

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH



HỆ CƠ SỞ DỮ LIỆU

ASSIGNMENT 2

GVHD: Trương Quỳnh Chi

SV: Võ Trịnh Xuân Nguyên – 1911736

Nguyễn Đăng Anh Khoa – 2010339

Trần Duy Chánh – 1912761

Nguyễn Kim Phương Trang – 1910620

Trần Quyết Thắng – 1915251

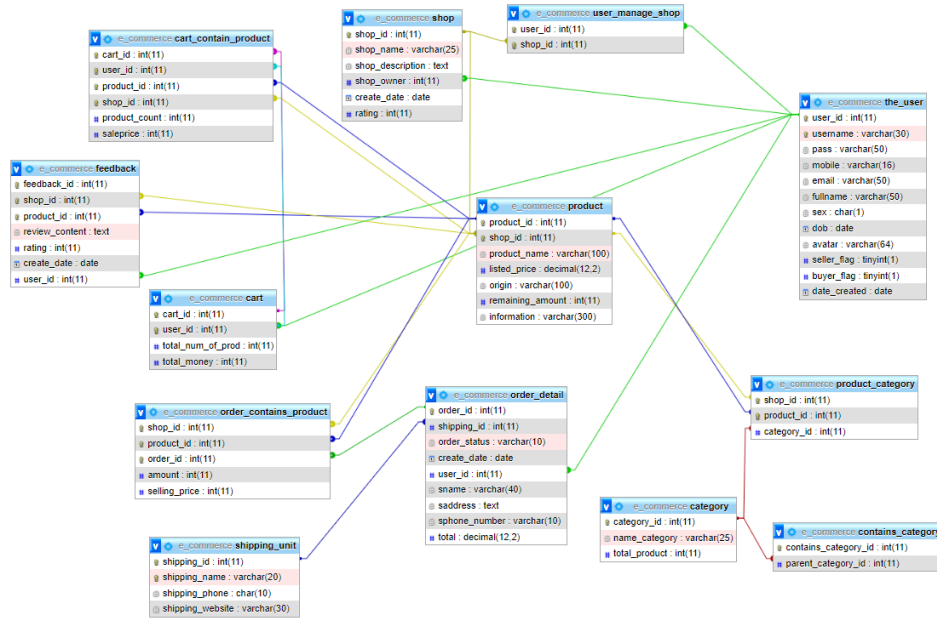
TP. Hồ Chí Minh, tháng 11 – 2021

I. Phần chung	3
1. Hiện thực các bảng dữ liệu đã thiết kế	3
2. Tạo các chỉ mục cần thiết hỗ trợ tăng tốc độ truy vấn: Phân tích lại câu truy vấn ban đầu cho ta kết quả:	4 5
3. Tạo dữ liệu mẫu ý nghĩa để minh họa:	5
II. Phần riêng	6
1. Thành viên 1:	6
2. Thành viên 2:	22
3. Thành viên 3:	37
4. Thành viên 4	65
5. Thành viên 5:	85

I. Phần chung

1. Hiện thực các bảng dữ liệu đã thiết kế

- Danh sách các bảng được tạo:
 - + the_user
 - + order_contains_product
 - + user_manage_shop
 - + shipping_unit
 - + cart_contain_product
 - + shop
 - + order_detail
 - + product_category
 - + feedback
 - + product
 - + cart
 - + category
 - + contains_category
- Code tạo bảng được đính kèm báo cáo.
- Lược đồ cơ sở dữ liệu:



2. Tạo các chỉ mục cần thiết hỗ trợ tăng tốc độ truy vấn:

a. Chỉ mục cho bảng the_user:

Vì bảng the_user có sử dụng câu truy vấn đối với cột fullname nên trường hợp này có thể sử dụng cột fullname làm index để tăng tốc độ truy vấn:

Xét câu truy vấn sau:

```
select * from the_user where fullname = 'Vo Trinh Xuan Nguyen';
```

Để phân tích cách MySQL thực hiện câu truy vấn này, ta sử dụng mệnh đề explain ở phía trước câu select.

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	the_user	ALL	NULL	NULL	NULL	NULL	10	Using where

Trước khi thêm index là cột fullname, MySQL phải duyệt toàn bộ 10 dòng của bảng the_user để tìm ra user nào có fullname là 'Vo Trinh Xuan Nguyen'.

Sau khi thêm index bằng câu lệnh sau:

```
alter table the_user
```

```
drop index if exists idx_fullname;

create index idx_fullname on the_user(fullname);
```

Phân tích lại câu truy vấn ban đầu cho ta kết quả:

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	the_user	ref	idx_fullname	idx_fullname	202	const	1	Using index condition

Như vậy sau khi thêm index thì MySQL chỉ cần tìm 1 hàng trong tổng cộng 10 hàng nhờ sử dụng index idx_fullname như được chỉ ra trong cột 'key' và 'possible_keys'.

Để kiểm tra lại các index được tạo trong bảng the_user, ta sử dụng câu lệnh

```
show index from the_user;
```

Kết quả:

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_commen
the_user 0		PRIMARY	1	user_id	A	10	NULL	NULL		BTREE		
the_user 0		username	1	username	A	10	NULL	NULL		BTREE		
the_user 1		idx_fullname	1	fullname	A	10	NULL	NULL		BTREE		

Như ta có thể thấy, ngoài hai index được tạo ra khi create table là index ở cột user_id (primary key) và cột username (unique constraint), ta còn có index là idx_fullname ở cột fullname vừa mới tạo ở câu lệnh trên.

3. Tạo dữ liệu mẫu ý nghĩa để minh họa:

- Code insert được đính kèm báo cáo.

II. Phần riêng

1. Thành viên 1:

Họ tên: Trần Quyết Thắng

MSSV: 1915251

1. Thủ tục insert dữ liệu:

- Mô tả chức năng:

+ Thêm một record vào bảng product với các trường shop_id, product_name, listed_price, origin, remaining_amount, information.

+ Kiểm tra các giá trị shop_id > 0, listed_price > 0, remaining_amount >= 0.

- Câu lệnh tạo thủ tục:

```
drop procedure if exists add_product;
delimiter //
create procedure add_product (
    in shop_id int,
    in product_name varchar(100),
    in listed_price decimal(12,2),
    in origin varchar(100),
    in remaining_amount int,
    in information varchar(300)
)
begin
    if shop_id < 1 then
        signal sqlstate '45000' set message_text = 'shop_id must be larger than 0';
    end if;

    if listed_price <= 0 then
        signal sqlstate '45000' set message_text = 'listed_price must be larger than 0';
    end if;

    if remaining_amount < 0 then
        signal sqlstate '45000' set message_text = 'listed_price must be larger than or
equal 0';
    end if;

    insert into
product(shop_id,product_name,listed_price,origin,remaining_amount,information)
    values (shop_id,product_name,listed_price,origin,remaining_amount,information);
end//
delimiter ;
```

- Câu lệnh thực thi thủ tục mẫu:

+ Nhập một record hợp lệ:

```
call add_product(5, 'Hat cat thoi gian', 12000.00, 'Campuchia', 10, 'Color: Yellow')
```

✓ MySQL trả về kết quả là trống rỗng (vd. không có hàng nào). (Truy vấn cần 0,0040 giây.)

```
call add_product(5, 'Hat cat thoi gian', 12000.00, 'Campuchia', 10, 'Color: Yellow');
```

[Sửa chung dòng] [Sửa] [Tạo mã PHP]

<input type="checkbox"/>	Sửa	Chép	Xóa bỏ	32	5	Hat cat thoi gian	12000.00	Campuchia	10	Color: Yellow
--------------------------	-----	------	--------	----	---	-------------------	----------	-----------	----	---------------


+ Nhập một record không hợp lệ:

```
call add_product(-2, 'Hat cat thoi gian', 0, 'Campuchia', -5, 'Color: Yellow')
```

Lỗi

Truy vấn SQL: [Chép](#)

```
call add_product(-2, 'Hat cat thoi gian', 0, 'Campuchia', -5, 'Color: Yellow');
```

MySQL đã nói: 

#1644 - shop_id must be larger than 0

2. Trigger:

2.1. Trigger BEFORE INSERT cập nhập lượng hàng còn lại của shop và lượng hàng người dùng đặt hàng, kiểm tra giá bán của sản phẩm.

- Mô tả chức năng:

+ Không được đặt hàng nhiều hơn lượng hàng còn lại, nếu đặt số lớn hơn tự động trả về bằng lượng hàng còn lại.

+ Tính toán lượng hàng còn lại sau khi người dùng đặt hàng thành công, kiểm tra nếu < 0 thì chỉ cho phép người dùng đặt hàng tối đa lượng hiện có và cập nhập lượng hàng còn lại về 0, ngược lại nếu ≥ 0 thì cho phép đặt hàng như bình thường và cập nhập lại lượng hàng còn lại của shop.

+ Kiểm tra nếu giá bán của sản phẩm nhỏ hơn giá niêm yết thì cập nhập giá bán bằng giá niêm yết.

- Câu lệnh tạo trigger:

```
DROP TRIGGER IF EXISTS before_insert_order_contains_product;
DELIMITER $$
CREATE TRIGGER before_insert_order_contains_product
BEFORE INSERT ON order_contains_product
FOR EACH ROW
BEGIN
    DECLARE newAmount int;
    DECLARE curAmount int;
    DECLARE curPrice decimal(12,2);

    SELECT remaining_amount INTO curAmount FROM product WHERE (product.product_id =
new.product_id and product.shop_id = new.shop_id);
    SET newAmount = curAmount - new.amount;
    IF newAmount < 0 THEN
        SET new.amount = curAmount;
        UPDATE product SET remaining_amount = 0 WHERE (product_id = new.product_id and
shop_id = new.shop_id);
    ELSE
        UPDATE product SET remaining_amount = newAmount WHERE (product_id =
new.product_id and shop_id = new.shop_id);
    END IF;

    SELECT listed_price INTO curPrice FROM product WHERE (product.product_id =
new.product_id and product.shop_id = new.shop_id);
    IF new.selling_price < curPrice THEN
        SET new.selling_price = curPrice;
    END IF;

END$$
DELIMITER ;
```

- Câu lệnh kiểm tra trigger hoạt động:

```
insert into order_contains_product(shop_id, product_id, order_id, amount, selling_price)
values (1, 1, 9, 2, 117000);
```

+ Trước khi thực hiện:

product_id	shop_id	product_name	listed_price	origin	remaining_amount	information
1	1	Áo thun nữ	200000.00	Hàn Quốc	20	Vải cotton, Màu trắng, Size M

+ Sau khi thực hiện:

product_id	shop_id	product_name	listed_price	origin	remaining_amount	information
1	1	Áo thun nữ	200000.00	Hàn Quốc	18	Vải cotton, Màu trắng, Size M

- Câu lệnh kiểm tra trigger hoạt động:

```
insert into order_contains_product(shop_id, product_id, order_id, amount, selling_price)
values (1, 1, 10, 100, 10);
```

+ Trước khi thực hiện

product_id	shop_id	product_name	listed_price	origin	remaining_amount	information
1	1	Áo thun nữ	200000.00	Hàn Quốc	18	Vải cotton, Màu trắng, Size M

+ Sau khi thực hiện

product_id	shop_id	product_name	listed_price	origin	remaining_amount	information
1	1	Áo thun nữ	200000.00	Hàn Quốc	0	Vải cotton, Màu trắng, Size M

shop_id	product_id	order_id	amount	selling_price
1	1	10	18	200000

2.2. Trigger AFTER DELETE cập nhập lại lượng hàng còn lại của shop khi người dùng xóa đơn đặt hàng.

- Mô tả chức năng:

+ Khi người dùng xóa hàng khỏi đơn hàng, lượng hàng còn lại sẽ được cộng tương ứng:

- Câu lệnh tạo trigger:

```
DROP TRIGGER IF EXISTS after_delete_order_contains_product;
DELIMITER $$
CREATE TRIGGER after_delete_order_contains_product
AFTER DELETE ON order_contains_product
FOR EACH ROW
BEGIN
    UPDATE product SET remaining_amount = remaining_amount + old.amount WHERE (product_id
    = old.product_id and shop_id = old.shop_id);
END$$
DELIMITER ;
```

- Câu lệnh kiểm tra trigger hoạt động:

```
DELETE FROM order_contains_product WHERE order_id = 10 AND shop_id = 1 AND product_id =
1;
```

- Kết quả:

+ Trước khi thực hiện:

		product_id	shop_id	product_name	listed_price	origin	remaining_amount	information
<input type="checkbox"/>	 Sửa  Chép  Xóa bỏ	1	1	Áo thun nữ	200000.00	Hàn Quốc	0	Vải cotton, Màu trắng, Size M

+ Sau khi thực hiện:

		product_id	shop_id	product_name	listed_price	origin	remaining_amount	information
<input type="checkbox"/>	Sửa Chép Xóa bỏ	1	1	Áo thun nữ	200000.00	Hàn Quốc	18	Vải cotton, Màu trắng, Size M

3. Thủ tục chứa câu SQL:

3.1. Thủ tục lấy danh sách các sản phẩm đã được đặt hàng bởi người dùng và sắp xếp theo thứ tự giảm dần về giá.

- Mô tả chức năng:

+ Thủ tục nhận tham số truyền vào là user_id và trả về danh sách các sản phẩm đã được đặt hàng bởi người dùng đó sắp xếp theo thứ tự giảm dần về giá.

- Câu lệnh tạo thủ tục:

```
DROP PROCEDURE IF EXISTS get_products_ordered_by_user_in_order_of_price;
DELIMITER $$
CREATE PROCEDURE get_products_ordered_by_user_in_order_of_price(
    in user_idx int
)
BEGIN
    SELECT DISTINCT product.product_id, product.product_name, product.listed_price FROM
    order_detail
    INNER JOIN order_contains_product AS ocp ON ocp.order_id = order_detail.order_id
    INNER JOIN product ON product.product_id = ocp.product_id AND product.shop_id =
    ocp.shop_id
    WHERE order_detail.user_id = user_idx ORDER BY product.listed_price DESC;
END;
```

- Câu lệnh kiểm tra thủ tục hoạt động:

```
call get_products_ordered_by_user_in_order_of_price(1);
```

- Kết quả:

product_id	product_name	listed_price
8	iPhone 12	20000000.00
1	Áo thun nữ	200000.00
25	Son lì	180000.00
3	Áo thun nam	150000.00
19	Tập Đọc 1	20000.00

3.2. Thủ tục lấy danh sách các khách hàng sắp xếp theo thứ tự giảm dần lượt đặt hàng của một shop.

- Mô tả chức năng:

+ Thủ tục nhận tham số truyền vào là shop_id và trả về danh sách danh sách các khách hàng sắp xếp theo thứ tự giảm dần lượt đặt hàng của shop đó.

- Câu lệnh tạo thủ tục:

```
DROP PROCEDURE IF EXISTS get_users_ordered_by_number_of_order_from_a_shop;
DELIMITER $$
CREATE PROCEDURE get_users_ordered_by_number_of_order_from_a_shop(
    in shop_idx int
)
BEGIN
    SELECT the_user.user_id, the_user.username, SUM(newtable.num) AS total_num FROM
the_user
    INNER JOIN (SELECT order_detail.user_id, COUNT(order_detail.user_id) AS num FROM shop
        INNER JOIN order_contains_product ON shop.shop_id =
order_contains_product.shop_id
        INNER JOIN order_detail ON order_detail.order_id =
order_contains_product.order_id
        WHERE shop.shop_id = 1 GROUP BY order_contains_product.order_id) AS
newtable
    ON the_user.user_id = newtable.user_id GROUP BY the_user.user_id ORDER BY total_num
DESC;
END;
```

- Câu lệnh kiểm tra thủ tục hoạt động:

```
call get_users_ordered_by_number_of_order_from_a_shop(1);
```

- Kết quả:

+ Tùy chọn

user_id	username	total_num
3	hotruongluong	4
4	nguyenthanhcong	3
1	voxuannguyen2001	2
2	nguyenphuoctri	1
8	linhchidepgai	1

4. Hàm:

4.2. Hàm tính tổng lượng tiền khách hàng đã đặt hàng thành công của một shop trong một năm dùng vòng lặp.

- Mô tả chức năng:

+ Hàm nhận 2 tham số là shop_id và năm.

+ Trả về tổng lượng tiền khách hàng đã đặt hàng thành công của shop đó trong năm đó.

- Câu lệnh tạo hàm:

```
DROP FUNCTION IF EXISTS calculate_total_sales_of_shop_a_year;
DELIMITER $$
CREATE function calculate_total_sales_of_shop_a_year(shop_idx int, yearx int)
RETURNS decimal
READS SQL DATA
DETERMINISTIC
BEGIN
    DECLARE total decimal(12,2) default 0;
    DECLARE exit_loop BOOLEAN default FALSE;
    DECLARE cur CURSOR FOR
        SELECT order_contains_product.selling_price
        FROM order_contains_product
        INNER JOIN order_detail ON order_detail.order_id =
order_contains_product.order_id
        WHERE year(order_detail.create_date) = yearx AND order_contains_product.shop_id =
shop_idx;
    DECLARE CONTINUE HANDLER FOR NOT FOUND SET exit_loop = TRUE;

    OPEN cur;
    the_loop: LOOP
        FETCH cur INTO iselling_price;
        IF exit_loop THEN
            LEAVE the_loop;
        END IF;
        SET total = total + iselling_price;
    END LOOP the_loop;
    CLOSE cur;
```

```
RETURN total;  
END $$  
DELIMITER ;
```

- Câu lệnh kiểm tra hàm:

```
select calculate_total_sales_of_shop_a_year(1,2021);
```

- Kết quả:

+ Tùy chọn

```
calculate_total_sales_of_shop_a_year(1,2021)  
786000
```

4.2. Hàm xếp hạng sản phẩm dựa trên rating.

- Mô tả chức năng:

+ Hàm nhận vào 2 tham số là shop_id và product_id.

+ Hàm trả về xếp hạng của sản phẩm, nếu rating trung bình của sản phẩm đó:

Lớn hơn 4.5: xếp hạng Great

Trong khoảng 4.0 - 4.5: xếp hạng Good

Trong khoảng 3.0 – 4.0: xếp hạng Fine

Thấp hơn hoặc bằng 3.0: xếp hạng Bad

- Câu lệnh tạo hàm:

```
DROP FUNCTION IF EXISTS rank_product_based_rating;  
DELIMITER $$  
CREATE FUNCTION rank_product_based_rating(shop_idx int, product_idx int)  
RETURNS varchar(10)  
READS SQL DATA  
DETERMINISTIC  
BEGIN  
    DECLARE total decimal(8,4) default 0;  
    DECLARE countx int default 0;  
    DECLARE average decimal(8,4);  
    DECLARE result varchar(10);  
    DECLARE irating int;  
    DECLARE exit_loop BOOLEAN;  
    DECLARE cur CURSOR FOR
```

```
SELECT rating
FROM feedback
WHERE shop_id = shop_idx AND product_id = product_idx;
DECLARE CONTINUE HANDLER FOR NOT FOUND SET exit_loop = TRUE;

OPEN cur;
the_loop: LOOP
    FETCH cur INTO irating;
    IF exit_loop THEN
        LEAVE the_loop;
    END IF;
    SET total = total + irating;
    SET countx = countx + 1;
END LOOP the_loop;
CLOSE cur;

IF countx > 0 THEN
    SET average = total/countx;
    IF average > 4.5 THEN
        SET result = 'Great';
    ELSEIF average > 4.0 THEN
        SET result = 'Good';
    ELSEIF average > 3.0 THEN
        SET result = 'Fine';
    ELSE
        SET result = 'Bad';
    END IF;
ELSE
    SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = "There are no feedback for this
product";
END IF;

RETURN result;
END $$
DELIMITER ;
```

- Câu lệnh kiểm tra hàm:

```
select rank_product_based_rating(1,1);
```

- Kết quả:

Tùy chọn

				feedback_id	shop_id	product_id	review_content	rating	create_date	user_id	
<input type="checkbox"/>		Sửa	 Chép	 Xóa bỏ	1	1	1	okay	3	2021-11-27	1
<input type="checkbox"/>		Sửa	 Chép	 Xóa bỏ	2	1	3	fine	4	2021-11-27	1
<input type="checkbox"/>		Sửa	 Chép	 Xóa bỏ	3	1	1	normal	3	2021-11-27	3
<input type="checkbox"/>		Sửa	 Chép	 Xóa bỏ	4	1	1	awesome	5	2021-11-27	8
<input type="checkbox"/>		Sửa	 Chép	 Xóa bỏ	5	1	1	amazming in shop_id = 1	4	2021-11-27	4

rank_product_based_rating(1,1)

Fine

5. Lập trình ứng dụng:

5.1. Giao diện hiển thị bảng order_detail:

Manage Orders									
TASK	All Orders								
	Order ID	Shipping ID	Date	Name	Phone	Address	Total Price	Status	Action
	1	1	2021-11-23	Nguyen Trang	0123456789	Phu Yen	99858030	success	Delete View
	2	2	2021-11-23	Tran Thang	0123456798	TP.HCM	147000	success	Delete View
	3	3	2021-11-27	Le Hoang Phuc	0123456700	TP.HCM	263000	success	Delete View
	4	4	2021-11-27	Le Tuyet Anh	0123456700	TP.HCM	126000	success	Delete View
TASK	5	1	2021-11-23	Nguyen Bao	0123456789	Phu Yen	348000	waiting	Delete View
	6	2	2021-11-23	Tran Hung	0123456798	TP.HCM	348000	success	Delete View
	7	3	2021-11-27	Thu Vi	0123456700	TP.HCM	348000	success	Delete View
	8	4	2021-11-27	Dan Anh	0123456700	TP.HCM	428000	success	Delete View
	9	3	2021-11-27	Lam Truong	0123456700	TP.HCM	526000	success	Delete View
	10	4	2021-11-27	Le Trung	0123456700	TP.HCM	3825000	success	Delete View
	11	3	2021-11-27	Tan Truong	0123456700	TP.HCM	285000	success	Delete View
	12	4	2021-11-27	Ngoc Trung	0123456700	TP.HCM	403000	success	Delete View
Delete All Insert Order									

- Nhấn nút Delete tại mỗi hàng để xóa khỏi table
- Nhấn Insert Order để thêm mới một Order
- Nhấn Delete All để xóa hết dữ liệu Order
- Nhấn View để xem chi tiết đơn hàng, tương ứng với table order_contains_product

5.2. Giao diện thêm mới vào bảng order_detail:

Manage Orders

TASK

Manage Orders

Manage Feedback

Manage Products

Manage Product in cart

Manage User

Insert Order

Shipping ID:

Date:

User ID:

Name:

Phone:

Address:

Status:

Insert New

- Thêm thành công:

Manage Orders

TASK

Manage Orders

Manage Feedback

Manage Products

Manage Product in cart

Manage User

Insert Order

Success!!

Shipping ID:

Date:

User ID:

Name:

Phone:

Address:

Status:

Insert New

5.3. Giao diện hiển thị bảng order_contains_product:

Manage Orders

TASK

Manage Orders

Manage Feedback

Manage Products

Manage Product in cart

Manage User

The Order Has:

Shop ID	Product ID	Amount	Price	Action
1	3	1	51000	<div style="display: inline-block; background-color: red; color: white; padding: 2px 5px; border-radius: 3px;">Delete</div> <div style="display: inline-block; background-color: green; color: white; padding: 2px 5px; border-radius: 3px; margin-left: 5px;">Edit</div>
2	7	3	22000	<div style="display: inline-block; background-color: red; color: white; padding: 2px 5px; border-radius: 3px;">Delete</div> <div style="display: inline-block; background-color: green; color: white; padding: 2px 5px; border-radius: 3px; margin-left: 5px;">Edit</div>
2	8	1	15000	<div style="display: inline-block; background-color: red; color: white; padding: 2px 5px; border-radius: 3px;">Delete</div> <div style="display: inline-block; background-color: green; color: white; padding: 2px 5px; border-radius: 3px; margin-left: 5px;">Edit</div>

[Delete All](#)
[Insert New](#)

5.4. Giao diện chỉnh sửa một record bảng order_contains_product:

Edit

TASK

Manage Orders

Manage Feedback

Manage Products

Manage Product in cart

Manage User

Success!!

Order ID:

2

Shop ID:

1

Product ID:

3

Amount:

10

Price:

1000000

Submit

5.5. Giao diện thêm mới vào bảng order_contains_product:

- Thêm thành công:

TASK

Manage Orders

Manage Feedback

Manage Products

Manage Product in cart

Manage User

Insert new order contains product

Success!!

Order ID:

Shop ID:

Product ID:

Amount:

Price:

Insert New

- Thêm thất bại:

TASK

Manage Orders

Manage Feedback

Manage Products

Manage Product in cart

Manage User

Insert new order contains product

Error! The values invalid

Order ID:

Shop ID:

Product ID:

Amount:

Price:

Insert New

5.6 Giao diện hiển thị product:

Manage Products

TASK

Manage Orders

Manage Feedback

Manage Products

Manage Product in cart

Manage User

Products Table

Choose the shop

Get all products ordered by user in order of price:

User ID

Submit

Get all users ordered by the number of order from a shop:

Shop ID

Submit

- Chọn shop để hiển thị các sản phẩm của shop đó.

Manage Products

TASK

Manage Orders

Manage Feedback

Manage Products

Manage Product in cart

Manage User

Products Table

Choose the shop

Lilyeyewear
FPT
Houseware
Book World
Cosmetics

Lilyeyewear Shop

	Listed Price	Origin	Amount	Information
	200000.00	Hàn Quốc	0	Vải cotton, Màu trắng, Size M
	200000.00	Hàn Quốc	15	Vải cotton, Màu đen, Size S
	150000.00	Việt Nam	20	Vải cotton, Màu đen, Size L
4 Áo thun nam	150000.00	Việt Nam	10	Vải cotton, Màu đen, Size XL
5 Giày thể thao nữ	500000.00	Việt Nam	5	Giày thể thao nữ buộc dây siêu nhẹ, Màu trắng, Size 38
6 Giày thể thao nam	550000.00	Việt Nam	8	Giày thể thao nam êm ái, Màu đen, Size 42

5.7. Tính năng lấy danh sách sản phẩm đã đặt hàng của người dùng sắp xếp theo giá tiền giảm dần:

TASK

Manage Orders

Manage Feedback

Manage Products

Manage Product in cart

Manage User

Products Table

Get all products ordered by user in order of price:

User ID

Submit

ID	Name	Listed Price	Origin	Amount	Information
8	iPhone 12	20000000.00			
10	Samsung Galaxy A03	4990000.00			
1	Áo thun nữ	200000.00			
25	Son lì	180000.00			
3	Áo thun nam	150000.00			
19	Tập Đọc 1	20000.00			

5.8. Tính năng lấy danh sách người dùng đã đặt hàng tại một shop theo thứ tự giảm dần lượt đặt hàng:

TASK

Manage Orders

Manage Feedback

Manage Products

Manage Product in cart

Manage User

Products Table

Get all users ordered by the number of order from a shop:

Shop ID

Submit

User ID	UserName	Total Num
3	hotruongluong	4
4	nguyenthanhcong	3
1	voxuannguyen2001	2
2	nguyenphuoctri	1
8	linhchidepgai	1

2. Thành viên 2:

Họ tên: Võ Trịnh Xuân Nguyên

MSSV: 1911736

1. Thủ tục để thêm dữ liệu vào một bảng dữ liệu

- Chức năng: thêm một user mới vào bảng the_user, có validate input và hiển thị thông báo lỗi có ý nghĩa.
- Câu lệnh tạo thủ tục:

```
drop procedure if exists add_user;

delimiter //

create procedure add_user (
    in username varchar(30),
    in pass varchar(50),
    in mobile varchar(12),
    in email varchar(50),
    in fullname varchar(50),
    in sex char(1),
    in dob date,
    in avatar varchar(64),
    in seller_flag boolean,
    in buyer_flag boolean
)
begin
    if not (username regexp '^[a-z0-9_]*$') then
        signal sqlstate '45000'
```

```
        set message_text = 'username can only contains lowercase characters,
numbers, dot(.) and underscore(_)';

    end if;

    if char_length(username) < 6 or char_length(username) > 30 then

        signal sqlstate '45000'

        set message_text = 'username\'s length must be between 6 and 30';

    end if;

    if char_length(pass) < 8 then

        signal sqlstate '45000'

        set message_text = 'password\'s length must be at least 8
characters';

    end if;

    if not (mobile regexp '^[0-9]*$') then

        signal sqlstate '45000'

        set message_text = 'mobile number can only contains numbers';

    end if;

    if not (email regexp '@.*\.') then

        signal sqlstate '45000'

        set message_text = 'email is not valid';

    end if;
```

```
insert into the_user (username, pass, mobile, email, fullname, sex, dob,
avatar, seller_flag, buyer_flag)

values (username, pass, mobile, email, fullname, sex, dob, avatar,
seller_flag, buyer_flag);

end//

delimiter ;
```

- Câu lệnh thực thi thủ tục mẫu:

```
call add_user (
'voxuannguyen2001',
'12345678',
'0397003301',
'voxuannguyen2001@gmail.com',
'Vo Trinh Xuan Nguyen',
'M',
date('2001-02-18'),
concat('img/voxuannguyen2001/avatar_', date_format(now(), "%Y_%j_%H_%i_%s")),
false,
true
);
```

- Kết quả từ DBMS:

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0050 seconds.)

```
call add_user ( 'voxuannguyen01', '12345678', '0397003301', 'voxuannguyen2001@gmail.com', 'Vo Trinh Xuan Nguyen', 'M', date('2001-02-18'),
concat('img/voxuannguyen2001/avatar_', date_format(now(), "%Y_%j_%H_%i_%s")), false, true );
```

[\[Edit inline \]](#) [\[Edit \]](#) [\[Create PHP code \]](#)

2. Trigger để kiểm soát các hành động INSERT, UPDATE, DELETE trên một số bảng đã tạo:

2.1. Trigger 1:

- Chức năng: Sau khi tạo một cửa hàng mới trong bảng shop, thêm một entry vào bảng user_manage_shop để thể hiện quan hệ chủ cửa hàng quản lý cửa hàng đó.
- Câu lệnh tạo trigger:

```
drop trigger if exists on_create_shop;

delimiter //

create trigger on_create_shop after insert on shop
for each row
begin
    update the_user set seller_flag = true
    where user_id = new.shop_owner;

    insert into user_manage_shop values (shop_owner, new.shop_id);
end;

delimiter ;
```

- Câu lệnh kiểm tra trigger hoạt động:

```
insert into shop (shop_name, shop_description, shop_owner, create_date)
values ("Nguyen's store", "Everything you need for your computers", 13,
CURRENT_DATE);
```

- Kết quả từ DBMS:

Table shop:

shop_id	shop_name	shop_description	shop_owner	create_date
1	Nguyen's store	Everything you need for your computers	13	2021-12-02

Table user_manage_shop:

user_id	shop_id	start_date
13	1	2021-12-02

2.2. Trigger 2:

- Chức năng: Sau khi thêm một entry mới vào bảng user_manage_shop, tìm người dùng tương ứng trong bảng the_user, set seller_flag của người dùng đó thành true và tăng giá trị của cột shop_manage_count lên 1 đơn vị.
- Câu lệnh tạo trigger:

```
drop trigger if exists on_insert_user_manage_shop;

delimiter //

create trigger on_insert_user_manage_shop before insert on user_manage_shop
for each row
begin
    update the_user set seller_flag = true, shop_manage_count = shop_manage_count
+ 1
    where user_id = new.user_id;
end;//

delimiter ;
```

- Câu lệnh kiểm tra trigger hoạt động:

```
insert into user_manage_shop(user_id, shop_id) values (1,1);
```

- Kết quả từ DBMS:

Table user_manage_shop:

user_id	shop_id	start_date
1	1	0000-00-00

Table the_user:

username	pass	mobile	email	fullname	sex	dob	avatar	seller_flag	buyer_flag	date_created	shop_manage_count
voxuannnguyen2001	12345678	0397003301	voxuannnguyen2001@gmail.com	Vo Trinh Xuan Nguyen	M	2001-02-18	img/voxuannnguyen2001/avatar_2021_337_00_06_18	1	1	2021-12-03	1

3. Thủ tục trong đó chỉ chứa các câu truy vấn để hiển thị dữ liệu và tham số đầu vào là các giá trị trong mệnh đề WHERE và/hoặc Having

3.1. Thủ tục 1:

- Chức năng: Lấy thông tin các shop được quản lý bởi một user nào đó (bao gồm: tài khoản chủ shop, tên chủ shop, ngày được tạo shop, ngày bắt đầu làm việc)
- Câu lệnh tạo thủ tục:

```
drop procedure if exists get_shops_managed_by_user;

delimiter //

create procedure get_shops_managed_by_user (in _user_id int)
begin
    select owner.username as username, owner.fullname as owner_name, s.shop_name
as shop_name, s.create_date as create_date, ums.start_date as start_date
    from the_user as u, user_manage_shop as ums, shop as s, the_user as owner
    where u.user_id = _user_id and u.user_id = ums.user_id and ums.shop_id =
s.shop_id and s.shop_owner = owner.user_id
    order by start_date;
end //

delimiter ;
```

- Câu lệnh thực thi thủ tục:

```
call get_shops_managed_by_user(2);
```

- Kết quả từ DBMS:

username	owner_name	shop_name	create_date	start_date
voxuannguyen2001	Vo Trinh Xuan Nguyen	SAMSUNG	2021-11-01	0000-00-00
nguyenphuoctri	Nguyen Phuoc Tri	Lilyeyewear	2021-11-02	2021-12-03

3.2. Thủ tục 2:

- Chức năng: Lấy thông tin về các người dùng có số lượng order lớn hơn một mức nhất định (bao gồm: user id, username, họ và tên, và số lượng order của người dùng ấy).
- Câu lệnh tạo thủ tục:

```
drop procedure if exists get_order_count_all_users;
delimiter //
create procedure get_order_count_all_users (in min_count int)
begin
    select u.user_id as user_id, u.username as username, u.fullname as fullname,
count(*) as order_count
    from the_user as u, order_detail as od
    where u.user_id = od.user_id
    group by u.user_id
    having count(*) >= min_count
    order by count(*), u.user_id;
end //
delimiter ;
```

- Câu lệnh thực thi thủ tục:

```
call get_order_count_all_users(0);
```

- Kết quả từ DBMS:

user_id	username	fullname	order_count
1	voxuannguyen2001	Vo Trinh Xuan Nguyen	2
2	nguyenphuoctri	Nguyen Phuoc Tri	2
3	hotruongluong	Ho Truong Luong	2
4	nguyenthanhcong	Nguyen Thanh Cong	2

4. Hàm tính toán:

4.1. Hàm 1:

- Chức năng: Xác định loại khách hàng (silver/gold/platinum) của một user dựa trên số order mà người đó đặt trong năm nay.
- Câu lệnh tạo hàm:

```
drop function if exists get_buyer_membership;

delimiter //

create function get_buyer_membership (_user_id integer)
returns varchar(8)
deterministic
begin
    declare cur_year int default year(curdate());
    declare order_cnt int default 0;
    declare membership varchar(8) default 'Bronze';

    set order_cnt = (select count(*) from order_detail
                    where user_id = _user_id and year(create_date) =
cur_year);

    if order_cnt >= 10 and order_cnt < 30 then
        set membership = 'Silver';
    elseif order_cnt >= 30 and order_cnt < 50 then
        set membership = 'Gold';
    else
        set membership = 'Platinum';
    end if;

    return membership;
```

```
end //
```

```
delimiter ;
```

- Câu lệnh thực thi thủ tục:

```
select get_buyer_membership(0);
```

- Kết quả từ DBMS:

```
get_buyer_membership(0)
```

```
Silver
```

4.2. Hàm 2:

- Chức năng: Tính tổng số tiền đã sử dụng của một người dùng dựa trên user id
- Câu lệnh tạo hàm:

```
drop function if exists get_total_money_spent;
```

```
delimiter //
```

```
create function get_total_money_spent(_user_id integer)
```

```
returns integer
```

```
deterministic
```

```
begin
```

```
    declare done bool default false;
```

```
    declare total_money int default 0;
```

```
    declare price, amt int;
```

```
    declare cur cursor for select selling_price, amount from  
order_contains_product where order_id = _order_id;
```

```
    declare continue handler for not found set done = true;
```

```
if not exists (select count(*) from the_user where user_id = _user_id) then
    signal sqlstate '45000'
    set message_text = 'Cannot found a user';
end if;

open cur;

read_loop: loop
    fetch cur into price, amt;
    if done then
        leave read_loop;
    end if;
    set total_money = total_money + price * amt;
end loop;
close cur;
return total_money;
end //
```

delimiter ;

- Câu lệnh thực thi hàm:

```
select get_total_money_spent(1);
```

- Kết quả từ DBMS:

get_total_money_spent(1)

370000

5. Lập trình ứng dụng

5.1 Giao diện trang quản lý người dùng: Hiển thị tất cả các người dùng trong CSDL

Manage Users

TASK

Manage Orders

Manage Feedback

Manage Products

Manage Product in cart

Manage User

Category

All Users

Search

Search

User ID	UserName	Password	Mobile	Email	Fullname	Sex	Date of birth	Avatar
1	voxuannnguyen2001	12345678	0397003301	voxuannnguyen2001@gmail.com	Vo Trinh Xuan Nguyen	M	2001-02-18	img/vc
2	nguyenphuotri	12345689	0914370975	ngphuotri.bmt@gmail.com	Nguyen Phuoc Tri	M	2001-08-28	img/ng
3	hotruongluong	htluong123	01678934556	truongluongho.ctnd@gmail.com	Ho Truong Luong	M	2001-10-08	img/hc
4	nguyenthanhcong	ntcong01	01697053312	ntcong@hcmut.edu.vn	Nguyen Thanh Cong	M	2001-07-15	img/ng
5	nguyenhaison	sonhainguyenbmt	0915559871	nhson2001@hcmut.edu.vn	Nguyen Hai Son	M	2001-02-24	img/ng
6	nguyenminhhai	minhhaivippro	0913737095	minhhaideptrai@gmail.com	Nguyen Minh Hai	M	2001-10-19	img/ng
7	yennhi_01	12345678	091334987	yennhi_le@hcmut.edu.vn	Le Nauven	F	2001-04-19	img/ye

Delete All
Add User
User statistics

Manage Users

TASK

Manage Orders

Manage Feedback

Manage Products

Manage Product in cart

Manage User

Category

All Users

Search

Search

Avatar	Seller	Buyer	Date Created	Action
img/voxuannnguyen2001/avatar_2021_337_00_29_00	Yes	Yes	2021-12-03	Delete Edit Get Shops Get Orders
img/nguyenphuotri/avatar_2021_337_00_29_00	Yes	Yes	2021-12-03	Delete Edit Get Shops Get Orders
img/hotruongluong/avatar_2021_337_00_29_00	Yes	Yes	2021-12-03	Delete Edit Get Shops Get Orders
img/nguyenthanhcong/avatar_2021_337_00_29_00	Yes	Yes	2021-12-03	Delete Edit Get Shops Get Orders
img/nguyenhaison/avatar_2021_337_00_29_00	Yes	Yes	2021-12-03	Delete Edit Get Shops Get Orders
img/nguyenminhhai/avatar_2021_337_00_29_00	No	Yes	2021-12-03	Delete Edit Get Shops Get Orders
img/yennhi_le/avatar_2021_337_00_29_00	No	Yes	2021-12-03	Delete Edit Get Shops Get Orders

Delete All
Add User
User statistics

- Chức năng:
 - + Xóa một người dùng khỏi CSDL (nút delete)
 - + Thay đổi thông tin của một người dùng (nút edit)
 - + Xem thông tin các shop mà người đó đang quản lý (nút view shops)
 - + Thêm một người dùng mới (nút add user)
 - + Xem thông tin xếp hạng người dùng theo số đơn hàng đã đặt (user statistics)
 - + Xóa tất cả người dùng (delete all)
 - + Tìm kiếm người dùng theo username (input bar đầu tiên)
 - + Tìm kiếm người dùng theo fullname (input bar thứ hai)
 - + Xem danh sách các đơn hàng của một người dùng (nút get orders)
- Tìm kiếm người dùng theo họ và tên:

Manage Users

TASK

Manage Orders

Manage Feedback

Manage Products

Manage Product in cart

Manage User

Category

All Users

Search

Search

User ID	UserName	Password	Mobile	Email	Fullname	Sex	Date of birth	Avatar
1	voxuannguyen2001	12345678	0397003301	voxuannguyen2001@gmail.com	Vo Trinh Xuan Nguyen	M	2001-02-18	img/vc
2	nguyenphuoctri	12345689	0914370975	ngphuoctri.bmt@gmail.com	Nguyen Phuoc Tri	M	2001-08-28	img/ng
3	hotruongluong	htluong123	01678934556	truongluongho_ctnd@gmail.com	Ho Truong Luong	M	2001-10-08	img/hc
4	nguyenthanhcong	ntcong01	01697053312	ntcong@hcmut.edu.vn	Nguyen Thanh Cong	M	2001-07-15	img/ng
5	nguyenhaison	sonhainguyenbmt	0915559871	nhson2001@hcmut.edu.vn	Nguyen Hai Son	M	2001-02-24	img/ng
6	nguyenminhthai	minhhaivippro	0913737095	minhhaideptrai@gmail.com	Nguyen Minh Hai	M	2001-10-19	img/ng
7	yennhi_01	12345678	091334987	yennhi_le@hcmut.edu.vn	Le Nouven	F	2001-04-19	img/ye

Delete All
Add User
User statistics

5.2. Giao diện thay đổi thông tin người dùng:

Các trường có thể thay đổi: fullname, mobile, email, date of birth, avatar, sex, user type.

TASK

Manage Orders

Manage Feedback

Manage Products

Manage Product in cart

Manage User

Category

Edit user

Username *

voxuannguyen2001

Fullname *

Vo Trinh Xuan Nguyen

Mobile

0397003301

Email

voxuannguyen2001@gmail.com

Date of birth

02/18/2001

Select Avatar:

Choose File

No file chosen

Sex

☒ Male
 ☐ Female

User type

☒ Buyer
 ☒ Seller

Edit User

5.3. Giao diện xem thông tin các shop mà người dùng quản lý:

TASK

Manage Orders

Manage Feedback

Manage Products

Manage Product in cart

Manage User

Category

Shops Managed By Users

Owner	Owner's fullname	Shop name	Shop creation date	Start date
voxuannguyen2001	Vo Trinh Xuan Nguyen	SAMSUNG	2021-11-01	0000-00-00
nguyenphuoctri	Nguyen Phuoc Tri	Lilyeyewear	2021-11-02	2021-12-03

5.4. Giao diện thêm người dùng mới:

Manage Users

TASK

Manage Orders

Manage Feedback

Manage Products

Manage Product in cart

Manage User

Category

Add a new user

Username *

Password *

Fullname *

Mobile

Email

Date of birth

Select Avatar

Choose File

No file chosen

Sex
☐ Male ☐ Female

User type
☒ Buyer ☐ Seller

Add New User

- Thông báo lỗi có ý nghĩa khi thông tin người dùng nhập không hợp lệ

Manage Users

TASK

Manage Orders

Manage Feedback

Manage Products

Manage Product in cart

Manage User

Category

Add a new user

Username *

Password *

Fullname *

Mobile

Email

Date of birth

Select Avatar

Choose File

No file chosen

Sex
☐ Male ☐ Female

User type
☒ Buyer ☐ Seller

Add New User

username can only contains lowercase characters, numbers, dot(.) and underscore(_)

5.5. Giao diện xếp hạng người dùng theo số đơn hàng đã đặt: có chức năng tìm kiếm với số đơn hàng tối thiểu

Manage Users

TASK
Manage Orders
Manage Feedback
Manage Products
Manage Product in cart
Manage User
Category

User statistics
Minimum Number of Orders:

User ID	Username	Fullname	Number of orders
1	voxuannguyen2001	Vo Trinh Xuan Nguyen	2
2	nguyenphuoctri	Nguyen Phuoc Tri	2
3	hotruongluong	Ho Truong Luong	2
4	nguyenthanhcong	Nguyen Thanh Cong	2

- Tìm các người dùng đã đặt trên 3 đơn hàng:

Manage Users

TASK
Manage Orders
Manage Feedback
Manage Products
Manage Product in cart
Manage User
Category

User statistics
There are no users whose number of orders is at least 3

3. Thành viên 3:

Họ tên: Nguyễn Đặng Anh Khoa

MSSV: 2010339

- Mô tả Table cart_contain_product

a. Column (các thuộc tính)

- cart_id: ID giỏ hàng
- user_id: ID người dùng sở hữu giỏ hàng đó
- product_id: ID của loại sản phẩm có trong giỏ hàng
- shop_id: ID của shop bán loại sản phẩm có ID là product_id
- product_count: số sản phẩm của loại sản phẩm có ID là product_id trong giỏ hàng.
- saleprice: giá bán của loại sản phẩm

**** Primary key: (cart_id, user_id, product_id, shop_id)**

- b. Row (các record):* các loại sản phẩm (bản bởi shop) có trong giỏ hàng của người dùng trong database với các thuộc tính gồm ID sản phẩm, số sản phẩm thuộc loại và giá bán.

c. Table hiển thị từ DBMS (qua câu lệnh select sau khi insert một vài record)

550 • `select * from cart_contain_product`

	cart_id	user_id	product_id	shop_id	product_count	saleprice
▶	1	1	1	1	2	10000
	1	1	7	2	1	100000
	1	1	13	3	5	21000
	2	2	28	5	1	10000
	2	2	29	5	2	100000
	2	2	30	5	2	200000
	3	3	2	1	4	10000
	3	3	8	2	1	50000
	3	3	14	3	2	20000
	4	4	19	4	2	10000

- Mô tả Table cart (cart là thực thể yếu của the_user)

a. Column (các thuộc tính)

- cart_id: ID giỏ hàng
- user_id: ID người dùng sở hữu giỏ hàng đó

**** Primary key: (cart_id, user_id)**

- b. Row (các record):* ID giỏ hàng và ID người dùng sở hữu giỏ hàng đó.

1. Thủ tục insert dữ liệu

- Mô tả chức năng: Thêm mới một record vào table cart_contain_product (khi một người dùng nào đó thêm một loại sản phẩm vào giỏ hàng của họ) với các tham số đầu vào là cart_id, user_id, product_id, shop_id, product_count, saleprice.
- Câu lệnh tạo thủ tục:

-- tạo constraint giới hạn số lượng sản phẩm của một loại sản phẩm trong giỏ hàng từ 1 -> 10.

```
alter table cart_contain_product
add constraint product_range
check (product_count between 1 and 10);
```

-- tạo procedure

```
drop procedure if exists add_product_into_cart
delimiter //
create procedure add_product_into_cart (
    in cart_id int,
    in user_id int,
    in product_id int,
    in shop_id int,
    in product_count int,
    in saleprice integer
)
begin
    insert into cart_contain_product (cart_id, user_id, product_id, shop_id,
product_count, saleprice)
    values (cart_id, user_id, product_id, shop_id, product_count, saleprice);
end//
delimiter ;
```

- Câu lệnh thực thi thủ tục mẫu:
 - a. Record được insert có product_count hợp lệ (thuộc đoạn 1 đến 10).

```
call add_product_into_cart(2,2,30,5,1,200000);
```

- Kết quả màn hình hiển thị từ DBMS.

```

401 call add_product_into_cart(2,2,30,5,1,200000);
402
403 • select * from cart_contain_product;
404
405

```

Result Grid

	cart_id	user_id	product_id	shop_id	product_count	saleprice
	1	1	1	1	2	10000
	1	1	7	2	1	100000
	1	1	13	3	5	21000
▶	2	2	30	5	1	200000
	3	3	2	1	4	10000
	3	3	8	2	1	50000
	3	3	14	3	2	20000
	4	4	19	4	2	10000
	4	4	25	5	2	20000
	5	5	20	4	2	30000
	5	5	26	5	2	40000
	7	7	3	1	2	60000

Output

Action Output

#	Time	Action
✓ 103	15:31:18	alter table cart_contain_product add constraint product_range check (product_count bet...
✓ 104	15:31:50	create procedure add_product_into_cart (in cart_id int, in user_idint, in product_id int, i...
✓ 105	15:35:31	call add_product_into_cart(2,2,30,5,1,200000)
✓ 106	15:36:26	select * from cart contain product LIMIT 0, 1000

b. Record được insert có product_count không hợp lệ (product_count = 11).

```
call add_product_into_cart(2,2,30,5,11,200000);
```

```

401
402 call add_product_into_cart(2,2,30,5,11,200000);
403
404

```

Output

Action Output

#	Time	Action	Message
✓ 104	15:31:50	create procedure add_product_into_cart (in cart_id int, in user_idint, in product_id int, i...	0 row(s) affected
✓ 105	15:35:31	call add_product_into_cart(2,2,30,5,1,200000)	1 row(s) affected
✓ 106	15:36:26	select * from cart_contain_product LIMIT 0, 1000	16 row(s) returned
✗ 107	15:41:32	call add_product_into_cart(2,2,30,5,11,200000)	Error Code: 3819. Check constraint 'product_range' is violated.

2. Trigger

2.1 Trigger BEFORE INSERT kiểm tra điều kiện về tổng số loại sản phẩm trong giỏ hàng khi một user_id thêm loại sản phẩm (product_id, shop_id) vào giỏ hàng cart_id.

- Mô tả chức năng: Giả sử một giỏ hàng chỉ có thể chứa tối đa 3 loại sản phẩm. Khi một người dùng có user_id thêm loại sản phẩm (product_id, shop_id) vào giỏ hàng cart_id của mình, kiểm tra xem giỏ hàng của họ đã đầy chưa. Dùng aggregate function COUNT(*) để đếm số record trong table cart_contain_product có user_id là user_id của lệnh insert mới. Nếu đã đầy thì ném ra lỗi với message "cart of user has been fulfilled. Can't add product!"
- Câu lệnh tạo trigger

```
drop trigger if exists cart_full_check
delimiter $$
create trigger cart_full_check before insert on cart_contain_product
for each row
begin
    DECLARE prodtype_count INT DEFAULT 0;
    SELECT COUNT(*) INTO prodtype_count
    FROM cart_contain_product
    WHERE user_id = new.user_id;

    if (prodtype_count = 3) then
        begin
            signal sqlstate '45000' set message_text = "cart of user has
            been fulfilled. Can't add product!";
        end;
    end if;
end;
$$
Delimiter ;
```

- Câu lệnh kiểm tra trigger hoạt động
- 1. Thủ tục insert dữ liệu với các trường dữ liệu cần nhập thỏa mãn điều kiện trigger
`call add_product_into_cart(2,2,29,5,2,100000);`
- Kết quả màn hình hiển thị từ DBMS.


```

427 • call add_product_into_cart(2,2,29,5,2,100000);
428 • select * from cart_contain_product;
429

```

Result Grid

	cart_id	user_id	product_id	shop_id	product_count	saleprice
▶	1	1	1	1	2	10000
	1	1	7	2	1	100000
	1	1	13	3	5	21000
	2	2	29	5	2	100000
	2	2	30	5	1	200000
	3	3	2	1	4	10000
	3	3	8	2	1	50000
	3	3	14	3	2	20000
	4	4	19	4	2	10000
	4	4	25	5	2	20000
	5	5	20	4	2	30000

cart_contain_product 3

Output

Action Output

#	Time	Action	Message
✓ 109	16:07:01	create trigger cart_full_check before insert on cart_contain_product for each row begin D...	0 row(s) affected
✓ 110	16:08:26	select * from cart_contain_product LIMIT 0, 1000	16 row(s) returned
✓ 111	16:09:56	call add_product_into_cart(2,2,29,5,2,100000)	1 row(s) affected
✓ 112	16:10:01	select * from cart_contain_product LIMIT 0, 1000	17 row(s) returned

- Trước khi gọi thủ tục insert, trong giỏ hàng của người dùng có user_id = 2 **chỉ có 1 loại sản phẩm** (product_id, shop_id) = (30,5) nên giỏ hàng **vẫn chưa đầy**. Vì vậy, lệnh gọi thủ tục insert thỏa mãn điều kiện trigger và một record mới được insert vào table cart_contain_product.

- Thủ tục insert dữ liệu với các trường dữ liệu cần nhập không thỏa mãn điều kiện trigger

```
call add_product_into_cart(1,1,29,5,1,100000);
```

- Kết quả màn hình hiển thị từ DBMS.

```

428 • call add_product_into_cart(1,1,29,5,1,100000);
429 • select * from cart_contain_product;

```

Output

Action Output

#	Time	Action	Message
✓ 111	16:09:56	call add_product_into_cart(2,2,29,5,2,100000)	1 row(s) affected
✓ 112	16:10:01	select * from cart_contain_product LIMIT 0, 1000	17 row(s) returned
✓ 113	16:14:35	select * from cart_contain_product LIMIT 0, 1000	17 row(s) returned
✗ 114	16:15:09	call add_product_into_cart(1,1,29,5,1,100000)	Error Code: 1644. cart of user has been fulfilled. Can't add product!

- Trước khi gọi thủ tục insert, trong giỏ hàng của người dùng có user_id = 1 **đã có 3 loại sản phẩm** (product_id, shop_id) = (1,1); (product_id, shop_id) = (7,2); (product_id, shop_id) = (13,3) nên giỏ hàng **đã đầy**. Vì vậy, lệnh gọi thủ tục insert không thỏa mãn điều kiện trigger và ném ra lỗi với message "cart of user has been fulled. Can't add product!"

cart_id	user_id	product_id	shop_id	product_count	saleprice
1	1	1	1	2	10000
1	1	7	2	1	100000
1	1	13	3	5	21000
2	2	29	5	2	100000
2	2	30	5	1	200000
3	3	2	1	4	10000
3	3	8	2	1	50000
3	3	14	3	2	20000
4	4	19	4	2	10000

2.2 Trigger AFTER INSERT cập nhật giá trị cột cho table cart sau khi một user_id thêm loại sản phẩm (product_id, shop_id) vào giỏ hàng cart_id.

- Mô tả chức năng: Sau khi một user_id thêm loại sản phẩm (product_id, shop_id) vào giỏ hàng cart_id của mình thì cập nhật giá trị cho cột total_num_of_prod (tổng số sản phẩm của tất cả các loại sản phẩm) và total_money (tổng tiền sản phẩm) của record ứng với giỏ hàng của người dùng trong table **cart**. total_num_of_prod và total_money vì là các thuộc tính dẫn xuất nên sẽ được thêm vào table **cart** sau khi create table.
 - + total_num_of_prod = total_num_of_prod hiện tại + new.product_count (từ lệnh insert mới vào cart_contain_product)
 - + total_money = total_money hiện tại + new.product_count*new.saleprice
- Câu lệnh tạo trigger:

```
-- add column total_num_of_prod and total_money to table cart
ALTER TABLE cart ADD COLUMN total_num_of_prod INT DEFAULT 0;
ALTER TABLE cart ADD COLUMN total_money INT DEFAULT 0;

drop trigger if exists update_cart_with_total_quantity
delimiter $$
create trigger update_cart_with_total_quantity after insert on
cart_contain_product
for each row
```

```
begin
-- update total number of products
  update cart
  set total_num_of_prod = (cart.total_num_of_prod + new.product_count)
  where cart.cart_id = new.cart_id;

-- update total money of products in cart
  update cart
  set total_money = (cart.total_money + new.product_count*new.saleprice)
  where cart.cart_id = new.cart_id;
end;
$$
Delimiter ;
```

- Câu lệnh kiểm tra trigger hoạt động
- Giả sử table cart_contain_product đang không có record và giá trị default của các cột total_num_of_prod và total_money trong table cart là 0. Người user_id = 1 sẽ lần lượt thêm vào giỏ hàng của mình 3 loại sản phẩm. Ta sẽ xem xét quá trình insert vào table cart_contain_product các record và xem sự cập nhật giá trị total_num_of_prod và total_money của giỏ hàng người user_id = 1 ở table cart qua các lần insert.

+ Insert lần 1:

```
insert into cart_contain_product (cart_id, user_id, product_id, shop_id,
product_count, saleprice) values (1,1,1,1,2,10000);
```

⇒ Kết quả màn hình hiển thị từ DBMS.

```

460 • insert into cart_contain_product (cart_id, user_id, product_id, shop_id, product_count, saleprice)
461 • values (1,1,1,1,2,10000);
462 • select * from cart;
463

```

Result Grid				
Filter Rows: <input type="text"/>				
Edit:				
Export/Import:				
Wrap Cell Content:				
cart_id	user_id	total_num_of_prod	total_money	
1	1	2	20000	
2	2	0	0	
3	3	0	0	
4	4	0	0	
5	5	0	0	
6	6	0	0	

cart 1 x Apply

Output

Action Output			
#	Time	Action	Message
90	16:53:40	ALTER TABLE cart ADD COLUMN total_money INT DEFAULT 0	0 row(s) affected Records: 0 Duplicates: 0 Warning
91	16:53:50	drop trigger if exists update_cart_with_total_quantity	0 row(s) affected
92	16:53:59	create trigger update_cart_with_total_quantity after insert on cart_contain_product for eac...	0 row(s) affected
93	16:54:12	insert into cart_contain_product (cart_id, user_id, product_id, shop_id, product_count, sal...	1 row(s) affected
94	16:54:33	select * from cart LIMIT 0, 1000	10 row(s) returned

→ $\text{total_num_of_prod} = 0 + 2 = 2$

→ $\text{total_money} = 0 + 2 \times 10000 = 20000$

+ Insert lần 2







```

insert into cart_contain_product (cart_id, user_id, product_id, shop_id,
product_count, saleprice) values (1,1,7,2,1,10000);

```

⇒ Kết quả màn hình hiển thị từ DBMS.

```
462 • insert into cart_contain_product (cart_id, user_id, product_id, shop_id, product_count, saleprice)
463 values (1,1,7,2,1,100000);
464
465 • select * from cart;
466
```

Result Grid				
Filter Rows: <input type="text"/>				
Edit:   				
Export/Import:  				
Wrap Cell Content: 				
cart_id	user_id	total_num_of_prod	total_money	
1	1	3	120000	
2	2	0	0	
3	3	0	0	
4	4	0	0	
5	5	0	0	
6	6	0	0	

Output			
Action Output			
#	Time	Action	Message
✓ 93	16:54:12	insert into cart_contain_product (cart_id, user_id, product_id, shop_id, product_count, sal...	1 row(s) affected
✓ 94	16:54:33	select * from cart LIMIT 0, 1000	10 row(s) returned
✓ 95	16:56:06	insert into cart_contain_product (cart_id, user_id, product_id, shop_id, product_count, sal...	1 row(s) affected
✓ 96	16:56:15	select * from cart LIMIT 0, 1000	10 row(s) returned

→ total_num_of_prod = 2 + 1 = 3

→ total_money = 20000 + 1*100000 = 120000

+ Insert lần 3

```
insert into cart_contain_product (cart_id, user_id, product_id, shop_id,
product_count, saleprice) values (1,1,13,3,5,21000);
```

⇒ Kết quả màn hình hiển thị từ DBMS.

```

464 • insert into cart_contain_product (cart_id, user_id, product_id, shop_id, product_count, saleprice)
465 values (1,1,13,3,5,21000);
466 • select * from cart;
467
468

```

Result Grid

	cart_id	user_id	total_num_of_prod	total_money
▶	1	1	8	225000
	2	2	0	0
	3	3	0	0
	4	4	0	0
	5	5	0	0
	6	6	0	0

Output

Action Output

#	Time	Action	Message
✓ 95	16:56:06	insert into cart_contain_product (cart_id, user_id, product_id, shop_id, product_count, sal...	1 row(s) affected
✓ 96	16:56:15	select * from cart LIMIT 0, 1000	10 row(s) returned
✓ 97	16:57:09	insert into cart_contain_product (cart_id, user_id, product_id, shop_id, product_count, sal...	1 row(s) affected
✓ 98	16:57:09	select * from cart LIMIT 0, 1000	10 row(s) returned

→ $\text{total_num_of_prod} = 3 + 5 = 8$

→ $\text{total_money} = 120000 + 5 \times 21000 = 225000$

3. Thủ tục chứa câu SQL

3.1 Procedure truy vấn loại sản phẩm trong giỏ hàng của người dùng

- Mô tả chức năng: Truy vấn loại sản phẩm được lưu trong giỏ hàng của một người dùng với `p_user_id` là tham số đầu vào, đồng thời lấy thêm một số thông tin của sản phẩm từ table `product` như `product_name`, `information` và sắp xếp `saleprice` (giá bán) của loại sản phẩm theo thứ tự tăng hoặc giảm dần.
- + Nếu tham số đầu vào `type_order = "ASC"` → sắp xếp tăng dần.
- + Nếu tham số đầu vào `type_order = "DESC"` → sắp xếp giảm dần.
- Câu lệnh tạo thủ tục:

```

drop procedure if exists get_product_info_in_cart
delimiter $$
create procedure get_product_info_in_cart(
    in p_user_id int,
    in type_order varchar(4)
)
begin
    if type_order = "ASC"
    then

```

```

begin
    select user_id, product.product_id, product_name, saleprice, information
    from cart_contain_product, product
    where user_id = p_user_id and cart_contain_product.product_id =
product.product_id
    order by saleprice;
end;
end if;
if type_order = "DESC"
then
begin
    select user_id, product.product_id, product_name, saleprice, information
    from cart_contain_product, product
    where user_id = p_user_id and cart_contain_product.product_id =
product.product_id
    order by saleprice DESC;
end;
end if;
end;
$$
Delimiter ;

```

- Câu lệnh thực thi thủ tục:

- a. Truy vấn giỏ hàng của user có user_id = 3 và sắp xếp saleprice theo thứ tự giảm dần.

```
call get_product_info_in_cart(3,"DESC");
```

⇒ Kết quả màn hình hiển thị từ DBMS.

```
505 • call get_product_info_in_cart(3,"DESC");
```

```
506
```

Result Grid					
Filter Rows:		Export:		Wrap Cell Content:	
user_id	product_id	product_name	saleprice	information	
3	8	iPhone 12	50000	RAM: 6GB, ROM: 128GB, Năm sản xuất: 2020	
3	14	Nồi chiên không dầu	20000	Hãng: Toshiba, Chất liệu: Thép không gỉ	
3	2	Áo thun nữ	10000	Vải cotton, Màu đen, Size S	

- b. Truy vấn giỏ hàng của user có user_id = 1 và sắp xếp saleprice theo thứ tự tăng dần.

```
call get_product_info_in_cart(1,"ASC");
```

⇒ Kết quả màn hình hiển thị từ DBMS.

505 • `call get_product_info_in_cart(1,"ASC");`
506

<

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [IA](#)

	user_id	product_id	product_name	saleprice	information
	1	1	Áo thun nữ	10000	Vải cotton, Màu trắng, Size M
	1	13	Máy xay sinh tố	21000	Hãng: Sunhouse, Chất liệu: Nhựa siêu bền
▶	1	7	iPhone 13 Pro Max	100000	RAM: 8GB, ROM: 256GB, Năm sản xuất: 2021

3.2 Procedure truy vấn người dùng có tổng tiền các sản phẩm trong giỏ hàng $\geq x$ (VNĐ)

- Mô tả chức năng: Truy vấn thông tin người dùng gồm: tên tài khoản (username), ID người dùng (user_id), họ tên (fullname), số điện thoại (mobile) (lấy thêm thông tin từ table *the_user*) mà có tổng tiền của tất cả sản phẩm trong giỏ hàng của họ $\geq x$ (VNĐ). Trong câu lệnh SELECT, ta sẽ **group by** user_id, điều kiện **having** tổng tiền $\geq x$ (VNĐ), sử dụng **aggregate function** lấy tổng là SUM, **order by** sắp xếp theo tổng tiền theo thứ tự tăng hoặc giảm dần.
- Câu lệnh tạo thủ tục:

```
drop procedure if exists get_total_money_in_cart_and_user_info
delimiter $$
create procedure get_total_money_in_cart_and_user_info(
    in x int,
    in type_order varchar(4)
)
begin
    if type_order = "ASC"
    then
        begin
            select the_user.user_id, username, fullname, mobile,
sum(saleprice*product_count) as total_money
            from the_user, cart_contain_product
            where the_user.user_id = cart_contain_product.user_id
            group by cart_contain_product.user_id
```



```

        having total_money >= x
        order by total_money;
    end;
end if;
if type_order = "DESC"
then
begin
    select the_user.user_id, username, fullname, mobile,
sum(saleprice*product_count) as total_money
    from the_user, cart_contain_product
    where the_user.user_id = cart_contain_product.user_id
    group by cart_contain_product.user_id
    having total_money >= x
    order by total_money DESC;
end;
end if;
end;
$$
Delimiter ;

```

- Câu lệnh thực thi thủ tục:


a. x = 100000 VNĐ, sắp xếp tăng dần.

```
call get_total_money_in_cart_and_user_info(100000,"ASC");
```

⇒ Kết quả màn hình hiển thị từ DBMS.

545 • `call get_total_money_in_cart_and_user_info(100000,"ASC");`
546

<

Result Grid | Filter Rows: | Export: | Wrap Cell Content: 

	user_id	username	fullname	mobile	total_money
▶	3	nguyenphuoctri	Nguyen Phuoc Tri	0914370975	130000
	5	nguyenthanhcong	Nguyen Thanh Cong	01697053312	140000
	7	nguyenminhai	Nguyen Minh Hai	0913737095	190000
	1	voxuannguyen2001	Vo Trinh Xuan Nguyen	0397003301	225000
	2	nguyenphuoctri	Nguyen Phuoc Tri	0914370975	600000
	8	yennhi_01	Le Nguyen Yen Nhi	091334987	640000

b. x = 200000 VNĐ, sắp xếp giảm dần.

```
call get_total_money_in_cart_and_user_info(200000,"DESC");
```

⇒ Kết quả màn hình hiển thị từ DBMS.

```
545 • call get_total_money_in_cart_and_user_info(200000,"DESC");
546
```

Result Grid					
Filter Rows: <input type="text"/>					
Export:  Wrap Cell Content: 					
	user_id	username	fullname	mobile	total_money
▶	8	yennhi_01	Le Nguyen Yen Nhi	091334987	640000
	2	nguyenphuoctri	Nguyen Phuoc Tri	0914370975	600000
	1	voxuannguyen2001	Vo Trinh Xuan Nguyen	0397003301	225000

4. Hàm

4.1 Hàm tính và trả về số lượng loại sản phẩm thỏa mãn điều kiện trong một đơn hàng.

- Mô tả chức năng: Hàm tính và trả về số lượng loại sản phẩm có $\text{selling_price} \geq x$ VNĐ (với $x \leq 1.000.000$ VNĐ) trong một đơn hàng cụ thể với tham số đầu vào là ID đơn hàng `para_order_id`.
- + Nếu $x > 1.000.000$ VNĐ thì ném lỗi với message `"Invalid input x!"`
- + Ngược lại, tính toán và trả về giá trị.
- Câu lệnh tạo hàm:

```
DELIMITER $$
```

```
DROP function if exists num_of_product_type_in_order $$
```

```
CREATE function num_of_product_type_in_order(para_order_id int, x int)
```

```
returns int
```

```
READS SQL DATA
```

```
DETERMINISTIC
```

```
BEGIN
```

```
    if (x > 1000000) then
```

```
        signal sqlstate '45000' set message_text = "Invalid input x!";
```

```
    else
```

```
    begin
```

```
        declare num int default 0;
```

```
        declare func_selling_price int;
```

```
        DECLARE exit_loop BOOLEAN;
```

```
        DECLARE cur CURSOR FOR
```

```
            SELECT selling_price
```

```
            FROM order_contains_product
```

```
            where order_id = para_order_id
```

```

        having selling_price >= x;
DECLARE CONTINUE HANDLER FOR NOT FOUND SET exit_loop = TRUE;
OPEN cur;
order_loop: LOOP
    FETCH cur INTO func_selling_price;
    IF exit_loop THEN
        CLOSE cur;
        LEAVE order_loop;
    END IF;
    set num = num + 1;
END LOOP order_loop;
return num;
end;
end if;
END $$
DELIMITER ;

```

- Xét đơn hàng có order_id = 3, ta có chi tiết đơn hàng này bằng lệnh:

```
select * from order_contains_product where order_id = 3;
```

639 • `select * from order_contains_product where order_id = 3;`
640

<

Result Grid

	shop_id	product_id	order_id	amount	selling_price
▶	2	1	3	1	32000
	3	1	3	1	140000
	4	2	3	1	59000
*	NULL	NULL	NULL	NULL	NULL

- Câu lệnh SELECT minh họa gọi hàm và kết quả hiển thị từ DBMS
 - x = 40000 VNĐ: có 2 loại sản phẩm có selling_price ≥ 40000 VNĐ là (shop_id, product_id) = (3,1) hoặc (4,2).

```
select num_of_product_type_in_order(3,40000);
```

```
641 • select num_of_product_type_in_order(3,40000);
642
```

Result Grid | Filter Rows: | Export: | Wrap

	num_of_product_type_in_order(3,40000)
▶	2

- b. $x = 2.000.000$ VNĐ: không có loại sản phẩm có $\text{selling_price} \geq 2.000.000$ VNĐ nên ném lỗi.

```
select num_of_product_type_in_order(3,2000000);
```

```
647 • select num_of_product_type_in_order(3,2000000);
648
```

Output

Action Output

#	Time	Action	Message
✓ 208	00:34:14	select num_of_product_type_in_order(3,40000) LIMIT 0, 1000	1 row(s) returned
✗ 209	00:34:36	select num_of_product_type_in_order(3,2000000) LIMIT 0, 1000	Error Code: 1644. Invalid input x!

4.2 Hàm tính và trả về tổng tiền của các sản phẩm thuộc một shop trong giỏ hàng của tất cả người dùng

- Mô tả chức năng: Hàm tính và trả về tổng tiền của các sản phẩm thuộc một shop nào đó trong giỏ hàng của tất cả người dùng trong database với tham số đầu vào là ID của shop para_shop_id. Kiểm tra sự tồn tại của tham số đầu vào trong table *shop*:
- + Nếu tồn tại ID shop thì tính và trả về giá trị.
- + Nếu không tồn tại thì ném lỗi với message "Invalid shop_id!"
- Câu lệnh tạo hàm:

```
DELIMITER $$
DROP function if exists total_money_pay_shop_in_cart $$
CREATE function total_money_pay_shop_in_cart(para_shop_id int)
returns int
READS SQL DATA
DETERMINISTIC
BEGIN
    -- kiểm tra shop_id truyền vào có tồn tại ko
    declare input_valid bool default true;
    declare tmp_id int default -1;
```

```
declare cur_check cursor for
    select shop_id
    from shop
    where shop_id = para_shop_id;
DECLARE CONTINUE HANDLER FOR NOT FOUND SET input_valid = false;
OPEN cur_check;
check_loop: loop
    fetch cur_check into tmp_id;
    IF (input_valid = false) THEN
        CLOSE cur_check;
        LEAVE check_loop;
    END IF;
    if (tmp_id > -1) then
        CLOSE cur_check;
        LEAVE check_loop;
    END IF;
end loop check_loop;

-- neu shop_id truyen vao khong ton tai
if (input_valid = false) then
    signal sqlstate '45000' set message_text = "Invalid shop_id!";
else
begin
    DECLARE total int default 0;
    declare amount int;
    declare func_saleprice int;
    DECLARE exit_loop BOOLEAN;
    DECLARE cur CURSOR FOR
        SELECT product_count, saleprice
        FROM cart_contain_product
        where shop_id = para_shop_id;
    DECLARE CONTINUE HANDLER FOR NOT FOUND SET exit_loop = TRUE;
    OPEN cur;
    cart_loop: LOOP
        FETCH cur INTO amount,func_saleprice;
        IF exit_loop THEN
            CLOSE cur;
```

```

        LEAVE cart_loop;
    END IF;
    set total = total + amount*func_saleprice;
END LOOP cart_loop;
return total;
end;
end if;
END $$
DELIMITER ;

```

- Câu lệnh SELECT minh họa gọi hàm và kết quả hiển thị từ DBMS
- a. Xét shop có shop_id = 1, ta có chi tiết sản phẩm của shop này có trong giỏ hàng của các user này bằng lệnh:

```
select * from cart_contain_product where shop_id = 1;
```

706 • `select * from cart_contain_product where shop_id = 1;`
707

<

Result Grid | Filter Rows: | Edit: | Export/

	cart_id	user_id	product_id	shop_id	product_count	saleprice
▶	1	1	1	1	2	10000
	3	3	2	1	4	10000
	7	7	3	1	2	60000
	7	7	4	1	1	70000
*	NULL	NULL	NULL	NULL	NULL	NULL

- Câu lệnh SELECT minh họa gọi hàm:

```
select total_money_pay_shop_in_cart(1);
```

⇒ Kết quả hiển thị từ DBMS

707 • `select total_money_pay_shop_in_cart(1);`
708

<

Result Grid | Filter Rows: | Export:

	total_money_pay_shop_in_cart(1)
▶	250000

- $\text{total_money_pay_shop_in_cart} = 2 \times 10000 + 4 \times 10000 + 2 \times 60000 + 70000 = 250.000 \text{ VNĐ}$

b. para_shop_id = 31 → không tồn tại shop, ta ném lỗi

- Câu lệnh SELECT minh họa gọi hàm:

```
select total_money_pay_shop_in_cart(31);
```

⇒ Kết quả hiển thị từ DBMS

```
699 • select total_money_pay_shop_in_cart(31);
```

700

<

Output

Action Output

#	Time	Action	Message
309	13:28:51	select * from cart_contain_product where shop_id = 1 LIMIT 0, 1000	4 row(s) returned
310	13:29:09	select total_money_pay_shop_in_cart(31) LIMIT 0, 1000	Error Code: 1644. Invalid shop_id!

5. Giao diện ứng dụng và các hình ảnh minh họa

5.1 Giao diện hiện thực table *cart_contain_product*

a. Ngôn ngữ sử dụng

- Backend: PHP
- Frontend: HTML, CSS, Javascript
- Database: mySQL

b. Mô tả giao diện và hình ảnh minh họa

**** Trang 1: Manage Product in User's cart**

TASK	Manage Product in User's cart					
Manage Order	REFRESH					
Manage Feedback	Cart ID	User ID	Product ID	Shop ID	Amount	Saleprice
Manage Product in cart	1	1	1	1	6	20000
	1	1	7	2	1	100000
	1	1	13	3	5	21000
	3	3	2	1	9	10000
	3	3	8	2	1	50000
	3	3	14	3	2	20000
	4	4	19	4	2	10000
	4	4	25	5	2	20000
	5	5	20	4	2	30000
	5	5	26	5	2	40000
	7	7	3	1	2	60000
	7	7	4	1	1	70000
	8	8	15	3	3	80000

[Delete All](#)
[Insert a record](#)

\$\$ Input User_id to retrieve products in his/her cart:

USER ID:

Please fill in all box!

\$\$ Retrieve Users have total money of products in cart that have saleprice >= input:

ASCENDING ORDER OF TOTAL MONEY

Input:

Please fill in all box!

\$\$ Input record needed to update with attribute column:

CART ID:

USER ID:

PRODUCT ID:

SHOP ID:

Column needed to UPDATE:

UPDATE Value:

Please fill in all box!

ECOMMERCE - Database Learners

- Table
 - + Title: Manage Product in User's cart
 - + Column: Cart ID (ID giỏ hàng), User ID (ID người dùng), Product ID (ID sản phẩm), Shop ID (ID shop), Amount (số lượng sản phẩm của loại sản phẩm tương ứng), Saleprice (giá bán).
 - + Row: các record của table ***cart_contain_product*** trong database.
- Button
 - + REFRESH: refresh lại trang hiện tại
 - + DELETE (cạnh mỗi row trong table): xóa record ứng với row được chọn.
 - + Insert a record: Link đến trang **Insert record into Carts contain Products**
- Input field
 - **\$\$ Input User_id to retrieve products in his/her cart:**
 - Chức năng: truy xuất giỏ hàng của người dùng có USER_ID được nhập
 - Nhập: USER ID

- Xuất: các record trong table cart_contain_product có user_id = USER ID được nhập.
- + Nếu USER_ID chưa được nhập thì ném lỗi với message “Please fill in all box!”
- + Nếu USER_ID được nhập không tồn tại trong row nào của bảng thì ném lỗi với message “ID doesn’t exist in table!”
- \$\$ Input record needed to update with attribute column:
- Chức năng: update Amount column (số lượng) hoặc Saleprice (giá bán) của loại sản phẩm có (product_id, shop_id) được nhập trong giỏ hàng cart_id của người dùng có user_id được nhập.
- Nhập:
 - CART ID
 - USER ID
 - PRODUCT ID
 - SHOP ID
 - Column needed to UPDATE: chọn 1 trong 2 column cần được update: Amount hoặc Saleprice
 - UPDATE value: giá trị cần mới được cập nhật vào.
 - + amount thuộc đoạn [1,10].
 - + saleprice \geq 1000 VNĐ.
- Xuất: Nhấn UPDATE và REFRESH để cập nhật dữ liệu trong table.
- + Nếu các input hợp lệ thì update giá trị và in message “Record updated successfully!”
- + Nếu có ít nhất một input chưa được nhập thì ném lỗi với message “Please fill in all box!”
- + Nếu các input tự nhập vào box **Column needed to UPDATE** khác Amount và Saleprice thì ném lỗi với message “Invalid Input in Column needed to UPDATE!”
- + Nếu update amount nhập vào box UPDATE value \notin [1,10] thì ném lỗi với message “Update amount out of range!”
- + Nếu update saleprice nhập vào box UPDATE value $<$ 1000 VNĐ thì ném lỗi với message “Update saleprice out of range!”
- \$\$ Retrieve Users have total money of products in cart that have saleprice \geq input:
ASCENDING ORDER OF TOTAL MONEY

- Chức năng: truy xuất user_id của người dùng có tổng tiền của tất cả sản phẩm trong giỏ hàng của họ \geq giá trị input nhập vào. Ngoài ra còn in ra tổng tiền và sắp xếp theo thứ tự tăng dần.
- Nhập: Input
- Xuất: in ra User_id thỏa mãn và tổng tiền sản phẩm trong giỏ.
- + Nếu Input chưa được nhập thì ném lỗi với message “Please fill in all box!”
- + Nếu Input được nhập hợp lệ nhưng không tồn tại user nào thỏa thì in message “No user satisfy!”

**** Trang 2: Insert record into Carts contain Products**

Insert record into Carts contain Products

Cart ID:

User ID:

Product ID:

Shop ID:

Amount:

Saleprice:

Insert New

c. Minh họa các thao tác

- **Thêm một record mới vào table cart_contain_product.**
- Nhấn *Insert a record* ở trang **Manage Product in User's cart** để chuyển tới trang **Insert record into Carts contain Products**
- Nhập dữ liệu:

1. Dữ liệu hợp lệ

Cart ID:

User ID:

Product ID:

Shop ID:

Amount:

Saleprice:

- Kết quả:

1	1	13	3	5	21000
2	2	28	5	2	10000
3	3	2	1	0	10000

2. Không nhập đủ dữ liệu vào các box

Failed!! You should fill all input

3. Dữ liệu không hợp lệ hoặc insert 1 record trùng với record đã tồn tại

Error! Can't add into database

● **Xóa đi một record khỏi table `cart_contain_product`.**

- Nhấn vào nút DELETE ở bên cạnh mỗi row trong table trang **Manage Product in User's cart** để xóa đi record tương ứng.

+ Ban đầu:

1	1	13	3	5	21000	Delete
2	2	28	5	2	10000	Delete
3	3	2	1	9	10000	Delete

+ Xóa record (2,2,28,5,2,10000), thu được:

1	1	13	3	5	21000	Delete
3	3	2	1	9	10000	Delete

- **Update giá trị của column Amount hoặc Saleprice của một record trong table cart_contain_product.**
- Sử dụng Input field sau:

\$\$ Input record needed to update with attribute column:

CART ID:

USER ID:

PRODUCT ID:

SHOP ID:

Column needed to UPDATE:

UPDATE Value:

- Nhập dữ liệu:
- 1. Dữ liệu hợp lệ

CART ID:

USER ID:

PRODUCT ID:

SHOP ID:

Column needed to UPDATE:

UPDATE Value:

- Ban đầu:

Cart ID	User ID	Product ID	Shop ID	Amount	Saleprice
1	1	1	1	1	20000
1	1	7	2	1	100000

- Sau khi update: Amount ở row 1 đã thay đổi sang 8

1	1	1	1	8	20000	<input type="button" value="Delete"/>
1	1	7	2	1	100000	<input type="button" value="Delete"/>

- Message được in ra:

Record updated successfully

2. Lỗi “Please fill in all box!”

UPDATE Value:

Please fill in all box!

3. Lỗi “Invalid Input in Column needed to UPDATE!”

Invalid Input in Column needed to UPDATE!

4. Lỗi “Update amount out of range!”

PRODUCT ID:

SHOP ID:

Column needed to UPDATE:

UPDATE Value:

Update amount out of range!

5. Lỗi “Update saleprice out of range!”

PRODUCT ID:

SHOP ID:

Column needed to UPDATE:

UPDATE Value:

Update saleprice out of range!

- **Truy xuất giỏ hàng của người dùng có USER_ID được nhập**
- Sử dụng Input field sau:

\$\$ Input User_id to retrieve products in his/her cart:

USER ID:

1. Dữ liệu hợp lệ, tồn tại User ID nhập trong bảng

USER ID:

Select Result

Cart ID	User ID	Product ID	Shop ID	Amount	Saleprice	
1	1	1	1	7	20000	<input type="button" value="Delete"/>
1	1	7	2	1	100000	<input type="button" value="Delete"/>
1	1	13	3	5	21000	<input type="button" value="Delete"/>

2. Dữ liệu không hợp lệ hoặc User ID được nhập không tồn tại trong bảng

\$\$ Input User_id to retrieve products in his/her cart:

USER ID:

ID doesn't exist in table!

- Truy xuất user_id của người dùng có tổng tiền của tất cả sản phẩm trong giỏ hàng của họ \geq giá trị input x nhập vào.
- Sử dụng Input field sau:

\$\$ Retrieve Users have total money of products in cart that have saleprice \geq input:

ASCENDING ORDER OF TOTAL MONEY

Input:

1. Tồn tại user thỏa mãn

ASCENDING ORDER OF TOTAL MONEY

Input:

USER : 4 ----- 60000

USER : 5 ----- 140000

USER : 3 ----- 180000

USER : 7 ----- 190000

USER : 1 ----- 345000

USER : 8 ----- 640000

2. Không tồn tại user thỏa mãn

Input:

No user satisfy!

4. Thành viên 4

Họ và tên: Nguyễn Kim Phương Trang

MSSV: 1910620

```
DROP TABLE IF EXISTS feedback;
CREATE TABLE feedback (
    feedback_id int not null auto_increment,
    shop_id int not null,
    product_id int not null,
    review_content text,
    rating int not null,
    create_date date,
    user_id int not null,
    primary key(feedback_id, shop_id, product_id),
    constraint rating_ck check (rating between 0 and 5)
);

ALTER TABLE feedback
ADD FOREIGN KEY ( product_id,shop_id) REFERENCES product(product_id,shop_id);
ALTER TABLE feedback
ADD FOREIGN KEY ( user_id) REFERENCES users(user_id);
```

1. Thủ tục insert dữ liệu:

- Mô tả chức năng: Thêm mới một feedback với các đầu vào là shop_id, product_id, review_content, rating, create_date, user_id.
- Câu lệnh tạo thủ tục:

```
drop procedure if exists add_feedback;
delimiter //
create procedure add_feedback (
    in shop_id int,
    in product_id int,
    in review_content text,
    in rating int,
    in create_date date,
```

```
        in user_id int
    )
begin
    if rating < 1 or rating > 5 then
        signal sqlstate '45000'
        set message_text = 'rating must be between 1 and 5';
    end if;
    insert into feedback(shop_id, product_id, review_content, rating,
create_date, user_id)
        values (shop_id, product_id, review_content, rating,
create_date, user_id);
end//
delimiter ;
```

- Câu lệnh thực thi thủ tục mẫu:
- 1. Feedback nhập vào có số sao đánh giá hợp lệ (thuộc đoạn 0 đến 5).

```
call add_feedback(1,1,"it's okay", 4, date('2021-11-27'),2);
```

- Kết quả màn hình hiển thị từ DBMS.

```
94 • select * from feedback;
95 • call add_feedback(1,1,"it's okay", 4, date('2021-11-27'),2);
96
```

Result Grid							
	feedback_id	shop_id	product_id	review_content	rating	create_date	user_id
▶	1	1	1	it's okay	4	2021-11-27	2
•	NULL	NULL	NULL	NULL	NULL	NULL	NULL

feedback 55 x				
Output				
Action Output				
#	Time	Action	Message	
✓ 704	14:31:17	CREATE TABLE feedback (feedback_id int not nul...	0 row(s) affected	
✓ 705	14:31:17	ALTER TABLE feedback ADD FOREIGN KEY (produ...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	
✓ 706	14:31:17	ALTER TABLE feedback ADD FOREIGN KEY (user_j...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	
✓ 707	14:31:24	call add_feedback(1,1,"it's okay", 4, date(2021-11-27 ...	1 row(s) affected	
✓ 708	14:31:31	select * from feedback LIMIT 0, 1000	1 row(s) returned	

2. Feedback nhập vào có số sao đánh giá không hợp lệ.

```
95 • call add_feedback(1,1,"invalid feedback rating", 6, date('2021-11-27'),2);
96
97
```

Output				
Action Output				
#	Time	Action	Message	
✓ 710	14:34:22	call add_feedback(1,1,"invalid feedback rating", 4, date(2021-11-27),6)	1 row(s) affected	
✗ 711	14:34:30	call add_feedback(1,1,"invalid feedback rating", 6, date(2021-11-27),2)	Error Code: 3819. Check constraint 'rating_ck' is violated.	

2. Trigger

2.1 Trigger BEFORE INSERT kiểm tra điều kiện feedback được thêm vào bởi user_id đã mua thành công sản phẩm (shop_id, product_id).

- Mô tả chức năng:
- + Kiểm tra user_id có đơn hàng order_id chứa sản phẩm (shop_id, product_id) và order_status = "success" thì feedback mới được thêm vào bảng.
- + Mỗi user_id chỉ được gửi một feedback cho mỗi sản phẩm.
- Câu lệnh tạo trigger

```
drop trigger feedback_check;
delimiter |
create trigger feedback_check before insert on feedback
for each row
begin
    if (new.user_id not in (
        select user_id
        from order_detail, order_contains_product
        where order_detail.order_id = order_contains_product.order_id
        and new.product_id = order_contains_product.product_id
        and new.shop_id = order_contains_product.shop_id
        and order_detail.order_status = "success"
    )) then
        begin
            signal sqlstate '45000' set message_text = "this user hasn't purchased
this product successfully";
        end;
    end if;
    if new.user_id in (
        select user_id
        from feedback
        where user_id = new.user_id and shop_id = new.shop_id and product_id =
new.product_id)
    then
        begin
            signal sqlstate '45000' set message_text = "this user has sent feedback
for this product";
        end;
    end if;
end;
|
Delimiter ;
```

- Câu lệnh kiểm tra trigger hoạt động
- 1. Trigger kiểm tra điều kiện 1: User_id chỉ được gửi feedback cho sản phẩm đã nhận hàng thành công.

Bảng order_detail:

	order_id	shipping_id	order_status	create_date	user_id	sname	saddress	sphone_number
▶	2	1	waiting	2021-11-23	1	Nguyen Trang	Phu Yen	0123456789
	3	2	processing	2021-11-23	2	Tran Thang	TP.HCM	0123456798
	4	3	success	2021-11-27	3	Le Hoang Phuc	TP.HCM	0123456700
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Bảng order_contains_product:

	shop_id	product_id	order_id	amount	selling_price
▶	1	1	2	3	22000
	1	1	3	3	17000
	1	1	4	3	17000
	1	2	2	3	22000
	1	3	2	1	51000
	1	3	3	2	29000
	1	4	2	1	15000
*	NULL	NULL	NULL	NULL	NULL

user_id = 2 hiện tại đang có các sản phẩm (shop_id, product_id) = {(1,1), (1,3)} hiện chưa thanh toán xong (order_status = “processing”)

```

87 • select shop_id, product_id, order_detail.order_status
88   from order_detail, order_contains_product
89   where order_detail.order_id = order_contains_product.order_id and user_id = 2;
90

```

Result Grid			
shop_id	product_id	order_status	
▶ 1	1	processing	
1	3	processing	

Do đó, khi cố gắng insert feedback từ user_id = 2 gửi cho sản phẩm (1,3) hoặc (1,1) trigger trên sẽ báo lỗi.

```

92 • call add_feedback(1,3,"invalid feedback 1", 4, date('2021-11-27'),2);
93 • call add_feedback(1,1,"invalid feedback 2", 4, date('2021-11-27'),2);
94

```

Output			
#	Time	Action	Message
723	14:58:24	select shop_id, product_id, order_detail.order_status from order_detail, ...	2 row(s) returned
724	15:00:58	call add_feedback(1,3,"invalid feedback 1", 4, date(2021-11-27),2)	Error Code: 1644. this user hasn't purchased this product successfully

- Kết quả màn hình hiển thị từ DBMS.

Thêm feedback hợp lệ với user_id = 3 đã thanh toán thành công sản phẩm có id(1,1).

```
108 • call add_feedback(1,1,"this is feedback (order_status = 'success'", 4, date('2021-11-27'),3);
```

feedback_id	shop_id	product_id	review_content	rating	create_date	user_id
5	1	1	this is feedback (order_status = 'success'	4	2021-11-27	3

- Trigger kiểm tra điều kiện trước khi insert: một user_id chỉ được gửi một feedback cho mỗi sản phẩm.

```
108 • call add_feedback(1,1,"another feedback from user_id = 3 to product(1,1)", 4, date('2021-11-27'),3);
```

```
109
```

#	Time	Action	Message
789	19:12:05	select * from feedback LIMIT 0, 1000	1 row(s) returned
790	19:14:38	call add_feedback(1,1,"another feedback from user_id = 3 to product(1,1)", 4, date('2021-11-27'),3);	Error Code: 1644. this user has sent feedback for this product

2.2 Trigger cập nhật dữ liệu rating cho bảng Shop từ feedback rating cho sản phẩm của Shop.

- Mô tả chức năng: Ứng với mỗi feedback được thêm thành công vào bảng dữ liệu, số sao rating của bảng Shop cũng được cập nhật dựa vào rating của sản phẩm mà Shop đang kinh doanh.

Rating của mỗi sản phẩm là số nguyên có giá trị từ 0 - 5.

Rating của mỗi shop được tính = (tổng rating của tất cả sản phẩm mà shop kinh doanh)/(số feedback)

- Câu lệnh tạo trigger:

```
drop trigger feedback_update_to_shop
delimiter |
create trigger feedback_update_to_shop after insert on feedback
for each row
begin
    declare number_of_reviews int default 0;
    set number_of_reviews = (select count(*) from shop, feedback where
shop.shop_id = feedback.shop_id) - 1;
    update shop
    set rating = (shop.rating*number_of_reviews +
new.rating)/(number_of_reviews + 1)
```

```
where shop.shop_id = new.shop_id;

end;

Delimiter ;
```

- Câu lệnh kiểm tra trigger hoạt động.

Bảng Shop hiện tại, rating = 0 cho thấy shop chưa nhận được feedback đến bất kỳ sản phẩm nào của Shop.

	shop_id	shop_name	shop_description	shop_owner	create_date	rating
▶	1	SAMSUNG	this is description, cool here	1	2021-11-01	0.0
	2	Lilyeyewear	fashionista	2	2021-11-02	0.0
	3	PS	health care	3	2021-11-03	0.0
	4	Blue Light		4	2021-11-04	0.0
	5	Eye Plus		5	2021-11-05	0.0
*	NULL	NULL	NULL	NULL	NULL	NULL

Bảng Feedback cũng đang rỗng.

```
105 • select * from feedback;
```

feedback_id	shop_id	product_id	review_content	rating	create_date	user_id
*	NULL	NULL	NULL	NULL	NULL	NULL

Sau khi thêm feedback mới thành công.

```
110 • call add_feedback(1,1,"first feedback to product in shop_id = 1", 4, date('2021-11-27'),3);
111
```

feedback_id	shop_id	product_id	review_content	rating	create_date	user_id
▶ 6	1	1	first feedback to product in shop_id = 1	4	2021-11-27	3
*	NULL	NULL	NULL	NULL	NULL	NULL

Cột rating của shop có shop_id = 1 nhận được feedback tương ứng cũng được cập nhật.
Rating mới của shop_id = 1 được tính = $(0 \cdot 0 + 4) / (0 + 1) = 4.0$

	shop_id	shop_name	shop_description	shop_owner	create_date	rating
▶	1	SAMSUNG	this is description, cool here	1	2021-11-01	4.0
	2	Lilyeyewear	fashionista	2	2021-11-02	0.0
	3	PS	health care	3	2021-11-03	0.0
	4	Blue Light		4	2021-11-04	0.0
	5	Eye Plus		5	2021-11-05	0.0
✱	NULL	NULL	NULL	NULL	NULL	NULL

3. Thủ tục chứa câu SQL

- a) procedure hiển thị danh sách các feedback của một sản phẩm và thông tin sản phẩm sắp xếp theo thứ tự tăng/giảm dần của rating.

```
drop procedure if exists get_list_feedbacks_in_order_by_product;
delimiter |
create procedure get_list_feedbacks_in_order_by_product(
    in ishop_id int,
    in iproduct_id int,
    in type_order varchar(4)
)
begin
    if type_order = "ASC"
    then
        begin
            select *
            from feedback, product
            where feedback.shop_id = ishop_id and feedback.product_id = iproduct_id
            and product.product_id = iproduct_id
            order by rating;
        end;
    end if;
    -- duplicate code
    if type_order = "DESC"
    then
        begin
            select *
            from feedback, product
```



```

        where feedback.shop_id = ishop_id and feedback.product_id = iproduct_id
and product.product_id = iproduct_id
        order by rating DESC;
    end;
end if;
end;
|
Delimiter ;

```

226 `call get_list_feedbacks_in_order_by_product(1,1,"DESC")`
227

feedback_id	shop_id	product_id	review_content	rating	create_date	user_id	product_id	shop_id	product_name	listed_price	origin	remaining_amount	information
9	1	1	will have the next time	5	2021-11-27	1	1	1	Áo thun nữ	200000.00	Hàn Quốc	20	Vải cotton, Màu trắng, Size M
6	1	1	first feedback to product in shop_id = 1	4	2021-11-27	3	1	1	Áo thun nữ	200000.00	Hàn Quốc	20	Vải cotton, Màu trắng, Size M
8	1	1	normal	3	2021-11-27	2	1	1	Áo thun nữ	200000.00	Hàn Quốc	20	Vải cotton, Màu trắng, Size M

- b) procedure hiển thị danh sách các sản phẩm của mỗi cửa hàng sắp xếp theo thứ tự số feedback mà mỗi sản phẩm nhận được.
- Mô tả thủ tục: nhận vào shop_id là cửa hàng cần hiển thị danh sách sản phẩm dựa trên feedback, tham số type_order cho phép hiển thị danh sách sắp xếp tăng hay giảm dần.
 - Câu lệnh tạo thủ tục:

```

drop procedure if exists getFeedbackOfShop
delimiter |
create procedure getFeedbackOfShop(
    in ishop_id int,
    in type_order varchar(4)
)
begin
    if type_order = ""
    then
    begin
        select *
        from product,(
            select product.product_id, count(feedback.product_id) as
numFeedback
            from product

```

```
        left join feedback
        on product.product_id = feedback.product_id
        group by product.product_id
    ) f
    where product.product_id = f.product_id and product.shop_id = ishop_id;
end;
end if;
if type_order = "ASC"
then
begin
    select *
    from product,(
        select product.product_id, count(feedback.product_id) as
numFeedback
        from product, feedback
        where product.product_id = feedback.product_id
        group by product.product_id
        having count(product.product_id) > 0
        order by count(product.product_id)) f
    where product.product_id = f.product_id and product.shop_id = ishop_id;
end;
end if;
-- duplicate code
if type_order = "DESC"
then
begin
    select *
    from product,(
        select product.product_id, count(feedback.product_id) as
numFeedback
        from product, feedback
        where product.product_id = feedback.product_id
        group by product.product_id
        having count(product.product_id) > 0
        order by count(product.product_id) DESC) f
    where product.product_id = f.product_id and product.shop_id = ishop_id;
```

```
end;
end if;
end;
|
Delimiter ;
```

- Câu lệnh thực thi thủ tục:

```
call getFeedbackOfProduct(1,"DESC");
```

- Kết quả chạy DBMS:

Danh sách feedback của shop_id = 1:

290 • `select * from feedback where shop_id = 1;`

	feedback_id	shop_id	product_id	review_content	rating	create_date	user_id
▶	6	1	1	first feedback to product in shop_id = 1	4	2021-11-27	3
	7	1	3	okay fine	4	2021-11-27	2
	8	1	1	normal	3	2021-11-27	2
	9	1	1	will have the next time	5	2021-11-27	1
	10	1	2	will have the next time	5	2021-11-29	1
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Kết quả truy xuất danh sách sản phẩm theo thứ tự số feedback giảm dần của shop_id = 1.

```
182 • call getFeedbackOfProduct(1,"DESC");
183
```

	product_id	shop_id	product_name	listed_price	origin	remaining_amount	information	product_id	numFeedback
▶	1	1	Áo thun nữ	200000.00	Hàn Quốc	20	Vải cotton, Màu trắng, Size M	1	3
	2	1	Áo thun nữ	200000.00	Hàn Quốc	15	Vải cotton, Màu đen, Size S	2	1
	3	1	Áo thun nam	150000.00	Việt Nam	20	Vải cotton, Màu đen, Size L	3	1

4. Hàm

4.1 Hàm tính tổng tiền đơn hàng order dựa trên bảng order_contains_product, dùng loop.

- Mô tả chức năng: Hàm nhận đầu vào là order_id và trả về tổng tiền của hóa đơn.
- Câu lệnh tạo hàm

```
DELIMITER $$
CREATE FUNCTION `getTotal`(`iorder_id` INT)
RETURNS int(11)
DETERMINISTIC
BEGIN
    DECLARE mytotal int default 0;
    declare iamount int;
    declare iselling_price int;
    DECLARE exit_loop BOOLEAN;
    DECLARE item_cursor CURSOR FOR
        SELECT amount, selling_price
        FROM order_contains_product
        where order_id = iorder_id;
    DECLARE CONTINUE HANDLER FOR NOT FOUND SET exit_loop = TRUE;
    OPEN item_cursor;
    item_loop: LOOP
        FETCH item_cursor INTO iamount, iselling_price;
        set mytotal = mytotal + iamount*iselling_price;
        IF exit_loop THEN
            CLOSE item_cursor;
            LEAVE item_loop;
        END IF;
    END LOOP item_loop;
    update order_detail
    set total = mytotal
    where order_id = iorder_id;
    return mytotal;
END$$
DELIMITER ;
```

- Câu lệnh SELECT minh họa gọi hàm:
Xét order có order_id = 2

205 • `SELECT * FROM order_contains_product where order_id = 2;`
206

Result Grid | Filter Rows: | Edit: | Export/Import:

	shop_id	product_id	order_id	amount	selling_price
▶	1	1	2	3	22000
	1	2	2	3	22000
	1	3	2	1	51000
	1	4	2	1	15000
✱	NULL	NULL	NULL	NULL	NULL

Thực thi hàm `calculate_total_of_order` với đầu vào là `order_id = 2`, kết quả trả về là 21300.

207 • `select calculate_total_of_order(2);`
208

Result Grid | Filter Rows: | Export:

	calculate_total_of_order(2)
▶	213000

- Kết quả màn hình hiển thị từ DBMS.

4.2 Hàm trả về thứ hạng bán hàng của shop dựa trên tổng số doanh thu của shop.

- Mô tả chức năng: Hàm nhận vào `shop_id` và trả về thứ hạng bán hàng của shop trong sản phẩm có `shop_id` tương ứng. Trả về Top 1, Top 2, Top 3 nếu shop nằm trong Top 3 bán hàng, trả về 'Top N' nếu shop nằm ngoài top 3.
- Câu lệnh tạo hàm:

```
drop function if exists get_shop_level;
delimiter //
create function get_shop_level (_shop_id int)
returns varchar(5)
deterministic
begin

    declare cnt1 int default 0;
    declare cnt2 int default 0;
```

```
declare cnt3 int default 0;
declare _level varchar(5) default 'Top N';

set cnt1 = (select count(*)
            from (
                select *, sum(total) as sumTotal
                from (
                    select distinct shop_id, order_detail.order_id, total
                    from order_detail, order_contains_product
                    where order_detail.order_id =
order_contains_product.order_id ) f
                group by shop_id
                order by sumTotal DESC
                limit 1 ) e
            where e.shop_id = _shop_id);
set cnt2 = (select count(*)
            from (
                select *, sum(total) as sumTotal
                from (
                    select distinct shop_id, order_detail.order_id, total
                    from order_detail, order_contains_product
                    where order_detail.order_id =
order_contains_product.order_id ) f
                group by shop_id
                order by sumTotal DESC
                limit 2) e
            where e.shop_id = _shop_id);
set cnt3 = (select count(*)
            from (
select *, sum(total) as sumTotal
            from (
                select distinct shop_id, order_detail.order_id, total
                from order_detail, order_contains_product
                where order_detail.order_id =
order_contains_product.order_id ) f
                group by shop_id
```

```

        order by sumTotal DESC
        limit 3) e
        where e.shop_id = _shop_id);
if cnt1 >= 1 then
    set _level = 'Top 1';
elseif cnt2 >= 1 then
    set _level = 'Top 2';
elseif cnt3 >= 1 then
    set _level = 'Top 3';
end if;
return _level;
end //
delimiter ;

```

- Câu lệnh Select minh họa gọi hàm:

Hàm bên dưới trả về thứ hạng của shop có shop_id = 1.

```
select get_shop_level(1);
```

- Kết quả chạy từ DBMS.

Thông tin các đơn hàng thuộc về các shop cùng với tổng tiền đơn hàng tương ứng.

```



190 • select distinct shop_id, order_detail.order_id, total
191         from order_detail, order_contains_product
192        where order_detail.order_id = order_contains_product.order_id;

```

Result Grid			
Filter Rows:		Export:	Wrap Cell Content:
shop_id	order_id	total	
1	2	213000	
1	3	167000	
1	4	102000	
2	5	102000	



Shop có shop_id thuộc Top 1.

```
193 • select get_shop_level(1);  
194
```

Result Grid			Filter Rows: <input type="text"/>
	get_shop_level(1)		
▶	Top 1		



Shop có shop_id thuộc Top 2.

```
193 • select get_shop_level(2);  
194
```

Result Grid			Filter Rows: <input type="text"/>
	get_shop_level(2)		
▶	Top 2		

Các shop còn lại thuộc Top N.

```
193 • select get_shop_level(3);  
194
```

Result Grid			Filter Rows: <input type="text"/>
	get_shop_level(3)		
▶	Top N		

5. Lập trình ứng dụng.

- Giao diện insert dữ liệu vào bảng Feedback

TASK	Add feedback
Manage Orders	
Manage Feedback	
Manage Products	
Manage Product in cart	
Manage User	
	Product id: <input type="text" value="1"/> Content: <input type="text" value="great value"/> Rating(0-5): <input type="text" value="4"/> By User: <input type="text" value="1"/> <input type="button" value="Add new"/>

Nếu dữ liệu thêm vào không thỏa trigger sẽ được hiển thị lỗi có nghĩa như trên giao diện 1 - user_id này phải thanh toán sản phẩm thành công mới được viết feedback cho sản phẩm.

TASK	Add feedback
Manage Orders	
Manage Feedback	
Manage Products	
Manage Product in cart	
Manage User	
	<div style="background-color: #f8d7da; padding: 5px; border: 1px solid #f5c6cb;"> Sactionthis user hasn't purchased this product successfully </div> Product id: <input type="text" value="1"/> Content: <input type="text" value="1"/> Rating(0-5): <input type="text" value="1"/> By User: <input type="text" value="1"/> <input type="button" value="Add new"/>

2-Mỗi user_id chỉ được gửi một feedback cho mỗi sản phẩm.

TASK

Manage Orders

Manage Feedback

Manage Products

Manage Product in cart

Manage User

Add feedback

Sactionthis user has sent feedback for this product

Product id:
1

Content:
hihi

Rating(0-5):
5

By User:
3

Add new

- Giao diện hiển thị bảng dữ liệu Feedback:

TASK

Manage Orders

Manage Feedback

Manage Products

Manage Product in cart

Manage User

All feedbacks

Choose the shop

feedback ID	Shop ID	Product ID	Content	Rating	Created	By User	Action
14	1	1	great deal	4	2021-11-27	3	<div>Delete</div> <div>Edit</div>
15	1	2	great value	4	2021-11-28	3	<div>Delete</div> <div>Edit</div>
16	2	9	high quality	4	2021-11-29	3	<div>Delete</div> <div>Edit</div>
17	2	10	great price	4	2021-11-30	4	<div>Delete</div> <div>Edit</div>
18	3	14	pencils ever	5	2021-11-30	4	<div>Delete</div> <div>Edit</div>
19	2	11	good value	3	2021-12-01	3	<div>Delete</div> <div>Edit</div>
20	1	3	It sets us up for criticism. That said - there is no way we can recommend this paper.	1	2021-12-01	4	<div>Delete</div> <div>Edit</div>
21	2	8	If you work in an office and know your paper, YOU WILL NOT LIKE THIS PAPER	2	2021-12-02	4	<div>Delete</div> <div>Edit</div>
22	3	15	The paper is exactly what I expected.	5	2021-12-02	4	<div>Delete</div> <div>Edit</div>
23	1	4	waste of time.	1	2021-12-02	3	<div>Delete</div> <div>Edit</div>
24	2	12	Do *NOT* order this product! it may not work and you wont' get a refund.	1	2021-12-02	3	<div>Delete</div> <div>Edit</div>
25	4	20	Not new, broken packages.	1	2021-12-02	4	<div>Delete</div> <div>Edit</div>
26	1	6	Very much worth the cost	5	2021-12-02	3	<div>Delete</div> <div>Edit</div>

Delete All

Insert Feedback

- Web cũng cho phép truy xuất danh sách feedback của một sản phẩm với product_id cụ thể - Thủ tục 3.1-get_list_feedbacks_in_order_by_product

TASK

Manage Orders

Manage Feedback

Manage Products

Manage Product in cart

Manage User

All feedbacks

Get all feedbacks of product

Product ID

Submit

feedback ID	Shop ID	Product ID	Content	Rating	Created	By User	Action
14	1	1	great deal	4	2021-11-27	3	<div>Delete</div> <div>Edit</div>
30	1	1	6	4	2021-12-02	4	<div>Delete</div> <div>Edit</div>

- Giao diện hiển thị danh sách dữ liệu từ việc gọi thủ tục hiển thị danh sách sản phẩm của một cửa hàng cùng số lượng feedback tương ứng của sản phẩm đó - Thủ tục 3.2-getFeedbackOfShop

TASK

Manage Orders

Manage Feedback

Manage Products

Manage Product in cart

Manage User

All feedbacks in shop

Choose the shop

Lilyeyewear Shop

ID	Name	Listed Price	Origin	Amount	Information	Number of feedbacks
1	Áo thun nữ	200000.00	Hàn Quốc	20	Vải cotton, Màu trắng, Size M	2
2	Áo thun nữ	200000.00	Hàn Quốc	15	Vải cotton, Màu đen, Size S	1
3	Áo thun nam	150000.00	Việt Nam	20	Vải cotton, Màu đen, Size L	1
4	Áo thun nam	150000.00	Việt Nam	10	Vải cotton, Màu đen, Size XL	1
5	Giày thể thao nữ	500000.00	Việt Nam	5	Giày thể thao nữ buộc dây siêu nhẹ, Màu trắng, Size 38	0
6	Giày thể thao nam	550000.00	Việt Nam	8	Giày thể thao nam êm ái, Màu đen, Size 42	1

Delete All

Insert Feedback

- Chọn chế độ hiển thị số feedback tăng dần hay giảm dần. VD: DESC

TASK

Manage Orders

Manage Feedback

Manage Products

Manage Product in cart

Manage User

All feedbacks in shop

Choose the shop

Lilyeyewear Shop

ID	Name	Listed Price	Origin	Amount	Information	Number of feedbacks
1	Áo thun nữ	200000.00	Hàn Quốc	20	Vải cotton, Màu trắng, Size M	2
2	Áo thun nữ	200000.00	Hàn Quốc	15	Vải cotton, Màu đen, Size S	1
3	Áo thun nam	150000.00	Việt Nam	20	Vải cotton, Màu đen, Size L	1
4	Áo thun nam	150000.00	Việt Nam	10	Vải cotton, Màu đen, Size XL	1
6	Giày thể thao nam	550000.00	Việt Nam	8	Giày thể thao nam êm ái, Màu đen, Size 42	1

[Delete All](#)
[Insert Feedback](#)

5. Thành viên 5:

Họ và tên: Trần Duy Chánh

MSSV:1912761

• Mô tả Table category

a)Colum (Các thuộc tính)

-category_id: id danh mục


-name_category: tên danh mục

-total_product: tổng số lượng sản phẩm

*Primary key:(category_id)

b)Row(các Record): Mỗi sản phẩm thuộc danh mục khác nhau, trong danh mục có category_id, name_category, total_product.

c)Table hiển thị từ DTBS

Result Grid			
Filter Rows: <input type="text"/>			
Edit: 			
	category_id	name_category	total_product
▶	1	Quần	8
	2	Áo	8
	3	Giày	4
	4	Nón	5
	5	Đồng hồ	3
	6	Dép	4
	7	Bông tai	2
	8	Bàn học	5
	9	Casio	2
	10	Vở	5
✱	NULL	NULL	NULL

1.Thủ tục insert

- Mô tả chức năng: Thêm mới một record vào table category (khi một shop nào đó thêm một Danh mục) với các tham số đầu vào là category_id, name_category, total_product.

-Câu lệnh thủ tục:

```
drop procedure if exists add_category
```

```
delimiter //
```

```
create procedure add_category (
```

```
    in category_id int,  
    in name_category int,  
    in total_product int  
)  
begin  
    insert into category(category_id, name_category, total_product)  
    values (category_id, name_category, total_product);  
end//  
delimiter ;
```

- Câu lệnh thực thi thủ tục mẫu:

```
call add_category(11,"Thước",4);
```

- Kết quả:

category_id	name_category	total_product
1	Quần	8
2	Áo	8
3	Giày	4
4	Nón	5
5	Đồng hồ	3
6	Dép	4
7	Bông tai	2
8	Bàn học	5
9	Casio	2
10	Vở	5
11	Thước	4
NULL	NULL	NULL

2.Lập trình ứng dụng

-Giao diện Category

Category				
TASK	All Category			
Manage Orders	Category ID	Name Category	Total Product	
Manage Feedback	1	Quần	8	Delete View
Manage Products	2	Áo	8	Delete View
Manage Product in cart	3	Giày	4	Delete View
Manage User	4	Nón	5	Delete View
Category	5	Đồng hồ	3	Delete View
	6	Dép	4	Delete View

-Giao diện insert_category

Category	
Insert Category	
Category ID:	<input type="text"/>
Name Category:	<input type="text"/>
Total Product:	<input type="text"/>
Insert New	

-Giao diện View_prodict_of_category

Category

The Category Has:

Shop ID	Product ID	Category ID
1	1	1

[Insert New](#)