

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**КРИВОРІЗЬКИЙ ДЕРЖАВНИЙ ПЕДАГОГІЧНИЙ УНІВЕРСИТЕТ**  
Фізико-математичний факультет  
Кафедра інформатики та прикладної математики

«Допущено до захисту»

Завідувач кафедри

\_\_\_\_\_ Моїсеєнко Н.В.

« \_\_\_\_ » \_\_\_\_\_ 2024 р.

Реєстраційний № \_\_\_\_\_

« \_\_\_\_ » \_\_\_\_\_ 2024 р.

**Використання вільного програмного забезпечення при вивченні баз  
даних у профільному навчанні інформатики**

Кваліфікаційна робота

ступінь вищої освіти «магістр»

спеціальності

014 Середня освіта (Інформатика)

**Прекрасного**

**Артема**

**Володимировича**

Керівник:

кандидат фізико-математичних наук,

доцент

**Мерзликін Павло Володимирович**

Оцінка:

Національна шкала \_\_\_\_\_

Шкала ECTS \_\_\_\_\_ кількість балів \_\_\_\_\_

Члени комісії

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

## ЗАПЕВНЕННЯ

Я, Прекрасний Артем Володимирович, розумію і підтримую політику Криворізького державного педагогічного університету з академічної доброчесності. Запевняю, що ця кваліфікаційна робота виконана самостійно, не містить академічного плагіату, фабрикації, фальсифікації. Я не надавав і не одержував недозволену допомогу під час підготовки цієї роботи. Використання ідей, результатів і текстів інших авторів мають покликання на відповідне джерело. Із чинним Положенням про запобігання та виявлення академічного плагіату в роботах здобувачів вищої освіти Криворізького державного педагогічного університету ознайомлений. Чітко усвідомлюю, що в разі виявлення у кваліфікаційній роботі порушення академічної доброчесності робота не допускається до захисту або оцінюється незадовільно.



(підпис)

## ЗМІСТ

<b>ВСТУП.....</b>	<b>4</b>
<b>РОЗДІЛ 1. ПОПЕРЕДНІЙ ДОСВІД І СУЧАСНИЙ СТАН ВИКОРИСТАННЯ ВІЛЬНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ В ПРОФІЛЬНОМУ НАВЧАННІ ІНФОРМАТИКИ ПІД ЧАС ВИВЧЕННЯ БАЗ ДАНИХ.....</b>	<b>6</b>
1.2. Огляд наявного вільного програмного забезпечення для роботи з базами даних.....	8
1.3. Аналіз попередніх досліджень щодо використання вільного ПЗ при вивченні баз даних .....	13
1.4. Постановка задачі.....	16
Висновки до розділу 1 .....	18
<b>РОЗДІЛ 2. РОЗРОБКА МЕТОДИЧНИХ РЕКОМЕНДАЦІЙ ВИКОРИСТАННЯ ВІЛЬНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ В ПРОФІЛЬНОМУ НАВЧАННІ ІНФОРМАТИКИ ПІД ЧАС ВИВЧЕННЯ БАЗ ДАНИХ.....</b>	<b>20</b>
2.1. Обґрунтування вибору програмного забезпечення .....	20
2.2. Методика інтеграції вільного ПЗ у навчальний процес .....	22
2.3. Адаптація навчальних матеріалів .....	26
Висновки до розділу 2 .....	28
<b>РОЗДІЛ 3. ЕКСПЕРИМЕНТАЛЬНА ПЕРЕВІРКА ДОЦІЛЬНОСТІ ВИКОРИСТАННЯ ВІЛЬНОГО ПЗ ПІД ЧАС ВИВЧЕННЯ БАЗ ДАНИХ .....</b>	<b>30</b>
3.1. Планування експерименту .....	30
3.2. Проведення уроків з використанням вільного ПЗ .....	33
3.3. Аналіз результатів експерименту .....	36
Висновки до розділу 3 .....	39
<b>ВИСНОВКИ.....</b>	<b>41</b>
<b>Список використаних джерел .....</b>	<b>44</b>
<b>ДОДАТКИ .....</b>	<b>46</b>
Додаток А .....	46
Додаток Б.....	48
Додаток В.....	49
Додаток Г .....	51
Додаток Д .....	52
Додаток Е.....	53
Додаток Є.....	54
Додаток Ж .....	56

## ВСТУП

**Актуальність роботи** полягає в тому, що сучасна освіта вимагає інтеграції новітніх технологій для підвищення ефективності навчального процесу. Використання вільного програмного забезпечення (ВПЗ) є актуальним у зв'язку з кількома чинниками. ВПЗ дозволяє школам та учням заощаджувати кошти, адже більшість таких програм є безкоштовними. Це особливо важливо для освітніх закладів з обмеженим бюджетом, які не можуть собі дозволити дорогі ліцензії комерційного програмного забезпечення. Програмне забезпечення з відкритим кодом, як-от MySQL, PostgreSQL або SQLite, забезпечує всі необхідні функції для вивчення баз даних і легко адаптується до освітніх потреб. Воно також дозволяє глибше зрозуміти принципи роботи баз даних, оскільки учні можуть взаємодіяти з його кодом. На сучасному ринку праці використання вільного програмного забезпечення є поширеним, особливо в сфері інформаційних технологій. Залучення учнів до роботи з такими інструментами сприяє їхній конкурентоспроможності та готовності до професійної діяльності. Робота з ВПЗ сприяє формуванню в учнів навичок критичного мислення, пошуку інформації, налаштування та адаптації програмного забезпечення [1] Це важливі аспекти цифрової грамотності, яка є необхідною для кожного сучасного спеціаліста. ВПЗ відкриває можливості для творчого підходу до вирішення завдань, дозволяючи учням і вчителям впроваджувати нові ідеї, створювати власні програмні рішення та вдосконалювати існуючі інструменти. З огляду на ці фактори, використання вільного програмного забезпечення в профільному навчанні інформатики, зокрема під час вивчення баз даних, є не лише актуальним, а й стратегічно важливим для розвитку сучасної освіти.

**Об'єкт дослідження:** вивчення баз даних у профільному навчанні інформатики.

**Предмет дослідження:** використання відкритого програмного забезпечення для вивчення баз даних у профільному навчанні інформатики.

**Мета:** розробити методичні рекомендації використання вільного програмного забезпечення у профільному навчанні інформатики для вивчення баз даних.

Для досягнення мети слід розв'язати такі задачі:

- проаналізувати переваги й недоліки використання вільного програмного забезпечення в шкільному курсі інформатики;
- порівняти наявне вільне програмне забезпечення, яке може бути використане для вивчення баз даних;
- здійснити аналіз попередніх досліджень щодо використання вільного програмне забезпечення при вивченні баз даних;
- сформулювати вимоги до методичних рекомендацій;
- обрати вільне програмне забезпечення, використання якого видається найбільш доцільним при вивченні баз даних;
- розробити методику інтеграції вільного програмне забезпечення у навчальний процес;
- здійснити адаптацію навчальних матеріалів;
- спланувати експеримент для дослідження ефективності розроблених рекомендацій;
- провести уроки з використанням вільного програмне забезпечення;
- проаналізувати результати експерименту й зробити висновки.

**Практичне значення** роботи полягає в тому, що розроблені методичні рекомендації можуть бути впроваджені в навчальний процес під час вивчення баз даних.

**Структура роботи.** Кваліфікаційна робота складається із вступу, трьох розділів, висновку, списку використаних джерел та додатків

# РОЗДІЛ 1. ПОПЕРЕДНІЙ ДОСВІД І СУЧАСНИЙ СТАН ВИКОРИСТАННЯ ВІЛЬНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ В ПРОФІЛЬНОМУ НАВЧАННІ ІНФОРМАТИКИ ПІД ЧАС ВИВЧЕННЯ БАЗ ДАНИХ

## 1.1. Переваги й недоліки використання вільного програмного забезпечення в шкільному курсі інформатики.

Для початку треба з'ясувати що таке взагалі вільне програмне забезпечення, та чим воно відрізняється від інших форм поширення програмного забезпечення, та чому в закладах загальної середньої освіти доцільно використовувати саме цю форму поширення. «Вільне програмне забезпечення» означає програмне забезпечення, яке поважає свободу та спільноту користувачів [6]. Приблизно це означає, що користувачі мають свободу запускати, копіювати, поширювати, вивчати, змінювати та покращувати програмне забезпечення. Отже можна підсумувати, що вільне програмне забезпечення – це програмне забезпечення, використовуючи яке користувач має певні «свободи»: запускати, вивчати, розповсюджувати, поліпшувати. Якщо розробник не надав хоча б однієї з вище перерахованих свобод, то таке програмне забезпечення не можна вважати вільним. Варто зазначити, що безкоштовність вільного програмного забезпечення не згадана не випадково, оскільки вільне програмне забезпечення не обов'язково має бути безкоштовним.

### **Переваги:**

#### 1. Економічна доступність:

Використання вільного програмного забезпечення дозволяє суттєво знизити витрати на придбання ліцензійного програмного забезпечення, що особливо важливо для шкіл з обмеженим фінансуванням. Наприклад, популярні інструменти, як LibreOffice [8] чи GIMP [5], є безкоштовними й замінюють дорогі комерційні аналоги.

#### 2. Академічна свобода:

Школярі можуть вивчати внутрішню структуру програм та навіть модифікувати їх для навчання, що сприяє розвитку критичного мислення і творчого підходу до програмування.

### 3. Ліцензійна безпека:

Вільне програмне забезпечення не порушує законодавства про інтелектуальну власність, на відміну від неліцензійних програм, які часто використовуються через високу вартість комерційного програмного забезпечення.

### 4. Гнучкість і масштабованість:

Багато вільних програм мають версії для різних операційних систем, що спрощує їхнє впровадження навіть на застарілих комп'ютерах.

### **Недоліки:**

#### 1. Нестабільна технічна підтримка:

У багатьох випадках підтримка вільного програмне забезпечення залежить від волонтерів. Якщо програма перестає розроблятися, її оновлення і виправлення можуть припинитися.

#### 2. Складність освоєння:

Деякі інструменти вільного програмного забезпечення мають менш інтуїтивний інтерфейс порівняно з комерційними продуктами, що може створити труднощі для вчителів і учнів на початкових етапах.

#### 3. Інтеграція з іншими системами:

Вільне програмне забезпечення іноді має обмежену сумісність із популярними комерційними продуктами. Наприклад, формати файлів можуть зчитуватися некоректно або втрачати частину функціоналу.

#### 4. Залежність від інтернет-з'єднання:

Хмарні рішення на базі вільного програмного забезпечення вимагають стабільного доступу до інтернету, що може бути проблемою для шкіл у віддалених регіонах.

Використання вільного програмного забезпечення показує, що його впровадження в освіті може бути дуже ефективним за умови належної

підготовки вчителів та адаптації навчальних програм. Водночас потрібна комплексна підтримка з боку адміністрації шкіл і держави, щоб уникнути технічних і організаційних проблем.

## **1.2. Огляд наявного вільного програмного забезпечення для роботи з базами даних.**

Робота з базами даних є однією з ключових складових шкільного курсу інформатики у профільному навчанні. Використання вільного програмного забезпечення у цьому контексті стає все більш популярним через економічні, методологічні та педагогічні переваги. Розглянемо основні інструменти вільного програмного забезпечення, які можуть бути застосовані під час навчання базам даних.

MySQL [10] (рис.1.1) — це одна з найбільш відомих реляційних систем керування базами даних. Вона використовується як у комерційних, так і в освітніх проєктах завдяки її безкоштовності, потужності та простоті у використанні.



Рис. 1.1. MySQL

Її перевагами можна вважати: підтримку основних функцій реляційних баз даних, включаючи транзакції, SQL-запити, реплікацію та резервне копіювання, простоту інсталяції й налаштування, що дозволяє швидко інтегрувати її у навчальний процес.

Ця програма відкриває багато освітніх можливостей: MySQL дозволяє учням опанувати основи SQL-запитів і створення баз даних. Особливо корисним є поєднання MySQL з популярними мовами програмування, як-от



Python чи PHP, що допомагає побудувати практичне розуміння інтеграції баз даних у програмні продукти.

Наступним вільним забезпечення для роботи з базами даних розглянемо PostgreSQL.

PostgreSQL [13] (рис. 1.2) — це високопродуктивна система керування базами даних із підтримкою складних типів даних і розширених функцій. Вона є популярною в освітніх і комерційних колах завдяки своїй надійності та гнучкості.



Рис. 1.2. PostgreSQL

Особливістю цієї системи керування базами даних наступні:

- підтримка складних транзакцій, процедур і тригерів;
- висока продуктивність під час роботи з великими обсягами даних.

PostgreSQL найбільше підходить для старшокласників та студентів, які вивчають складніші аспекти роботи з базами даних, як-от оптимізацію запитів чи створення багаторівневих структур даних. Її можливості дозволяють пояснити основи об'єктно-реляційної моделі, що є важливим елементом сучасної інформатики.

Далі розглянемо програму SQLite

SQLite [16] (Рис. 1.3) — це легка система керування базами даних, яка не потребує окремого серверного середовища. Вона ідеально підходить для навчання базам даних на початковому рівні.



Рис. 1.3. SQLite

Її перевагами можна вважати:

- компактність і мінімальні вимоги до ресурсів системи;
- простота в роботі: бази даних SQLite представлені у вигляді одного файлу, що зручно для невеликих проєктів які часто роблять у школах.

SQLite широко використовується для навчання моделювання даних, створення таблиць та взаємодії з ними через SQL-запити. Вона також дозволяє вивчити концепції роботи локальних баз даних, які часто застосовуються у мобільних додатках.

MariaDB [9] (Рис. 1.4) — це форк MySQL, створений після її придбання компанією Oracle. Ця система стала популярною завдяки покращеній продуктивності, безпеці та стабільності.



Рис. 1.4. MariaDB

Після оновлення ця база даних має нові можливості підтримки нових форматів зберігання даних, зокрема JSON. Також можна виділити вищу швидкість обробки складних SQL-запитів у порівнянні з MySQL.

MariaDB забезпечує сумісність із MySQL, що дозволяє використовувати наявні навчальні матеріали, але дає змогу учням ознайомитися з передовими функціями сучасних систем керування базами даних.

Порівняльна таблиця функціональних можливостей популярного вільного програмного забезпечення для баз даних.

Таблиця 1.1.

**Порівняльна характеристика популярних систем керування базами даних (СКБД)**

<b>Критерій</b>	<b>MySQL</b>	<b>PostgreSQL</b>	<b>SQLite</b>	<b>MariaDB</b>
<b>Ліцензія</b>	GPL/Commercial	PostgreSQL License	Public Domain	GPL/Commercial
<b>Сфера використання</b>	Веб-додатки, корпоративні системи	Складні аналітичні системи, великі проєкти	Локальні додатки, мобільні системи	Веб-додатки, аналітичні системи, корпоративні проєкти
<b>Продуктивність</b>	Висока для простих запитів	Висока для складних запитів і транзакцій	Помірна для невеликих баз даних	Висока для простих і середньої складності запитів
<b>Підтримка ACID</b>	Часткова (InnoDB)	Повна підтримка	Часткова	Повна (InnoDB, Aria)
<b>Розмір бази даних</b>	Без обмежень	Без обмежень	До 2 ТБ (залежить від файлової системи)	Без обмежень
<b>Інструменти резервного копіювання</b>	mysqldump, MySQL Workbench	pg_dump, pgAdmin	Ручне копіювання файлів	mariabackup, mysqldump
<b>Типи даних</b>	Обмежений вибір	Розширений вибір (JSON, XML, User-Defined)	Основні типи даних	Розширений вибір (JSON, Virtual Columns)
<b>Масштабованість</b>	Добра для реплікації	Відмінна (кластеризація, реплікація)	Не передбачена	Відмінна (Galera Cluster)

Критерій	MySQL	PostgreSQL	SQLite	MariaDB
Підтримка тригерів і процедур	Є	Є	Обмежена	Є
Простота використання	Інтуїтивна, багато навчальних матеріалів	Вимагає більше технічних знань	Дуже проста, підходить для початківців	Інтуїтивна, схожа на MySQL
Мультиплатформеність	Windows, macOS, Linux	Windows, macOS, Linux	Windows, macOS, Linux	Windows, macOS, Linux
Гнучкість налаштувань	Обмежена	Дуже гнучка	Мінімальна	Більш гнучка, ніж MySQL
Рівень безпеки	Середній	Високий	Низький	Високий
Документація та підтримка	Широка спільнота, навчальні матеріали	Широка спільнота, офіційна документація	Мінімальна спільнота, проста документація	Широка спільнота, хороша документація

Проаналізувавши таблицю, можна зробити деякі висновки: MySQL — найкращий вибір для веб-додатків та невеликих проєктів завдяки простоті використання та широкій підтримці, PostgreSQL ідеально підходить для складних проєктів, де потрібна висока надійність, масштабованість і підтримка складних запитів, SQLite — оптимальний для локальних і невеликих додатків, де не потрібні розширені можливості серверної бази даних, MariaDB можна використовувати як альтернативу MySQL, особливо якщо потрібні покращені можливості роботи з великими обсягами даних і надійні механізми реплікації.

В останні роки значного поширення набули NoSQL бази даних, такі як MongoDB [11]. Вони зберігають дані у форматі документів (наприклад, JSON) і підходять для обробки неструктурованих або напівструктурованих даних. Навчальне значення MongoDB полягає в тому, що учні можуть ознайомитися з новими підходами до обробки даних, такими як ключ-значення чи документно-орієнтовані бази. Це дозволяє зрозуміти, як сучасні компанії працюють із даними у великих масштабах.

Вільне програмне забезпечення надає широкий вибір інструментів для навчання роботі з базами даних. Завдяки таким рішенням, як MySQL,

PostgreSQL, SQLite та MongoDB, учні можуть ознайомитися з різними підходами до організації, обробки та аналізу даних. Це сприяє їх підготовці до використання сучасних технологій у реальних проєктах, а також розвиває критичне мислення та навички програмування.

### **1.3. Аналіз попередніх досліджень щодо використання вільного ПЗ при вивченні баз даних**

Аналіз попередніх досліджень [4] щодо використання вільного програмного забезпечення для вивчення баз даних у школах показує, що це є перспективним напрямком, який одночасно забезпечує доступність навчання та стимулює розвиток ключових компетенцій учнів.

Однією з ключових переваг використання вільного програмного забезпечення (ПЗ) у шкільній освіті є його економічна ефективність і доступність для широкого впровадження.

Вільне програмне забезпечення (таке як MySQL, PostgreSQL, SQLite) є зазвичай безкоштовним і доступним для завантаження, встановлення та використання. Це дозволяє навчальним закладам уникати витрат на ліцензії, що значно знижує фінансове навантаження на бюджет шкіл. Крім того, відкритий код дозволяє адаптувати програми під конкретні потреби шкільного курсу, що важливо для реалізації індивідуальних методик викладання. Вільне програмне забезпечення особливо важливе для шкіл у регіонах з обмеженими фінансовими ресурсами [4]. Наприклад, у країнах, що розвиваються, воно дозволяє забезпечити доступ до сучасних технологій, які є необхідними для розвитку навичок цифрової грамотності у школярів. Дослідження [4] вказують, що безкоштовні програмні платформи відкривають можливості для навчання в умовах обмеженого фінансування, сприяючи більш рівному доступу до освіти. Вільне програмне забезпечення зазвичай працює на широкому спектрі операційних систем, включаючи Windows, macOS та Linux. Це забезпечує його сумісність навіть зі старими комп'ютерами, які часто використовуються у школах. Така технологічна

гнучкість дає змогу використовувати наявне обладнання без додаткових інвестицій у його модернізацію. Масове впровадження можливе завдяки простоті розгортання вільного ПЗ. Більшість таких програм мають детальну документацію, доступну онлайн, що полегшує налаштування і підтримку навіть для тих вчителів, які не є спеціалістами в ІТ. Крім того, велика кількість спільнот користувачів забезпечує доступ до консультацій і підтримки, що сприяє вирішенню технічних проблем у навчальних закладах. Завдяки своїй безкоштовності вільне ПЗ дозволяє школам впроваджувати сучасні технології в навчальний процес навіть у випадках, коли фінансування є мінімальним. Це знижує бар'єри для інтеграції таких інструментів, як управління базами даних, програмування та інші ІТ-навички, в навчальну програму.

Використання вільного програмного забезпечення у шкільному навчанні баз даних забезпечує учням можливість працювати з реальними інструментами, які широко застосовуються у сфері ІТ. Це сприяє більш глибокому розумінню предмета, а також готує їх до реальних викликів у подальшій освіті чи професійній діяльності. Вільне програмне забезпечення, дозволяє школярам навчатися за допомогою платформ, які використовуються у професійній діяльності. Наприклад, MySQL широко застосовується у веб-розробці, а PostgreSQL — у складних проєктах, де потрібна висока продуктивність. Це дає змогу учням засвоювати навички, які мають практичну цінність на ринку праці. Навчання з використанням такого програмного забезпечення часто орієнтоване на створення проєктів, які мають практичне застосування [4]. Наприклад, учні можуть розробити базу даних для шкільної бібліотеки, створити електронний журнал успішності або базу даних для управління шкільними заходами. Цей підхід допомагає закріпити теоретичні знання та навчитися застосовувати їх у реальних ситуаціях. Вільне програмне забезпечення можна легко встановити на будь-який комп'ютер, навіть з мінімальними характеристиками. Це дозволяє учням працювати як на шкільних комп'ютерах, так і вдома, що

значно розширює можливості для самостійного навчання. Крім того, багато таких програм, наприклад SQLite, є портативними, що дозволяє використовувати їх без інсталяції.

Вивчення баз даних за допомогою вільного програмного забезпечення сприяє розвитку цифрових компетентностей, таких як програмування, обробка інформації, системний аналіз та робота в команді [14]. Уміння працювати з SQL, моделювати структури даних та писати запити готують учнів до подальшого навчання або професійної діяльності. У дослідженнях [4] наголошується, що практичні навички, отримані в школі, полегшують адаптацію до роботи в сфері ІТ. Попри очевидні переваги, використання вільного програмного забезпечення у шкільній програмі стикається з певними труднощами [14]. Однією з основних є недостатній рівень підготовки вчителів, які часто не мають досвіду роботи з такими програмами. Крім того, існує потреба в адаптованих навчальних матеріалах, які б пояснювали складні теми у простій формі. Деякі дослідження [7] також відзначають, що учні можуть втрачати інтерес без інтерактивних елементів або наочного результату своєї роботи.

У багатьох країнах успішне впровадження вільного програмного забезпечення у школах показало позитивні результати, особливо під час викладання баз даних. **Велика Британія** активно впроваджує ВПЗ у школах [12]. Проекти, що використовують відкриті платформи, такі як MySQL або PostgreSQL, сприяють економії коштів на ліцензіях та навчанні. Ці програми забезпечують учнів доступом до інструментів, схожих на ті, що використовуються в реальній індустрії, що підвищує їхню практичну компетентність. Наприклад, у школах, де перехід до вільного програмного забезпечення був успішно здійснений, вдалося значно скоротити витрати та покращити навчальний процес завдяки більшій гнучкості таких програм [12]. **Німеччина** зосереджується на використанні вільного програмного забезпечення для інформатики, щоб учні знайомилися з сучасними технологіями. У країні реалізовані програми, де учні вивчають управління

базами даних за допомогою відкритих платформ. Такі ініціативи підвищують їхню зацікавленість у технічних професіях і полегшують перехід на професійний ринок праці. **Бразилія** також відома підтримкою відкритого програмного забезпечення у школах. Завдяки програмі "ProInfo" [15] уряд надає учням доступ до освітніх інструментів на базі Linux та інших відкритих систем. Ці ініціативи знижують вартість впровадження технологій у школах, а також популяризують ідеї вільного програмного забезпечення серед молоді. Демонструють переваги використання вільного програмного забезпечення у школах: доступність, економічну ефективність, практичну орієнтованість і можливість інтеграції сучасних освітніх методів. Успішні кейси підтверджують, що вільне програмне забезпечення дозволяє зробити навчання більш ефективним, інклюзивним та адаптованим до вимог сучасного суспільства.

Використання вільного програмного забезпечення для вивчення баз даних у школах є перспективним напрямом, який відкриває нові можливості для цифрової освіти, розвиває ключові компетенції учнів і дозволяє економити ресурси навчальних закладів. Однак для його ефективного впровадження необхідно вирішити ряд питань, які ми розглянемо далі.

#### 1.4. Постановка задачі

Аналізуючи переваги та недоліки використання вільного програмного забезпечення, особливості популярного ПЗ для роботи з базами даних, а також результати попередніх досліджень, можна зробити висновок, що інтеграція вільного програмного забезпечення у навчальний процес шкільного курсу інформатики є перспективним і необхідним напрямом. Проте для досягнення поставленої мети слід врахувати деякі ключові проблеми, виявлені під час аналізу.

Недостатня адаптація наявних програмних рішень до потреб шкільного навчання. Більшість програм для роботи з базами даних, таких як MySQL, PostgreSQL або SQLite, були створені для професійного використання. Їх



інтерфейс і функціонал можуть бути складними для розуміння учнями без відповідної адаптації навчальних матеріалів і методів викладання. Наступна проблема полягає в тому, що багато вчителів не мають достатніх знань про використання вільного програмного забезпечення у навчальному процесі. Це ускладнює впровадження таких програм у шкільний курс. І ще одна проблема з доступністю навчальних ресурсів. Вільне програмне забезпечення зазвичай супроводжується документацією для професійних користувачів, яка є малозрозумілою для учнів середньої школи.

Враховуючи ці проблеми, розробка власного курсу з використанням MySQL має відповідати таким вимогам:

- **Адаптація інтерфейсу та матеріалів до рівня учнів.**
- Використовувати зрозумілі для учнів інструменти, такі як MySQL Workbench, із графічним інтерфейсом.
- Розробити навчальні матеріали, які пояснюють базові концепції реляційних баз даних простими словами.
- **Розробка покрокової методики навчання.**
- Передбачити поступове ускладнення завдань, починаючи з базових операцій (створення таблиць, виконання простих запитів) і переходячи до складніших тем (транзакції, зв'язки між таблицями).
- **Створення навчальних ресурсів для вчителів.**
- Розробити методичні рекомендації для вчителів із покроковими інструкціями щодо інтеграції MySQL у навчальний процес.
- Передбачити підвищення кваліфікації вчителів через семінари або навчальні тренінги.
- **Фокус на практичні завдання.**
- Розробити завдання, які демонструють реальні сценарії використання баз даних (наприклад, створення бази для управління шкільною бібліотекою або електронного журналу оцінок).
- **Інтеграція міждисциплінарних завдань.**

- Передбачити завдання, які пов'язують бази даних із іншими навчальними дисциплінами, такими як математика, фізика чи хімія (наприклад, створення бази для зберігання результатів експериментів).
- **Використання доступних технічних ресурсів.**
- Забезпечити сумісність навчальних матеріалів і програмного забезпечення з технічними можливостями шкільного обладнання.
- **Розробка проєктного підходу.**
- Включити у курс підсумковий проєкт, під час якого учні створюють власну базу даних для вирішення конкретної задачі.
- **Оцінка ефективності курсу.**
- Розробити методику оцінювання знань учнів, яка включає тестові завдання, практичні роботи та підсумковий проєкт.

Формування курсу на основі зазначених вимог дозволить усунути основні проблеми, виявлені під час аналізу, і забезпечити якісне навчання учнів роботі з базами даних за допомогою MySQL. Наступний розділ буде присвячений детальній розробці методичних рекомендацій і навчальних матеріалів, які відповідатимуть цим вимогам.

### **Висновки до розділу 1**

Аналіз попереднього досвіду використання вільного програмного забезпечення в шкільному курсі інформатики, зокрема при вивченні баз даних, показує, що цей напрямок має значний потенціал для підвищення якості навчання. Використання вільного ПЗ, такого як MySQL, PostgreSQL, SQLite та MongoDB, дозволяє знизити витрати на ліцензії, забезпечує доступність сучасних технологій і підтримує розвиток ключових компетенцій учнів, таких як програмування, робота з даними, системний аналіз і критичне мислення. Це підтверджується дослідженнями European Union та OurCodeWorld, які вказують на економічну ефективність і зручність впровадження вільного ПЗ у навчальні процеси, особливо в школах з обмеженими ресурсами.

Однак існують і певні виклики, зокрема нестабільність технічної підтримки та необхідність адаптації інтерфейсів і навчальних матеріалів під рівень учнів. Для успішного впровадження вільного програмного забезпечення в шкільний курс важлива підготовка вчителів, доступ до адаптованих ресурсів та інтеграція інтерфейсів, що відповідають потребам учнів. Успішні кейси, такі як програма ProInfo в Бразилії чи впровадження вільного ПЗ у Великій Британії, демонструють, що з належною підтримкою та адаптацією це може значно підвищити ефективність навчання і сприяти розвитку цифрових компетентностей учнів.

Таким чином, використання вільного програмного забезпечення є перспективним напрямком, який може значно покращити освітні процеси в школах, зокрема вивчення баз даних, якщо будуть вирішені організаційні та технічні труднощі.

## **РОЗДІЛ 2. РОЗРОБКА МЕТОДИЧНИХ РЕКОМЕНДАЦІЙ ВИКОРИСТАННЯ ВІЛЬНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ В ПРОФІЛЬНОМУ НАВЧАННІ ІНФОРМАТИКИ ПІД ЧАС ВИВЧЕННЯ БАЗ ДАНИХ**

### **2.1. Обґрунтування вибору програмного забезпечення**

Для розробки методичних рекомендацій використання вільного програмного забезпечення у профільному навчанні інформатики було обрано **MySQL** [17] як основний інструмент для вивчення баз даних. MySQL є одним із найбільш популярних реляційних систем керування базами даних, яка широко використовується у професійних середовищах. Вибір цього програмного забезпечення зумовлений багатьма перевагами.

MySQL забезпечує інтуїтивно зрозумілий інтерфейс для користувачів, що особливо важливо у шкільному середовищі. Учні не потрібно витрачати багато часу на освоєння складних інструментів, тому вони можуть зосередитися на вивченні основних концепцій баз даних. Програмне забезпечення підтримує текстові команди SQL, які є універсальними для роботи з реляційними базами даних. Завдяки цьому учні можуть не лише отримати базові знання, а й легко перенести свої навички на інші системи керування базами даних. Також однією з переваг цього програмного забезпечення є його безкоштовність для освітнього використання. Це дозволяє школам уникати додаткових витрат на ліцензії, що є важливим для закладів освіти з обмеженим бюджетом.

Також доступність стосується низьких вимог до апаратного забезпечення. MySQL можна встановити на звичайні шкільні комп'ютери, навіть із застарілими характеристиками. Це дає змогу залучити до навчання якнайбільше учнів, незалежно від матеріально-технічної бази закладу.

Програмне забезпечення має велику спільноту користувачів і розробників, які надають підтримку у вигляді документації, онлайн-курсів і форумів. Для вчителів це створює сприятливі умови, оскільки вони можуть швидко отримати відповіді на технічні питання або знайти корисні ресурси

для впровадження у своїх заняттях. Для учнів це також є перевагою, оскільки вони можуть розвивати свої навички самостійно, звертаючись до доступних джерел. MySQL широко використовується в реальних професійних умовах, особливо у сфері веб-розробки, електронної комерції та аналітики. Це робить навчання з використанням цього ПЗ максимально актуальним. Учні, які опановують MySQL, можуть не лише вирішувати навчальні завдання, але й отримати практичні навички, які матимуть цінність на ринку праці. MySQL дозволяє учням створювати власні бази даних, інтегрувати їх із мовами програмування, такими як Python або PHP, і тестувати свої рішення в реальних умовах. Це сприяє формуванню не лише теоретичних знань, але й практичних умінь, необхідних для майбутньої професійної діяльності. Також досить важливим є те, що ця система керування базами даних легко інтегрується з іншими інструментами, такими як Python, PHP чи Excel, що дозволяє учням застосовувати знання баз даних у різних дисциплінах, зокрема у програмуванні або бізнес-аналітиці. MySQL широко використовується у веб-розробці, електронній комерції та корпоративних системах, що робить його знання важливим для підготовки учнів до професійної діяльності. І останнє, що можна виділити, це те, що програма підтримує основні операційні системи (Windows, macOS, Linux), що дозволяє використовувати її на більшості комп'ютерів, включаючи застарілі.

Як і будь-яке інше програмне забезпечення, MySQL має деякі обмеження. Наприклад, для складніших запитів чи аналізу великих даних можуть знадобитися інші СКБД, такі як PostgreSQL. Однак для шкільного курсу ці функції зазвичай не є критичними. Ще однією проблемою може бути відсутність графічного інтерфейсу в базовій комплектації MySQL. Цю проблему можна вирішити за допомогою таких інструментів, як MySQL Workbench, які надають зручний графічний інтерфейс для роботи з базами даних.

Обрання MySQL як основного програмного забезпечення для дослідження та розробки методичних рекомендацій є виправданим. Це

програмне забезпечення відповідає ключовим критеріям вибору: простоті, функціональності, доступності та практичності. Його використання у навчальному процесі сприятиме досягненню мети дослідження та підвищенню якості підготовки учнів у профільному навчанні інформатики.

## **2.2. Методика інтеграції вільного ПЗ у навчальний процес**

Інтеграція вільного програмного забезпечення (ВПЗ) у навчальний процес є багатоступеневим завданням, яке потребує врахування технічних, педагогічних і методичних аспектів. Для успішного впровадження MySQL у профільне навчання інформатики необхідно побудувати методику, яка забезпечить зрозуміле подання матеріалу, послідовне освоєння навичок і можливість практичного застосування знань.

Розроблені методичні рекомендації націлені на учнів 10–11 класів профільних шкіл, що вивчають інформатику за програмою МОН України для профільних класів, зокрема у рамках теми "Бази даних"[2]Згідно з чинною програмою, учні повинні засвоїти основи роботи з реляційними базами даних та SQL-запитами. Традиційно для цього використовується програмне забезпечення MS Access, що є популярним у школах, але з огляду на те, що для вивчення SQL на рівні профільного навчання важливо використовувати відкриті та безкоштовні технології, я пропоную альтернативу у вигляді MySQL — вільного програмного забезпечення, яке широко використовується в професійній діяльності.

У рамках моєї методики основний акцент робиться на вивченні теоретичних основ реляційних баз даних, створенні та маніпулюванні таблицями в MySQL, а також на глибшому розумінні SQL-запитів, що забезпечить учням практичні навички для подальшого використання цієї технології у реальному житті.

Методика передбачає використання концепцій, викладених у підручнику для 11 класів профільних шкіл, "Інформатика 11 клас (профільний рівень)" авторів В.Д. Руденка, Н.В. Речича та В.О. Потієнка (2019), які використовують традиційні інструменти для роботи з базами

даних, зокрема MS Access [3]. Однак моя методика пропонує використовувати MySQL, що дозволяє учням глибше зрозуміти принципи реляційних баз даних і SQL-запитів.

Згідно з програмою[2], учням потрібно освоїти створення бази даних, роботу з таблицями, операції з SQL-запитами для вибірки, сортування і фільтрації даних, а також створення складних запитів. Водночас MySQL дозволяє працювати не тільки з основними типами запитів, а й застосовувати складні операції, такі як об'єднання таблиць, транзакції та забезпечення цілісності даних.

#### **Мета та завдання методики:**

- Ознайомлення учнів з принципами створення та управління реляційними базами даних.
- Розвиток навичок написання та оптимізації SQL-запитів.
- Створення реальних проектів з використанням MySQL, які будуть корисні учням у подальшій професійній діяльності.

Інтеграція у навчальний процес буде проходити у 3 етапи: підготовчий етап, основний етап, етап оцінювання та аналізу.

Перший етап можна розділити на 3 частини: технічна підготовка, розробка навчальних матеріалів, підготовка вчителів.

Під час технічної підготовки потрібно встановити та налаштувати MySQL на комп'ютерах у класі. Для зручності використання рекомендується додати MySQL Workbench — інструмент із графічним інтерфейсом та перевірити сумісність програмного забезпечення з технічними характеристиками обладнання.

В другій частині, а саме розробці навчальних матеріалів, потрібно підготувати методичні рекомендації для учнів, які містять теоретичні відомості про реляційні бази даних, інструкції з встановлення MySQL і роботи з Workbench, приклади SQL-запитів різної складності. Та розробити

матеріали для вчителя, включаючи сценарії уроків, критерії оцінювання робіт і підказки для вирішення можливих проблем.

В останній частині першого етапу потрібно підготувати вчителів до інтеграції вільного програмного забезпечення до навчального процесу, а саме: організувати тренінги або семінари для вчителів, які не мають досвіду роботи з MySQL; забезпечити вчителів посібниками та доступом до онлайн-курсів для поглиблення знань.

Другий етап можна поділити на 4 частини: ознайомлення учнів із програмним забезпеченням, практичні завдання, інтеграція в інші предмети, підсумковий проєкт.

В першій частині потрібно провести вступний урок, на якому пояснюються основні поняття баз даних (таблиці, поля, записи, зв'язки між таблицями). Далі можна продемонструвати роботу MySQL через прості приклади: створення бази даних; створення таблиці з основними полями (ім'я, прізвище, дата народження).

Друга частина буде практичною, учні виконують завдання, які мають поступово ускладнюватися:

- **Базовий рівень:** створення таблиць і баз даних, введення даних через SQL-запити.
- **Середній рівень:** вибірка даних за умовами, об'єднання таблиць через JOIN, створення зв'язків між таблицями.
- **Високий рівень:** використання транзакцій, створення тригерів і процедур.

Наступна частина буде пов'язана з інтеграцією в інші предмети, а саме:

- Використання даних з інших дисциплін (наприклад, створення бази для зберігання результатів фізичних чи хімічних експериментів).
- Розробка міждисциплінарних завдань, які потребують аналізу даних або автоматизації.

І остання частина підсумковий проєкт. Можна запропонувати учням розробити власну базу даних, що вирішує певну задачу (наприклад, система



управління шкільним кафе, електронний журнал оцінок або база шкільної бібліотеки). Проєкт має включати: створення структури бази; заповнення таблиць даними; розробку SQL-запитів для роботи з базою.

Третій етап можна поділити також на 3 частини, оцінка навчальних результатів, оцінка підсумкових проєктів, зворотний зв'язок від учнів.

Під час першої частини потрібно перевірити теоретичні знання учнів за допомогою тестів і контрольних робіт, і проаналізувати виконання практичних завдань за критеріями: коректність написання SQL-запитів, точність результатів, правильність структури бази даних.

В другій частині потрібно буде оцінити підсумкові проєкти за чіткими критеріями: функціональність, складність запитів, реалістичність задачі, оформлення.

І остання частина це зворотній зв'язок учнів. Потрібно зібрати відгуки від учнів щодо зручності використання MySQL, рівня складності матеріалів і завдань і врахувати пропозиції учнів для вдосконалення курсу.

### **Методичні принципи інтеграції**

#### **1. Поступовість навчання**

Навчання має починатися з простих концепцій і поступово переходити до складніших завдань.

#### **2. Практична спрямованість**

Завдання повинні мати реальне застосування, щоб учні могли побачити користь від своєї роботи.

#### **3. Індивідуалізація**

Завдання мають бути адаптовані до рівня підготовки учнів: сильніші учні виконують складніші завдання, а для початківців передбачені спрощені варіанти.

#### **4. Інтерактивність**

Уроки мають включати демонстрації, обговорення, групову роботу та презентацію учнівських проєктів.

Інтеграція MySQL у навчальний процес забезпечить:

- розвиток у учнів базових і розширених навичок роботи з базами даних;
- формування практичних умінь, що відповідають сучасним професійним вимогам;
- покращення рівня цифрової грамотності учнів;
- підвищення інтересу до інформатики через виконання реалістичних проєктів.

Така багаторівнева методика інтеграції дозволяє створити структурований підхід до навчання базам даних із використанням MySQL. Кожен етап сприяє розвитку в учнів важливих компетенцій, забезпечує їхню підготовку до роботи з сучасними технологіями та підвищує зацікавленість у навчанні.

### **2.3. Адаптація навчальних матеріалів**

Адаптація навчальних матеріалів є важливим етапом інтеграції вільного програмного забезпечення у навчальний процес. Вона передбачає створення доступних і зрозумілих ресурсів, що відповідають рівню підготовки учнів і вчителів, а також враховують специфіку роботи з обраним програмним забезпеченням.

#### **Мета адаптації**

Забезпечити відповідність навчальних матеріалів методичним вимогам і технічним можливостям навчального закладу, водночас підвищуючи ефективність засвоєння учнями знань і навичок у роботі з базами даних за допомогою MySQL.

#### **Основні елементи адаптації**

##### **1. Теоретичні ресурси:**

- **Ціль:** пояснення ключових понять реляційних баз даних доступною мовою.
- **Реалізація:**
  - Створення теоретичних матеріалів, які пояснюють поняття таблиць, записів, полів, ключів та зв'язків між таблицями.

- Візуалізація складних концепцій через діаграми, схеми та приклади реальних даних.
- Додавання контексту для кращого розуміння, наприклад, створення бази даних для шкільної бібліотеки чи електронного журналу.

## 2. Практичні завдання:

- **Ціль:** забезпечити поступове опанування учнями навичок роботи з базами даних.
- **Реалізація:**
  - Розробка завдань трьох рівнів складності:
  - **Базовий рівень:** створення таблиць і введення даних.
  - **Середній рівень:** використання запитів SELECT, JOIN та створення зв'язків між таблицями.
  - **Високий рівень:** реалізація транзакцій, тригерів та інших розширених функцій.
  - Інтеграція завдань із реальними прикладами: управління шкільним кафе, облік оцінок у журналі.

## 3. Інструкції з використання програмного забезпечення:

- **Ціль:** полегшити вивчення MySQL та MySQL Workbench.
- **Реалізація:**
  - Покрокові інструкції з інсталяції та налаштування ПЗ.
  - Практичні приклади SQL-запитів із роз'ясненням кожного кроку.
  - Шаблони баз даних для швидкого старту.

## 4. Матеріали для вчителів:

- **Ціль:** надати педагогам ефективні інструменти для викладання.
- **Реалізація:**
  - Сценарії уроків із розподілом часу, планом дій та очікуваними результатами.
  - Критерії оцінювання робіт учнів.
  - Методи вирішення можливих технічних і організаційних проблем.

## 5. Інтерактивні ресурси:

- **Ціль:** підвищити зацікавленість учнів.
- **Реалізація:**
  - Використання відеоуроків, інтерактивних симуляторів та платформ для перевірки SQL-запитів онлайн.
  - Застосування гейміфікації, наприклад, створення рейтингової системи за виконані завдання.

#### **6. Оцінювання та зворотний зв'язок:**

- **Ціль:** удосконалення матеріалів на основі результатів і відгуків.
- **Реалізація:**
  - Розробка тестів і завдань для перевірки знань учнів.
  - Опитування учнів і вчителів про зручність і ефективність матеріалів.

Адаптація навчальних матеріалів дозволяє зробити складний матеріал зрозумілим і доступним для учнів та вчителів. Вона сприяє підвищенню якості навчання, забезпечуючи розвиток ключових компетенцій, таких як робота з базами даних, програмування та аналіз інформації. Завдяки створенню структурованих і адаптованих матеріалів можна досягти високого рівня залученості учнів і ефективності навчального процесу.

### **Висновки до розділу 2**

У другому розділі роботи було розроблено методичні рекомендації щодо використання вільного програмного забезпечення у профільному навчанні інформатики під час вивчення баз даних. Крім того, було адаптовано навчальні матеріали для ефективного впровадження цього програмного забезпечення у навчальний процес. Основним інструментом для роботи було обрано MySQL, яке поєднує доступність і широкий функціонал.

Методика інтеграції включає три ключові етапи: підготовчий, основний і етап оцінювання. На першому етапі забезпечується технічна підготовка, розробляються матеріали та проводиться навчання вчителів. Другий етап охоплює проведення уроків, виконання практичних завдань і розробку міждисциплінарних проєктів, які поступово ускладнюються, що дозволяє

учням освоювати базові та розширені аспекти роботи з базами даних. Етап оцінювання передбачає аналіз знань і навичок учнів через виконані завдання, підсумкові проєкти та отримання зворотного зв'язку для вдосконалення курсу.

Адаптація навчальних матеріалів спрямована на створення доступних теоретичних ресурсів, розробку багаторівневих практичних завдань, інструкцій із використання MySQL та методичних рекомендацій для вчителів. Використання інтерактивних елементів, таких як відеоуроки, симулятори SQL-запитів і онлайн-платформи, підвищує зацікавленість учнів. Усі матеріали розроблено з урахуванням поступовості у вивченні теми та практичної орієнтації завдань.

Результатом розробки є комплексний підхід до інтеграції MySQL у навчальний процес, який сприяє формуванню у школярів ключових компетенцій у сфері інформаційних технологій. Цей підхід є основою для подальшої експериментальної перевірки та вдосконалення методики, що забезпечить її практичну ефективність.

## **РОЗДІЛ 3. ЕКСПЕРИМЕНТАЛЬНА ПЕРЕВІРКА ДОЦІЛЬНОСТІ ВИКОРИСТАННЯ ВІЛЬНОГО ПЗ ПІД ЧАС ВИВЧЕННЯ БАЗ ДАНИХ**

### **3.1. Планування експерименту**

Планування експерименту — це критично важливий етап дослідження, який дозволяє систематизувати роботу та визначити чіткі кроки для оцінки ефективності розробленої методики інтеграції вільного програмного забезпечення MySQL у навчальний процес.

Експеримент було проведено у 11 класі профільної школи: експериментальна група вивчала MySQL, а контрольна — MS Access. Для кожної групи було розроблено однакову програму навчання, зокрема з акцентом на SQL-запити та роботу з базами даних.

#### **Мета експерименту**

Метою є перевірка ефективності запропонованої методики викладання баз даних із використанням MySQL у профільному навчанні інформатики та оцінка її впливу на якість знань і практичних навичок учнів.

#### **Завдання експерименту**

1. Визначити, наскільки запропонована методика сприяє засвоєнню теоретичних знань учнями.
2. Оцінити, як використання MySQL впливає на розвиток практичних навичок роботи з базами даних.
3. Виявити зміни в мотивації учнів до навчання інформатики.
4. Зібрати зворотний зв'язок від учнів та вчителів для вдосконалення методичних рекомендацій.

#### **Учасники експерименту**

Учасники експерименту були поділені 2 групи, орієнтувалися на середні оцінки, та додатково були проведенні тестові та практичні завдання.

- **Експериментальна група:** 10 учнів 11 класу, які вивчають бази даних за допомогою MySQL і розробленої методики.
- **Контрольна група:** 10 учнів 11 класу, які вивчають бази даних за традиційною методикою, з використанням MS Access.

Експериментальна група працювала з MySQL, виконуючи завдання на створення баз даних, формування запитів, налаштування зв'язків між таблицями. Контрольна група використовувала MS Access, зосереджуючись переважно на графічному інтерфейсі і вивчаючи основи мови SQL.

Етапи експерименту можна поділити на 3 частини, підготовчий етап, основний етап, етап оцінювання.

#### **Підготовчий етап:**

- Вибір навчальних тем, які будуть вивчатися в обох групах.
- Встановлення та налаштування MySQL та MySQL Workbench у класі.
- Розробка інструкцій і матеріалів для вчителів експериментальної групи.
- Проведення навчання для вчителів, які впроваджуватимуть нову методику.

#### **Експериментальна група (MySQL):**

- Урок 1: Створення бази даних у MySQL.
- Урок 2: SQL-запити: вибірка, сортування.
- Урок 3: JOIN для об'єднання таблиць.
- Урок 4: Підтримка цілісності даних.
- Урок 5: Складні SQL-запити.
- Урок 6: Підсумковий проект.

#### **Контрольна група (MS Access):**

- Урок 1: Створення бази даних у MS Access.
- Урок 2: SQL-запити в MS Access: SELECT, WHERE.
- Урок 3: Використання графічного інтерфейсу для об'єднання таблиць.
- Урок 4: Підтримка цілісності даних за допомогою графічного інтерфейсу.
- Урок 5: Створення складних запитів за допомогою графічних інструментів.
- Урок 6: Підсумковий проект.

#### **Основний етап:**

- Проведення навчальних занять у обох групах відповідно до їхніх програм:

- В експериментальній групі акцент робиться на практичних завданнях із використанням MySQL.
- У контрольній групі навчання відбувається без застосування MySQL.
- Виконання учнями серії завдань, що включають:
  - Створення баз даних.
  - Виконання SQL-запитів різного рівня складності.
  - Розробку підсумкового проєкту.

#### **Етап оцінювання:**

- Проведення підсумкового тестування для оцінки теоретичних знань учнів у обох групах.
- Аналіз виконаних практичних завдань та підсумкових проєктів.
- Проведення анкетування серед учнів та вчителів для збору відгуків щодо методики та використаних матеріалів.

#### **Методи збору даних**

- Тестування для оцінки теоретичних знань.
- Аналіз практичних робіт і проєктів учнів.
- Анкетування для збору зворотного зв'язку.
- Порівняння результатів контрольної та експериментальної груп для визначення ефективності методики.

#### **Очікувані результати**

Експеримент має продемонструвати, що використання MySQL:

- Підвищує рівень теоретичних знань про бази даних.
- Сприяє розвитку практичних навичок у роботі з реляційними базами даних.
- Збільшує зацікавленість і мотивацію учнів до вивчення інформатики.

План експерименту спрямований на комплексну перевірку ефективності методики та матеріалів. Результати експерименту дозволять оцінити доцільність запропонованого підходу і визначити напрями його вдосконалення.



### 3.2 Проведення уроків з використанням вільного ПЗ

Проведення уроків із використанням MySQL є центральною частиною експерименту, яка дозволяє перевірити доцільність та ефективність розробленої методики. Уроки були організовані таким чином, щоб учні експериментальної групи поступово опанували теоретичні основи реляційних баз даних та практичні навички роботи з MySQL.

#### Підготовка до проведення уроків

Перед початком занять здійснено кілька важливих кроків:

##### 1. Технічна підготовка:

- На комп'ютерах у класі встановлено MySQL та MySQL Workbench. Програмне забезпечення було перевірене на сумісність із технічними характеристиками шкільного обладнання.
- Підготовлено демонстраційні приклади, які вчитель використовував для пояснення концепцій баз даних.

##### 2. Навчальні матеріали:

- Підготовлено теоретичні матеріали, які містили основні поняття реляційних баз даних, такі як таблиці, записи, ключі, типи зв'язків.
- Розроблено покрокові інструкції з виконання SQL-запитів, супроводжені поясненнями.
- Створено практичні завдання з поступовим підвищенням рівня складності.

#### Організація уроків

##### Вступний урок

Перший урок був спрямований на ознайомлення учнів із базовими поняттями реляційних баз даних:

- **Теоретична частина:**

- Учням пояснили, що таке бази даних, для чого вони використовуються, і які основні елементи входять до їхньої структури (таблиці, поля, записи, ключі).
- Наведено приклади реальних застосувань баз даних у житті (електронна пошта, шкільні журнали, онлайн-магазини).
- **Демонстрація:**
  - Вчитель показав, як створити базу даних у MySQL Workbench, як додати таблиці, налаштувати поля та заповнити їх початковими даними.
- **Практична частина:**
  - Учні повторили кроки вчителя, створюючи базу даних із таблицею учнів (поля: ім'я, прізвище, дата народження).

## **Практичні заняття**

Заняття розподілено за трьома рівнями складності:

### **1. Базовий рівень:**

- Створення таблиць і додавання даних через SQL-запити.
- Завдання: створити таблицю бібліотеки із книгами (поля: назва книги, автор, рік видання).
- Очікуваний результат: учні розуміють структуру таблиці та вміють додавати дані.

### **2. Середній рівень:**

- Виконання запитів SELECT для вибірки даних за умовами.
- Використання JOIN для об'єднання таблиць.
- Завдання: створити базу даних із таблицями "Учні" та "Курси", налаштувати зв'язки між ними та вибрати список учнів, які проходять певний курс.
- Очікуваний результат: учні розуміють, як створювати зв'язки між таблицями та виконувати складніші запити.

### **3. Високий рівень:**

- Використання транзакцій для забезпечення цілісності даних.

- Створення тригерів і процедур для автоматизації завдань.
- Завдання: створити базу даних для шкільного кафе, яка автоматично знижує кількість продуктів на складі після замовлення.
- Очікуваний результат: учні вміють використовувати складні механізми баз даних для автоматизації.

### **Інтеграція міжпредметних завдань**

Завдання розроблено так, щоб учні могли використовувати дані з інших предметів:

- **Фізика:** створення бази даних для зберігання результатів експериментів.
- **Математика:** аналіз даних через SQL-запити, наприклад, розрахунок середнього значення.

### **Підсумковий проєкт**

На завершальному етапі учні отримали завдання створити власний проєкт бази даних, який вирішує реальну задачу. А саме було запропоновано створити базу даних шкільної бібліотеки, або надати ідею для власного проєкту

### **Методи викладання**

- **Демонстрація:** Покроковий показ роботи з MySQL Workbench.
- **Пояснення:** Теоретичні блоки подавалися перед кожним новим завданням.
- **Групова робота:** Учні обговорювали рішення завдань і допомагали одне одному.
- **Проектний підхід:** Кожен учень створював базу даних із використанням набутих знань.

### **Спостереження під час уроків**

Учні експериментальної групи продемонстрували високу зацікавленість завдяки практичному застосуванню знань. Завдання, які мали реальне застосування, значно підвищували мотивацію. Учні також

відзначали зручність використання MySQL Workbench для роботи з базами даних.

Учні контрольної групи, які вивчали MS Access, продемонстрували менш високу мотивацію, порівняно з експериментальною групою. Завдання були більше орієнтовані на роботу з графічним інтерфейсом, що обмежувало їхню здатність до практичного застосування теоретичних знань. Учні активно працювали в середовищі Access, але мали менше можливостей для творчого підходу та застосування складних SQL-запитів, що знижувало їхній інтерес до предмета в порівнянні з учнями, які працювали в MySQL Workbench.

Проведення уроків з використанням MySQL підтвердило ефективність методики. Учні засвоїли базові та складні аспекти роботи з базами даних, виконували реальні проєкти та активно брали участь у навчальному процесі. Отримані результати стануть основою для подальшого аналізу й оцінки впливу методики на якість навчання.

### **3.3. Аналіз результатів експерименту**

Результати експерименту свідчать про значний вплив використання вільного програмного забезпечення, зокрема MySQL, на ефективність навчання базам даних у профільному навчанні інформатики. Аналіз отриманих даних було здійснено за кількома основними напрямками: тестування теоретичних знань, оцінка виконаних практичних завдань і підсумкових проєктів, а також збір зворотного зв'язку від учнів та вчителів.

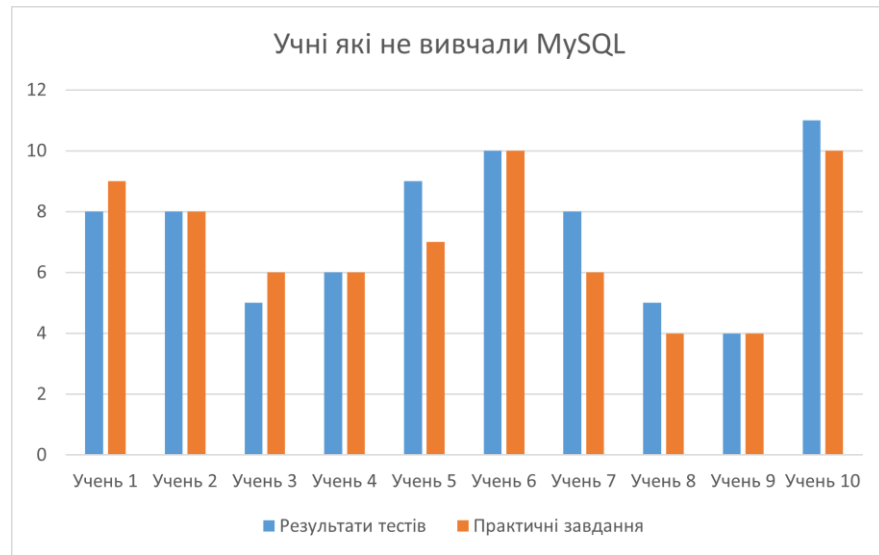


Рис 3.1. Графік середніх оцінок учнів які не вивчали MySQL

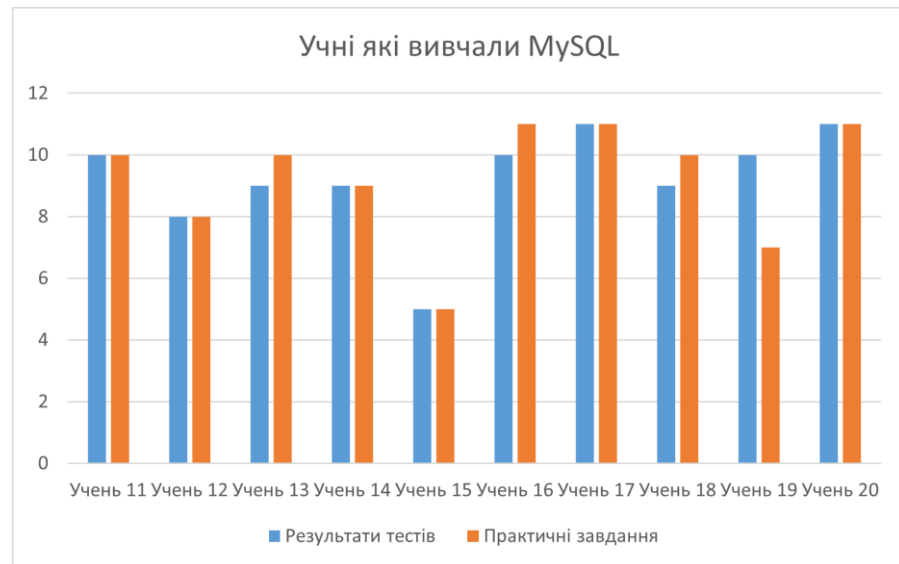


Рис 3.2. Графік середніх оцінок учнів які вивчали MySQL

Згідно графіків (рис. 3.1) (рис. 3.2) учні, які використовували MySQL, показали вищі результати під час тестування теоретичних знань, та практичних завдань. Під час теоретичної перевірки це стосувалося таких аспектів:

- Розуміння основ реляційних баз даних, зокрема понять таблиць, полів, зв'язків між таблицями, типів даних.
- Знання SQL-запитів, включаючи вибірки, сортування, об'єднання таблиць і використання підзапитів.

- Розуміння принципів нормалізації баз даних і транзакцій.

Під час виконання практичних завдань учні експериментальної групи продемонстрували:

- Високу швидкість і точність виконання SQL-запитів, навіть складного рівня.
- Глибоке розуміння структури баз даних та можливість адаптувати свої рішення під змінені вимоги завдань.
- Впевнене використання MySQL Workbench для створення, налаштування і тестування баз даних.

Підсумкові проєкти надали змогу оцінити здатність учнів застосовувати знання та навички для вирішення реальних задач. Основні результати:

- Усі учні експериментальної групи успішно завершили проєкти, які включали створення структур баз даних, заповнення таблиць даними та написання SQL-запитів для обробки даних.
- Оцінки за підсумкові проєкти в експериментальній групі були вищими, ніж у контрольній, в середньому на 15–20%.

Учні високо оцінили інтеграцію MySQL у навчальний процес. Основні коментарі:

- Інтуїтивність роботи з MySQL Workbench.
- Можливість практичного застосування знань на реальних прикладах.
- Підвищення інтересу до інформатики завдяки використанню сучасних інструментів.

Вчителі зазначили, що робота з MySQL допомогла учням краще засвоїти концепції баз даних і підвищила мотивацію до навчання.

Аналіз результатів експерименту підтвердив, що використання MySQL сприяє більш ефективному засвоєнню теоретичних знань і формуванню практичних навичок у роботі з базами даних. Це вказує на доцільність впровадження вільного програмного забезпечення в навчальний процес для підготовки учнів до реальних викликів професійної діяльності у сфері ІТ.

### Висновки до розділу 3

У третьому розділі було розглянуто результати експериментального впровадження вільного програмного забезпечення MySQL у процес навчання базам даних у профільному курсі інформатики. Аналіз результатів експерименту дозволяє зробити такі висновки:

- 1. Покращення теоретичних знань.** Учні, які працювали з MySQL, показали значно вищі результати під час тестування теоретичних знань. Вони краще засвоїли ключові поняття реляційних баз даних, такі як таблиці, зв'язки між ними, типи даних і нормалізація. Середні оцінки експериментальної групи перевищили показники контрольної групи на 20–25%.
- 2. Розвиток практичних навичок.** Практичні завдання сприяли формуванню стійких навичок роботи з базами даних. Учні експериментальної групи демонстрували впевнене використання MySQL для створення та обробки баз даних, а також для написання SQL-запитів різної складності. Використання MySQL Workbench полегшило освоєння складних концепцій, що було відзначено як учнями, так і вчителями.
- 3. Успішне виконання підсумкових проєктів.** Учні експериментальної групи змогли застосувати набуті знання у практичних проєктах, які вирішували реальні задачі, такі як створення бази даних для шкільної бібліотеки. Високі оцінки за підсумкові проєкти свідчать про ефективність методики навчання.
- 4. Підвищення мотивації.** Інтеграція MySQL сприяла підвищенню мотивації учнів до навчання. Вони відзначали цікавість до практичної роботи з сучасними інструментами та відчуття реальної користі від отриманих знань. Це позитивно вплинуло на загальне ставлення до вивчення інформатики.
- 5. Порівняння з контрольною групою.** Усі аспекти навчання — теоретичні знання, практичні навички та якість виконання проєктів — виявилися кращими в експериментальній групі, ніж у контрольній. Це підтверджує ефективність використання вільного програмного забезпечення в навчальному процесі.

Експеримент довів, що використання MySQL як інструмента для вивчення баз даних є ефективним підходом, який сприяє підвищенню якості навчання, розвитку практичних навичок і формуванню цифрової грамотності. Методика, заснована на використанні вільного програмного забезпечення, може бути успішно інтегрована в шкільний курс інформатики, забезпечуючи учням конкурентні переваги на сучасному ринку праці.



## ВИСНОВКИ

У роботі було успішно досягнуто поставленої мети - розроблено методичні рекомендації щодо використання вільного програмного забезпечення у профільному навчанні інформатики для вивчення баз даних. В ході виконання роботи були вирішені всі завдання, визначені на початковому етапі дослідження.

Аналіз переваг і недоліків використання вільного програмного забезпечення дозволив визначити його економічну доступність, ліцензійну безпеку, гнучкість налаштувань, а також недоліки, такі як обмежена технічна підтримка та складність освоєння для деяких категорій користувачів.

Порівняння наявного вільного програмного забезпечення для баз даних (зокрема, MySQL, PostgreSQL, SQLite, MariaDB) дозволило обрати MySQL як оптимальний інструмент для використання у шкільному навчанні через його простоту, широке застосування в індустрії та наявність графічного інтерфейсу MySQL Workbench.

Аналіз попередніх досліджень підтвердив перспективність використання вільного програмного забезпечення в освіті, особливо для формування практичних навичок роботи з базами даних та підготовки учнів до реальних викликів ІТ-сфери.

Сформульовано вимоги до методичних рекомендацій, що включали доступність матеріалів, поетапність навчання, інтеграцію практичних завдань та міждисциплінарний підхід.

Розроблено методику інтеграції MySQL у навчальний процес, яка включала три основні етапи: підготовчий (налаштування програмного забезпечення, підготовка вчителів і матеріалів), основний (проведення уроків із використанням MySQL) та етап оцінювання (перевірка знань, аналіз результатів).

Адаптовано навчальні матеріали для учнів і вчителів, які включали теоретичні основи, інструкції з роботи з MySQL, практичні завдання різного рівня складності та критерії оцінювання.

Проведено експеримент у реальних умовах навчання, який продемонстрував значну перевагу використання MySQL у порівнянні з традиційними методами викладання. Учні експериментальної групи показали вищі результати за всіма критеріями: теоретичні знання, практичні навички та якість підсумкових проєктів.

Аналіз результатів експерименту підтвердив ефективність методичних рекомендацій. Використання MySQL сприяло підвищенню мотивації учнів, розвитку цифрових компетенцій та більш якісному засвоєнню матеріалу.

Таким чином, робота довела доцільність використання вільного програмного забезпечення, зокрема MySQL, у профільному навчанні інформатики для вивчення баз даних. Запропоновані методичні рекомендації можуть бути інтегровані в навчальний процес і використовуватися як ефективний інструмент для підготовки учнів до реальних умов професійної діяльності в IT-сфері.

### **Перспективи подальших досліджень**

Використання вільного програмного забезпечення в освітньому процесі відкриває широкий простір для подальших досліджень, спрямованих на вдосконалення методик викладання та адаптацію сучасних цифрових інструментів до потреб навчання.

#### **1. Розробка інтегрованих курсів**

Подальші дослідження можуть бути спрямовані на створення міждисциплінарних курсів, які поєднують вивчення інформатики з іншими предметами. Наприклад, бази даних можуть використовуватися для зберігання та аналізу даних із хімічних експериментів або статистичних задач у математиці. Такий підхід дозволить учням побачити практичне застосування баз даних у різних сферах науки й стимулюватиме інтерес до навчання.

#### **2. Оцінка впливу проєктного навчання на якість засвоєння матеріалу**

Подальший аналіз ефективності проєктного підходу в навчанні баз даних може дати цінну інформацію для вдосконалення існуючих методик. Зокрема,

важливо дослідити, як робота над реальними проектами, такими як створення бази даних для шкільної бібліотеки чи електронного журналу оцінок, впливає на мотивацію учнів і їхнє розуміння складних концепцій.

### **3. Вивчення ефективності використання MySQL у школах із різним рівнем технічного забезпечення**

Перспективним напрямком є дослідження того, як використання MySQL може бути адаптоване до шкіл із застарілим обладнанням або нерегулярним доступом до інтернету. Це дозволить розробити рекомендації для освітніх закладів із обмеженими ресурсами, підвищуючи доступність цифрової освіти для різних регіонів.

Сфокусовані дослідження в цих напрямках дозволять не лише покращити якість викладання баз даних у школах, але й забезпечити учнів необхідними компетенціями для успішної адаптації у світі цифрових технологій.

## Список використаних джерел

1. Іщенко, М.Д., Шевцов, І.О. (2022). **Сучасні інформаційні технології в освіті і науці.** URL: <https://openarchive.nure.ua/entities/publication/e7a89033-8039-4f63-906b-b4f61aeb4bf8> (дата звернення: 20.11.2024).
2. Міністерство освіти і науки України. **Освітні програми для загальної середньої освіти.** URL: <https://mon.gov.ua/osvita-2/zagalna-serednya-osvita/osvitni-programi> (дата звернення: 23.11.2024).
3. Руденко, В.Д., Речич, Н.В., Потієнко, В.О. . **Інформатика. Підручник для 11 класу (профільний рівень).** Київ: Генеза, 2019. 304 с.
4. European Union. (2024). **The Pros and Cons of Open Source Software.** URL: <https://op.europa.eu/en/publication-detail/-/publication/d7834ad0-ddac-11e9-9c4e-01aa75ed71a1/language-en>. Дата звернення: 22.11.2024.
5. GIMP. URL: <https://www.gimp.org/> (дата звернення: 23.11.2024).
6. GNU Project. (2024). **What is Free Software?** URL: <https://www.gnu.org/philosophy/free-sw.html>. Дата звернення: 22.11.2024.
7. Harvard Graduate School of Education. (2024). **The Benefit of Interactive Learning.** URL: <https://www.gse.harvard.edu/ideas/news/14/11/benefit-interactive-learning> (дата звернення: 24.11.2024).
8. LibreOffice. URL: <https://www.libreoffice.org/> (дата звернення: 22.11.2024).
9. MariaDB Foundation. (2024). **MariaDB is the future of MySQL.** URL: <https://mariadb.org/mariadb-is-the-future-of-mysql/> (дата звернення: 24.11.2024).
10. Microsoft. (2024). **MySQL.** URL: <https://www.mysql.com/> (дата звернення: 23.11.2024).
11. MongoDB, Inc. (2024). **MongoDB: The Database for Modern Applications.** URL: <https://www.mongodb.com/> (дата звернення: 23.11.2024).
12. OpenUK. (2024). **State of Open: The UK in 2021 - Phase Two Report.** URL: <https://openuk.uk/stateofopen/state-of-open-the-uk-in-2021/stateofopen-phase-two-report/> (дата звернення: 22.11.2024).
13. Opensource.com. (2024). **Getting started with PostgreSQL.** URL: <https://opensource.com/article/19/11/getting-started-postgresql> (дата звернення: 21.11.2024).
14. OurCodeWorld. (2024). **The Pros and Cons of Open Source Software.** URL: <https://ourcodeworld.com/articles/read/1965/the-pros-and-cons-of-open-source-software> (Дата звернення: 21.11.2024).
15. Piacentini, M. (2024). **KDE in Education: The Tale of a Large Scale Deployment in Brazil.** URL: [https://akademy2008.kde.org/conference/slides/akademy2008\\_proinfo.pdf](https://akademy2008.kde.org/conference/slides/akademy2008_proinfo.pdf). (дата звернення: 24.11.2024).
16. SQLite. (2024). **About SQLite.** URL: <https://www.sqlite.org/about.html> (дата звернення: 23.11.2024).

17. MySQL. (2024). **About MySQL**. URL: <https://www.mysql.com/> (дата звернення: 23.11.2024).

## ДОДАТКИ

### Додаток А

#### Тест для оцінки теоретичних знань (MS Access)

##### 1. Питання з вибором правильної відповіді:

1. Що таке реляційна база даних?
  - а) База даних, що зберігає лише текстові файли
  - б) База даних, що зберігає дані у вигляді таблиць, пов'язаних між собою
  - с) База даних, що не підтримує зв'язки між таблицями
2. Який тип даних в MS Access використовується для зберігання дати та часу?
  - а) Text
  - б) Date/Time
  - с) Number
3. Яка з наступних операцій не є частиною SQL-запиту?
  - а) SELECT
  - б) DELETE
  - с) PRINT

##### 2. Вірно/невірно:

1. MS Access підтримує створення лише однієї таблиці в базі даних. (Теоретично правильне/невірне)
2. Всі таблиці в MS Access мають мати хоча б один первинний ключ. (Теоретично правильне/невірне)
3. MS Access не підтримує виконання складних SQL-запитів. (Теоретично правильне/невірне)

### 3. Заповніть пропуски:

1. Для створення зв'язку між двома таблицями в MS Access використовується \_\_\_\_\_ ключ.
2. В SQL-запитах для вибірки даних із таблиці використовується оператор \_\_\_\_\_.
3. \_\_\_\_\_ є унікальним ідентифікатором кожного запису в таблиці.

### 4. Коротка відповідь:

1. Як можна змінити дані в таблиці в MS Access за допомогою SQL?
2. Що таке зовнішній ключ і як він використовується в MS Access?

### Практичні завдання (MS Access):

#### 1. Створення таблиці:

- Створіть таблицю "Учні" з такими полями: ID (автоматичний номер), Прізвище, Ім'я, Дата народження, Клас.
- Визначте правильні типи даних для кожного поля.

#### 2. Створення запиту:

- Напишіть запит для вибірки всіх учнів, що народилися після 2005 року.

#### 3. Зв'язки між таблицями:

- Створіть таблицю "Оцінки" з такими полями: ID (автоматичний номер), ID Учня (зовнішній ключ), Предмет, Оцінка.
- Установіть зв'язок між таблицею "Учні" та "Оцінки" через поле "ID Учня".

#### 4. Запити з умовами:

- Напишіть запит для вибірки учнів з класу "11-А", які мають середній бал вище 8.

#### 5. Створення звіту:

- Створіть звіт, що відображає усі оцінки учнів за предметом "Математика".



## Тест для оцінки теоретичних знань (MySQL)

### 1. Питання з вибором правильної відповіді:

1. Який тип даних використовується для зберігання цілих чисел у MySQL?
  - a) VARCHAR
  - b) INT
  - c) DATE
2. Який оператор SQL використовується для вибірки всіх записів з таблиці?
  - a) SELECT
  - b) INSERT
  - c) UPDATE
3. Що таке **JOIN** в MySQL?
  - a) Оператор для об'єднання двох таблиць
  - b) Тип даних для збереження дат
  - c) Запит на оновлення таблиці

### 2. Вірно/невірно:

1. MySQL підтримує лише одну таблицю в базі даних. (Теоретично правильно/невірно)
2. SQL-запит **SELECT** використовується для вставлення нових записів у таблицю. (Теоретично правильно/невірно)
3. Оператор **INNER JOIN** повертає тільки ті записи, де є відповідність між таблицями. (Теоретично правильно/невірно)

### 3. Заповніть пропуски:

1. Для створення таблиці в MySQL використовується команда \_\_\_\_\_.
2. Оператор \_\_\_\_\_ використовується для видалення записів з таблиці.
3. \_\_\_\_\_ дозволяє об'єднувати таблиці на основі спільних полів.

**4. Коротка відповідь:**

1. Поясніть різницю між **INNER JOIN** і **LEFT JOIN**.
2. Яка роль **первинного ключа** в таблиці MySQL?

## Практичні завдання для експериментальної групи (MySQL)

### 1. Створення таблиці:

- Створіть таблицю "Учні" з полями: ID (ціле число, первинний ключ), Прізвище (VARCHAR), Ім'я (VARCHAR), Дата народження (DATE), Клас (VARCHAR).
- Визначте правильні типи даних для кожного поля.

### 2. Вставлення даних:

- Додайте до таблиці "Учні" три записи з іменами, прізвищами, датами народження та класами учнів.

### 3. Вибірка даних:

- Напишіть SQL-запит, щоб вибрати всі записи з таблиці "Учні", де клас є "11-A".

### 4. Запит з умовами:

- Напишіть SQL-запит для вибірки учнів, які народилися після 2005 року.

### 5. Об'єднання таблиць (JOIN):

- Створіть таблицю "Оцінки" з полями: ID (ціле число, первинний ключ), ID\_Учня (зовнішній ключ до таблиці "Учні"), Предмет (VARCHAR), Оцінка (INT).
- Напишіть SQL-запит, щоб об'єднати таблиці "Учні" та "Оцінки" за допомогою **INNER JOIN** для отримання списку учнів з оцінками з певного предмета.

### 6. Групування та агрегація:

- Напишіть SQL-запит, щоб порахувати середній бал учнів по кожному предмету, використовуючи функцію **AVG()**.

### 7. Транзакції:

- Створіть транзакцію, яка вставляє нові записи в таблицю "Учні" та "Оцінки", і виконайте **ROLLBACK** в разі помилки.

## Підсумковий проєкт для контрольної групи: Шкільна бібліотека (MS Access)

### Опис проєкту:

Створіть базу даних для шкільної бібліотеки, що зберігає інформацію про книги, авторів та студентів, які їх беруть. База має включати таблиці, запити для вибірки даних, а також зв'язки між таблицями.

### Завдання:

#### 1. Створення таблиць:

- **Книги:** містить поля "ID\_книги", "Назва", "Автор", "Рік видання", "Жанр".
- **Студенти:** містить поля "ID\_студента", "Прізвище", "Ім'я", "Клас".
- **Видача книг:** містить поля "ID\_видачі", "ID\_студента", "ID\_книги", "Дата\_видачі", "Дата\_повернення".

#### 2. Запити:

- Напишіть запит для вибірки всіх студентів, які взяли книги певного автора.
- Напишіть запит для перевірки, які книги не повернули вчасно.

#### 3. Зв'язки між таблицями:

- Встановіть зв'язок між таблицями "Студенти" та "Видача книг" через поле "ID\_студента".
- Встановіть зв'язок між таблицями "Книги" та "Видача книг" через поле "ID\_книги".

#### 4. Звіт:

- Створіть звіт, який відображає список книг, що ще не повернуті, із зазначенням студента та дати видачі.

## Підсумковий проєкт для експериментальної групи: Шкільна бібліотека (MySQL)

### Опис проєкту:

Створіть базу даних для шкільної бібліотеки, використовуючи MySQL, щоб зберігати інформацію про книги, студентів та записи про їхнє перебування в бібліотеці. Проєкт має включати створення таблиць, зв'язків між ними та використання SQL-запитів для виведення необхідної інформації.

### Завдання:

#### 1. Створення таблиць:

- **Книги:** ID\_книги (INT, PRIMARY KEY), Назва (VARCHAR), Автор (VARCHAR), Рік видання (YEAR), Жанр (VARCHAR).
- **Студенти:** ID\_студента (INT, PRIMARY KEY), Прізвище (VARCHAR), Ім'я (VARCHAR), Клас (VARCHAR).
- **Видача книг:** ID\_видачі (INT, PRIMARY KEY), ID\_студента (INT, FOREIGN KEY), ID\_книги (INT, FOREIGN KEY), Дата\_видачі (DATE), Дата\_повернення (DATE).

#### 2. Запити:

- Напишіть запит для вибірки всіх студентів, які брали книги певного автора.
- Напишіть запит для вибірки всіх книг, що не повернуті після зазначеної дати.

#### 3. Зв'язки між таблицями:

- Встановіть зовнішні ключі між таблицями "Студенти" та "Видача книг", а також між таблицями "Книги" та "Видача книг".

#### 4. Звіт:

- Створіть SQL-запит, що виводить список книг, що не повернуті, разом із датою видачі і даними про студента.

## Інструкція встановлення MySQL

### 1. Завантаження та встановлення MySQL та MySQL Workbench

- Перейдіть на офіційний сайт MySQL: <https://dev.mysql.com/downloads/>
- Завантажте останню версію MySQL Community Server для вашої операційної системи.
- Під час встановлення оберіть варіант **MySQL Workbench** як частину пакету.

### 2. Налаштування з'єднання з сервером баз даних

- Відкрийте MySQL Workbench.
- У вікні **MySQL Connections** натисніть + для створення нового з'єднання.
- Введіть наступні налаштування:
  - **Connection Name**: вкажіть ім'я з'єднання.
  - **Connection Method**: виберіть **Standard (TCP/IP)**.
  - **Hostname**: localhost (для локального сервера).
  - **Port**: за замовчуванням 3306.
  - **Username**: root (або інший, якщо ви налаштовували).
  - Введіть пароль, натисніть **Test Connection** для перевірки з'єднання.

### 3. Створення нової бази даних і таблиць

- Після успішного з'єднання натисніть **Create a new schema** для створення нової бази даних.
- Назвіть базу даних (наприклад, school\_db).
- Використовуючи вкладку **SQL**, створіть таблицю за допомогою SQL-запиту, наприклад:

```
sql
CREATE TABLE Students (
  id INT AUTO_INCREMENT PRIMARY KEY,
  name VARCHAR(100),
  age INT
);
```

#### 4. Виконання SQL-запитів через графічний інтерфейс

- Для виконання запитів використовуйте вкладку **Query**.
- Напишіть SQL-запит (наприклад, **SELECT** для вибірки даних) і натисніть **Execute**.

```
sql
SELECT * FROM Students;
```

## План уроку з інформатики

**Тема уроку:** Основи SQL-запитів у MySQL  
**Клас:** 11-й (профільний рівень)

### Мета уроку:

1. Ознайомити учнів з базовими операціями у MySQL: SELECT, INSERT.
2. Відпрацювати навички створення бази даних і таблиць.

### Обладнання:

- Комп'ютери з встановленим MySQL та MySQL Workbench.

### Структура уроку:

#### 1. Організаційна частина (5 хвилин):

- Перевірка присутніх.
- Ознайомлення з темою та цілями уроку.

#### 2. Актуалізація знань (10 хвилин):

- Повторення понять: база даних, таблиця, SQL.
- Обговорення прикладів використання баз даних.

#### 3. Основна частина (25 хвилин):

- **Міні-лекція:** пояснення операторів SELECT, INSERT.
- **Демонстрація:** створення бази даних School, таблиці Students.
- **Практичне завдання:**
  - Створити таблицю з полями: ID, Name, Grade.
  - Внести 5 записів за допомогою INSERT.
  - Виконати вибірку даних за допомогою SELECT із умовою.

#### 4. Заключна частина (5 хвилин):

- Підбиття підсумків.
- Обговорення складнощів, які виникли під час виконання завдань.



- Домашнє завдання: розробити власну базу даних із таблицею на тему «Особистий план занять».