

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТІ
“ЛЬВІВСЬКА ПОЛІТЕХНІКА”
Кафедра систем штучного інтелекту



Розрахунково-графічна робота
з дисципліни:
«Організація баз даних та знань»
на тему:
«Проектування БД для проекту RESTAVRATSIA»

Виконав студент групи КН-209

Качмар О.І.

Балів	Дата

Викладач

Мельникова Н. І.

Львів – 2020р

Тема проекту:

Створення веб-сервісу, призначеного для швидкої, зручної та безболісної комунікації та укладення угоди про реставрацію будь-якого виду об'єктів між замовником та компанією-виконавцем. Створення на платформі цього сервісу форуму відгуків та репортів про діяльність конкретної реставраційної компанії.

Зміст

Вступ.....	3
1. Логічна схема.....	5
2. Опис структури.....	6
3. Фізична модель.....	8
4. Ділова модель.....	12
5. Запити.....	15
6.Висновки.....	25
7.Список використаної літератури.....	26

Вступ

Кожен з нас має старовинні речі, які нам дорогі. Чи це антикварний кулон, який дістався вам від бабусі, чи годинник, що передав вам дідусь, який пройшов війну або ж пам'ятка вашого улюбленого міста, де ви народились і живете, що має надзвичайно вагому культурну та історичну цінність.

До прикладу, я громадянин Львова. Наше місто славиться багатьма перлинами архітектури, які занесені до пам'ятки ЮНЕСКО. Це старовинні будівлі, речі які в них знаходяться, пам'ятники, які, на щастя, неможливо приватизувати.

На жаль, ніщо не вічне. І старовинні речі, пам'ятні для нас потребують не тільки постійного догляду, але й відновлення, конкретніше кажучи, реставрації.

Реставрація – певні дії спрямовані на відновлення об'єктів та запобігання подальшому їх знищенню та забезпечення їх подальшої довговічності.

Так як, об'єкти, що підлягають реставрації дуже дорогі нам саме за їх пам'ять, антикварність тому, звісно ж реставраційні роботи проводяться з мінімальним втручанням в історичний матеріал об'єкта. Хороший реставратор береже культурну спадщину, тісно співпрацюючи з ученими, (технічними) мистецтвознавцями, археологами та антропологами. Автентичність, цілісність та стійкість - основні цінності, що формують догляд за спадщиною це міра того, наскільки успішною вона передається в майбутнє покоління.

Звісно ж, фахівці витрачають багато зусиль, щоб зберегти пам'ятки мистецтва і архітектури, так як кожен з них частина нашої історії, а як це зробить аматор?

Як людина, що не має жодного уявлення про процес, який дійсно можна назвати мистецтвом, володіючи річчю що має неоціненне культурно-історичне значення.

Уявіть собі, що станеться, якщо відбудеться втручання аматора в оригінальну давньоримську архітектуру або що станеться, коли дилетант почне якось змінювати старовинні скульптури доби італійського відродження.

Ми переконані, що ніщо з нічого не виникає і безслідно не зникає.

Саме тому ми командно створили незалежний сервіс за допомогою якого як звичайний обиватель, який володіє старовинною, дорогою його серцю річчю так і цілі громади, спільноти людей, що потребують реставрації об'єктів будь-якого типу, будь-якого значення можуть знайти справжніх майстрів, справжніх професіоналів своєї справи, які тільки і шукають, як щось комусь відреставрувати.

Наш проект сервіс має user-friendly інтерфейс, отже його вікова категорія людей є необмежена. Все, що потрібно зробити зі сторони замовника – це перейти до ресурсу, зареєструватись та сформулювати своє оголошення про потребу реставрації певного об'єкта з описом і фотографією та чекати. Через деякий час на замовлення подадуть заявку чимало майстрів, з пропозицією про співпрацю в якій вони вкажуть ціну за реставрацію вашого зазначеного в оголошенні об'єкта. Вам залишиться тільки обрати того кому довіритись і погодити пропозицію в ціновому аспекті.

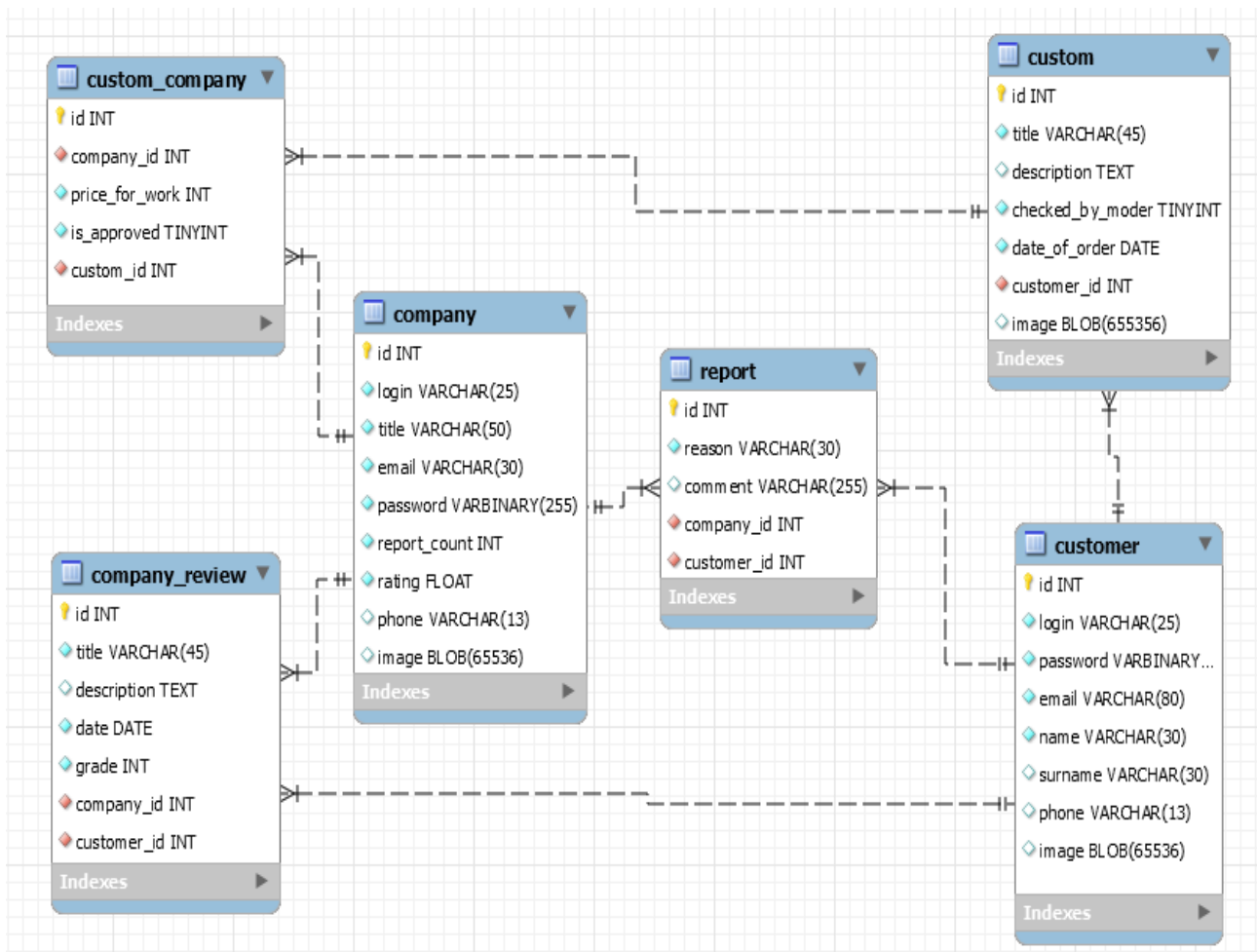
Якщо ж ви виконавець, а не замовник - ви так само маєте зареєструватись на нашому веб-ресурсі та здійснити пошук оголошень. При знаходженні цікавого вам оголошення перейдіть за оголошенням, огляньте його і підпишіться.

Ясна річ, що як і в реальному житті, так і в віртуальному середовищі є чимало шахраїв. Ці люди або не виконують своєї роботи, яка була домовлена, або виконують не приділяючи достатньо уваги. Ми не толеруємо таку поведінку і як хороші розробники розробили протидію шахраям.

Якщо вас на нашому ресурсі було обмануто чи вас чимось образила компанія, що виконувала ваше замовлення – ви можете сформулювати скаргу(репорт) і надіслати її адміністраторам. Також залишіть негативний відгук про цю компанію, це знизить її статус і, можливо її жертвами стане суттєва менша кількість людей.

Вдалого користування нашим ресурсом!

Логічна схема БД проекту



На схемі представлено 6 сутностей:

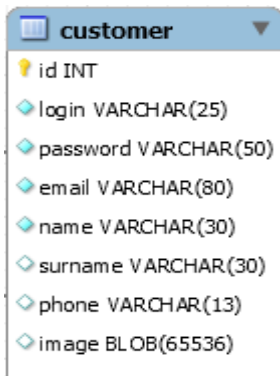
customer,company,report,custom,custom_company,company_review.

Ця схема представлена в нотації Баркера[6], бо сутності в ній представлені прямокутником, який містить ім'я сутності, її атрибути. Обов'язкові для вказання атрибути є NOT NULL. Реляційні зв'язки один-до-багатьох тут позначені відповідними стрілками. Для зв'язку багато-до-багатьох створені проміжні таблиці.

Опис структури БД

Задана БД *restavratsia* формується з наступних таблиць:

Таблиця `customer`



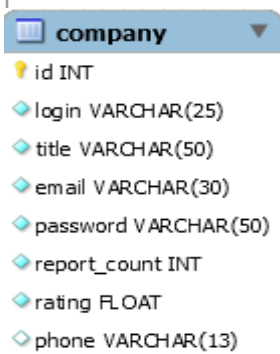
customer	
id	INT
login	VARCHAR(25)
password	VARCHAR(50)
email	VARCHAR(80)
name	VARCHAR(30)
surname	VARCHAR(30)
phone	VARCHAR(13)
image	BLOB(65536)

В таблиці *customer* містяться дані про сутність замовник. Сутність[1] характеризується строковими атрибутами, які є NOT NULL, це: *login*, *password*, *email*, *name* (ім'я замовника).

Строкові атрибути, які є NULL - це *surname* (прізвище), *phone*.

Крім того в цій таблиці зберігається інформація про особисте зображення замовника. Зображення є також NULL, типу BLOB з обмеженням даних на 64КБ.

Таблиця `company`



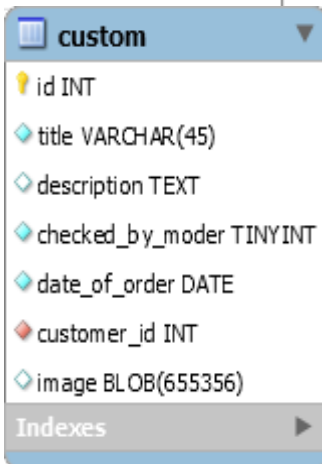
company	
id	INT
login	VARCHAR(25)
title	VARCHAR(50)
email	VARCHAR(30)
password	VARCHAR(50)
report_count	INT
rating	FLOAT
phone	VARCHAR(13)

Ця таблиця містить дані про сутність компанія-виконавець замовлення. Виконавець має наступні строкові NOT NULL[2] атрибути: *login*, *title* (назва компанії), *email*, *password*.

Атрибут *phone* – телефон компанії строкового типу NULL.

Атрибути *report_count* (кількість скарг про порушення норм) є цілочисельного типу NOT NULL, та *rating* (рейтинг компанії) типу числа з плаваючою комою NOT NULL. Значення цих атрибутів обчислюється відповідними тригерами.

Таблиця `custom`

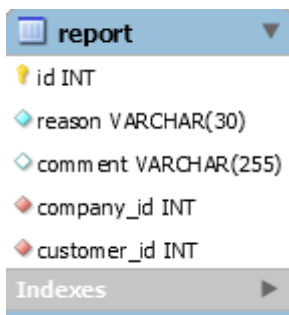


custom	
id	INT
title	VARCHAR(45)
description	TEXT
checked_by_moder	TINYINT
date_of_order	DATE
customer_id	INT
image	BLOB(65536)

Indexes

Custom описує сутність замовлення. Замовлення характеризується строковими NOT NULL атрибутами заголовок, опис замовлення. Кожне замовлення має дату публікації, цей атрибут ми назвали *date_of_order* типу DATE NOT NULL. При виведенні всіх замовлень ми використовували TINYINT(BOOL) поле *checked_by_moder* – перевірка на схвалення модератором такого замовлення. Кожне замовлення має замовника, тому ми вказали *customer_id* типу INT, зовнішній ключ на таблицю `customer` (customer реалізує зв'язок один-до-багатьох з таблицею custom).

Таблиця `report`



report	
id	INT
reason	VARCHAR(30)
comment	VARCHAR(255)
company_id	INT
customer_id	INT

Indexes

Report описує сутність репортів (скарг). Скарга має строкові NOT NULL атрибути *reason* (причина) та *comment*. Так як репорт про компанію формує замовник, тому таблиця `report` має атрибути-зовнішні ключі `company_id` та `customer_id` типу INT, таким чином реалізували зв'язок багато-до-багатьох між таблицями *Company* та *Customer*.

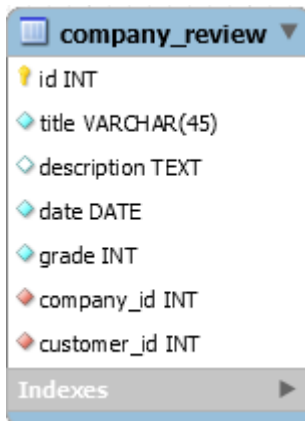
Таблиця `custom_company`



custom_company	
id	INT
company_id	INT
price_for_work	INT
is_approved	TINYINT
custom_id	INT

В таблиці custom_company зберігається інформація про сутність замовлення-компанія. Таблиця зберігає пропозиції аплікантів на певне замовлення з пропонованою ціною, price_for_work INT NOT NULL, за виконання реставраційної роботи. Отже, доречно буде використати атрибути-зовнішні ключі 'company_id' та 'custom_id' на відповідні таблиці, таким чином формуючи зв'язок багато-до-багатьох між компанією, що 'подається' на замовлення. Якщо замовник затвердить компанію, то в значення атрибуту is_approved TINYINT NOT NULL відповідного запису таблиці буде значення 1, інакше 0.

Таблиця `company_review`



company_review	
id	INT
title	VARCHAR(45)
description	TEXT
date	DATE
grade	INT
company_id	INT
customer_id	INT

В цій таблиці зберігається інформація про сутність відгуку про компанію. Користувач може конфігурувати свій відгук задаючи значення для строкових NOT NULL атрибутів - title, description. Дата створення відгуку буде збережено для атрибуту date типу DATE NOT NULL. Оцінка компанії буде визначатись полем grade INT NOT NULL. Так як відгук про компанію формує замовник, тому таблиця 'company_review' має атрибути-зовнішні ключі 'company_id' та 'customer_id' типу INT, таким чином реалізували зв'язок багато-до-багатьох між таблицями Company та Customer.

Крім того, що всі зовнішні ключі ми зробили індексованими (всі первинні за замовчуванням індексовані PRIMARY), що мало б пришвидшити операцію JOIN для таблиць, ми встановили для всіх Foreign Key ON DELETE CASCADE ON UPDATE CASCADE. Тобто при видаленні запису в таблиці company відповідний запис з таблиці company_review буде видалятися, так само при зміні Id запису з company відповідний foreign key з таблиці company_review зміниться.

Таблиці нашої БД нормалізовані за першими 3-ома формами нормалізації[2]:

- 1) Всі вони мають унікальний ключ, в нашому випадку id, всі значення атомарні.
- 2) Відповідність першій формі, незалежність частинно від складеного ключа (так як таких наші таблиці не містять, тому відповідає).
- 3) Відповідність другій формі, незалежність неключових полів.[5]

Фізична модель БД

```
CREATE DATABASE IF NOT EXISTS restavratsia DEFAULT CHARACTER SET utf8 ;  
USE restavratsia ;
```

```
use restavratsia;
```

```
CREATE TABLE IF NOT EXISTS company (  
    id INT NOT NULL AUTO_INCREMENT,  
    login VARCHAR(25) NOT NULL,  
    title VARCHAR(50) NOT NULL,  
    email VARCHAR(30) NOT NULL,  
    password VARCHAR(25) NOT NULL,  
    report_count INT NOT NULL,  
    rating FLOAT NOT NULL,  
    phone VARCHAR(13) NULL,  
    image BLOB(65536) NULL,  
    PRIMARY KEY ( id ))  
ENGINE = InnoDB;
```

```
CREATE TABLE IF NOT EXISTS customer (  
    id INT NOT NULL AUTO_INCREMENT,  
    login VARCHAR(25) NOT NULL,  
    password VARCHAR(25) NOT NULL,  
    email VARCHAR(30) NOT NULL,  
    name VARCHAR(30) NOT NULL,  
    surname VARCHAR(30) NULL,  
    phone VARCHAR(13) NULL,  
    image BLOB(65536) NULL,  
    PRIMARY KEY ( id ))  
ENGINE = InnoDB;
```

```
CREATE TABLE IF NOT EXISTS custom (  
    id INT NOT NULL AUTO_INCREMENT,
```

```

title VARCHAR(45) NOT NULL,
description TEXT NULL,
checked_by_moder TINYINT NOT NULL,
date_of_order DATE NOT NULL,
customer_id INT NOT NULL,
image BLOB(655356),
PRIMARY KEY ( id ),
INDEX fk_custom_customer1_idx ( customer_id ASC),
CONSTRAINT fk_custom_customer1
FOREIGN KEY ( customer_id )
REFERENCES restavratsia . customer ( id )
ON DELETE CASCADE #[2]
ON UPDATE CASCADE)
ENGINE = InnoDB; #[3]

```

```

CREATE TABLE IF NOT EXISTS customcompany (
id INT NOT NULL AUTO_INCREMENT,
company_id INT NOT NULL,
price_for_work INT NOT NULL,
is_approved TINYINT NOT NULL,
custom_id INT NOT NULL,
PRIMARY KEY ( id ),
INDEX company_id_idx ( company_id ASC),
INDEX fk_customcompany_custom1_idx ( custom_id ASC),
CONSTRAINT fk_Company_company_id
FOREIGN KEY ( company_id )
REFERENCES restavratsia . company ( id )
ON DELETE CASCADE
ON UPDATE CASCADE,
CONSTRAINT fk_customcompany_custom1
FOREIGN KEY ( custom_id )
REFERENCES restavratsia . custom ( id )

```

```
ON DELETE CASCADE
ON UPDATE CASCADE)
ENGINE = InnoDB;
```

```
CREATE TABLE IF NOT EXISTS company_review (
  id INT NOT NULL,
  title VARCHAR(45) NOT NULL,
  description TEXT NULL,
  date DATE NOT NULL,
  grade INT NOT NULL,
  company_id INT NOT NULL,
  customer_id INT NOT NULL,
  PRIMARY KEY ( id ),
  INDEX fk_company_review_company1_idx ( company_id ASC),
  INDEX fk_company_review_customer1_idx ( customer_id ASC),
  CONSTRAINT fk_company_review_company1
    FOREIGN KEY ( company_id )
    REFERENCES restavratsia . company ( id )
    ON DELETE CASCADE
    ON UPDATE CASCADE,
  CONSTRAINT fk_company_review_customer1
    FOREIGN KEY ( customer_id )
    REFERENCES restavratsia . customer ( id )
    ON DELETE CASCADE
    ON UPDATE CASCADE)
ENGINE = InnoDB;
```

```
CREATE TABLE IF NOT EXISTS report (
  id INT NOT NULL,
  reason VARCHAR(30) NOT NULL,
  comment VARCHAR(255) NULL,
```

```
company_id INT NOT NULL,  
customer_id INT NOT NULL,  
PRIMARY KEY ( id ),  
INDEX fk_report_company1_idx ( company_id ASC),  
INDEX fk_report_customer1_idx ( customer_id ASC),  
CONSTRAINT fk_report_company1  
FOREIGN KEY ( company_id )  
REFERENCES restavratsia . company ( id )  
ON DELETE CASCADE  
ON UPDATE CASCADE,  
CONSTRAINT fk_report_customer1  
FOREIGN KEY ( customer_id )  
REFERENCES restavratsia . customer ( id )  
ON DELETE CASCADE  
ON UPDATE CASCADE)  
ENGINE = InnoDB;
```

Ділова модель

Таблиці Функції	Компанія- виконавець	Замовник	Замовлення	Заяви (custom_company)	Репорти	Відгуки
Хешування паролів при створенні користувачів	*	*				
Реєстрація, Видалення, Апдейт, селекція даних замовника		*				
Реєстрація, Видалення, Апдейт, селекція даних виконавця	*					
Створення, Видалення, Апдейт, селекція даних конкретного замовлення			*			
Вибірка замовлень певного замовника		*	*			
Подача заявок на виконання реставраційної роботи.				*		
Перерахунок рейтингу компанії(тригер)	*					*
Створення відгуків користувачами про компанії						*
Вибірка компаній які подали заяву на конкретне оголошення	*	*		*		
Конфігурація репорту про компанію					*	

Перерахунок кількості репортів (тригер)	*				*	
Вибірка за допомогою курсору відгуків про конкретну компанію	*	*				*

1) Хешування паролю при створенні користувачів.

Ця операція буде виконана за допомогою тригера, який перед вставкою даних про користувача буде хешувати його пароль. За типом події – insert. За часом опрацювання-before. Працює на рівні рядка. Працює в тій же транзакції, а не автономно.

2) Реєстрація, Видалення, Апдейт, селекція даних замовника.

Це відповідно Insert , Delete ,Update та Select операції , які виконуються з таблицею customer, стосуються всіх атрибутів таблиці . Операції update може також бути здійснена вкладеним select для update за іменем login замовника.

3) Реєстрація, Видалення, Апдейт, селекція даних компанії-виконавця.

Це відповідно Insert , Delete ,Update та Select операції , які виконуються з таблицею company, стосуються всіх атрибутів таблиці.

4) Створення, Видалення, Апдейт, селекція даних конкретного замовлення

Це відповідно Insert , Delete та Select операції , які виконуються з таблицею custom, стосуються всіх атрибутів таблиці. Update здійснюється при передачі всіх даних про замовлення. (так працював серверний вбудований метод).

5) Вибірка замовлень певного замовника

Це операція селекції замовлення з таблиці custom , де значення атрибуту customer_id , замовника , що створив замовлення буде рівне тому, яке є в запиті.

6) Подача заявки на виконання реставраційної роботи

Це insert запит, який спрацьовує при подачі компанією заявки про виконання реставраційної роботи. Створює запис в таблиці custom_company. Отже, вказуються значення атрибутів company_id, custom_id. Крім того, компанія вказує ціну за виконання роботи для price_for_work-атрибуту. Також вказується при insert значення атрибуту is_approved=false, коли ж замовник схвалює компанію is_approved змінюється на true.

7) Перерахунок рейтингу компанії-

Ця операція здійснюється за рахунок тригера, який при додаванні відгуку про компанію здійснює перерахунок оцінки компанії, знаходячи середнє арифметичне атрибутів grade таблиці company_review та виконує update поля rating таблиці company. За типом події – Update. За часом дії – after. За об'єктом бд – на рівні рядка. В цій же транзакції.

8) Створення відгуку про компанію.

Це insert-запит, який створює запис в таблиці company_review, відгук замовника про компанію, отже мають бути вказані значення атрибутів company_id, customer_id та тих, які формують відгук : title, description, date, grade.

9) Вибірка компаній, які подали заяву на конкретне оголошення.

Ця операція здійснюється злиттям таблиць custom, company та custom_company за id. Результатом є вибірка значень атрибутів: title, description таблиці custom і price_for_work таблиці custom_company, тобто вивід даних компанія, опис замовлення і ціну реставраційної роботи.

10) Перерахунок кількості репортів.

Ця операція здійснюється за рахунок тригера, який при додаванні репорту про компанію здійснює перерахунок кількості репортів на компанію (кількість значень конкретної компанії атрибуту company_id) і виконує update company. За типом події – Update. За часом дії – after. За об'єктом бд – на рівні рядка. В цій же транзакції

11) Конфігурування репорту про компанію.

Це insert-запит, який створює запис в таблиці report, скарга замовника на компанію, отже мають бути вказані значення атрибутів company_id, customer_id та тих, які конфігурують скаргу : reason (причини: спам, ненормативна лексика і ін.), comment (докладний коментар скарги).

12) Вибірка за допомогою курсору відгуків про конкретну компанію

Виконуючи цю операцію, було створено procedure, в якій було задекларовано курсор та локальні змінні : title_review, description_review, title_company, full_name_user, report_counter. В ці змінні при циклічному зчитуванні з курсору (попередньо його відкривши) записувались результати query. Як тільки в змінні було зчитано дані результат виводився в консоль, курсор закривався, процедура завершувалась. Запит стосувався таблиць company_review, company, custom – виведення інформації про company_review.title as review, company_review.description as description, company.title as company, Concat(name, " ", surname) as user_full_name, report_count за допомогою декількох straight_join. З групуванням за компанією, сортуванням за датою відгуку. Розумію, що використання курсорів [4] знижує продуктивність прикладних програм [4], але ми використали подібний за суттю реалізації код в нашому проекті. Детальніший опис біля відповідної квері.

Запити до БД

1) Хешування паролю при створенні користувачів.

```
mysql> CREATE TRIGGER customer_password BEFORE
-> INSERT ON customer
-> FOR EACH ROW
-> SET NEW.password = AES_ENCRYPT(NEW.password, 'my_password');
Query OK, 0 rows affected (0.15 sec)
```

2) Реєстрація, Видалення, Апдейт, селекція даних замовника [7].

Реєстрація замовників

```
mysql> insert into customer
-> values(1,"AlexKach","29060leksii","loxaskorost@gmail.com","Oleksii","Kachmar","0932345818",""),
-> (2,"RestoMen","8124guch","resto@gmail.com","Taras","Borulkof","0932323418",""),
-> (3,"KapitanNemo123","34249Philosoph","cap121@gmail.com","Steve","Gerrard","0234345818",""),
-> (4,"ArniJackson","NetStars332","arnold@gmail.com","Arnold","Jackson","0834345818",""),
-> (5,"MaxGioMan","MaxMaximus41","Maxim@gmail.com","Max","Pulyk","0983243891",""),
-> (6,"ChakNarson","VarChak110","AndrewChak@gmail.com","Andrew","Yarchak","0954839103",""),
-> (7,"Mikado225","Tort100501","Imperator226@gmail.com","Vova","Rudyy","03748201384",""),
-> (8,"NazarKov","BestAlgorithmer21","KovalNazar@gmail.com","Nazar","Koval","0923454538","");
Query OK, 8 rows affected (0.09 sec)
Records: 8 Duplicates: 0 Warnings: 0
```

```
mysql> select * from customer1
-> ^C
mysql> select * from customer;
```

id	login	password	email	name	surname	phone	image
1	AlexKach	0x81f696912c1a63aa6f7b99b57c5bac64	loxaskorost@gmail.com	Oleksii	Kachmar	0932345818	0x
2	RestoMen	0x493fd6ef2582157696ec0806a057b2c0	resto@gmail.com	Taras	Borulkof	0932323418	0x
3	KapitanNemo123	0xb45a15c816d144029118d28d143af088	cap121@gmail.com	Steve	Gerrard	0234345818	0x
4	ArniJackson	0x67068bc50399f537d6de85cb002e24f6	arnold@gmail.com	Arnold	Jackson	0834345818	0x
5	MaxGioMan	0xf15fe94e06fb332b7ebc33ae39ba916a	Maxim@gmail.com	Max	Pulyk	0983243891	0x
6	ChakNarson	0x441e68ef82d74fc24847eb54aee2aded	AndrewChak@gmail.com	Andrew	Yarchak	0954839103	0x
7	Mikado225	0x3cf24a110a8fb1c63d05929d42010571	Imperator226@gmail.com	Vova	Rudyy	03748201384	0x
8	NazarKov	0x338fc26aad5a1e9cb9472dba1074e68eb7fb364979d08559e743304a315475019	KovalNazar@gmail.com	Nazar	Koval	0923454538	0x

8 rows in set (0.00 sec)

Апдейт, селекція замовника за id.

```
mysql> update customer
-> set login = "MaxRestavrator",
-> password = "MaxGio2906",
-> email = "Max_maximus@gmail.com"
-> ,phone = "0932358526"
-> where id = 5;
```

```
Query OK, 0 rows affected (0.06 sec)
Rows matched: 1 Changed: 0 Warnings: 0
```

```
mysql> select * from customer where id = 5;
```

id	login	password	email	name	surname	phone	image
5	MaxRestavrator	0x4d617847696f32393036	Max_maximus@gmail.com	Max	Pulyk	0932358526	0x

1 row in set (0.00 sec)

Видалення замовника за id-полем.

```
mysql> delete from customer where id = 5;
Query OK, 1 row affected (0.14 sec)

mysql> select * from customer;
```

id	login	password	email	name	surname	phone	image
1	AlexKach	0x81f696912c1a63aa6f7b99b57c5bac64	loxaskorost@gmail.com	Oleksii	Kachmar	0932345818	0x
2	RestoMen	0x493fd6ef2582157696ec0806a05782c0	resto@gmail.com	Taras	Borulkof	0932323418	0x
3	KapitanNemo123	0xb45a15c816d14402911bd28d143af088	cap121@gmail.com	Steve	Gerrard	0234345818	0x
4	ArniJackson	0x670688c50399f537d6deb5c8002e24f6	arnold@gmail.com	Arnold	Jackson	0834345818	0x
6	ChakNarson	0x441e68ef82d74fc24847eb54aee2aded	AndrewChak@gmail.com	Andrew	Yarchak	0954839103	0x
7	Mikado225	0x3cf24a110a8fb1c63d05929d42010571	Imperator226@gmail.com	Vova	Rudyy	03748201384	0x
8	NazarKov	0x338fc26aad5a1e9cb9472dba1074e68eb7fb364979d8559e743304a315475019	KovalNazar@gmail.com	Nazar	Koval	0923454538	0x

7 rows in set (0.00 sec)

3)Реєстрація,Видалення,Апдейт,селекція даних компанії-виконавця.

Реєстрація компаній-виконавців.

```
mysql> insert into company
-> values(1,"3Doors20","Door Restavr","doors3@gmali.com","0302bestDoors",0,0,"0923241423",""),
-> (2,"FurnitureBoys","Furniture Master","furniture23@gmail.com","weFur2020",0,0,"0853352965",""),
-> (3,"Facadeexperts","Facade Experts","facade664@gmail.com","FacadeYours47",0,0,"0413352982",""),
-> (4,"BalconyWorkers","Balcony Worker","balcon9843y@gmail.com","BalconTop21",0,0,"0663352923",""),
-> (5,"PlitkaMasters","Plitka Master","plitka1234@gmail.com","plitka1248",0,0,"0813352975",""),
-> (6,"BramaRestav","Brama Restavrer","brama432@gmail.com","getBramaRest25",0,0,"0323352916",""),
-> (7,"WindowChecker","Window Repair","windows668@gmail.com","windowseXp20",0,0,"0443352931",""),
-> (8,"Architects","Architect in architecture","architecture223@gmail.com","ArchitectProfi22",0,0,"0823352993",""),
-> (9,"BatteriesExp","Batteries and co","batteries112@gmail.com","batteries4401",0,0,"09623944",""),
-> (10,"KnifeProfi","Knife Professional","knifeExp446@gmail.com","knifeIsTop25",0,0,"0943352934","");
Query OK, 10 rows affected (0.22 sec)
Records: 10 Duplicates: 0 Warnings: 0
```

id	login	title	email	password	report_count	rating	phone	image
1	3Doors20	Door Restavr	doors3@gmali.com	0x3033303262657374446f6f7273	0	0	0923241423	0x
2	FurnitureBoys	Furniture Master	furniture23@gmail.com	0x776546757232303230	0	0	0853352965	0x
3	Facadeexperts	Facade profs	facadeProfs@gmail.com	0x466163616465596f7572733437	0	0	0942340192	0x
4	BalconyWorkers	Balcony Worker	balcon9843y@gmail.com	0x42616c636f6e546f703231	0	0	0663352923	0x
5	PlitkaMasters	Plitka Master	plitka1234@gmail.com	0x706c6974686131323438	0	0	0813352975	0x
6	BramaRestav	Brama Restavrer	brama432@gmail.com	0x6765744272616061526573743235	0	0	0323352916	0x
7	WindowChecker	Window Repair	windows668@gmail.com	0x77696e646f77736558703230	0	0	0443352931	0x
8	Architects	Architect in architecture	architecture223@gmail.com	0x41726368697465637450726f66693232	0	0	0823352993	0x
9	BatteriesExp	Batteries and co	batteries112@gmail.com	0x626174746572696573343031	0	0	09623944	0x
10	KnifeProfi	Knife Professional	knifeExp446@gmail.com	0x686e6966654973546f703235	0	0	0943352934	0x

Оновлення даних про конкретну компанію(що спеціалізується по певній області)

```
mysql> update company
-> set login = "CeramicsGuru",
-> title = "Ceramics Restavr",
-> email = "ceramics10@gmail.com"
-> where id = (select id where title like 'Door%');
Query OK, 1 row affected (0.12 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

```
mysql> select * from company;
```

id	login	title	email	password	report_count	rating	phone	image
1	CeramicsGuru	Ceramics Restavr	ceramics10@gmail.com	0x3033303262657374446F7273	0	0	0923241423	0x
2	FurnitureBoys	Furniture Master	furniture23@gmail.com	0x77654675723203230	0	0	0853352965	0x
3	Facadeexperts	Facade profs	facadeProfs@gmail.com	0x466163616465596F7572733437	0	0	0942340192	0x
4	BalconyWorkers	Balcony Worker	balcon9843y@gmail.com	0x42616C636F6E546F703231	0	0	0663352923	0x
5	PlitkaMasters	Plitka Master	plitka1234@gmail.com	0x706C6974606131323438	0	0	0813352975	0x
6	BramaRestav	Brama Restavr	brama432@gmail.com	0x6765744272616061526573743235	0	0	0323352916	0x
7	WindowChecker	Window Repair	windows668@gmail.com	0x77696E646F77736558703230	0	0	0443352931	0x
8	Architects	Architect in architecture	architecture223@gmail.com	0x41726368697465637450726F66693232	0	0	0823352993	0x
9	BatteriesExp	Batteries and co	batteries112@gmail.com	0x6261747465726965733434031	0	0	09623944	0x
10	KnifeProfi	Knife Professional	knifeExp446@gmail.com	0x6B6E6966654973546F703235	0	0	0943352934	0x

Вибірка даних про компанію відбувається з вказуванням Id.

```
mysql> select * from company where Id = 5;
```

id	login	title	email	password	report_count	rating	phone	image
5	PlitkaMasters	Plitka Master	plitka1234@gmail.com	0x706C6974606131323438	0	0	0813352975	0x

1 row in set (0.00 sec)

Delete відбувався теж за id.

```
mysql> delete from company where Id = 5;
Query OK, 1 row affected (0.27 sec)

mysql> select * from company;
```

id	login	title	email	password	report_count	rating	phone	image
2	FurnitureBoys	Furniture Master	furniture23@gmail.com	0x77654675723203230	0	0	0853352965	0x
3	Facadeexperts	Facade profs	facadeProfs@gmail.com	0x466163616465596F7572733437	0	0	0942340192	0x
4	BalconyWorkers	Balcony Worker	balcon9843y@gmail.com	0x42616C636F6E546F703231	0	0	0663352923	0x
6	BramaRestav	Brama Restavr	brama432@gmail.com	0x6765744272616061526573743235	0	0	0323352916	0x
7	WindowChecker	Window Repair	windows668@gmail.com	0x77696E646F77736558703230	0	0	0443352931	0x
8	Architects	Architect in architecture	architecture223@gmail.com	0x41726368697465637450726F66693232	0	0	0823352993	0x
9	BatteriesExp	Batteries and co	batteries112@gmail.com	0x6261747465726965733434031	0	0	09623944	0x
10	KnifeProfi	Knife Professional	knifeExp446@gmail.com	0x6B6E6966654973546F703235	0	0	0943352934	0x

8 rows in set (0.00 sec)

4) Створення,Видалення,Апдейт,селекція даних конкретного замовлення

Створення замовлень

```
mysql> insert into custom
-> values(1,"Need Furniture restavration","My table is quite old and I'd like you to ...",0,'13-05-03',2),
-> (2,"Crave for Architecture specialist","I have a sculpture...",0,'13-03-01',6),
-> (3,"Appreciate batteries exp","I got an ancient battery...",0,'17-01-09',4),
-> (4,"Brame masters are privileged","My bram has recently broken...",0,'05-07-05',8),
-> (5,"Need a Knife profi","These knives were tranported to me ...",0,'18-03-02',2),
-> (6,"Windows guru highly needed","My windows' old handle ...",0,'16-02-02',1),
-> (7,"If you're an expert of Balcony check it up","I'd like to glaze the balcony ...",0,'12-02-02',3),
-> (8,"Facade masters are respected","Please review my facade it's totally ...",0,'14-02-03',2),
-> (9,"Doors","...",0,'12-02-02',7),
-> (10,"Need Furniture restavration","My table is quite old and I'd like you to ...",0,'11-08-09',6);
Query OK, 10 rows affected (0.09 sec)
Records: 10 Duplicates: 0 Warnings: 0
```

Оновлення замовлення, відбувається тільки при передачі даних про попередню його версію. (Так працював серверний метод проекту)

```
mysql> update custom
-> set title = "Need Architecture solution",
-> description = "We are Lviv architecture department and out aim is to renew view of sculpture...."
-> where title = "Crave for Architecture specialist" and description = "I have a sculpture..." and checked_by_moder = 0;
Query OK, 1 row affected (0.23 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

Перехід до конкретного замовлення.

```
mysql> select * from custom where id = 3;
+-----+-----+-----+-----+-----+-----+
| id | title                | description                | checked_by_moder | date_of_order | customer_id |
+-----+-----+-----+-----+-----+-----+
| 3 | Appreciate batteries exp | I got an ancient battery... | 0 | 2017-01-09 | 4 |
+-----+-----+-----+-----+-----+-----+
1 row in set (0.09 sec)
```

Видалення замовлення можна зробити як зі списку замовлень:

```
mysql> delete from custom where title = "Need a Knife profi" and description = "These knives were tranported to me ...";
Query OK, 1 row affected (0.24 sec)
```

Так і перейшовши до конкретного замовлення і відіславши ріквестом на сервер його id.

```
mysql> delete from custom where id = 4;
Query OK, 1 row affected (0.22 sec)
```

```
+-----+-----+-----+-----+-----+
| id | title                | description                | checked_by_moder | date_of_order | customer_id |
+-----+-----+-----+-----+-----+-----+
| 1 | Need Furniture restavrnation | My table is quite old and I'd like you to ... | 0 | 2013-05-03 | 2 |
| 2 | Need Architecture solution | We are Lviv architecture department and out aim is to renew view of sculpture.... | 0 | 2013-03-01 | 6 |
| 3 | Appreciate batteries exp | I got an ancient battery... | 0 | 2017-01-09 | 4 |
| 6 | Windows guru highly needed | My windows' old handle ... | 0 | 2016-02-02 | 1 |
| 7 | If you're an expert of Balcony check it up | I'd like to glaze the balcony ... | 0 | 2012-02-02 | 3 |
| 8 | Facade masters are respected | Please review my facade it's totally ... | 0 | 2014-02-03 | 2 |
| 9 | Doors | ... | 0 | 2012-02-02 | 7 |
| 10 | Need Furniture restavrnation | My table is quite old and I'd like you to ... | 0 | 2011-08-09 | 6 |
+-----+-----+-----+-----+-----+-----+
```

5) Вибірка замовлень певного замовника,це дані які будуть показані в списку замовлень користувача.[10]

```
mysql> SET @customerId = 3;
Query OK, 0 rows affected (0.03 sec)

mysql> select title,description,date_of_order from custom
-> where customer_id = @customerId;
```

title	description	date_of_order
If you're an expert of Balcony check it up	I'd like to glaze the balcony ...	2012-02-02
Old Balcony repair project	My wood balcony is actually done so I need you to	2013-03-05
Brick Balcony restoration	I'll check every applicant for this job...	2010-08-01
My balcony is exhausted	As you can see I've not served my balcony for...	2011-11-01
Glass balcony improvement	In the city center our department...	2017-03-01
Build brand-new balcony on base of elder one	We got a basis for...	2019-02-05

```
6 rows in set (0.00 sec)
```

6) Подача заявок на виконання реставраційної роботи.

```
mysql> Insert into custom_company
-> values(1,2,300,0,1),
-> (2,12,800,0,1),
-> (3,13,400,0,1),
-> (4,11,600,0,1),
-> (5,14,1500,0,8),
-> (6,15,7200,0,8),
-> (7,3,8300,0,8),
-> (8,10,1020,0,16),
-> (9,16,900,0,16),
-> (10,17,800,0,16),
-> (11,18,600,0,16),
-> (12,4,1300,0,13),
-> (13,6,1270,0,6),
-> (14,7,850,0,1),
-> (15,8,700,0,2);
Query OK, 15 rows affected (0.20 sec)
Records: 15 Duplicates: 0 Warnings: 0
```

```
mysql> select * from custom_company;
```

id	company_id	price_for_work	is_approved	custom_id
1	2	300	0	1
2	12	800	0	1
3	13	400	0	1
4	11	600	0	1
5	14	1500	0	8
6	15	7200	0	8
7	3	8300	0	8
8	10	1020	0	16
9	16	900	0	16
10	17	800	0	16
11	18	600	0	16
12	4	1300	0	13
13	6	1270	0	6
14	7	850	0	1
15	8	700	0	2

```
15 rows in set (0.00 sec)
```

7)Перерахунок рейтингу компанії.

```
mysql> delimiter $$
mysql> CREATE TRIGGER recalculate_rating
  -> After insert on company_review
  -> For each row
  -> Begin
  -> Set @new_company_rate = (select sum(grade)/count(company_id) from company_review where company_id = new.company_id)
  -> update company
  -> set rating = @new_company_rate
  -> where id = NEW.company_id;
  -> end$$
Query OK, 0 rows affected (0.19 sec)

mysql> delimiter ;
mysql> delete from company_review;
Query OK, 2 rows affected (0.15 sec)
```

8)Створення відгуків користувачами про компанії.

```
mysql> insert into company_review
  -> values(1,"Not Bad furniture service","This was normal job everything...","20-03-01",5,2,1),
  -> (2,"Good Furniture skills!","", "20-06-04",8,2,7),
  -> (3,"Facade is done not good enough(","I was not satisfied with your facade proection due to next reasons","19-01-08",6,3,4),
  -> (4,"You guys did facade of my home well","", "18-03-01",7,3,6),
  -> (5,"Your job with my balcony is done not well","There are some downsides in your job","20-02-05",4,4,3),
  -> (6,"Very Bad Work!!!","", "20-03-01",1,4,2),
  -> (7,"Well done!","I recommend these Brama restavrators","19-01-06",9,6,8),
  -> (8,"You're amazing!","", "16-03-01",10,6,1),
  -> (9,"My Windows were bad furnished!(","", "13-03-01",4,7,4),
  -> (10,"Great job with my windows!","", "17-03-01",7,7,7),
  -> (11,"NOT Recommend this furniture service!","", "12-03-01",4,2,3);
Query OK, 11 rows affected (0.14 sec)
Records: 11 Duplicates: 0 Warnings: 0
```

```
mysql> select * from company_review;
```

id	title	description	date	grade	company_id	customer_id
1	Not Bad furniture service	This was normal job everything...	2020-03-01	5	2	1
2	Good Furniture skills!		2020-06-04	8	2	7
3	Facade is done not good enough(I was not satisfied with your facade proection due to next reasons	2019-01-08	6	3	4
4	You guys did facade of my home well		2018-03-01	7	3	6
5	Your job with my balcony is done not well	There are some downsides in your job	2020-02-05	4	4	3
6	Very Bad Work!!!		2020-03-01	1	4	2
7	Well done!	I recommend these Brama restavrators	2019-01-06	9	6	8
8	You're amazing!		2016-03-01	10	6	1
9	My Windows were bad furnished!(2013-03-01	4	7	4
10	Great job with my windows!		2017-03-01	7	7	7
11	NOT Recommend this furniture service!		2012-03-01	4	2	3

Оцінки Компанії після відгуків

```
mysql> select * from company;
```

id	login	title	email	password	report_count	rating	phone	image
2	FurnitureBoys	Furniture Master	furniture23@gmail.com	0x776546757232303230	0	5.66667	0853352965	0x
3	Facadeexperts	Facade profs	facadeProfs@gmail.com	0x466163616465596F7572733437	0	6.5	0942340192	0x
4	BalconyWorkers	Balcony Worker	balcon9843y@gmail.com	0x42616C636F6E546F703231	0	2.5	0663352923	0x
6	BramaRestav	Brama Restavrer	brama432@gmail.com	0x6765744272616D61526573743235	0	9.5	0323352916	0x
7	WindowChecker	Window Repair	windows668@gmail.com	0x77696E646F77736558703230	0	5.5	0443352931	0x

8) Вибірка компаній які подали заяву на конкретне оголошення.(вибірка тих, ціна роботи яких від 300 до 1500 з сортуванням.)[8]

```
mysql> select company.title as company,custom.title as custom,custom_company.price_for_work
-> from custom_company
-> Straight_join company
-> Straight_join custom
-> On custom_company.custom_id = custom.Id
-> And custom_company.company_id = company.Id
-> Where price_for_work between 300 and 1500 and custom.id in (select id from custom where title Like '%Furniture%')
-> Order By price_for_work;
```

company	custom	price_for_work
Furniture Master	Need Furniture restavration	300
Best furni experts	Need Furniture restavration	400
Furniture restavrnrs	Need Furniture restavration	600
Furniture and co	Need Furniture restavration	800
Window Repair	Need Furniture restavration	850

5 rows in set (0.09 sec)

Конфігурування репорту про компанію.Перерахунок кількості репортів.

```
mysql> delimiter $$
mysql> Create trigger add_number_of_reports
-> After insert on report
-> for each row
-> Begin
-> update company
-> set report_count = report_count + 1
-> where id = new.company_id;
-> End$$
Query OK, 0 rows affected (0.13 sec)

mysql> delimiter ;
```

```
mysql> insert into report
-> values(1,"Bullying","",4,1),
-> (2,"Racism","",4,2),
-> (3,"Profanity","A lot of bad words and aggression being in contact",3,1),
-> (4,"Cheating company","I've lost a lot of money because of its cheating papers",2,1),
-> (5,"Deception company","Please check out its diploma",4,3),
-> (6,"Wrong data","This company gives wrong prices for work",2,2);
Query OK, 6 rows affected (0.30 sec)
Records: 6 Duplicates: 0 Warnings: 0
```

```
mysql> select * from company;
```

id	login	title	email	password	report_count	rating	phone	image
2	FurnitureBoys	Furniture Master	furniture23@gmail.com	0x776546757232303230	2	5.6667	0853352965	0x
3	Facadeexperts	Facade prosfs	facadeProfs@gmail.com	0x466163616465596f7572733437	1	6.5	0942340192	0x
4	BalconyWorkers	Balcony Worker	balcon9843y@gmail.com	0x426160636f6e546f703231	3	2.5	0663352923	0x
6	BramaRestav	Brama Restavrer	brama432@gmail.com	0x6765744272616061526573743235	0	9.5	0323352916	0x
7	WindowChecker	Window Repair	windows668@gmail.com	0x77696e646f77736558703230	0	5.5	0443352931	0x
8	Architects	Architect in architecture	architecture223@gmail.com	0x41726368697465637450726f66693232	0	0	0823352993	0x

9)Вибірка за допомогою курсору відгуків про конкретну компанію.[9]

```
mysql> delimiter $$
mysql> Create Procedure Cursor_Join()
-> Begin
-> Declare title_review varchar(45);
-> Declare description_review TEXT;
-> Declare title_company varchar(50);
-> Declare full_name_user varchar(100);
-> Declare report_counter int;
-> Declare done int default 0;
->
-> Declare my_cur Cursor For
-> select company_review.title as review,company_review.description as description,
-> company.title as company,Concat(name," ",surname) as user_full_name,report_count
-> from company_review
-> Straight_Join company
-> Straight_Join customer
-> On company_review.company_id = company.id
-> And company_review.customer_id = customer.id
-> Group By company.id
-> Order By date;
->
-> DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;
-> Open my_cur;
->
-> read_loop:LOOP
-> IF done THEN
-> LEAVE read_loop;
-> End if;
-> FETCH my_cur INTO title_review,description_review,
-> title_company,full_name_user,report_counter;
->
-> Select title_review,description_review,
-> title_company,full_name_user,report_counter;
-> END LOOP;
->
-> Close my_cur;
-> End$$
Query OK, 0 rows affected (0.13 sec)
```

Робота процедури з курсором.[11]

```
mysql> call Cursor_Join();
```

title_review	description_review	title_company	full_name_user	report_counter
My Windows were bad furnished!		Window Repair	Arnold Jackson	0

```
1 row in set (0.00 sec)
```

title_review	description_review	title_company	full_name_user	report_counter
Well done!	I recommend these Brama restavrators	Brama Restavner	Nazar Koval	0

```
1 row in set (0.01 sec)
```

title_review	description_review	title_company	full_name_user	report_counter
Facade is done not good enough	I was not satisfied with your facade proection due to next reasons	Facade profs	Arnold Jackson	1

```
1 row in set (0.02 sec)
```

title_review	description_review	title_company	full_name_user	report_counter
Your job with my balcony is done not well	There are some downsides in your job	Balcony Worker	Steve Gerrard	3

```
1 row in set (0.04 sec)
```

title_review	description_review	title_company	full_name_user	report_counter
Not Bad furniture service	This was normal job everything...	Furniture Master	Oleksii Kachmar	2

```
1 row in set (0.05 sec)
```

Подібний код запиту в нашому проєкті на ASP.NET Core

```
[Microsoft.AspNetCore.Mvc.HttpPost]
[Microsoft.AspNetCore.Mvc.Route("login")]
0 references
public async Task<ActionResult> Login([Microsoft.AspNetCore.Mvc.FromBody] LoginViewModel model)
{
    if (ModelState.IsValid)
    {
        var result = await _signInManager.PasswordSignInAsync(model.Login, model.Password, model.RememberMe, false);
        if (result.Succeeded)
        {
            string mySelectQuery = "SELECT id, login, isCompany FROM user where login = '" + model.Login + "'";
            MySqlConnection myConnection = new MySqlConnection("Server=localhost;port=3306;" +
                "user=root;password=leomessii021;database=mydb");
            MySqlCommand myCommand = new MySqlCommand(mySelectQuery, myConnection);
            myCommand.Connection.Open();
            //Так декларується курсор і йому присвоюється результат query вище
            var ob = myCommand.ExecuteReader();
            var arr = new List<string>();
            while (ob.Read())//Та ж операція , що і Fetch
            {
                //зчитування даних тільки не в змінні, а в Масив.
                arr.Add(ob[0].ToString() + " " + ob[1].ToString() + " " + ob[2].ToString());
            }
            return Ok(Json(arr));
        }
    }
}
```


Цей код схожий до коду нашої процедури `Cursor_Join`, адже ми так само створюємо `query`, результат роботи якої записується в об'єкт класу `MySqlDataReader` і цей об'єкт по факту і є операцією `declare` та `open` нашого курсора. `Ob.Read()` –аналог операції `Fetch` нашого курсору, в цьому коді ми так само циклічно перебираємо результати , записані в курсор.Щоправда, ми додаємо результати в масив, а не записуємо в локальні змінні, як в `mysql` скрипті.Знищується об'єкт `Ob` прихованим методом `Dispose` – це аналог `Close` для нашого курсора.

Висновки до роботи

Отже, виконавши цю роботу, я описав основні методи проектування бази даних нашого проекту та сформував відповідні запити для її створення та обробки. Основними сутностями моєї бази даних були:

- 1)Замовник , описує особу,якій потрібна реставрація
- 2)Компанія-виконавець, описує компанію,яка виконувала цю реставрацію
- 3)Замовлення,описує замовлення,яке формує замовник на своїй сторінці(має виконавця)
- 4)Репорт_на_компанію,характеризує конфігурацію скарги,які складає замовник на компанію(багато-до-багатьох замовника і компанії)
- 5)Відгуки_про_компанію,описує відгук замовника про компанію(багато-до-багатьох замовника і компанії)
- 6)Компанія_замовлення,формується з пропертей домовленості між компанією і замовленням(багато-до-багатьох компанії і замовлення).

Основними операціями з моєю базою даних були CRUD-операції з замовником,компанією-виконавцем,замовленням(використав INSERT,UPDATE,DELETE,SELECT до відповідних таблиць);подання компанії на певне замовлення(INSERT до custom_company);створення скарг на компанію та відгуків про неї(INSERT'и до report,company_review відповідно);вибірка замовлень,на які підписана компанія(злиттям таблиць company,custom,custom_company); щоб симулювати код проекту, було створено курсор, в який записувались результати query,що вибирала інформацію про відгуки про компанію.

Також, для підтримки цілісності моєї бд в процесі було створено тригери:хешування паролів користувачів,перерахунок рейтингу компанії при вставці відгуку з оцінкою та підрахунку кількості репортів про компанія,на основі яких адміністратор ресурсу виносить рішення про блокування компанії на нашій сторінці.

Список використаної літератури

- 1) Конспект лекцій Мельникова Н.І.
- 2) Пасічник В.В., Резніченко В.А. Організація баз даних та знань
- 3) Kroenke D.M., Auer D.J. Database Processing: Fundamentals, Design, and Implementation. 14th ed
- 4) Powell G. Beginning Database Design
- 5) <https://habr.com/ru/post/254773/>
- 6) <http://enisey.name/umk/teis/ch18s04s09.html>
- 7) <https://www.w3schools.com/sql/default.asp>
- 8) [https://uk.wikipedia.org/wiki/Join_\(SQL\)](https://uk.wikipedia.org/wiki/Join_(SQL))
- 9) <https://www.mysqltutorial.org/mysql-cursor/>
- 10) http://www1.udel.edu/evelyn/SQL-Class3/SQL3_AggrSyntax.html
- 11) <https://www.w3resource.com/mysql/mysql-procedure.php>