МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ВОДНОГО ГОСПОДАРСТВА ТА ПРИРОДОКОРИСТУВАННЯ

Кафедра комп’ютерних наук та прикладної математики

КУРСОВА РОБОТА

з дисципліни “Бази даних”

на тему

“База даних картинної галереї ”

Виконала студентка ННІАКОТ:

Група КН-31

Войцеховська Діана Вікторівна

Керівник:

к. е. н. Бачишина Л. Д.

Рівне-2021

**Зміст**

[Вступ 3](#_Toc74620850)

[Опис предметної області 4](#_Toc74620851)

[Нормалізація бази даних 6](#_Toc74620852)

[Умови збереження цілісності 8](#_Toc74620853)

[Інфологічна модель даних 10](#_Toc74620854)

[Форми для введення та корегування даних 10](#_Toc74620855)

[Запити до бази даних: 13](#_Toc74620856)

[Тригери 14](#_Toc74620857)

[Процедури 15](#_Toc74620858)

[Звіти 16](#_Toc74620859)

[Висновки 18](#_Toc74620860)

[Список використаної літератури 19](#_Toc74620861)

# Вступ

Бази даних є систематичним збором даних або інформації, спеціально організованої для швидкого пошуку. Вони підтримують електронне зберігання, редагування, пошук та формування даних. A [система](https://www.britannica.com/technology/database-management-system) керування базами [даних](https://www.britannica.com/technology/database-management-system) (СКБД) витягує інформацію з бази даних у відповідь на запити. Безсумнівно бази даних полегшують управління даними. Саме через це вони є досить поширеними у використанні.

Сьогодні у світі існує безліч динамічних веб-сайтів, які обробляються через бази даних. Для роботи з даними, що зберігаються в базі даних, використовується SQL або мова структурованих запитів. SQL – це мова програмування, що використовується в більшості реляційних баз даних в якості ефектного способу збереження даних, пошуку їх частин, поновлення, вилучення з бази і видалення. В основному, SQL є головним інструментом оптимізації та обслуговування бази даних, хоча він і не обмежений цими цілями.

У свою чергу, для зручності роботи з СУБД використовуються спеціальні веб-додатки, які дозволяють за допомогою графічного інтерфейсу виконувати адміністрування сервера баз даних, запускати спеціальні команди, а також працювати з контентом таблиць і баз даних в цілому. Прикладом такого додатку є phpMyAdmin, який використовується для адміністрування [системи](https://www.britannica.com/technology/database-management-system) керування базами [даних](https://www.britannica.com/technology/database-management-system) MySQL. Він написаний на мові PHP, що дозволяє працювати з базами даних і саме за допомогою цього веб-додатку була виконана дана робота.

Метою даної курсової роботи є отримання навичок використання баз даних в створенні різних сайтів та веб-додатків, а також створення самих баз даних і роботи з ними.

# Опис предметної області

Нехай потрібно розробити модель бази даних картинної галереї.

База потрібна містити в собі дані про картини, художника, виставки, білети та здійснювати облік придбаних білетів.

Дані в найбільш поширених типах сучасних баз даних зазвичай зберігаються у вигляді рядків і стовпців, що формують таблицю.

Всю інформацію про базу даних можна помістити в одну таблицю, проте поля в процесі проектування можуть бути змінені.

|  |  |  |
| --- | --- | --- |
| **Назва поля** | **Тип поля** | **Розмірність** |
| [Код Картини](#_bookmark36) | Числовий(Ціле) | 11 |
| [Назва](#_bookmark37) картини | Текстовий | 45 |
| [Рік написання](#_bookmark42) | Дата |  |
| Код художника | Числовий(Ціле) | 11 |
| [Код Направлення в мистецтві](#_bookmark47) | Числовий(Ціле) | 17 |
| [номер білету](#_bookmark60) | Числовий(Ціле) | 11 |
| Ціна | Грошовий | 10,2 |
| Кількість проданих білетів | Числовий(Ціле) | 5 |
| Код Виставки | Числовий(Ціле) | 11 |
| Тема Виставки | Текстовий | 45 |
| Дата початку | Дата |  |
| Дата закінчення | Дата |  |
| Номер білету | Числовий(Ціле) | 11 |
| Номер залу | Текстовий | 5 |
| Місце виставки | Текстовий | 45 |
| Назва Направлення | Текстовий | 45 |
| Прізвище художника | Текстовий | 45 |
| Ім’я | Текстовий | 45 |
| По-батькові | Текстовий | 45 |
| Місце народження | Текстовий | 45 |
| Діяльність | Текстовий | 45 |

Щоб структура бази даних була ефективною, потрібно забезпечити швидкий доступ до таблиць, виключити непотрібне повторення даних та забезпечити цілісність даних так, щоб при видаленні відбувалися автоматичні зміни в об’єктах.

Також база даних картинної галереї повинна мати реєстрацію для нових художників, щоб і вони мали змогу запропонувати свої роботи, та форми для того, щоб запропонувати свої картини чи виставки.

Користувач сайту повинен володіти інформацією про всі проведення виставок, список зареєстрованих художників, та інформацією про наявність картин, які присутні на виставках.

# Нормалізація бази даних

Щоб таблиця відповідала всім умовам і була без повторюваних даних, нормалізуємо базу даних згідно теорії нормалізації Кодда.

Для цього поділимо таблицю на декілька інших таблиць, які поєднанні між собою за допомогою даних з первинними ключами.

**Таблиця 1**

**Картини**

Поле **Код Картини** є ключовим, первинним та автоінкриментом, оскільки воно повинно бути унікальним і не повторюватися, а також автоматично додаватися при введені нового запису у таблицю.

|  |  |  |
| --- | --- | --- |
| **Назва поля** | **Тип поля** | **Розмірність** |
| Код Картини | Числовий(Ціле) | 11 |
| Назва | Текстовий | 45 |
| Рік написання | Дата |  |
| Код Художника | Числовий(Ціле) | 11 |
| Код Направлення в мистецтві | Числовий(Ціле) | 17 |

**Таблиця 2**

**Білети**

Поле **номер білету** є ключовим.

|  |  |  |
| --- | --- | --- |
| **Назва поля** | **Тип поля** | **Розмірність** |
| Номер білету | Числовий(Ціле) | 11 |
| ціна | Грошовий | 10,2 |
| Кількість проданих білетів | Числовий(Ціле) | 5 |

**Таблиця 3**

**Виставки**

Поле **Код Виставки** є ключовим.

|  |  |  |
| --- | --- | --- |
| **Назва поля** | **Тип поля** | **Розмірність** |
| Код Виставки | Числовий(Ціле) | 11 |
| Тема Виставки | Текстовий | 45 |
| Дата початку | Дата |  |
| Дата закінчення | Дата |  |
| Номер білету | Числовий(Ціле) | 11 |

**Таблиця 4**

**Виставлення картини**

Поле **Код Виставки** є ключовим.

|  |  |  |
| --- | --- | --- |
| **Назва поля** | **Тип поля** | **Розмірність** |
| Код Виставки | Числовий(Ціле) | 11 |
| Код Картини | Числовий(Ціле) | 11 |
| Номер залу | Текстовий | 5 |
| Місце виставки | Текстовий | 45 |

**Таблиця 5**

**Направлення в мистецтві**

Поле **Код Направлення в мистецтві** є ключовим.

|  |  |  |
| --- | --- | --- |
| **Назва поля** | **Тип поля** | **Розмірність** |
| Код Направлення в мистецтві | Числовий(Ціле) | 17 |
| Назва Направлення | Текстовий | 45 |

**Таблиця 6**

**Художники**

Поле **Код Художника** є ключовим.

|  |  |  |
| --- | --- | --- |
| **Назва поля** | **Тип поля** | **Розмірність** |
| Код Художника | Числовий(Ціле) | 11 |
| Прізвище | Текстовий | 45 |
| Ім’я | Текстовий | 45 |
| По-батькові | Текстовий | 45 |
| Місце Народження | Текстовий | 45 |
| Діяльність | Текстовий | 45 |

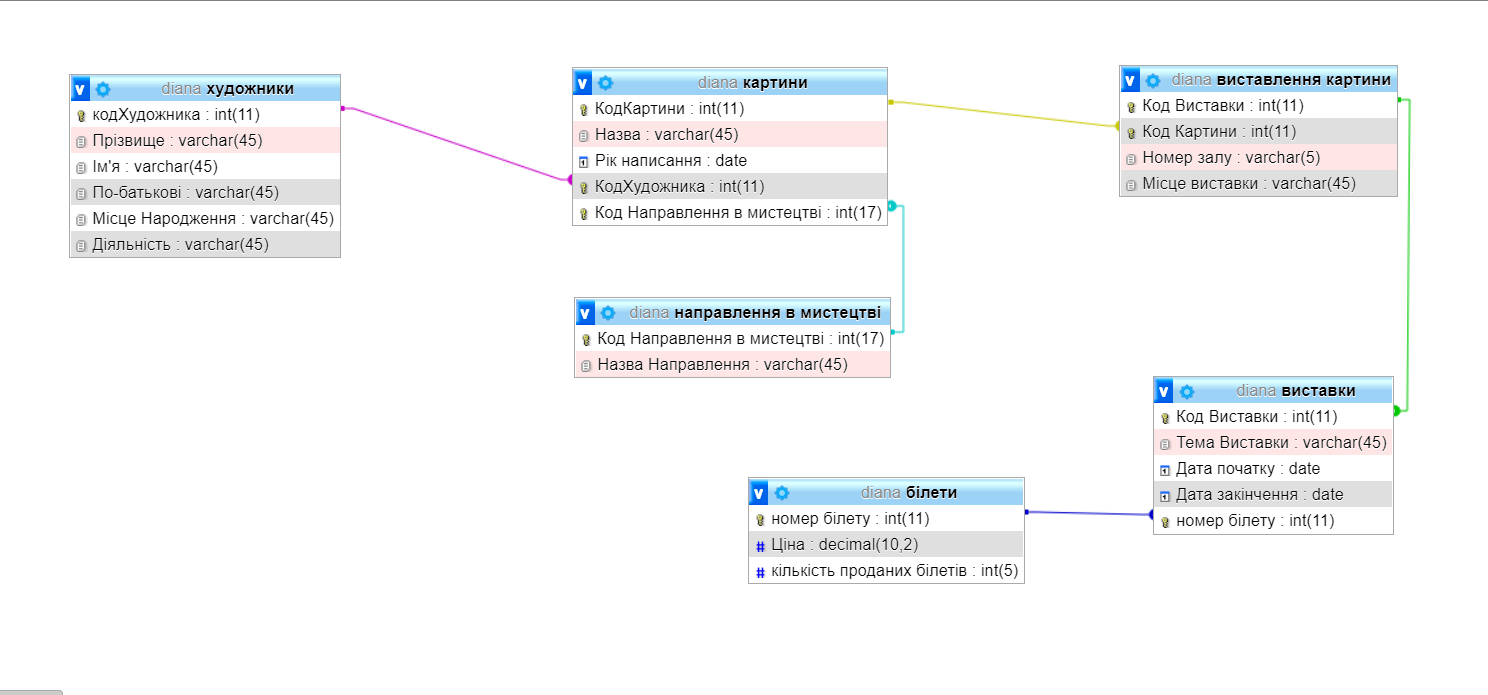


Рис.1. Схема зв’язків між таблицями

# Умови збереження цілісності

Умовами збереження цілісності даних в таблиці **Картини** є збереження даних про картини. Знищення даних можливе тільки при попередньому знищенні даних з таблиць **Художники**, **Виставлення картини** та **Направлення в мистецтві.** (Каскадне знищення не встановлено, встановлено знищення RESTRICT). Якщо змінити значення в таблиці, зміни будуть відображені і в таблицях **Художники, Виставлення картини** та **Направлення в мистецтві.** (Встановлено каскадне оновлення даних, зв’язаних з іншими таблицями). Неможливим буде внесення даних в таблицю **Виставлення картини** без внесення даних попередньо в таблицю **Картини**.

Умовами збереження цілісності даних в таблиці **Білети** є збереження даних про білети. Знищення даних можливе також тільки при попередньому знищенні даних з таблиці **Виставки** і відповідно, оновлення даних будуть відображатися і в цій таблиці (Не встановлено каскадне знищення, але встановлено каскадне оновлення даних). Неможливим буде внесення даних в таблицю **Виставки** без внесення даних спочатку в таблицю **Білети**.

Умовами збереження цілісності даних в таблиці **Виставки** є збереження даних про виставки. Знищення даних можливе при попередньому знищенні даних з таблиць **Виставлення картини** та **Білети**, і відповідно оновлення даних також будуть відображатися в цих таблицях. (Каскадне знищення не встановлено, але встановлено каскадне оновлення даних).

Умовами збереження цілісності даних в таблиці **Виставлення картини** є збереження даних про виставлення картин на виставках. Знищення даних можливе при попередньому знищенні даних з таблиць **Виставки** та **Картини**, і відповідно оновлення даних також будуть відображатися в цих таблицях. (Каскадне знищення не встановлено, але встановлено каскадне оновлення даних). Неможливим буде внесення даних в таблицю **Виставлення картини** без внесення даних спочатку в таблицю **Картини**.

Умовами збереження цілісності даних в таблиці **Направлення в мистецтві** є збереження даних про направлення. Знищення даних можливе при попередньому знищенні даних з таблиці **Картини**, і відповідно оновлення даних також буде відображатися в цій таблиці. (Каскадне знищення не встановлено, але встановлено каскадне оновлення даних).

Умовами збереження цілісності даних в таблиці **Художники** є збереження даних про художників. Знищення даних можливе при попередньому знищенні даних з таблиці **Картини**, і відповідно оновлення даних також буде відображатися в цій таблиці. (Каскадне знищення не встановлено, але встановлено каскадне оновлення даних). Неможливим буде внесення даних в таблицю **Художники** без внесення даних спочатку в таблицю **Картини**.

Зв’язок між таблицями Художники і Картини здійснено по полю КодХудожника, зв’язок між Картини і Направлення в мистецтві ⸺ по полю Код Направлення в мистецтві, між таблицями Картини і Виставлення Картини ⸺ по полю Код Картини, між таблицями Виставлення картини і Виставки ⸺ по полю Код Виставки та зв’язок між таблицями Виставки та Білети здійснено по полю номер білету. Задано умови цілісності даних.

# Інфологічна модель даних



# Форми для введення та корегування даних

Для того, щоб забезпечити введення даних, створюємо необхідні форми введення, при зберіганні яких, дані в таблиці бази будуть створені і збережені. Їх призначення забезпечити створення, запропонованої користувачами веб-сайту, нової інформації про виставки, картини та художників, які хочуть прийняти участь у створення виставки чи просто з метою поширення власних робіт на сайті.

Форма для того, щоб запропонувати виставку (Рис.2) надає користувачеві змогу запропонувати проведення своєї виставки, при цьому користувач може вказати тему виставки, дату проведення та який номер білету буде відповідати для цієї виставки.

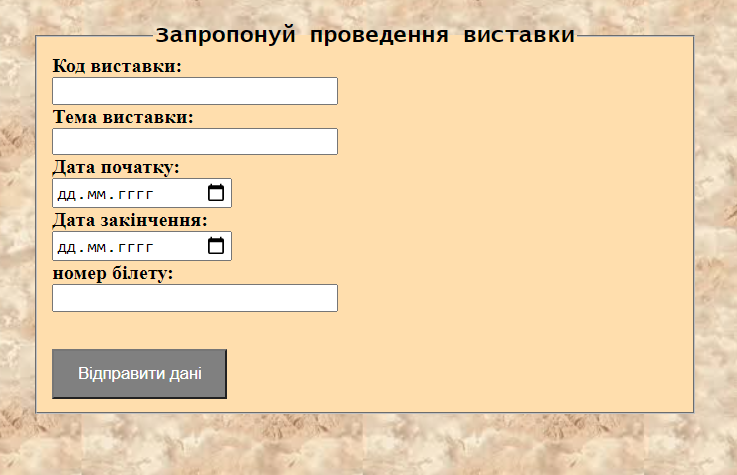


Рис.2. Форма для того, щоб користувач зміг запропонувати своє проведення виставки.

Форма для того, щоб користувач зміг запропонувати свої картини для виставок (Рис.3. ) складається з введення назви картини, дати написання , посилання на форму реєстрації виконавця картини (Рис.4), та введення коду художника (якщо він уже зареєстрований), а також введення коду направлення в мистецтві (потрібний код користувач може вибрати перейшовши по посиланню, де є список всіх направлень).

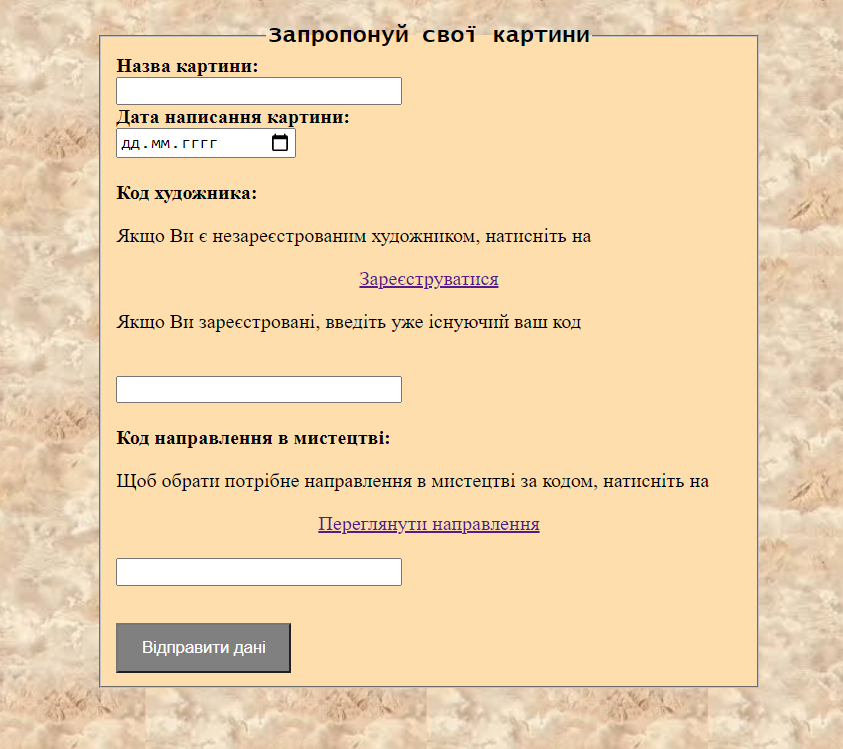


Рис.3. Форма для того, щоб користувач зміг запропонувати свої картини для виставок.

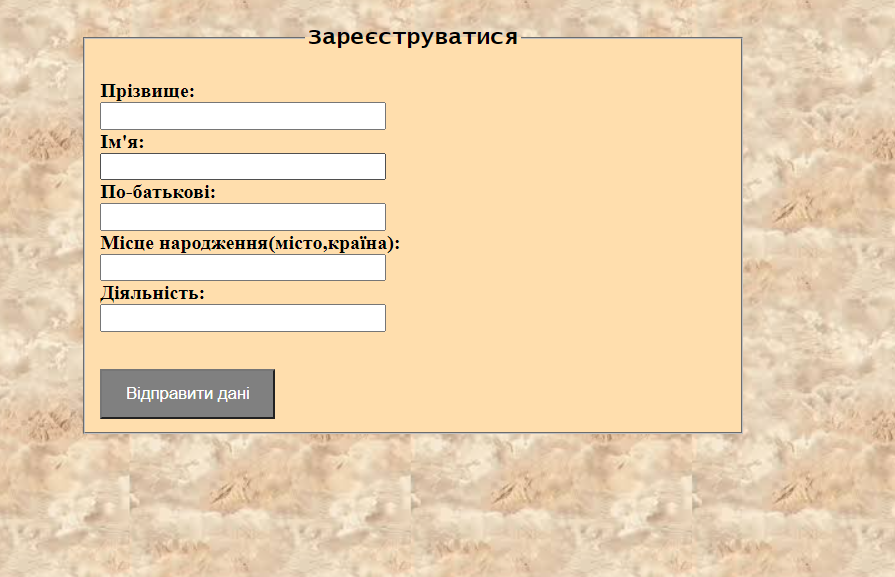


Рис.4. Форма для реєстрації художників.

Для того, щоб редагувати та видаляти дані, виводимо дані у вигляді таблиці та забезпечуємо видалення та редагування за допомогою натиску на відповідні слова «Видалити» та «Редагувати»:

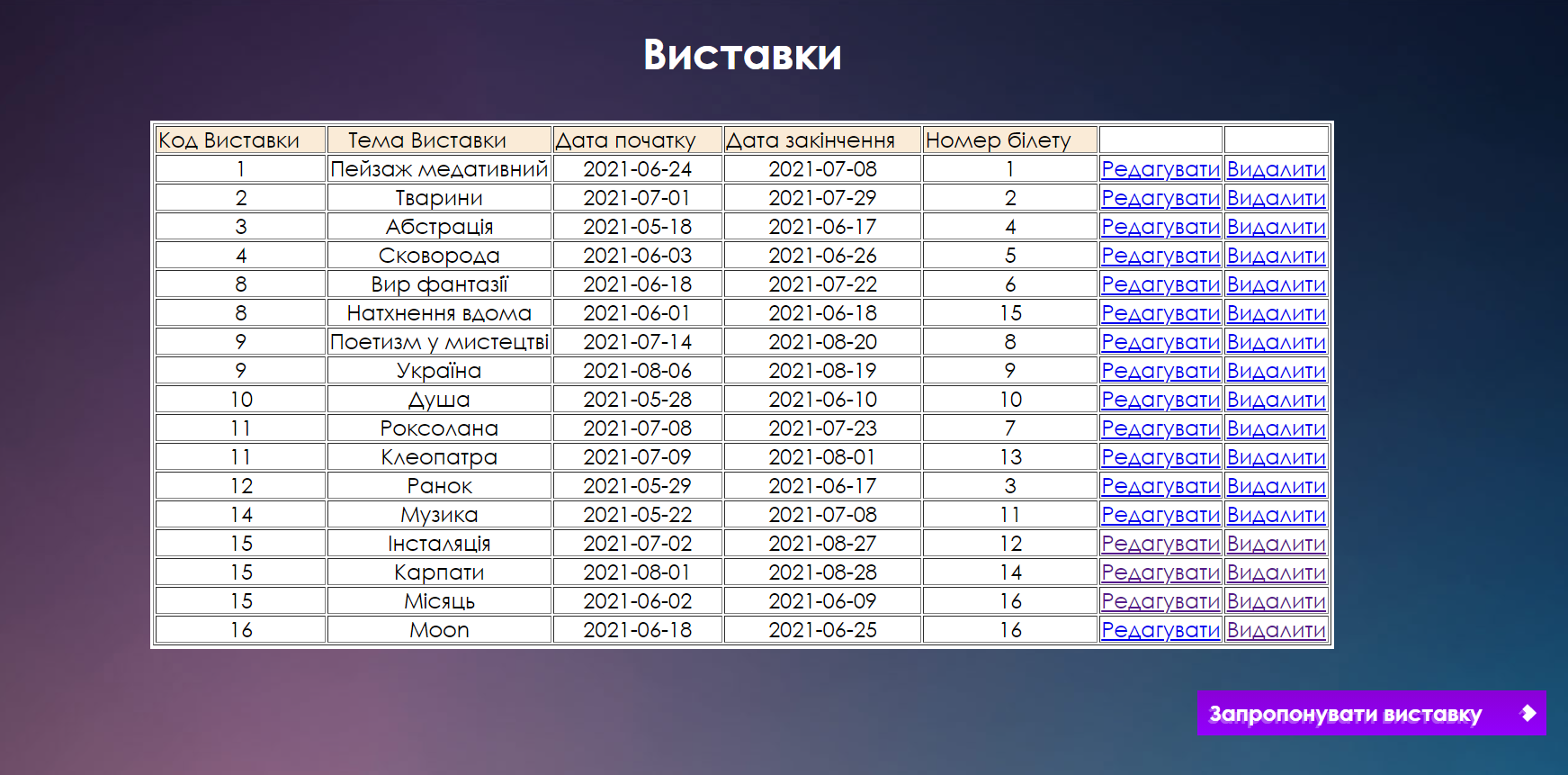


Рис.5. Вивід таблиці «Виставки» та забезпечення редагування та видалення даних.

# Запити до бази даних:

1. Перший запит дозволяє отримати інформацію про виставки :

SELECT `виставки`.`Код Виставки`, `виставки`.`Тема Виставки`,`виставки`.`Дата початку`,`виставки`.`Дата закінчення`,`виставки`.`номер білету` FROM `виставки` WHERE 1

1. Другий запит дозволяє отримати інформацію про тему виставки та білети які продалися на дану виставку:

SELECT `виставки`.`Тема Виставки`,`білети`.`номер білету`,`білети`.`кількість проданих білетів`,`білети`.`Ціна` FROM `виставки`,`білети`

1. Третій запит дозволяє отримати інформацію про художника, його виставки і ціну білету на дану виставку по Прізвищу художника в алфавітному порядку:

SELECT `художники`.`Прізвище`,`виставки`.`Тема Виставки`,`виставки`.`Дата початку`,`виставки`.`Дата закінчення`,`виставки`.`номер білету`,`білети`.`Ціна` FROM `художники`,`виставки`,`білети` ORDER BY `художники`.`Прізвище`

1. Четвертий запит дозволяє вивести суму з усіх проданих білетів:

SELECT SUM(`білети`.`кількість проданих білетів`) FROM `білети`

1. П’ятий запит дозволяє вивести загальну суму отриманих коштів від продажу білетів:

SELECT SUM(`білети`.`кількість проданих білетів`\*`білети`.`Ціна`) As `Загальна сума отриманих коштів від продажу білетів` FROM `білети`

# Тригери

Для того, щоб полегшити роботу з базою даних і задати обмеження можна використовувати тригери.

Тригери представляють спеціальний тип збереженої процедури, яка викликається автоматично при виконанні певної дії над таблицею або поданням, зокрема, при додаванні, зміну або видалення даних, тобто при виконанні команд INSERT, UPDATE, DELETE.

Особливість тригерів полягає в тому, що SQL код , написаний в тілі тригера, буде виконаний після того, як в базі даних відбудеться яка-небудь подія .

Створюємо тригер, який при введені прізвища художника «Косач» виводить повідомлення про помилку :

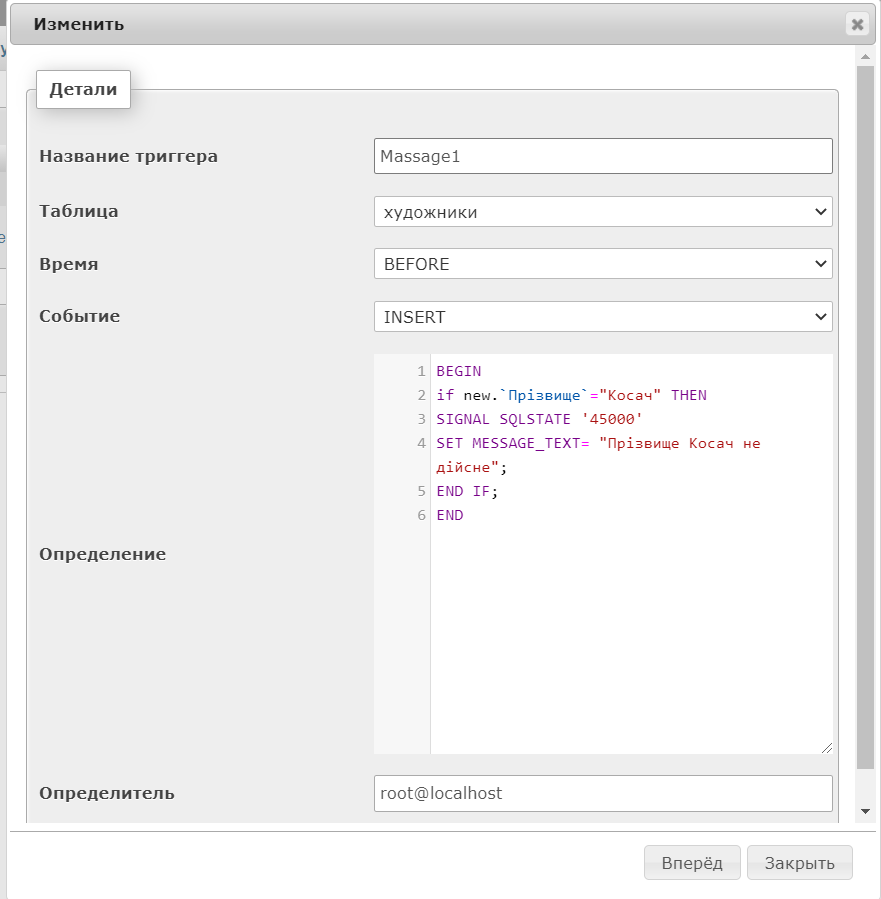


Рис.6. Створення тригера.

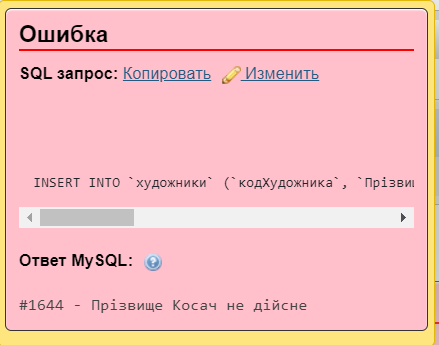


Рис.7. Виведення повідомлення про помилку.

# Процедури

Процедури є також засобом полегшення роботи з базами даних, оскільки завдяки їм можна вивести ту інформацію , яку хочеш не прописуючи окремі команди.

Створюємо процедуру у якій виводимо назву картини , рік написання якої = «2008-02-05»:

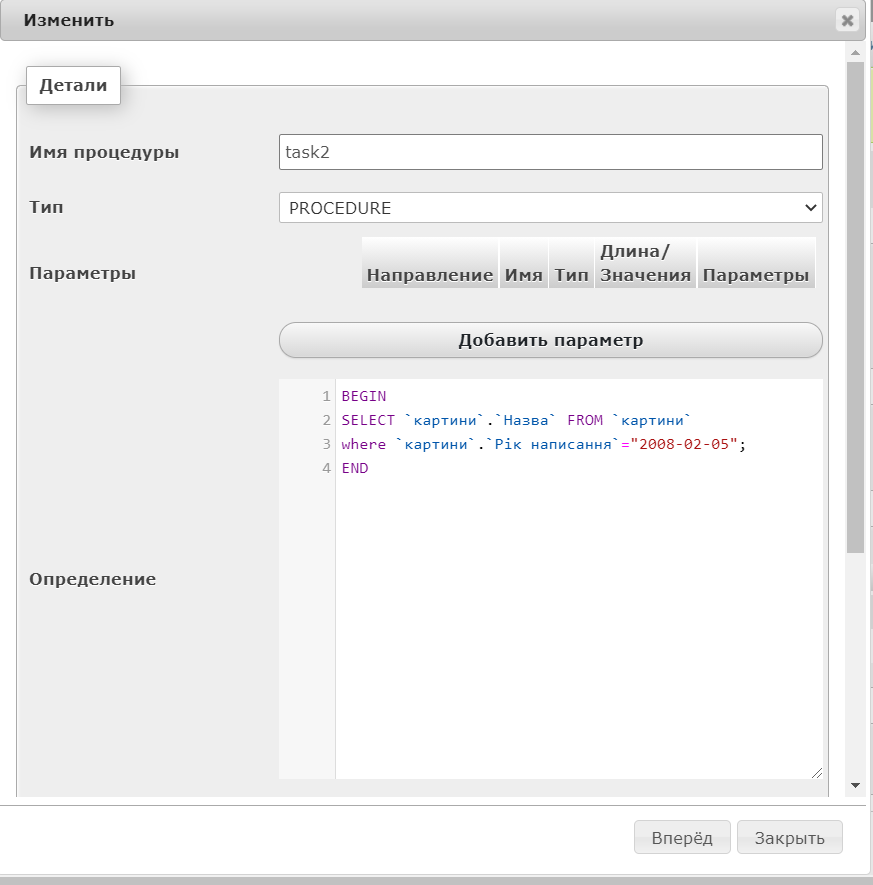


Рис.8. Створення процедури

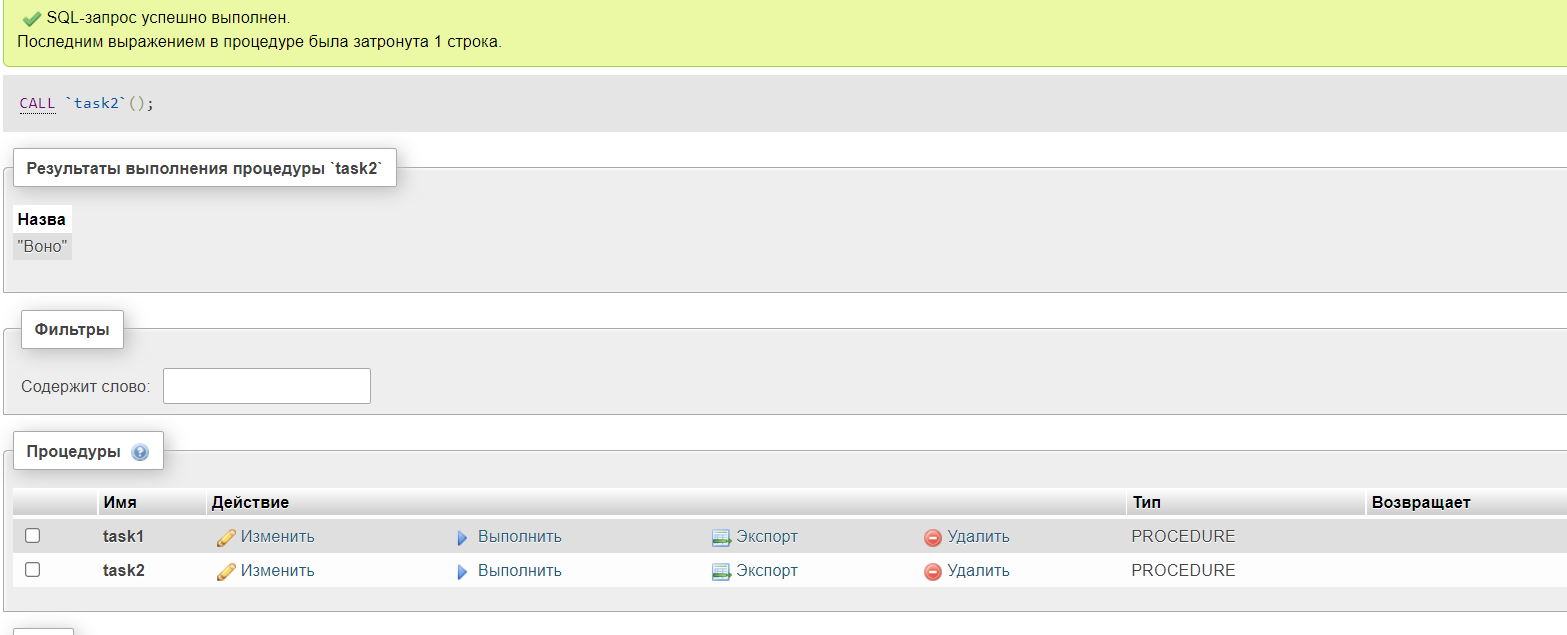
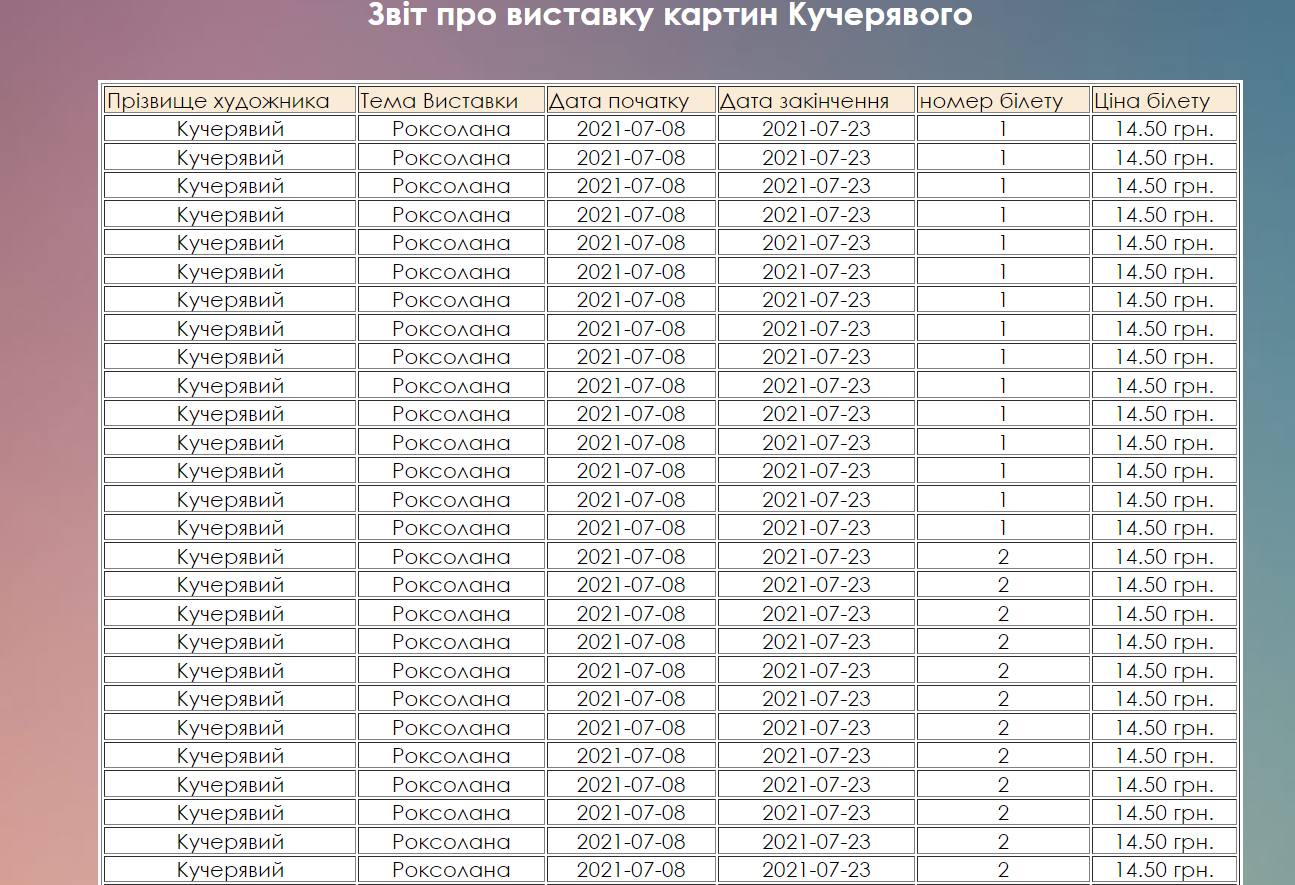


Рис.9. Виконання процедури.

# Звіти

1. Звіт про виставку картин художника Кучерявого. Даний звіт виводить інформацію про виставку картин даного художника і інформацію про номер білету і ціну.



1. Звіт про виставки за серпень. Даний звіт виводить всі виставки за місяць серпень, кількість проданих білетів і ціну, а також підраховує загальну суму проданих всіх білетів за весь час та загальну суму отриманих коштів від проданих білетів за весь час.



# Висновки

Під час виконання даної курсової роботи було створено базу даних картинної галереї. Для її створення необхідно було завантажити програму XAMPP, щоб під’єднатися до сервера і підключитися до веб-додатку phpMyAdmin і вже потім була можливість працювати з базою даних. Дана база даних містить в собі 6 таблиць, які поєднанні між собою, тобто мають між собою зв’язки. Для створення інтерфейсу для бази даних картинної галереї було використано мову розмітки HTML, CSS, PHP та JavaScript. Також під час виконання роботи базу даних було нормалізовано згідно теорії нормалізації Кодда, побудовано ER-діаграму та інфологічну модель бази даних.

Також при виконанні даної роботи ми вивчили як створювати базу даних і підключати її до додатків або до веб-сайтів, редагувати та видаляти дані з неї, вивчити теорію проектування баз даних, які існують види баз даних, моделі, теорію нормалізації Кодда і Фагіна, а також дізналися, що таке структурована мова запитів SQL.

Набуті нами навички при створенні даної курсової роботи допоможуть нам у майбутньому працювати з базами даних, використовувати їх у подальшій роботі, а також легко працювати з бізнесовою інформацією.

# Список використаної літератури

1. Що таке бази даних - види і приклади їх використання.

URL: <https://ktonanovenkogo.ru/voprosy-i-otvety/baza-dannyh-chto-ehto-takoe.html>

1. Що таке база даних.

URL: <https://hostiq.ua/wiki/database/>

1. Що таке бази даних (БД) і навіщо вони потрібні.

URL: <https://htmlacademy.ru/tutorial/php/databases>

1. Що таке база даних.

URL: <https://habr.com/ru/post/555760/>

1. Методичні вказівки до виконання курсової роботи з навчальної дисципліни «Бази даних» для здобувачів вищої освіти першого (бакалаврського) рівня за освітньо-професійною програмою «Комп’ютерні науки» спеціальності 122 «Комп’ютерні науки» денної, заочної та дистанційної форм навчання [Електронне видання] / Бачишина Л. Д., Харів Н. О. – Рівне : НУВГП, 2020. – 24 ст.
2. Що таке реляційна база даних?

URL: <https://aws.amazon.com/ru/relational-database/>