

Міністерство освіти і науки України
Національний університет «Львівська політехніка»
Інститут комп'ютерних наук та інформаційних технологій
Кафедра «Системи штучного інтелекту»



Лабораторна робота №7
з дисципліни: «ОБДЗ»

*Виконав студент
групи КН-208*

Жеребецький О.В.

Прийняла:

асистент

Якимішин Х.М.

Львів-2020

Хід роботи

1. Виберемо всі активні товари та коментарі пов'язані з ними. Для цього використаємо ліве з'єднання. Якщо користувач не має квитка, то в полі `comment_id` буде порожнє значення(**NULL**).

```
SELECT active_goods.active_goods_id, active_goods.product_id, active_goods.arrive_address_city,  
active_goods.delivery_service_id, comment.comment_text, comment.comment_mark  
FROM active_goods LEFT JOIN comment ON  
active_goods.comment_id = comment.comment_id;
```

	active_goods_id	product_id	delivery_service_id	sell/buy	comment_id	activate_time	arrive_address_city	arrive_address_street
▶	1	1	2	BLOB	1	2020-10-02 00:00:00	address1	street1
	2	2	3	BLOB	2	2020-09-02 00:00:00	address2	street2
	3	3	2	BLOB	3	2020-02-02 00:00:00	address3	street4
	4	4	4	BLOB	4	2020-01-02 00:00:00	address4	street3
	6	4	2	BLOB	NULL	2020-01-02 00:00:00	address4	street3
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

2. Виберемо місце куди відправляється товар і ціну прописану сервісом доставки. Використовуючи директиву **INNER JOIN**.

```
SELECT active_goods.arrive_address_city, delivery_service.delivery_service_prices  
FROM active_goods INNER JOIN delivery_service ON  
active_goods.delivery_service_id = delivery_service.delivery_service_id;
```

	arrive_address_city	delivery_service_prices
▶	address1	10
	address2	20
	address3	10
	address4	30
	address4	10

Тепер товар як має адресу в *addrees1*.

```
SELECT active_goods.arrive_address_city, delivery_service.delivery_service_prices  
FROM active_goods INNER JOIN delivery_service ON  
active_goods.delivery_service_id = delivery_service.delivery_service_id  
where active_goods.arrive_address_city="address1"
```

	arrive_address_city	delivery_service_prices
▶	address1	10

3. Виберемо адресу складу і адресу куди потрібно доставити продукт. Товари мають бути створенні після 2020-08-01. Для цього умовне з'єднання таблиць *active_goods* і *product* за атрибутом *product_id*, та таблиці *product* і *storage* за атрибутом *storage_id*, використовуючи директиву **INNER JOIN**.

```
SELECT active_goods.arrive_address_city, storage.storage_adress_city
FROM (active_goods INNER JOIN product) inner join storage ON
active_goods.product_id = product.product_id
AND product.storage_id = storage.storage_id
where active_goods.activate_time>"2020-08-01"
```

	arrive_address_city	storage_adress_city
▶	address1	address1
	address2	address2

4. Виберемо всі активні замовлення для яких відповідний їм продукт лежить на складі з ціною оренди менше 7. Для цього замість директиви **JOIN** використаємо підзапит в умові відбору, який буде повертати потрібні продукти.

```
SELECT active_goods.active_goods_id, product.product_info
FROM active_goods INNER JOIN product
ON active_goods.product_id = product.product_id
WHERE product.storage_id IN (SELECT storage.storage_id FROM storage
WHERE storage.storage_rental_price < 7);
```

	active_goods_id	product_info
▶	1	info1
	2	info2
	4	info4
	6	info4
	3	info3

Виберемо перші три продукти

```
SELECT active_goods.active_goods_id, product.product_info
FROM active_goods INNER JOIN product
ON active_goods.product_id = product.product_id
WHERE product.storage_id IN (SELECT storage.storage_id FROM storage
WHERE storage.storage_rental_price < 7)
order by product.product_info desc limit 3;
```

	active_goods_id	product_info
▶	4	info4
	6	info4
	3	info3

5. Визначимо замовлення, яке не має жодного коментарю.

```
SELECT product_id, activate_time FROM active_goods
WHERE NOT EXISTS
(SELECT * FROM comment WHERE comment.comment_id = active_goods.comment_id);
```

	product_id	activate_time
▶	4	2020-01-02 00:00:00

6. Визначимо продукти, які мають назву теми довшу ні 3 символи або в ціні знаходяться букви.

```
SELECT product_id, product_info, product_price, product_theme
FROM product
WHERE NOT (CHAR_LENGTH(product_theme)>3) OR product_price REGEXP '[A-Z]';
```

	product_id	product_info	product_price	product_theme
•	NULL	NULL	NULL	NULL

```
SELECT product_id, product_info, product_price, product_theme
FROM product
WHERE NOT (CHAR_LENGTH(product_theme)=3) OR product_price REGEXP '[A-Z]';
|
```

	product_id	product_info	product_price	product_theme
▶	1	info1	10	theme1
	2	info2	12	theme2
	3	info3	11	theme3
	4	info4	6	theme4
	5	info5	8	theme5

Висновок: на цій лабораторній роботі було вивчено методи вибору даних зі з'єднаних таблиць БД засобами SQL та виконано запити до бази даних з використанням директив *SELECT* та *JOIN*, а також складних критеріїв в умові вибірки.