latin1	varchar(10)			
ADDRESS_ID	int	11	11	latin1_swedish_ci	NO	8	10	0	latin1	int(11)	PRI		
ADDRESS	varchar	45	45	latin1_swedish_ci	NO	9	45	0	latin1	varchar(45)			
CITY	varchar	30	30	latin1_swedish_ci	NO	10	30	0	latin1	varchar(30)			
STATE	varchar	2	2	latin1_swedish_ci	NO	11	2	0	latin1	varchar(2)			
POSTAL_CODE	varchar	10	10	latin1_swedish_ci	NO	12	10	0	latin1	varchar(10)			
COUNTRY_ID	char	2	2	latin1_swedish_ci	NO	13	2	0	latin1	char(2)			
ROW_SOURCE	varchar	10	10	latin1_swedish_ci	NO	14	10	0	latin1	varchar(10)			

TUẦN 4.2: THỰC HÀNH XÂY DỰNG ĐƯỢC CẤU TRÚC CỦA MỘT CƠ SỞ DỮ LIỆU

Bài 1: Thiết lập bảng dữ liệu (PK, UK, DataType, Null, constraint)

Bài 2: Thiết lập quan hệ dữ liệu (Relationship)

Mã nguồn:

```

CREATE DATABASE BaiTuan4;

CREATE TABLE BaiTuan4.Persons (personID int NOT NULL, name VARCHAR(255) NOT NULL,
age int, address VARCHAR(255) NOT NULL, city VARCHAR(255),
city VARCHAR(255), CONSTRAINT uc_name_address UNIQUE (name, address),
PRIMARY KEY (personID));

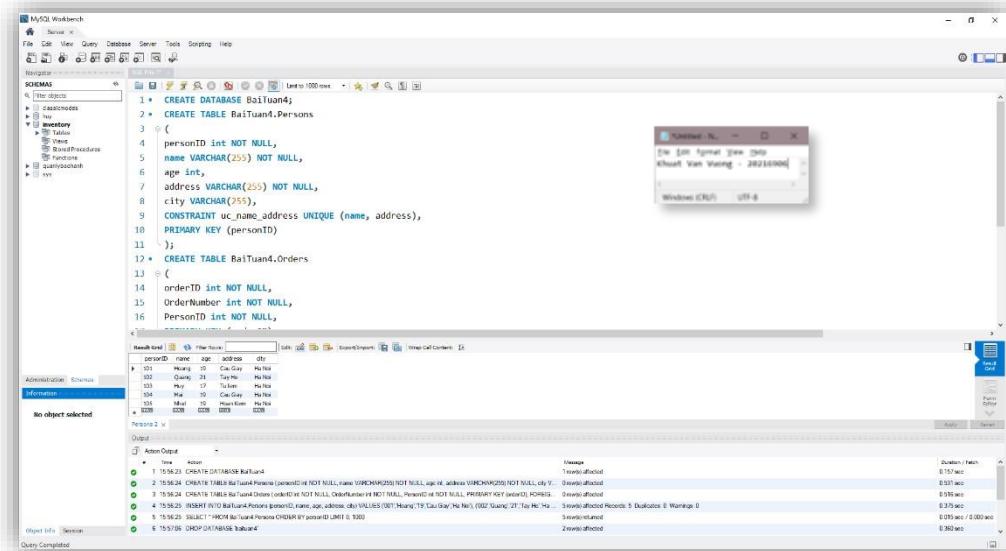
CREATE TABLE BaiTuan4.Orders (orderID int NOT NULL, OrderNumber int NOT
NULL, PersonID int NOT NULL, PRIMARY KEY (orderID), FOREIGN KEY (personID) REFERENCES Persons (PersonID));

INSERT INTO BaiTuan4.Persons (personID, name, age, address, city)
VALUES ('101', 'Hoang', '19', 'Cau Giay', 'Ha Noi'),
('102', 'Quang', '21', 'Tay Ho', 'Ha Noi'),
('103', 'Huy', '17', 'Tu liem', 'Ha Noi'),
('104', 'Mai', '19', 'Cau Giay', 'Ha Noi'),
('105', 'Nhat', '19', 'Hoan Kiem', 'Ha Noi');

SELECT * FROM BaiTuan4.Persons ORDER BY personID

```

Giao diện kết quả:

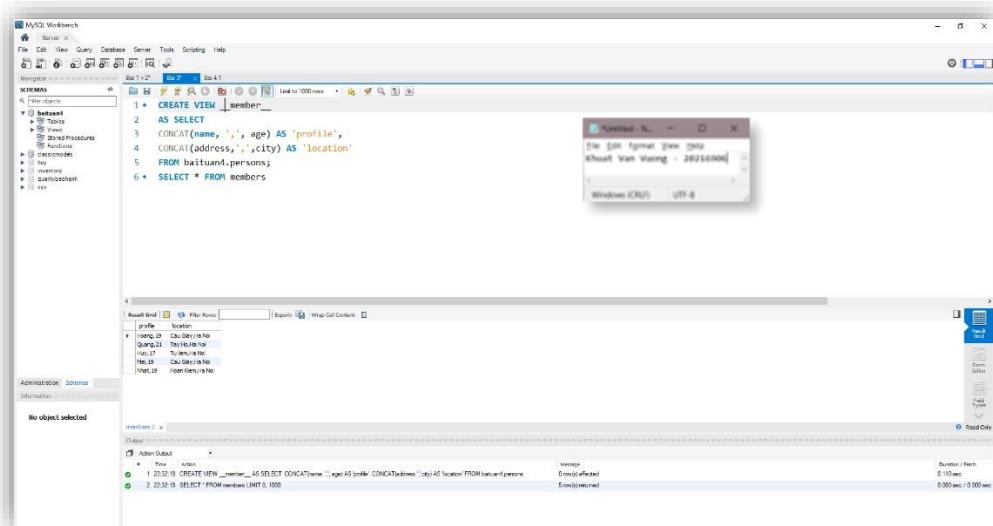


Bài 3: Thiết lập khung nhìn (Views)

Mã nguồn:

```
CREATE VIEW members AS
SELECT CONCAT(name, ',', age) AS 'profile',
       CONCAT(address, ',', city) AS 'location'
  FROM baituan4.persons;
SELECT * FROM members
```

Giao diện kết quả:



Bài 4: Thiết lập thủ tục và hàm (procedure & function)

Mã nguồn:

```
USE baituan4;
DELIMITER //
CREATE PROCEDURE count_person (OUT param1 int) BEGIN
  SELECT COUNT(*) INTO param1
  FROM persons; END//
DELIMITER ;
CALL count_person(@a);
SELECT @a;
```

```

USE classicmodels;

DELIMITER //

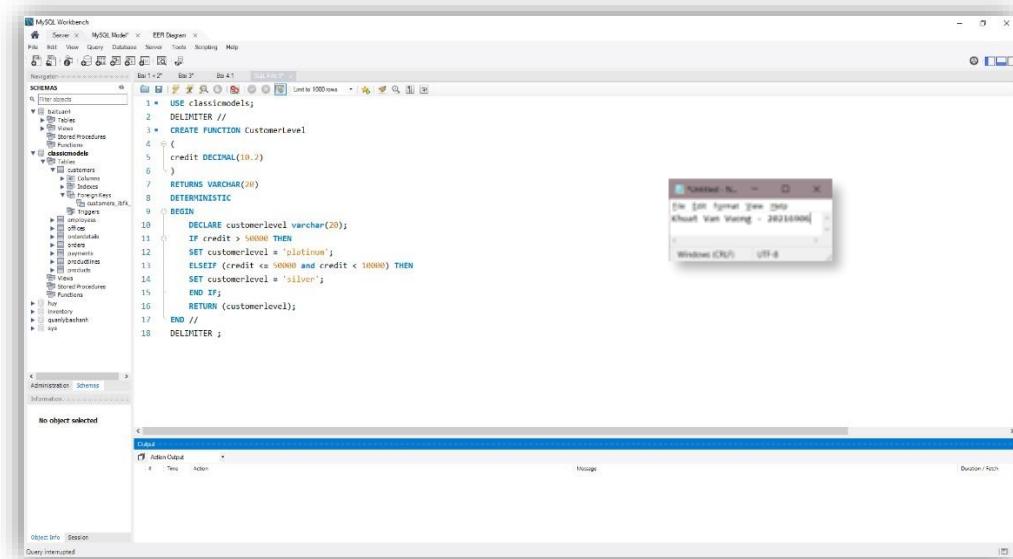
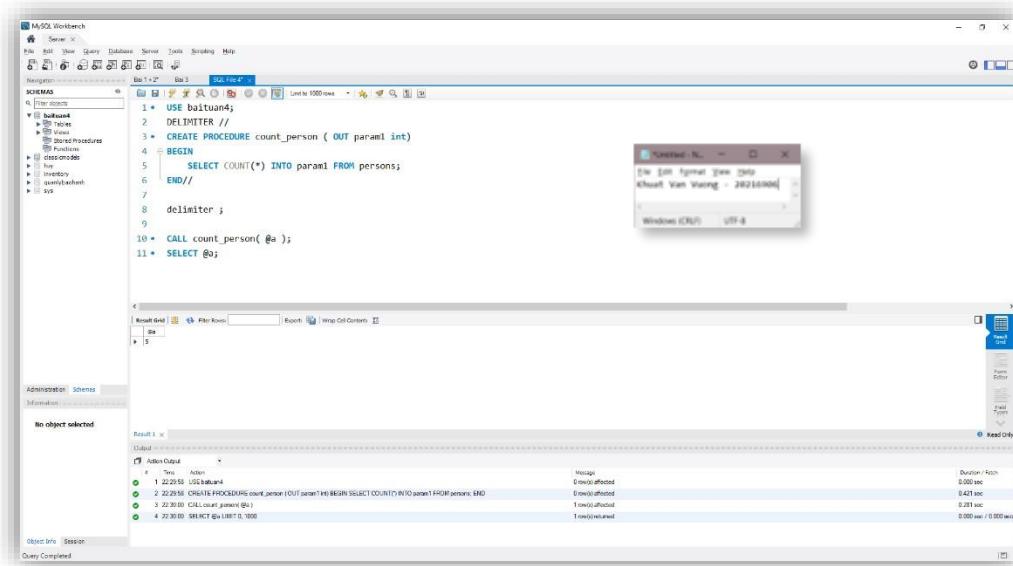
CREATE FUNCTION CustomerLevel (credit DECIMAL(10.2)) RETURNS VARCHAR(20) DETERMINISTIC
BEGIN DECLARE customerlevel varchar(20); IF credit > 50000 THEN

SET customerlevel = 'platinum'; ELSEIF (credit <= 50000 AND credit < 10000) THEN SET customerlevel = 'silver'; END IF; RETURN (customerlevel); END //

DELIMITER ;

```

Giao diện kết quả:



TUẦN 5: THỰC HÀNH CÁC CÂU LỆNH THAO TÁC DỮ LIỆU (PHẦN MYSQL DATA MANIPULATION)

Bài 1: Thực hiện các câu lệnh cập nhật (insert, update, delete dữ liệu)

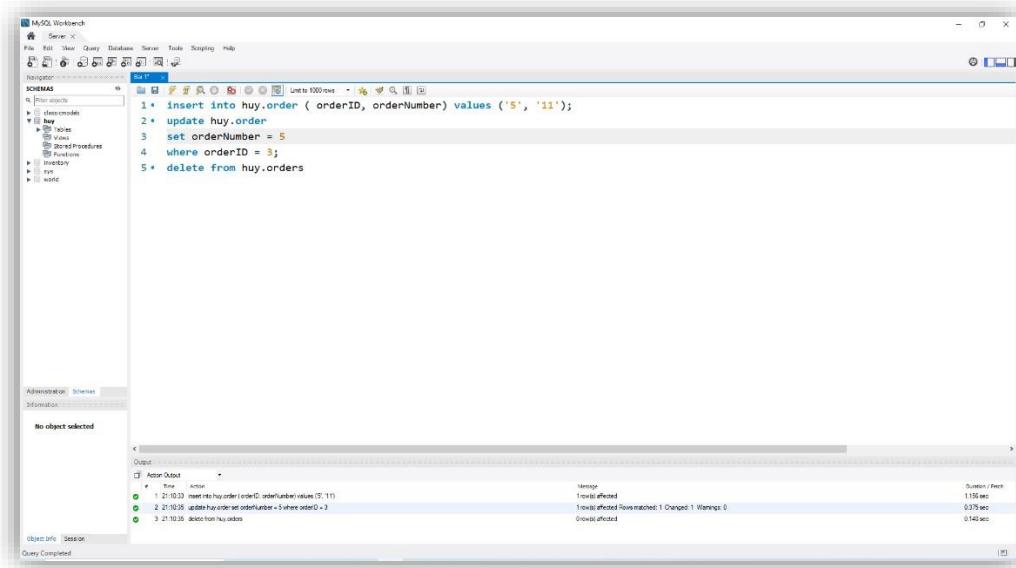
Mã nguồn:

```
INSERT INTO huy.order (orderID, orderNumber)
VALUES ('3', '10');

UPDATE huy.order SET orderNumber = 3 WHERE orderID = 3;

DELETE FROM huy.orders
```

Giao diện kết quả:



Bài 2: Cập nhật dữ liệu từ 1 bảng excel qua sinh tự động SQL Script

Mã nguồn:

```
USE huy;

DELETE FROM huy.order;

INSERT INTO huy.order (orderId, orderNumber, personId)
VALUES ('1', '111', '123');

INSERT INTO huy.order (orderId, orderNumber, personId) VALUES ('2', '222',
'234');

INSERT INTO huy.order (orderId, orderNumber, personId)
VALUES ('3', '333', '345');

INSERT INTO huy.order (orderId, orderNumber, personId)
VALUES ('4', '444', '456');
```

```
INSERT INTO huy.order (orderId, orderNumber, personId)
VALUES ('5', '555', '567');
```

UPDATE huy.order

```
SET personID = 1 WHERE orderId = 1;
UPDATE huy.order SET personID = 2 WHERE orderId = 2;
UPDATE huy.order SET personID = 3 WHERE orderId = 3;
UPDATE huy.order SET personID = 4 WHERE orderId = 4;
UPDATE huy.order SET personID = 5
WHERE orderId = 5;
```

Giao diện kết quả:

Bài 3: Cập nhật dữ liệu từ một bảng khác

Mã nguồn:

```
USE classicmodels;

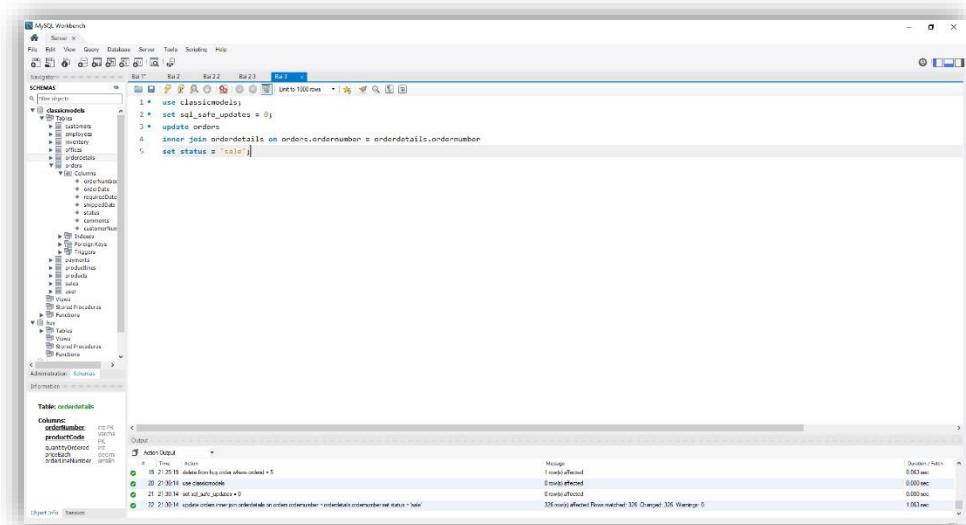
SET sql_safe_updates = 0;

UPDATE orders

INNER JOIN orderdetails ON orders.ordernumber = orderdetails.ordernumber

SET status = 'sale';
```

Giao diện kết quả:



Bài 4: Cập nhật dữ liệu qua một procedure

Mã nguồn:

```
USE huy;

DELIMITER //

CREATE PROCEDURE insertid_ (in orderid int, in ordernumber int, personId int) BEGIN

INSERT INTO huy.order (orderid, ordernumber, personid)

VALUES (orderid, ordernumber, personid); END //

DELIMITER ;

CALL insertid_ ('11', '11', '111');

SELECT * FROM huy.order
```

Giao diện kết quả:

The screenshot shows the MySQL Workbench interface. In the top-left pane, the 'Schemas' tree is visible, showing the 'huy' database selected. In the main area, a SQL editor window displays the following code:

```
1 * user huy;
2 delimiter //
3 * create procedure insertid_(in orderid int, in ordernumber int, personid int )
4 * begin
5 *     insert into huy.order (orderid, ordernumber, personid)
6 *         values (orderid, ordernumber, personid);
7 * end //
8 * delimiter ;
9 *
10 * call insertid_(111,111,111);
11 * select * from huy.order;
```

Below the SQL editor, a 'Result Grid' shows the data in the 'order' table:

OrderID	OrderNumber	PersonID
111	111	111
111	111	111

At the bottom of the interface, the 'Output' pane displays the execution log:

```
order t: x
Output:
1 21:41:25 user huy
2 21:41:25 [Info] execute procedure insertid_(in orderid int, in ordernumber int, personid int) [begin insert into huy.order (orderid, ordernumber, personid) values (orderid, ordernumber, personid);]
3 21:41:25 [Info] call insertid_(111,111,111)
4 21:41:25 [Info] select * from huy.order LIMIT 0, 1000
5 21:41:25 [Info] 2 rows(s) returned
```

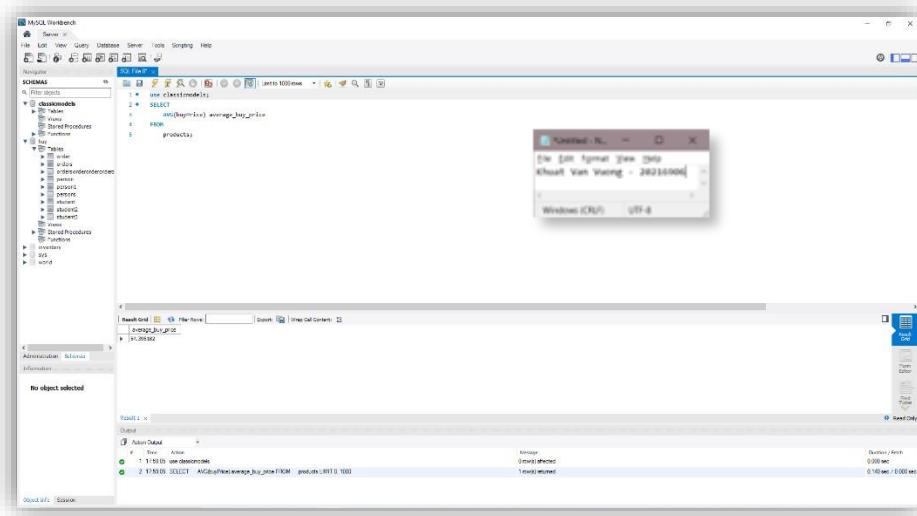
The log shows the procedure being called with parameters (111, 111, 111), the insertion of two rows into the 'order' table, and the final query selecting all rows from the table.

TUẦN 6: THỰC HÀNH CÁC CÂU LỆNH SỬ DỤNG HÀM TRONG TRUY VẤN

Bài 1: Aggregate Functions, Math Functions, Comparison Functions

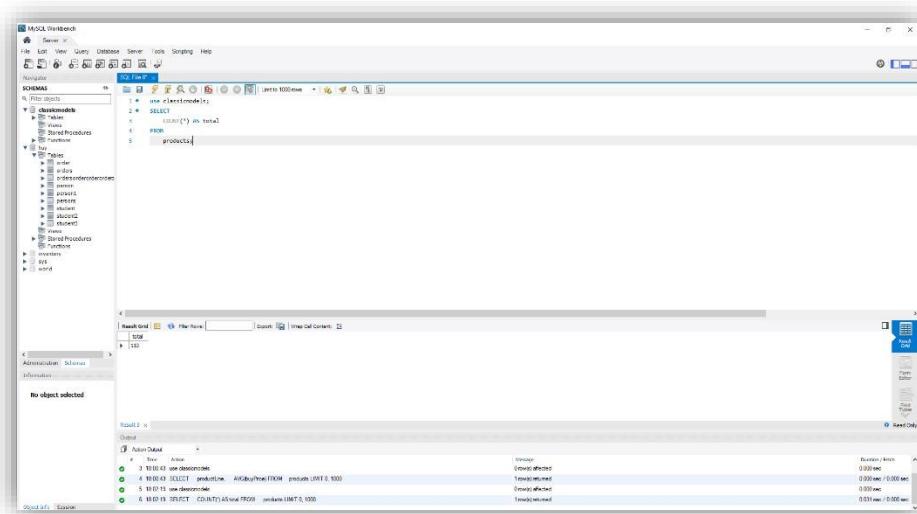
- Aggregate Functions:

```
USE classicmodels;
SELECT AVG(buyPrice) average_buy_price
FROM products;
```

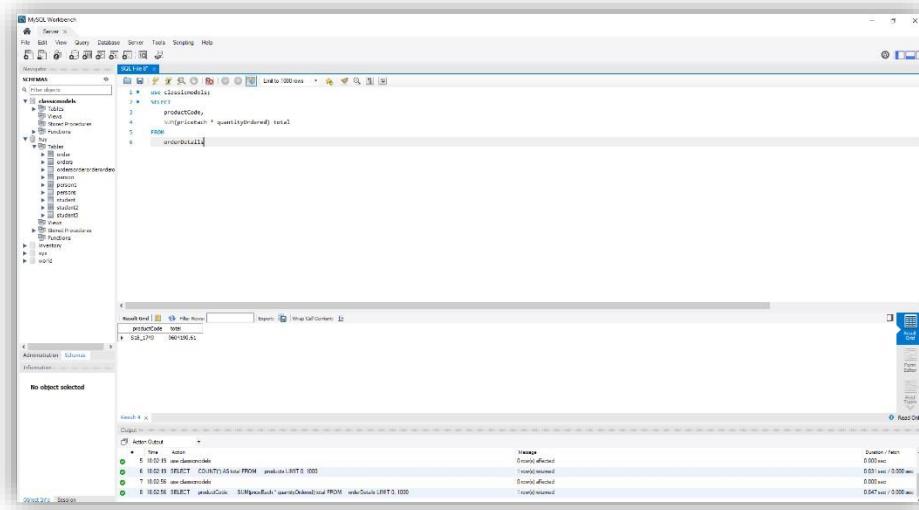


- Math Functions:

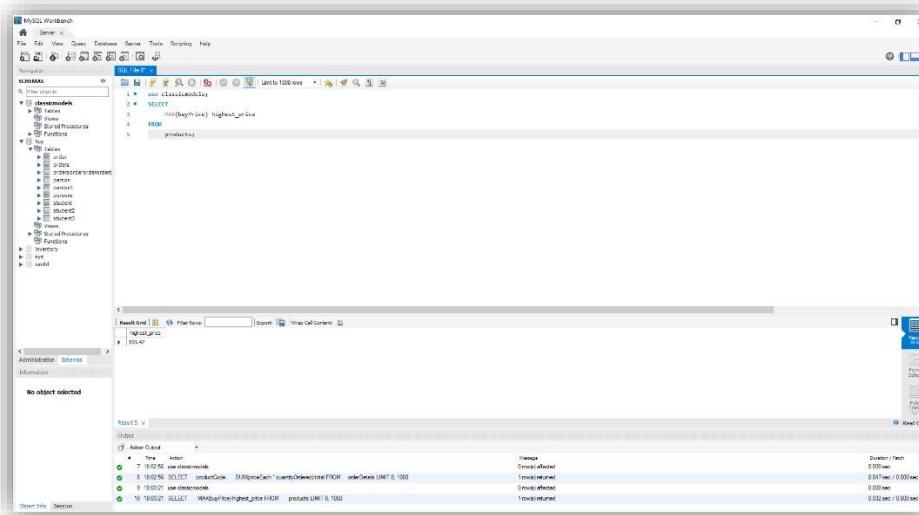
```
USE classicmodels;
SELECT COUNT(*) AS total
FROM products;
```



```
SELECT productCode, SUM(priceEach * quantityOrdered) total
FROM orderDetails;
```



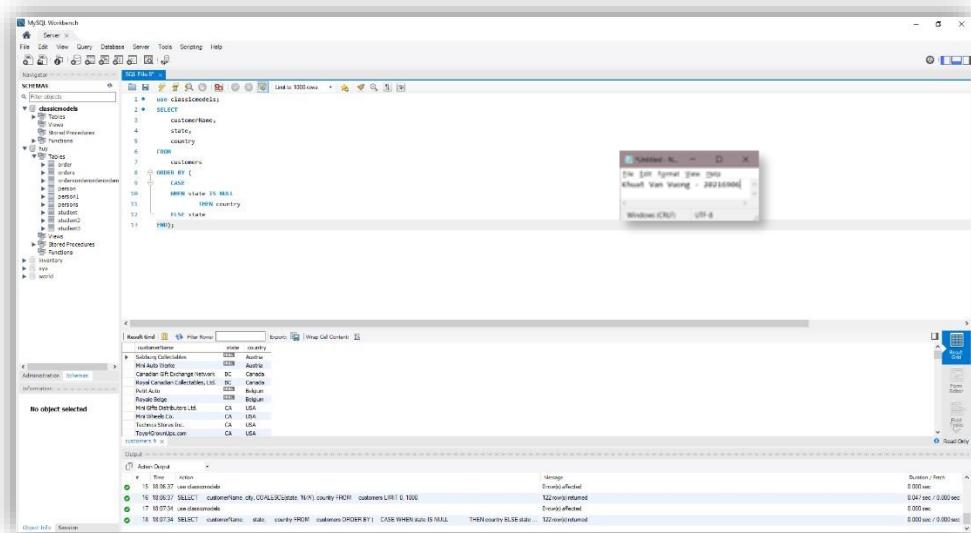
```
USE classicmodels;
SELECT MAX(buyPrice) highest_price
FROM products;
```



Bài 2: Control Flow Functions and Expressions

```
USE经典模型;

SELECT customerName, state, country
FROM customers
ORDER BY(CASE
    WHEN state IS NULL THEN country
    ELSE state
END);
```



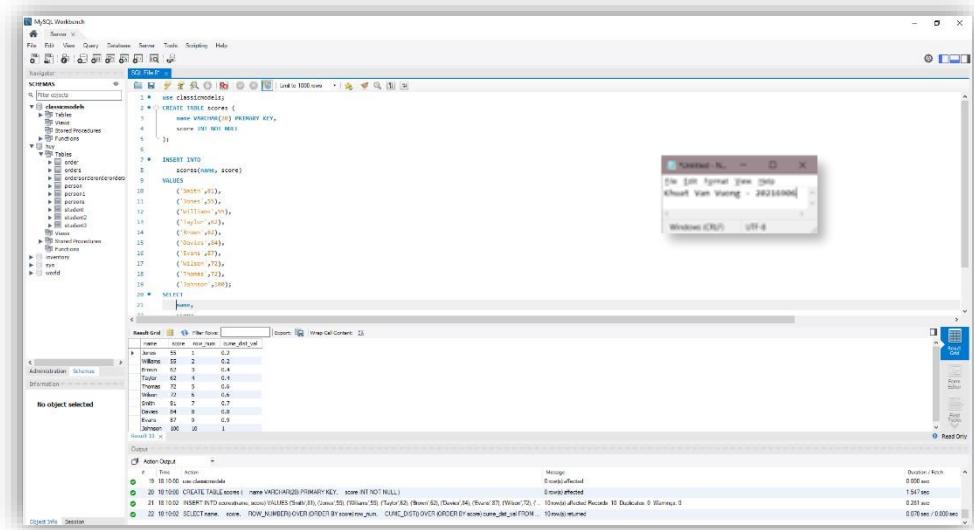
Bài 3: Window Functions

```
USE经典模型;

CREATE TABLE scores(name VARCHAR(20) PRIMARY KEY score INT NOT NULL);

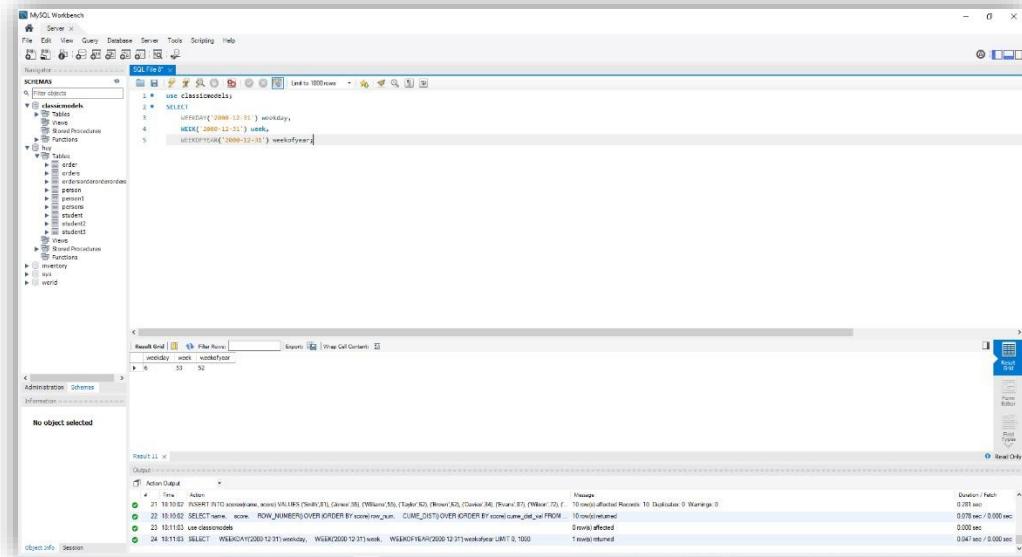
INSERT INTO score(name, score)

VALUES ('Smith', 81), ('Jones', 55), ('Williams', 55), ('Taylor', 62), ('Brown',
62), ('Davies', 84), ('Evans', 87), ('Wilson', 72);
```



Bài 4: Date Functions, String Functions

```
SELECT WEEKDAY('2000-12-31') weekday, WEEK ('2000-12-31') WEEK, WEEKOFYEAR('2000-12-31') weekofyear;
```



Bài 5: Đánh chỉ mục dữ liệu một bảng

```
USE classicmodels;
SELECT concat(contactFirstName, ' ', contactLastName) Fullname
FROM customers;
```

The screenshot shows the MySQL Workbench interface. In the SQL Editor, the following query is run:

```
use classicmodels;
SELECT concat(contactFirstName, ' ', contactLastName) Fullname
FROM customers;
```

The results pane displays the concatenated full names for all customers in the database.

```
EXPLAIN SELECT employeeNumber, lastName, firstName FROM employees
WHERE jobTitle = 'Sales Rep';
SHOW INDEXES FROM employees;
```

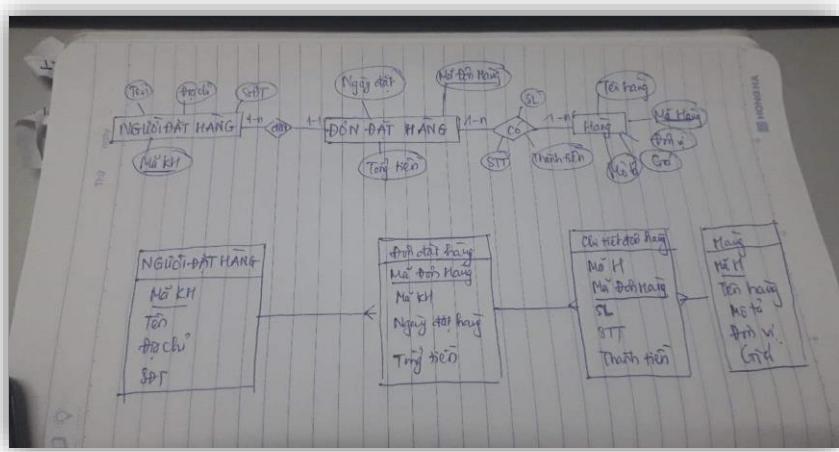
The screenshot shows the MySQL Workbench interface. The user runs the EXPLAIN command on the previous query:

```
EXPLAIN SELECT employeeNumber, lastName, firstName FROM employees
WHERE jobTitle = 'Sales Rep';
SHOW INDEXES FROM employees;
```

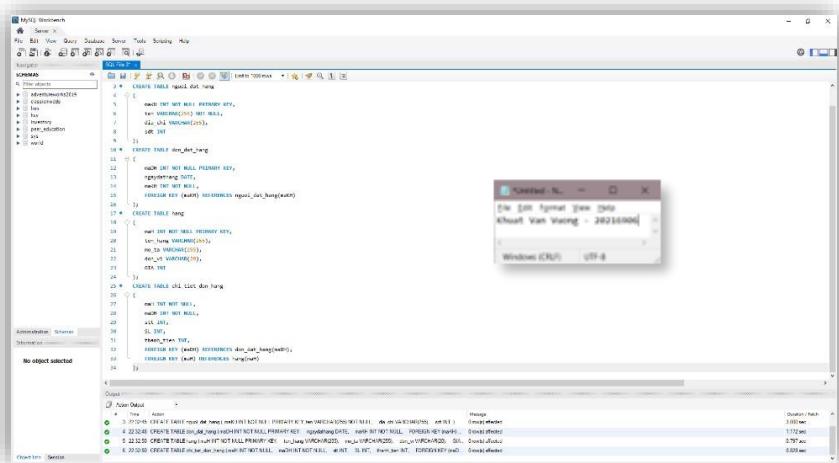
The EXPLAIN output provides details about the query's execution plan, and the SHOW INDEXES output lists the indexes available on the employees table.

TUẦN 7: THỰC HÀNH THIẾT KẾ CƠ SỞ DỮ LIỆU LUU TRỮ ĐƠN HÀNG

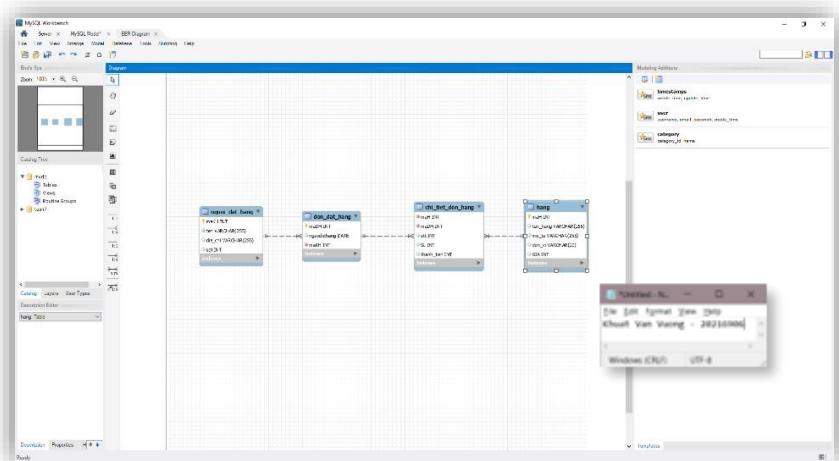
Bài 1 + 2 : Xây dựng lược đồ R-E và E-R



Bài 3: Thiết kế trên hệ quản trị cơ sở dữ liệu



Bài 4: In diagram cơ sở dữ liệu



Bài 5: Viết câu lệnh thêm dữ liệu và bảng

The screenshot shows the MySQL Workbench interface with the following details:

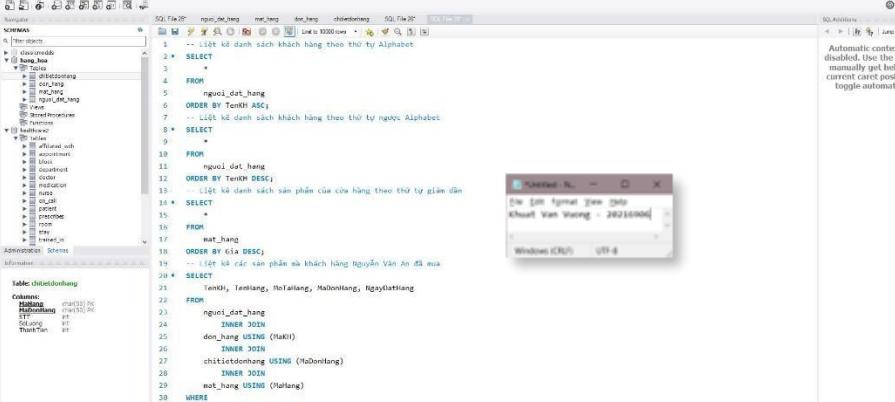
- MySQL Workbench** window title.
- Server** tab selected.
- SQL** tab selected.
- SQL Editor** pane:
 - Text area containing the SQL code for inserting data into the 'dept_emp' table.
 - Execution results pane showing the inserted rows.
- Logs** pane:
 - Shows the log entries for the executed SQL statements.
 - Log entry 36: 30 rows affected, 0 warnings, 0 errors.
 - Log entry 37: 30 rows affected, 0 warnings, 0 errors.
 - Log entry 38: 30 rows affected, 0 warnings, 0 errors.
- Information** pane: "No object selected".
- Properties** pane: "Object" section.
- Output** pane: "Auton Output" section.
- Messages** pane: "Info/Warning/Error" section.
- Details** pane: "Duration / Size" section.

Bài 6: Viết câu lệnh truy vấn Thay đổi cấu trúc CSDL

The screenshot shows the MySQL Workbench interface with several tabs open: Local Instance MySQL 8.0.22, MySQL Model, CCI Diagram, Local Reference MySQL 8.0.22, and CCI Diagram. The main window displays a complex SQL query for analyzing product sales:

```
SQL File 20*          nhan_hang      mat_hang      don_hang      mua_hang      SO_Ton_Hang
1 -- LIST KHACH DANH SACH KHACH HANG DA MUA O CUA HANG
2 * SELECT
3   matHang, TenKH
4   FROM
5   nhan_hang
6   INNER JOIN
7   don_hang USING (matHang)
8 -- LIST KHACH DANH SACH SAN PHAM CUA CUA HANG
9 * SELECT
10 *
11   FROM
12   matHang
13   ... LIST KHACH DANH SACH DON DIET HANG CUA HANG
14 * SELECT
15   donHang, MaDonHang,
16   NgayTatBang,
17   Tontang,
18   NoTatBang,
19   ChiTiet,
20   SoLuong,
21   TenKH,
22   DiaChi,
23   SoDienThoai
24   FROM
25   don_hang
26   INNER JOIN
27   chitietdonhang USING (MaDonHang)
28   INNER JOIN
29   mat_hang USING (MaHang)
30   INNER JOIN
31   nguoi_dai_hang USING (MaKH);
```

A tooltip is visible in the center of the screen, providing information about the current column 'Khach Van Vieng - 28216000'. The status bar at the bottom right shows 'Windows (CR/LF) - UTF-8'.



MySQL Workbench

File Edit View Query Database Server Tools Databases Help

Schemas

Tables

Views

Procedures

Functions

Information

Table: chitietdonhang

Columns:

- DonHangID: int(10) PK
- TenHang: varchar(100)
- HieuUngHieuTrang: decimal(10,2)
- TongTien: decimal(10,2)
- Soluong: int
- ThamMinh: text

SQL Editor

```
SELECT * FROM nguoi_dat_hang ORDER BY TenTen ASC;
```

```
-- Lист. kd danh sach khach hang theo thu tu Alphabet
```

```
* SELECT *
```

```
FROM
```

```
nguoi_dat_hang
```

```
ORDER BY TenTen ASC;
```

```
-- Lист. kd danh sach khach hang theo thu tu ngoc Alphabet
```

```
* SELECT *
```

```
FROM
```

```
nguoi_dat_hang
```

```
ORDER BY TenTen DESC;
```

```
-- Lict. kd danh sach san pham cua hang theo thu tu giam den
```

```
* SELECT *
```

```
FROM
```

```
mat_hang
```

```
ORDER BY gia DESC;
```

```
-- Lict. so cac tan san pham cua hang tang hanh
```

```
* SELECT *
```

```
FROM
```

```
nguoi_dat_hang
```

```
INNER JOIN
```

```
don_hang USING (DonHangID)
```

```
INNER JOIN
```

```
chitietdonhang USING (DonHangID)
```

```
INNER JOIN
```

```
mat_hang USING (MatHangID)
```

```
WHERE
```

```
TenTen = 'Nguyen Van An';
```

Results

DonHangID	TenHang	HieuUngHieuTrang	TongTien	Soluong	ThamMinh
1	May tinh	0.00	10000000.00	1	Nguyen Van An

No Actions

Automatic context help is disabled. Use the toolbar manually or get help for the current caret position or the tooltip-automatic-help.

The screenshot shows the MySQL Workbench interface with a query editor containing the following SQL code:

```
SELECT
    COUNT(*) AS SoKhachHang
FROM
    nguo_danh_hang
WHERE
    MaKH IN (SELECT DISTINCT
        nguo_danh_hang.MaKH
    FROM
        don_hang
        INNER JOIN
        nguo_danh_hang USING (MaKH)
    WHERE
        don_hang.MaDH IS NOT NULL);
-- Số mặt hàng mà cửa hàng bán

SELECT
    COUNT(MaHang) AS SoHungHuong
FROM
    mat_hang;
-- Số lượng tên của đơn hàng 123

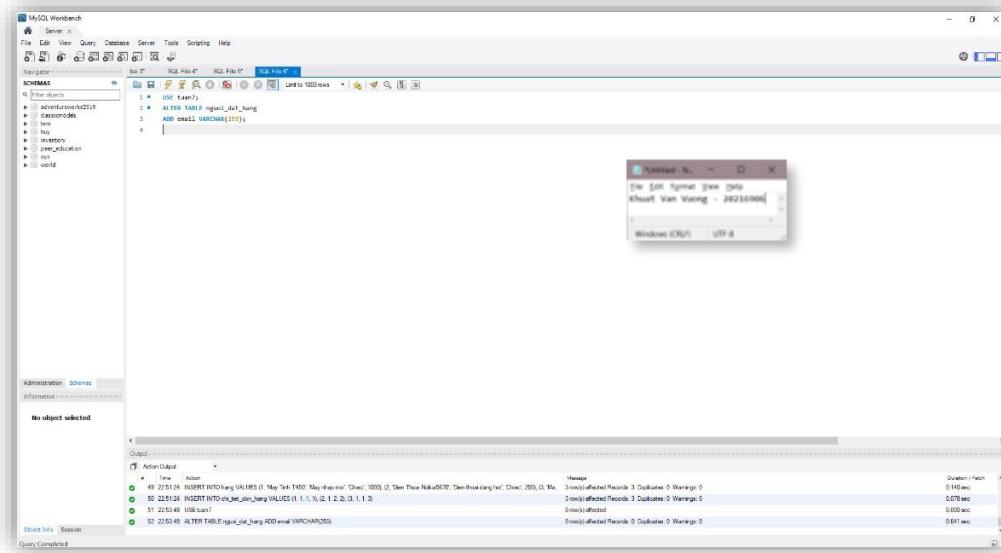
SELECT
    MaDonHang, SUM(Gia + Suluong) AS TongTien
FROM
    chitietdonhang
    INNER JOIN
    mat_hang USING (MaHang)
WHERE
    MaDonHang = '123';
```

A tooltip is visible in the center-right of the screen, titled "Automatic completion help", which says: "Click on the toolbar to automatically get help for the current caret position or to toggle automatic help."

Bài 7: Thay đổi cấu trúc CSDL

USE tuan7;

```
ALTER TABLE nguoi_dat_hang ADD email VARCHAR (255);
```



TUẦN 8

Đề bài:

Bài 1. Cho quan hệ R trên tập thuộc tính U = {A, B, C, D, E, G} và tập các phụ thuộc hàm F = {AB → C, AC → D, D → EG, G → B, A → D, CG → A}.

- Chứng minh rằng nếu R thoả mãn F thì R cũng thoả mãn các phụ thuộc hàm AB → E và AD → BC.
- Tính bao đóng $\{A\}^+$ của tập thuộc tính {A}.
- Tập các phụ thuộc hàm F đã là tối thiểu chưa? Vì sao? Nếu chưa, tìm một phụ tố tối thiểu của F.

Bài 2. Cho quan hệ R(U): U = {A, B, C, D, E, G, H} và tập phụ thuộc hàm F = {A → C, AB → G, B → DE, G → H, GH → A}.

- Chứng minh rằng nếu R thoả mãn F thì R cũng thoả mãn các phụ thuộc hàm AB → H và G → C.
- Tính bao đóng $\{G\}^+$ của tập thuộc tính {G}.

Bài 3. Cho quan hệ R trên tập thuộc tính U = {G, H, I, K, L, M} và tập các phụ thuộc hàm F = {GH → L, I → M, L → K, HM → G, GK → I, H → L}.

- Chứng minh rằng nếu R thoả mãn F thì R cũng thoả mãn các phụ thuộc hàm HI → G và GH → KM.
- Tính bao đóng $\{G, H\}^+$ của tập thuộc tính {G, H}.
- Tập các phụ thuộc hàm F đã là tối thiểu chưa? Vì sao? Nếu chưa, tìm một phụ tố tối thiểu của F.

Bài 4. Cho quan hệ R(U): U = {H, I, K, L, M, N} và tập phụ thuộc hàm F = {I → LM, HI → K, K → N, KN → I}.

- Tìm một khoá tối thiểu của quan hệ R.
- Kiểm tra tính mất mát thông tin khi tách R thành các quan hệ: R₁(HIN); R₂(HIK), R₃(ILM).
- Chuẩn hoá quan hệ R về dạng chuẩn 3NF.

Bài 5. Cho quan hệ R(U): U = {A, B, C, D, E, G, H} và tập phụ thuộc hàm F = {A → C, AB → G, B → DE, G → H, GH → A}.

- Tìm một khoá tối thiểu của quan hệ R.
- Kiểm tra tính mất mát thông tin khi tách R thành các quan hệ: R₁(AC); R₂(BDE); R₃(ABGH).
- Chuẩn hoá quan hệ R về dạng chuẩn 3NF.

Bài 6. Cho quan hệ R(U): U = {A, B, C, D, E, G, H} và tập phụ thuộc hàm F = {A → C, AB → G, B → DE, G → H, GH → A}.

- Kiểm tra tính mất mát thông tin khi tách R thành các quan hệ: R₁(AC); R₂(BDE); R₃(ABGH).
- Chuẩn hoá quan hệ R về dạng chuẩn 3NF.

Bài 1:

- $D \rightarrow EG \rightarrow AD \rightarrow EG$ và $G \rightarrow B \rightarrow EG \rightarrow B \rightarrow AD \rightarrow B$
 $AD \rightarrow B \rightarrow AD \rightarrow AB$ và $AB \rightarrow C \rightarrow AB \rightarrow BC \rightarrow AD \rightarrow BC$
- $\{A\}^+ = \{A, B, C, D, E, G\}$
- Phân rã các phụ thuộc hàm có nhiều thuộc tính
 $F_1 = \{AB \rightarrow C, AC \rightarrow D, D \rightarrow E, D \rightarrow G, G \rightarrow B, A \rightarrow D, CG \rightarrow A\}$

Loại các thuộc tính dư thừa

$AB \rightarrow C$ loại B

$F_2 = \{ A \rightarrow C, AC \rightarrow D, D \rightarrow E, D \rightarrow G, G \rightarrow B, A \rightarrow D, CG \rightarrow A \}$

$AC \rightarrow D$ loại C

$F_3 = \{ A \rightarrow C, A \rightarrow D, D \rightarrow E, D \rightarrow G, G \rightarrow B, A \rightarrow D, CG \rightarrow A \}$

$CG \rightarrow A$ loại G

$F_4 = \{ A \rightarrow C, A \rightarrow D, D \rightarrow E, D \rightarrow G, G \rightarrow B, C \rightarrow A \}$

Một phủ tối thiểu của F là F_4 .

Bài 2:

a) $AB \rightarrow G, G \rightarrow H \rightarrow AB \rightarrow H$

$GH \rightarrow A, A \rightarrow C$ và $G \rightarrow H \rightarrow GH \rightarrow H \rightarrow GH \rightarrow C$ $\{G\}^+ = \{A, C, G, H\}$

Bài 3:

a) $I \rightarrow M \rightarrow HI \rightarrow HM$ mà $HN \rightarrow G \rightarrow HI \rightarrow$

$GGH \rightarrow L, L \rightarrow K \rightarrow GH \rightarrow GK$

$GK \rightarrow I, I \rightarrow M \rightarrow GK \rightarrow M$

$\Rightarrow GH \rightarrow M$ và $GH \rightarrow L \rightarrow GH \rightarrow KM$

b) $\{G, H\}^+ = \{G, H, L, K, I, M\}$

Loại các thuộc tính dư thừa, $HM \rightarrow G$ loại được M vì

$\{H\}^+ = \{H, G, L, K, I, M\}$ có $F_1 = \{GH \rightarrow L, I \rightarrow M, L \rightarrow K, H \rightarrow G, GK \rightarrow I, H \rightarrow L\}$

Loại bỏ các thuộc tính dư thừa, bỏ được $GH \rightarrow L$

F chưa tối thiểu. Một phủ tối thiểu của F là $\{I \rightarrow M, L \rightarrow K, H \rightarrow G, GK \rightarrow I, H \rightarrow L\}$

Bài 4:

a) Tập nguồn {H}

Tập trung gian {I, K, N}

Một khóa tối thiểu là {H,

I}

b) Ta có bảng:

	H	I	K	L	M	N
R ₁	a ₁₁	b ₁₃	b ₁₄	b ₁₄	b ₁₅	a ₆
R ₂	a ₁	a ₁	b ₂₄	b ₂₄	b ₂₅	b ₂₆
R ₃	b ₃₁	b ₃₃	a ₄	a ₄	a ₅	b ₃₆

	H	I	K	L	M	N
R ₁	a ₁	a ₁	a ₁	b ₁₄	a ₅	a ₆
R ₂	a ₁	a ₂	a ₃	b ₂₄	a ₅	a ₆
R ₃	b ₃₁	a ₃	b ₃₃	a ₄	a ₅	a ₆

Bài 5:

a) $F_1 = \{ A \rightarrow C, AB \rightarrow G, B \rightarrow D, B \rightarrow E, G \rightarrow H, G \rightarrow A \}$

Gọi P là tập các thuộc tính xuất hiện ở vế phải của các phụ thuộc hàm.

$$P = \{ C, G, D, E, H, A \}$$

Do P không chứa B nên khóa tối thiểu của R phải chứa B

$$\text{Xét } \{B\}^+ = BDE$$

$$\{BA\}^+ = ABCDEGH \rightarrow \{B, A\} \text{ là khóa tối thiểu của quan hệ R}$$

$$\{BC\}^+ = BCDE$$

$$\{BD\}^+ = BDE$$

$$\{BE\}^+ = BDE$$

$$\{BG\}^+ = BDEGHAC \rightarrow \{B, G\} \text{ là khóa tối thiểu của quan hệ R}$$

$$\{BH\}^+ = BHDE$$

b)

	A	B	C	D	E	G	H
R ₁	A	a ₁₂	C	a ₁₄	a ₁₅	a ₁₆	a ₁₇
R ₂	a ₂₁	B	a ₂₃	D	E	a ₁₆	a ₁₇
R ₃	A	B	a ₃₃	a ₃₄	a ₃₅	G	H

	A	B	C	D	E	G	H
R ₁	A	a ₁₂	C	a ₁₄	a ₁₅	a ₁₆	a ₁₇
R ₂	a ₂₁	B	a ₂₃	D	E	a ₁₆	a ₁₇
R ₃	A	B	C	D	E	G	H

c) Tập phụ thuộc hàm F₁ là phủ tối thiểu của tập phụ thuộc hàm F

{BA} là khóa tối thiểu của quan hệ R

Để quan hệ R đạt chuẩn 3NF, tách R thành các quan hệ

$$R_1(BAG), R_2(BDE), R_3(AC), R_4(GHA)$$

TUẦN 9: TRUY VẤN DỮ LIỆU TRÊN CƠ SỞ DỮ LIỆU OLTP

Bài 1: Truy vấn thông tin khách hàng phần nàn

Mã nguồn:

```
SELECT * FROM customers

WEHRE phone = '+49 69 66 90 2555'
```

Giao diện kết quả:

customerNumber	customerName	contactLastName	contactFirstName	phone	addressLine1	addressLine2	city	state	postalCode	country	salesRepEmployeeNumber
128	Bauer See Auto, Co.	Kiel	Roland	+49 69 66 90 2555	Lyonerstr. 34		Frankfurt	WES	60528	Germany	1504

Bài 2: Truy vấn ra thông tin đơn hàng

Mã nguồn:

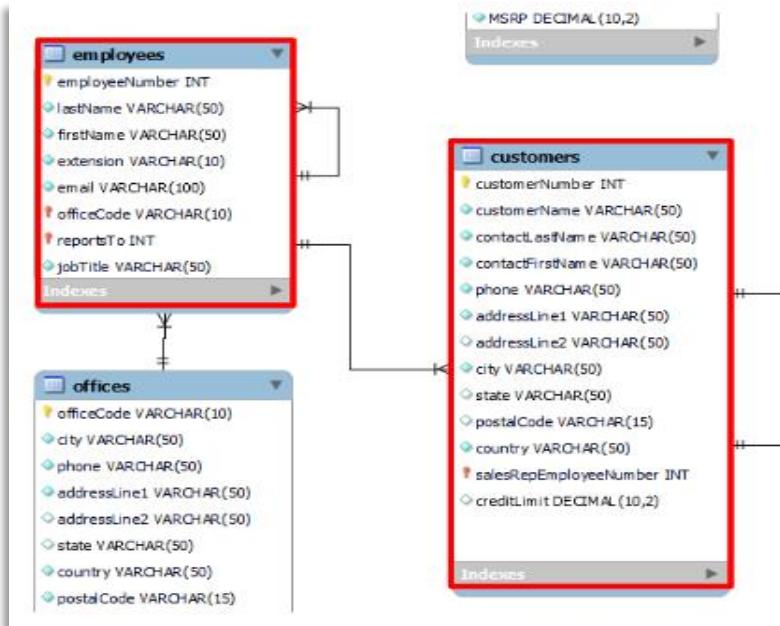
```
SELECT * FROM orders

WEHRE customerNumber = '128' AND orderDate = '2003-1-9'
```

Giao diện kết quả:

orderNumber	orderDate	requiredDate	shippedDate	status	comments	customerNumber
10101	2003-01-09	2003-01-18	2003-01-11	Shipped	Check on availability.	128

Bài 3: Truy vấn nhân viên đã chăm sóc khách hàng đơn hàng này



Mã nguồn:

```
SELECT * FROM employees AS e INNER JOIN customers AS c
WHERE e.employeeNumber = c.salesRepEmployeeNumber AND customerNumber = '128'
```

Giao diện kết quả:

The screenshot shows the MySQL Workbench interface with the following details:

- Navigator:** Shows the database schema with the **Tables** node expanded, displaying tables like customers, employees, offices, etc.
- SQL Editor:** Contains the query:

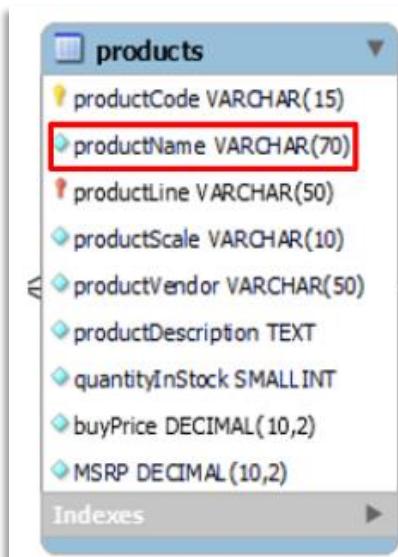

```
1 SELECT *FROM employees AS e INNER JOIN customers AS c
2 WHERE e.employeeNumber = c.salesRepEmployeeNumber
3 AND customerNumber = '128'
4 -- Khuất Văn Vuong 20216906
```
- Result Grid:** Displays the result of the query, showing one row:

employeeNumber	lastName	firstName	extension	email	officeCode	reportsTo	jobTitle	customerNumber	customerName	contactLastName
1504	Jones	Berry	x102	bjones@classicmodels.com	7	1102	Sales Rep	128	Blauer See Auto, Co.	Kettell
- Output:** Shows the execution message: "4 08:08:30 SELECT *FROM orders WHERE customerNumber = '128' AND orderDate > '2003-1-9' - Khuất Văn Vuong... 1 rows(s) returned".

Bài 4: Truy vấn thông tin sản phẩm bị phàn nàn

Mã nguồn:

```
SELECT productCode, productName, productLine, buyPrice
FROM products
WHERE productName LIKE '%1928 Mercedes-Benz%'
```



Giao diện kết quả:

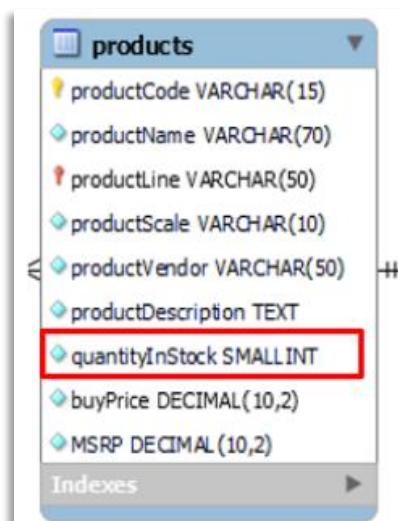
The screenshot shows the MySQL Workbench interface with the results of the query execution. The results grid displays the following data:

productCode	productName	productLine	buyPrice
S18_2795	1928 Mercedes-Benz SK	Vintage Cars	72.56

Bài 5: Kiểm tra kho hàng còn sản phẩm đó không

Mã nguồn:

```
SELECT productName, quantityInStock
FROM products
WHERE productCode = 'S18_2795'
```



Giao diện kết quả:

No object selected

products 6 x

Action Output

Object Info Session

Output

Time Action Message Duration / Fetch

8 08:18:28 SELECT productName, quantityInStock FROM products WHERE productCode = '1928' - Khuất Văn Vương... 0 rows returned 0.000 sec / 0.000 sec

Bài 6: Đưa ra những dòng sản phẩm có cùng mức giá, chênh lệch giá nhỏ để tư vấn

Mã nguồn:

```
SELECT productName, productLine, buyPrice
FROM products
WHERE ABS(72.56 - buyPrice)<=5
```

Giao diện kết quả:

No object selected

products 7 x

Action Output

Object Info Session

Output

Time Action Message Duration / Fetch

8 08:18:44 SELECT productName, quantityInStock FROM products WHERE productCode = 'S18_2795' - Khuất Văn Vương... 0 rows returned 0.016 sec / 0.000 sec

products

productCode VARCHAR(15)

productName VARCHAR(70)

productLine VARCHAR(50)

productScale VARCHAR(10)

productVendor VARCHAR(50)

productDescription TEXT

quantityInStock SMALLINT

buyPrice DECIMAL(10,2)

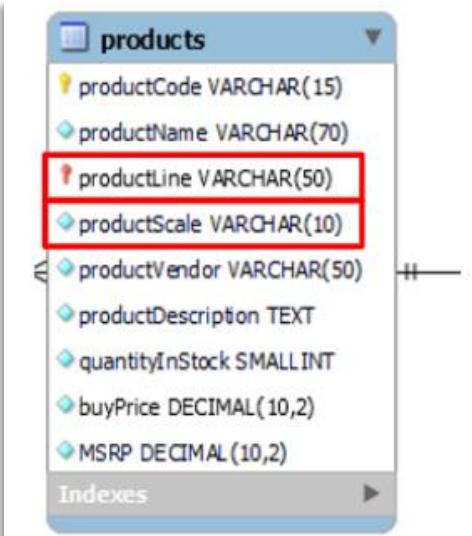
MSRP DECIMAL(10,2)

Indexes

Bài 7: Đưa ra những dòng xe có cùng một số đặc điểm với xe trước

Mã nguồn:

```
SELECT productName, productScale, productVendor,
       productDescription
  FROM products
 WHERE productLine = 'Vintage Cars'
   AND productScale = '1:18'
```



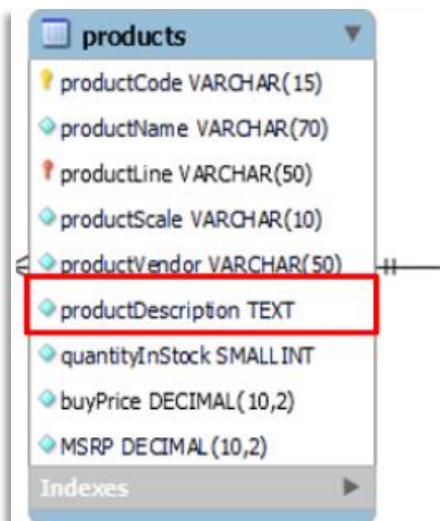
Giao diện kết quả:

productCode	productScale	productVendor	productDescription
1910 Lincoln Berlin	1:18	Motor City Art Classics	Features opening engine cover, doors, trunk, a...
1936 Mercedes-Benz 500K Special Roadster	1:18	Studio M Art Models	This 1:18 scale replica is constructed of heavy d...
1917 Grand Touring Sedan	1:18	Welly Diecast Productions	This 1:18 scale replica of the 1917 Grand Tourin...
1911 Ford Town Car	1:18	Motor City Art Classics	Features opening hood, opening doors, openin...
1932 Model A Ford 3-Coupe	1:18	Aubart Studio Design	This model features grille-mounted chrome horn...
1928 Mercedes-Benz SSK	1:18	Gearbox Collectibles	This 1:18 replica features grille-mounted chrom...
1913 Ford Model T Speedster	1:18	Carson DieCast Legends	This 25th Anniversary edition of the 1913 Ford...
1929 Ford Model A	1:18	Red Star Diecast	Hand crafted diecast model horse carriage a...
18th Century Vintage Horse Carriage	1:18	Unimax Art Galleries	Features opening trunk, working steering system...
1903 Ford Model A	1:18	Exoto Designs	Features Gold Trim, Full Size Spare Tire, Chrom...
1917 Maxwell Touring Car	1:18	Exoto Designs	Features opening hood, opening doors, openin...
1941 Chevrolet Special Deluxe Cabriolet	1:18	Exoto Designs	This 1:18 scale precision die cast replica feature...
1932 Alfa Romeo 8C2300 Spider Sport	1:18	Exoto Designs	Features opening trunk, working steering system...
1904 Buick Runabout	1:18	Exoto Designs	Features opening trunk, working steering system...

Bài 8: Đưa ra những dòng xe có cùng một số đặc điểm với xe trước

Mã nguồn:

```
SELECT *FROM products
 WHERE (productDescription LIKE '%white%'
        OR productDescription LIKE '%black%')
   AND productDescription LIKE '%opening hood%'
```



Giao diện kết quả:

The screenshot shows the SSMS interface with a query window containing the following SQL code:

```

SELECT * FROM products
WHERE (productDescription LIKE '%white%' 
      OR productDescription LIKE '%black%')
      AND productDescription LIKE '%opening hood%'
-- Khuất Văn Vương 20216906

```

The results grid displays the following data:

productCode	productName	productline	productScale	productVendor	productDescription	quantityInStock	buyPrice
S18_2246	1911 Ford Town Car	Vintage Cars	1:18	Motor City Classics	Features opening hood, opening doors, open...	540	33.45
S18_2247	1941 Lincoln Special Deluxe Cabriolet	Vintage Cars	1:18	Brown Diecast	Features opening hood, opening doors, open...	3378	64.85
S24_3151	1912 Ford Model T Delivery Wagon	Vintage Cars	1:24	Min Lin Diecast	This model features chrome trim and grille, open...	9173	46.9
S24_3420	1937 Horch 930V Limousine	Vintage Cars	1:24	Autobahn Studio Design	Features opening hood, opening doors, open...	2902	26.3
S24_3816	1940 Ford Delivery Sedan	Vintage Cars	1:24	Carousel DieCast Legends	Chrome Trim, Chrome Grille, Opening Hood, Op...	6521	48.6

Below the results grid, the message pane shows: "11 08:22:39 SELECT productName, productScale, productVendor, productDescription FROM products WHERE productDescription LIKE '%opening hood%' returned 15 rows(s)." The duration is listed as "0.000 sec / 0.000 sec".

Bài 9: Tìm 1 nhân viên đã có kinh nghiệm để tư vấn cho khách hàng

The screenshot shows the MySQL Workbench graphical interface with the 'customers' table selected. The table structure is displayed as follows:

customerNumber	customerName	contactLastName	contactFirstName	phone	addressLine1	addressLine2	city	state	postalCode	country	salesRepEmployeeNumber	creditLimit
----------------	--------------	-----------------	------------------	-------	--------------	--------------	------	-------	------------	---------	------------------------	-------------

The column 'salesRepEmployeeNumber' is highlighted with a red rectangle.

Mã nguồn:

```

SELECT salesRepEmployeeNumber, COUNT(salesRepEmployeeNumber) AS SLTV
FROM customers
GROUP BY SalesRepEmployeeNumber
ORDER BY COUNT(salesRepEmployeeNumber) DESC

```

Giao diện kết quả:

The screenshot shows the SSMS interface. In the top pane, a query window displays the following SQL code and its execution results:

```

SELECT salesRepEmployeeNumber, COUNT(salesRepEmployeeNumber) AS SLTV FROM customers
GROUP BY salesRepEmployeeNumber
ORDER BY COUNT(salesRepEmployeeNumber) DESC
-- Khuất Văn Vương 20216906

```

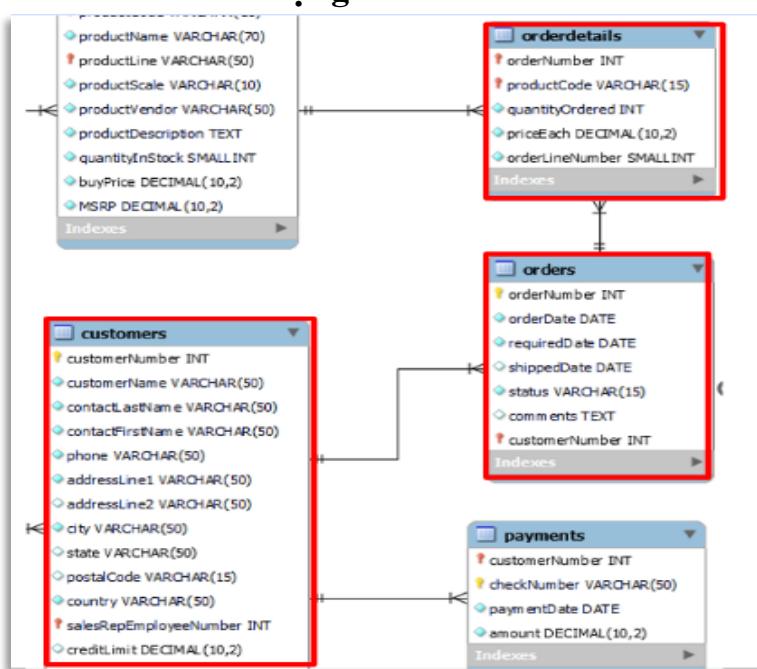
The results grid shows the following data:

salesRepEmployeeNumber	SLTV
1401	10
1504	9
1501	8
1323	8
1206	7
1370	7
1365	6
1366	6
1388	6
1216	6
1702	6
1337	6
1611	5
1612	5

In the bottom pane, the Query History shows two recent queries:

- 14 08:27:59 SELECT salesRepEmployeeNumber, COUNT(salesRepEmployeeNumber) AS SLTV FROM customers GR... 16 row(s) returned Duration / Fetch 0.000 sec / 0.000 sec
- 15 08:28:47 SELECT * FROM data.employees LIMIT 0.50000 23 row(s) returned Duration / Fetch 0.000 sec / 0.000 sec

Bài 10: Hiển thị những khách hàng đã mua sản phẩm này để tiến hành khảo sát chất lượng



Mã nguồn:

```

SELECT *FROM customers AS c
INNER JOIN orders AS o ON o.customerNumber = c.customerNumber
INNER JOIN orderdetails AS od ON od.orderNumber = o.customerNumber
WHERE productCode = 'S18_2795'

```

Giao diện kết quả:

The screenshot shows the SQL Server Management Studio interface. In the top navigation bar, the 'File', 'Edit', 'View', 'Query', 'Database', 'Server', 'Tools', 'Scripting', and 'Help' menus are visible. Below the menu is a 'Navigator' pane showing the 'SCHEMAS' section with 'data' selected, containing tables like 'customers', 'employees', 'orderdetails', 'payments', and 'products'. The main area displays a 'Result Grid' with data from a query. The query itself is:

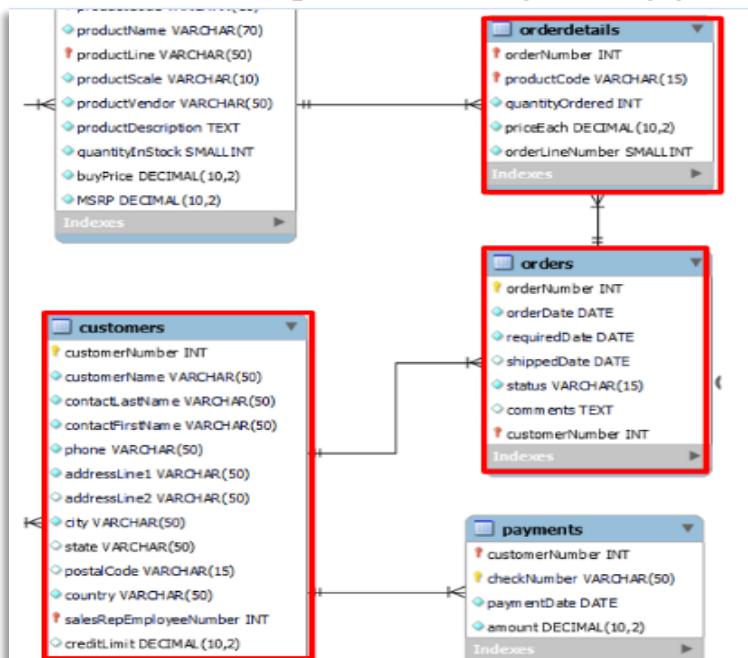
```

1 • SELECT *FROM customers AS c
2     INNER JOIN orders AS o ON o.customerNumber = c.customerNumber
3     INNER JOIN orderdetails AS od ON od.orderNumber = o.orderNumber
4     WHERE productCode = '$10_2795'
5 -- Khuất Văn Vương 20216906

```

The result grid contains 28 rows of customer information, including columns like 'customerNumber', 'customerName', 'contactLastName', 'contactFirstName', 'phone', 'addressLine1', 'addressLine2', 'city', 'state', 'postalCode', and 'country'. Below the grid is an 'Output' pane showing the execution time and message.

Bài 11: Hiển thị top 5 khách hàng có tổng giá trị đơn hàng lớn nhất



Mã nguồn:

```

SELECT SUM(quantityOrdered * priceEach) AS total, c.customerNumber, customerName,
       phone FROM customers AS c

INNER JOIN orders AS o ON o.customerNumber = c.customerNumber

INNER JOIN orderdetails AS od ON od.orderNumber = o.orderNumber

GROUP BY c.customerNumber

ORDER BY total DESC LIMIT 5

```

Giao diện kết quả:

The screenshot shows the SSMS interface with a query window containing the following SQL code:

```

SELECT SUM(quantityOrdered * priceEach) AS total, c.customerNumber, c.customerName, phone
FROM customers AS c
INNER JOIN orders AS o ON o.customerNumber = c.customerNumber
INNER JOIN orderdetails AS od ON od.orderNumber = o.orderNumber
GROUP BY c.customerNumber
ORDER BY total DESC
LIMIT 5
-- Khuất Văn Vương 20216906
    
```

The results grid displays five rows of data:

total	customerNumber	customerName	phone
820689.54	141	Euro+ Shopping Channel	(01) 555 94 44
591827.34	124	Mini Gifts Distributors Ltd.	415551450
180585.07	114	Australian Collectors, Co.	03 9520 4555
177913.95	151	Muscle Machine Inc	212557413
15873.12	119	La Rodelle Gifts	40.67.8555

The message pane at the bottom shows two log entries:

- Action Output: 18 08:34:47 SELECT SUM(quantityOrdered * priceEach) AS total, c.customerNumber, c.customerName, phone FROM ... Error Code: 1054. Unknown column 'od.customerNumber' in 'on clause'
- 19 08:35:22 SELECT SUM(quantityOrdered * priceEach) AS total, c.customerNumber, c.customerName, phone FROM c... 5 rows returned Duration / Fetch: 0.000 sec / 0.015 sec / 0.000 sec

Bài 12: Hiển thị top 5 sản phẩm có tỷ lệ doanh số cao nhất**Mã nguồn:**

```

SELECT productName, (SUM(quantityOrdered * priceEach)*100) /
(SELECT SUM(quantityOrdered * priceEach) FROM orderdetails)
AS percentage

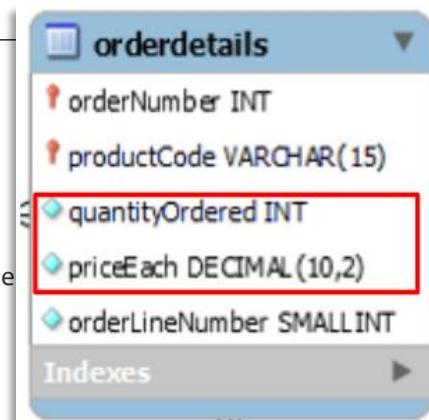
FROM products AS p

INNER JOIN orderdetails AS od od.productCode = p.productCode

INNER JOIN orders AS o od.o.orderNumber = od.orderNumber

GROUP BY p.productCode

ORDER BY percentage DESC LIMIT 5
    
```



The screenshot shows the MySQL Workbench interface. In the top-left, the Navigator pane displays the schema structure. The central area contains a SQL editor window with the following query:

```

1 • SELECT productName, (SUM(quantityOrdered * priceEach) * 100) / (SELECT SUM(quantityOrdered * priceEach) FROM orderdetails) AS percentage
2   FROM products AS p
3   INNER JOIN orderdetails AS od ON od.productCode = p.productCode
4   INNER JOIN orders AS o ON o.orderNumber = od.orderNumber
5   GROUP BY p.productCode
6   ORDER BY percentage DESC
7   LIMIT 5
8   -- Khuất Văn Vương 20216906

```

The Result Grid shows the output of the query:

productName	percentage
1992 Ferrari 360 Spider red	2.882492
2001 Ferrari Enzo	1.986173
1952 Alpine Renault 1300	1.978490
2003 Harley-Davidson Eagle Drag Bike	1.777203
1968 Ford Mustang	1.681885

The bottom pane shows the Action Output log:

- 20 08:39:19 SELECT productName, (SUM(quantityOrdered * priceEach) * 100) / (SELECT SUM(quantityOrdered * priceEach) FROM orderdetails) AS percentage ... Error Code: 1054. Unknown column 'quantity' in field list' Duration / Fetch: 0.000 sec / 0.000 sec
- 21 08:39:33 SELECT productName, (SUM(quantityOrdered * priceEach) * 100) / (SELECT SUM(quantityOrdered * priceEach) FROM orderdetails) AS percentage ... 5 row(s) returned Duration / Fetch: 0.000 sec / 0.000 sec

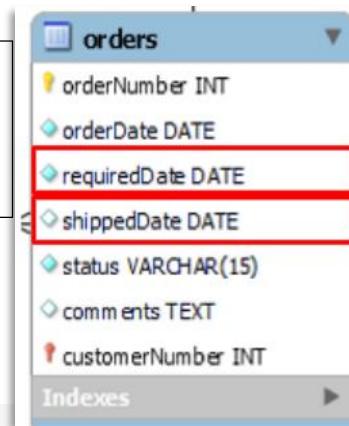
Bài 13: Kiểm tra giao vận đã đúng thời gian yêu cầu hay chưa, hiển thị đơn hàng giao trễ

Mã nguồn:

```

SELECT orderNumber, orderDate, requireDate, shippedDate,
       timestampdiff(DAY, requiredDate, shippedDate) AS delayDate
FROM orders WHERE requiredDate < shippedDate

```



Giao diện kết quả:

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```

1 • SELECT orderNumber, orderDate, requireDate, shippedDate, timestampdiff(DAY, requiredDate, shippedDate) AS delayDate
2   FROM orders WHERE requiredDate < shippedDate
3   -- Khuất Văn Vương 20216906

```

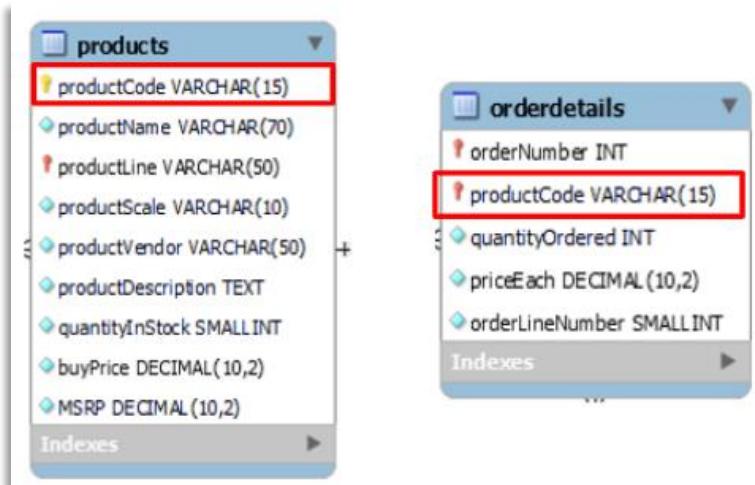
The Result Grid shows the output of the query:

orderNumber	orderDate	requireDate	shippedDate	delayDate
10165	2003-10-22	2003-10-31	2003-12-26	56

The bottom pane shows the Action Output log:

- 21 08:39:33 SELECT orderNumber, orderDate, requireDate, shippedDate, timestampdiff(DAY, requiredDate, shippedDate) AS delayDate ... 1 row(s) returned Duration / Fetch: 0.000 sec / 0.000 sec
- 22 08:42:01 SELECT orderNumber, orderDate, requireDate, shippedDate, timestampdiff(DAY, requiredDate, shippedDate) AS delayDate ... 1 row(s) returned Duration / Fetch: 0.000 sec / 0.000 sec

Bài 14: Đưa các sản phẩm không có mặt trong bất kỳ một đơn hàng nào



Mã nguồn:

```
SELECT *FROM products
WHERE productCode
NOT IN (SELECT productCode FROM orderdetails)
```

Giao diện kết quả:

The screenshot shows the SSMS interface with the following details:

- Navigator:** Shows the database schema with 'Tables' expanded, including customers, employees, offices, orderdetails, orders, payments, productlines, products, and test.
- SQL File 15:** Contains the query:


```
1 • SELECT *FROM products
2 WHERE productCode
3 NOT IN (SELECT productCode FROM orderdetails)
```
- Result Grid:** Displays the results of the query, showing a single row of data from the products table:

productCode	productName	productline	productScale	productVendor	productDescription	quantityInStock	buyPrice	MSRP
S18_3223	1985 Toyota Supra	Classic Cars	1:18	Highway 66 Mini Classics	This model features soft rubber tires, working s...	7733	57.91	107.57
- Output:** Shows the execution log with two entries:
 - 22 08:42:01 SELECT orderNumber, orderDate, requiredDate, shippedDate, timestampdf(DAY, requiredDate, shipped... 1 row(s) returned
 - 23 09:01:44 SELECT *FROM products WHERE productCode NOT IN (SELECT productCode FROM orderdetails) - K... 1 row(s) returned

Bài 15: Đưa ra các sản phẩm có số lượng trong kho lớn hơn trung bình số lượng trong kho của các sản phẩm cùng loại

Mã nguồn:

```
SELECT *FROM products WHERE quantityInStock >
(SELECT AVG(quantityInStock) FROM products
WHERE productLine = products.productLine) delayDat
```

The screenshot shows the MySQL Workbench interface. On the left, the 'Schemas' tree view is expanded to show the 'data' schema, which contains several tables like customers, employees, offices, orderdetails, orders, payments, products, and so on. In the center, the 'products' table is selected. The table has columns: productCode (VARCHAR(15)), productName (VARCHAR(70)), productLine (VARCHAR(50)), productScale (VARCHAR(10)), productVendor (VARCHAR(50)), productDescription (TEXT), quantityInStock (SMALLINT), buyPrice (DECIMAL(10,2)), and MSRP (DECIMAL(10,2)). The 'quantityInStock' column is highlighted with a red box. On the right, there's a 'Indexes' section.

Giao diện kết quả:

The screenshot shows the MySQL Workbench interface during query execution. The SQL editor window contains the query: `SELECT *FROM products WHERE quantityInStock > (SELECT AVG(quantityInStock) FROM products WHERE productLine = products.productLine)`. The results grid shows 58 rows of product data, each with columns: productCode, productName, productLine, productScale, productVendor, productDescription, and quantityInStock. The results are ordered by productLine and productVendor. The status bar at the bottom indicates the execution time and duration.

Bài 16: Thống kê tổng số lượng sản phẩm trong kho theo từng dòng sản phẩm của từng nhà cung ứng

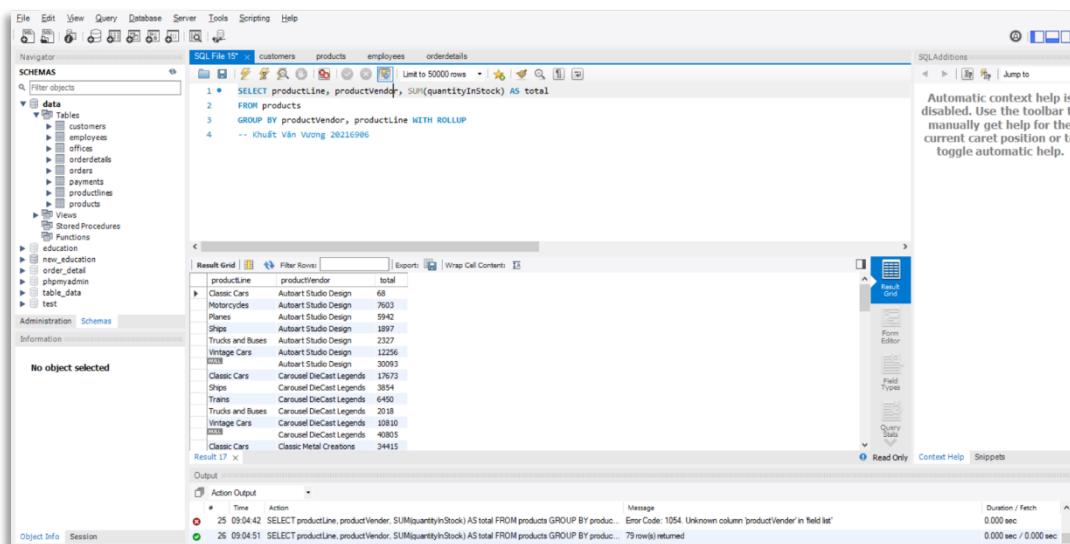
Mã nguồn:

```
SELECT productLine, productVendor SUM(quantityInStock) AS total
FROM products
GROUP BY productVendor, productLine WITH ROLLUP
```

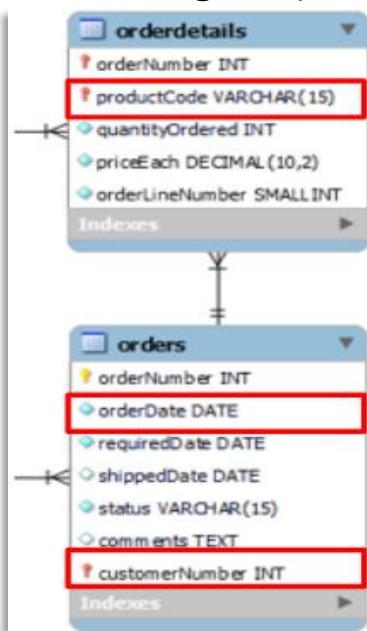
The screenshot shows the MySQL Workbench interface. The 'products' table structure is displayed again, with specific columns highlighted: productLine (VARCHAR(50)), productVendor (VARCHAR(50)), quantityInStock (SMALLINT), buyPrice (DECIMAL(10,2)), and MSRP (DECIMAL(10,2)). These columns are highlighted with red boxes. The 'Indexes' section is also visible.

Giao diện kết quả:

Khuất Văn Vương – 20216906



Bài 17: Thống kê ra mỗi sản phẩm được đặt hàng lần cuối vào thời gian nào và khách hàng đã đặt hàng



Mã nguồn:

```

SELECT productCode, orderDate, customerNumber
FROM(
  SELECT productCode, orderDate, customerNumber, row_number() over
    (partition by productCode ORDER BY orderDate DESC) AS RowNum
  FROM order AS o INNER JOIN orderdetails AS od ON o.orderNumber = od.orderNumber
)AS TB
WHERE RowNum = 1

```

Giao diện kết quả:

Khuất Văn Vương – 20216906

58

Lớp: 142295

The screenshot shows the SSMS interface with the following details:

- File Bar:** File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Navigator:** Shows the database schema with the "data" schema expanded, containing tables like customers, employees, orderdetails, orders, payments, products, and productlines.
- SQL Editor:** A query window titled "SQL File 15" with the following T-SQL code:

```
1 • SELECT productCode, orderDate, customerNumber
2   FROM (
3       SELECT productCode, orderDate, customerNumber, row_number() over (partition by productCode ORDER BY orderDate DESC ) AS RowNum
4   FROM orders AS o INNER JOIN orderdetails AS od ON o.orderNumber = od.orderNumber
5   ) AS TB
6   WHERE RowNum = 1
7 -- Khuất Văn Vương 20216906
```
- Result Grid:** Displays the results of the query, showing columns: productCode, orderDate, and customerNumber. The data includes rows such as S10_1676, S10_1949, S10_2016, S10_2018, S10_2579, S10_4962, S12_1098, S12_1666, S12_2823, S12_3148, S12_3891, and S12_3990, with dates ranging from 2005-05-13 to 2005-05-17 and customer numbers 141, 141, 141, 141, 362, 119, 382, 175, 141, 141, 141, 382, 175, 382.
- Output Window:** Shows the execution history with two entries:
 - Time: 2009-13-23 Action: "SELECT * FROM data.orders LIMIT 0, 50000" Message: 326 row(s) returned Duration / Fetch: 0.000 sec / 0.000 sec
 - Time: 2009-16-30 Action: "SELECT productCode, orderDate, customerNumber FROM (SELECT productCode, orderDate, customer..." Message: 109 row(s) returned Duration / Fetch: 0.000 sec / 0.000 sec
- Object Info:** Session

TUẦN 10: XÂY DỰNG DASHBOARD VÀ PHÂN TÍCH DỮ LIỆU TỪ CSDL OLTP

Bài 1: Thống kê doanh số của các nhân viên/quản lý theo phòng ban (cụ thể là phòng ban có officeCode = 4)

Giao diện kết quả:

officeCode	employeeNumber	Sales
4	1337	569485.75
4	1370	1258577.81
4	1401	868220.55
4	1702	387477.47
Tổng doanh số		3083761.58

So sánh kết quả thực hiện trên MySQL:

officeCode	employeeNumber	Sales
4	1337	569485.75
4	1370	1258577.81
4	1401	868220.55
4	1702	387477.47

Bài 2: Thống kê doanh số theo các office

Giao diện kết quả:

	Mã phòng ban	Doanh số
1	Mã phòng ban	1429063.57
2		892538.62
3		1157589.72
4		3083761.58
5		457110.07
6		1147176.35
7		1436950.70
9	Tổng doanh số	9604190.61

So sánh kết quả thực hiện trên MySQL:

officeCode	Sales
1	1429063.57
2	892538.62
3	1157589.72
4	3083761.58
5	457110.07
6	1147176.35
7	1436950.70

Bài 3: Thống kê doanh số theo nội địa tại các office

Giao diện kết quả:

country	Doanh số
USA	1429063,57
USA	89538,62
USA	951677,06
France	1007374,02
Japan	167909,95
Australia	562582,59
UK	430547,44
Tổng doanh số	5448994,05

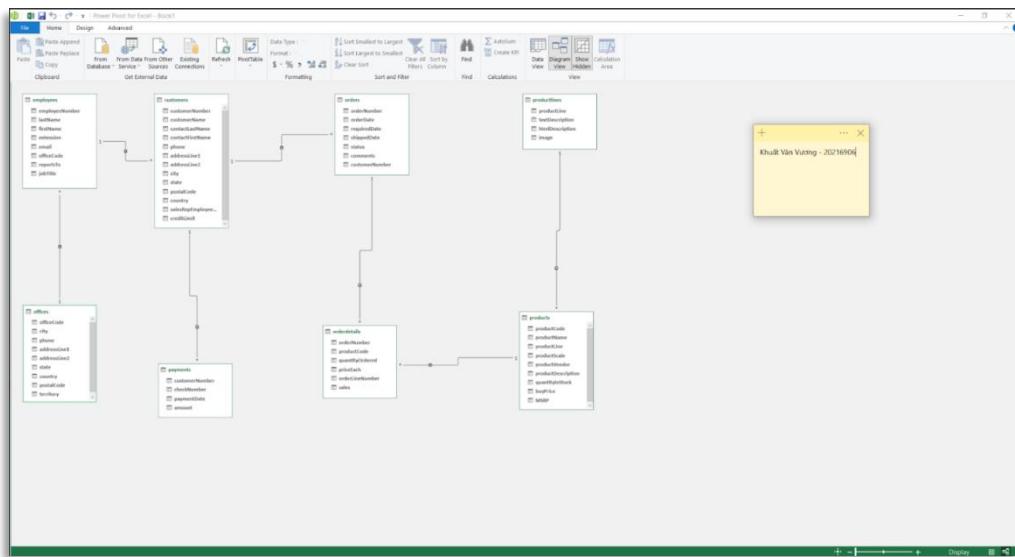
Bài 4: Thống kê top 5 những sản phẩm được khách hàng mua nhiều nhất theo khu vực cụ thể

Giao diện kết quả:

productName	Số lượng bán ra
1936 Chrysler Airflow	487
1987 Chevy Pickup	523
1997 Ford Taurus Sedan	491
1997 BMW R 1100 S	475
2002 Suzuki XREO	499
Tổng số lượng bán ra	2475

Bài 5: Xây dựng dashboard trên Excel/Power BI

Giao diện kết quả:



TỔNG KẾT MÔN HỌC

Trong môn học này, ngoài những kiến thức đã được thầy giảng dạy trong tiết học thì những bài thực hành hàng tuần thầy giao cũng giúp em biết thêm rất nhiều về học phần Cơ sở dữ liệu này.

Tổng kết những việc làm được:

- Thiết lập được môi trường thực hành là hệ quản trị cơ sở dữ liệu MySQL.
- Thực hành được các câu lệnh truy vấn dữ liệu.
- Thực hành xây dựng được cấu trúc của một cơ sở dữ liệu.
- Thực hành các câu lệnh thao tác dữ liệu.
- Thực hành các câu lệnh sử dụng hàm trong truy vấn.
- Thiết kế cơ sở dữ liệu lưu trữ.
- Luyện tập về phụ thuộc hàm trong hệ quản trị cơ sở dữ liệu.
- Quản trị cơ sở dữ liệu.

Tổng kết những điều học được:

- Ứng dụng môn học để biết cách sắp xếp, phân tích và chọn lọc thông tin, dữ liệu trong nhiều lĩnh vực.
- Việc thực hành hàng tuần vừa giúp ôn tập lại các kiến thức được học trên lớp đồng thời bổ sung nhiều điều mới trong quá trình làm bài.
- Hiểu thêm về các ứng dụng của cơ sở dữ liệu trong thực tế (Giám sát truy cập, thiết kế, quản lý và lưu trữ thông tin có hệ thống, đảm bảo an toàn dữ liệu...)
- Tìm hiểu được cấu trúc của một cơ sở dữ liệu (Xây dựng lược đồ, truy vấn các bảng hệ thống trong CSDL)
- Khi thực hành đã gặp các lỗi về câu truy vấn (khóa, sai logic, sai cú pháp, timeout,...) qua đó rút ra được kinh nghiệm cho những lần sau.
- Biết thêm được nhiều kiến thức mới được thầy lồng ghép thông qua các bài thực hành.
- Biết cách tra cứu thông tin, xử lý vấn đề gặp phải.
- Nâng cao kỹ năng trình bày, thiết kế báo cáo.

Trong quá trình học tập có gặp nhiều khúc mắc và khó khăn, kiến thức bị bỏ sót nên bài báo cáo này đã giúp em tổng hợp lại kiến thức. Cuối cùng em xin gửi lời cảm ơn đến thầy Nguyễn Danh Tú và các bạn trong lớp đã giúp đỡ em trong việc học tập. Em xin chân thành cảm ơn!

Tài liệu tham khảo:

- 1) <https://www.mysqltutorial.org/>
- 2) <https://w3schools.com>
- 3) Slide bài giảng thầy Nguyễn Danh Tú.