

Лабораторна робота 4

СТВОРЕННЯ ТА ВИКОРИСТАННЯ ПРЕДСТАВЛЕНЬ

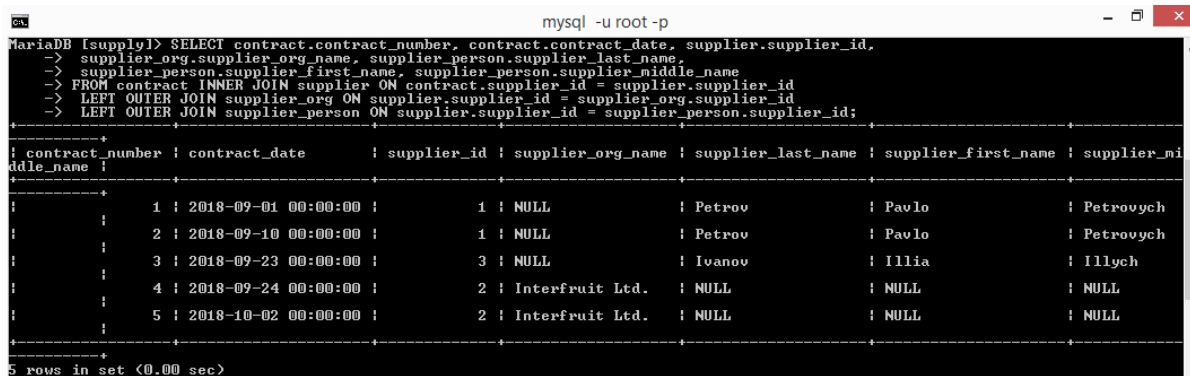
Мета роботи: навчитися створювати та застосовувати представлення (view) у базі даних на прикладі СУБД MySQL.

Хід роботи

1. Створити представлення, що дозволяє при перегляді списку договорів бачити назву постачальника

Створення представлень здійснюється за допомогою оператора CREATE VIEW. Таким чином, створити представлення, яке дозволить переглядати список договорів із зазначенням назви постачальника, можна на основі наступного запиту (рисунок 4.1).

```
SELECT contract.contract_number, contract.contract_date, supplier.supplier_id,  
       supplier_org.supplier_org_name, supplier_person.supplier_last_name,  
       supplier_person.supplier_first_name, supplier_person.supplier_middle_name  
FROM contract INNER JOIN supplier ON contract.supplier_id = supplier.supplier_id  
LEFT OUTER JOIN supplier_org ON supplier.supplier_id = supplier_org.supplier_id  
LEFT OUTER JOIN supplier_person ON supplier.supplier_id = supplier_person.supplier_id;
```



contract_number	contract_date	supplier_id	supplier_org_name	supplier_last_name	supplier_first_name	supplier_middle_name
1	2018-09-01 00:00:00	1	NULL	Petrov	Pavlo	Petrovych
2	2018-09-10 00:00:00	1	NULL	Petrov	Pavlo	Petrovych
3	2018-09-23 00:00:00	3	NULL	Ivanov	Illia	Illych
4	2018-09-24 00:00:00	2	Interfruit Ltd.	NULL	NULL	NULL
5	2018-10-02 00:00:00	2	Interfruit Ltd.	NULL	NULL	NULL

5 rows in set (0.00 sec)

Рисунок 4.1

Результат такого запиту має певний недолік – дані постачальників – юридичних та фізичних осіб знаходяться у різних стовпцях, а також присутні значення NULL. Цей недолік можна виправити за допомогою наступного запиту (рисунок 4.2).

```
SELECT contract.contract_number, contract.contract_date, supplier.supplier_id,
IFNULL(supplier_org.supplier_org_name, CONCAT(supplier_person.supplier_last_name, ' ',
supplier_person.supplier_first_name, ' ', supplier_person.supplier_middle_name)) AS `Supplier`
FROM contract INNER JOIN supplier ON contract.supplier_id = supplier.supplier_id
LEFT OUTER JOIN supplier_org ON supplier.supplier_id = supplier_org.supplier_id
LEFT OUTER JOIN supplier_person ON supplier.supplier_id = supplier_person.supplier_id;
```

```
mysql -u root -p
MariaDB [supply]> SELECT contract.contract_number, contract.contract_date, supplier.supplier_id,
-> IFNULL(supplier_org.supplier_org_name, CONCAT(supplier_person.supplier_last_name, ' ',
-> supplier_person.supplier_first_name, ' ', supplier_person.supplier_middle_name)) AS `Supplier`
-> FROM contract INNER JOIN supplier ON contract.supplier_id = supplier.supplier_id
-> LEFT OUTER JOIN supplier_org ON supplier.supplier_id = supplier_org.supplier_id
-> LEFT OUTER JOIN supplier_person ON supplier.supplier_id = supplier_person.supplier_id;
```

contract_number	contract_date	supplier_id	Supplier
1	2018-09-01 00:00:00	1	Petrov Pavlo Petrovych
2	2018-09-10 00:00:00	1	Petrov Pavlo Petrovych
3	2018-09-23 00:00:00	3	Ivanov Illia Illych
4	2018-09-24 00:00:00	2	Interfruit Ltd.
5	2018-10-02 00:00:00	2	Interfruit Ltd.

5 rows in set (0.00 sec)

Рисунок 4.2

Тепер можна створити дане представлення з назвою `contract_supplier` за допомогою відповідної команди мови SQL (рисунок 4.3).

```
mysql -u root -p
MariaDB [supply]> SHOW TABLES;
```

Tables_in_supply
contract
contract_supplier
supplied
supplier
supplier_org
supplier_person

6 rows in set (0.00 sec)

```
MariaDB [supply]> SELECT * FROM contract_supplier;
```

contract_number	contract_date	supplier_id	Supplier
1	2018-09-01 00:00:00	1	Petrov Pavlo Petrovych
2	2018-09-10 00:00:00	1	Petrov Pavlo Petrovych
3	2018-09-23 00:00:00	3	Ivanov Illia Illych
4	2018-09-24 00:00:00	2	Interfruit Ltd.
5	2018-10-02 00:00:00	2	Interfruit Ltd.

5 rows in set (0.01 sec)

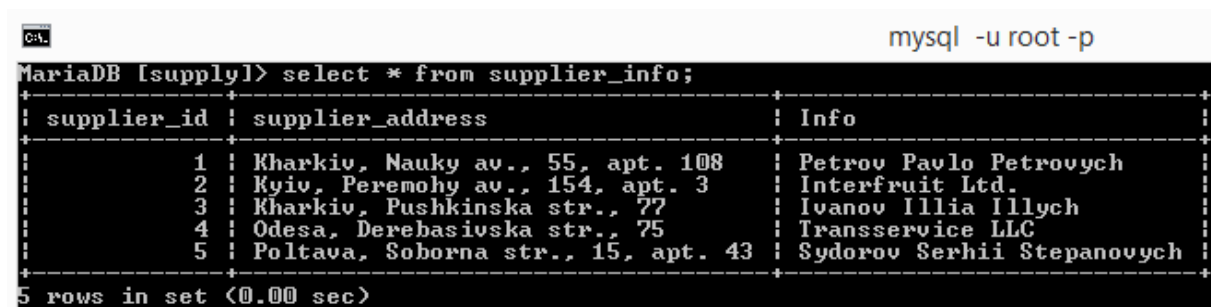
Рисунок 4.3

2. Створити представлення, що дозволяє користувачу працювати з обмеженими даними про постачальників

Припустимо, що для певних користувачів повинна бути доступна не уся загальна інформація про постачальників (збережена у таблиці `supplier`), а лише інформація про код та адресу постачальника. При цьому

користувач повинен мати можливість бачити дані постачальника як суб'єкта підприємницької діяльності (для юридичних осіб – назва, для фізичних – прізвище, ім'я, по батькові) (рисунок 4.4).

```
CREATE VIEW supplier_info AS
SELECT supplier.supplier_id, supplier.supplier_address,
    IFNULL(supplier_org.supplier_org_name, CONCAT(supplier_person.supplier_last_name, ' ',
        supplier_person.supplier_first_name, ' ', supplier_person.supplier_middle_name)) AS `Info`
FROM supplier LEFT OUTER JOIN supplier_org ON supplier.supplier_id = supplier_org.supplier_id
LEFT OUTER JOIN supplier_person ON supplier.supplier_id = supplier_person.supplier_id;
```



supplier_id	supplier_address	Info
1	Kharkiv, Nauky av., 55, apt. 108	Petrov Pavlo Petrovych
2	Kyiv, Peremohy av., 154, apt. 3	Interfruit Ltd.
3	Kharkiv, Pushkinska str., 77	Ivanov Illia Illych
4	Odesa, Derebasivska str., 75	Transservice LLC
5	Poltava, Soborna str., 15, apt. 43	Sydorov Serhii Stepanovych

5 rows in set (0.00 sec)

Рисунок 4.4

В разі виникнення необхідності, видалити представлення можна за допомогою оператора DROP VIEW.

3. Оформити звіт з лабораторної роботи

У звіт включити основні етапи виконання лабораторної роботи та знімки екрану, що їх демонструють.

4. Питання для самоконтролю

1. Що таке представлення?
2. Назвати переваги та недоліки представлень.
3. Який оператор мови SQL використовується для створення представлень?
4. Який оператор мови SQL використовується для видалення представлень?
5. Яким чином можна перевірити наявність представлення у базі даних?
6. Як вказати список стовпців при створенні представлення?
7. Що таке вертикальне представлення?
8. Що таке горизонтальне представлення?