ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Кафедра інженерії програмного забезпечення

КУРСОВИЙ ПРОЕКТ

з дисципліни «Бази даних»

.(назва дисципліни)

на тему: «База даних автоматизованої системи оцінювання знань студентів»

КПІПЗ.2101103.01.05.ПЗ

(шифр, назва)

Рівень вищої освіти             Перший (бакалаврський)

Галузь знань                      12 «Інформаційні технології»

Спеціальність     121 «Інженерія програмного забезпечення»

Освітня програма   «Інженерія програмногозабезпечення»

Студента. І курсу, група . ІПЗc-21-1. \_\_\_\_\_\_\_\_\_\_\_ Є.О. Сідельник.

(шифр) (підпис) (ініціали, прізвище)

Керівник. канд. техн. наук, доцент . \_\_\_\_\_\_\_\_\_\_\_ Ю.В. Форкун .

(посада, вчене звання, науковий ступінь) (підпис) (ініціали, прізвище)

Кількість балів \_\_\_\_\_\_\_

Оцінка за шкалою:

національною\_\_\_\_\_\_\_\_\_/ЄКТС\_\_\_\_

Члени комісії: \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис) (ініціали, прізвище)

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

(підпис) (ініціали, прізвище)

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

(підпис) (ініціали, прізвище)

Хмельницький, 2021

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет програмування та комп’ютерних і телекомунікаційних систем.....

Кафедра інженерії програмного забезпечення

*Спеціальність.* 121 «Інженерія програмного забезпечення»

Шифр, назва

Затверджую

Завідувач кафедри \_\_\_\_\_\_\_\_\_\_\_\_\_

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

\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_202\_\_\_ року

**ЗАВДАННЯ**

**НА КУРСОВИЙ ПРОЕКТ**

Сідельник Євген Олександрович \_

Прізвище, ім’я, по батькові студента

1. Тема проекту База даних автоматизованої системи оцінювання знань студентів

керівник проекту Форкун Юрій Вікторович канд. техн. наук, доцент

Прізвище, ім’я, по батькові науковий ступінь, вчене звання

2. Строк подання студентом проекту на кафедру . 22.12.2021 .

3. Вихідні дані до проекту\_\_методичні вказівки до курсового проектування для студентів спеціальності 121 «Інженерія програмного забезпечення»» з курсу «Бази даних»

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити) дослідження предметної області, проектування та реалізація бази даних на фізичному рівні, детальне проектування, реалізація БД та розробка програмного забезпечення з її ведення

5. Перелік графічного матеріалу (із зазначенням обов’язкових креслень) …………………............................................................................................................

6. Дата видачі завдання 08.10.2021========…………………………………….....

Календарний план

|  |  |  |  |
| --- | --- | --- | --- |
| № з/п | Назва етапів курсового проекту | Термін виконання етапів проекту  (тиждень) | Відмітка про виконання |
| 1 | Характеристика предметної області. Постановка задачі | 1 |  |
| 2 | Виділення сутностей предметної області (неформальна модель ПО) | 2 |  |
| 3 | Приведення до третьої нормальної форми | 3-4 |  |
| 4 | Побудова інформаційно-логічної моделі даних (концептуальна ER-діаграма БД) | 5 |  |
| 5 | Вибір та обґрунтування СКБД | 6-7 |  |
| 6 | Створення та заповнення бази даних. Фізична ER-діаграма бази даних | 8-9 |  |
| 7 | Вибір засобів розробки інтерфейсу програмного забезпечення | 10-11 |  |
| 8 | Створення інтерфейсу для роботи з базою даних | 12-14 |  |
| 9 | Налагодження та тестування | 15-16 |  |
| 10 | Інструкція користувача | 17 |  |
| 11 | Формування вимог до технічних засобів | 17 |  |

Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис)

Керівник проекту \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис)

**АНОТАЦІЯ**

Курсовий проект “ База даних автоматизованої системи оцінювання знань студентів”.

Автор роботи: Сідельник Є.О.

Керівник роботи: Форкун Ю.В.

// todo

Обсяг – 58 с., 35 рис., 4 табл., 2 додатки, 9 джерел.

Мета курсового проекту: розробка бази даних предметної області навчального закладу для автоматизації процесу оцінювання та контролю знань студентів.

У курсовому проекті виконана розробка бази даних бази даниих для приватного підприємця, який займається створенням меблів під індивідуальне замовлення, виконано аналіз основних задач: ввід даних про матеріали, ручки, ніжки, класифікації, замовників, складові меблів, знижки, перегляд та коригування даних, вилучення непотрібної інформації, відбір даних за деякими критеріями; виконано аналіз структури даних. Для розробки БД використане середовище розробки Access.

Дата Підпис

ЗМІСТ

Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

5

*КПІПЗ.2101103.01.05.ПЗ*

Розроб.

Сідельник Є. О.

Перевір.

Форкун Ю.В.

Реценз.

Н. Контр.

.

Затверд.

*База даних автоматизованої системи оцінювання   
знань студентів*

Літ.

Акрушів

43

ХНУ. ІПЗс-21-1

[ВСТУП 6](#_Toc9568877)

[1. Дослідження предметної області 7](#_Toc9568878)

[1.1 Характеристика предметної області. Постановка задачі 7](#_Toc9568879)

[1.2 Виділення сутностей предметної області (неформальна модель ПО) 8](#_Toc9568880)

[1.3 Приведення до третьої нормальної форми 11](#_Toc9568881)

[1.4 Побудова інформаційно-логічної моделі даних (концептуальна ER-діаграма БД) 15](#_Toc9568882)

[2. Проектування та реалізація бази даних на фізичному рівні 17](#_Toc9568883)

[2.1 Вибір та обґрунтування СКБД 17](#_Toc9568884)

[2.2 Створення та заповнення бази даних. Фізична ER-діаграма бази даних 19](#_Toc9568885)

[3. Реалізація БД та розробка програмного забезпечення з її ведення 26](#_Toc9568886)

[3.1 Вибір засобів розробки інтерфейсу програмного забезпечення 26](#_Toc9568887)

[3.2 Створення інтерфейсу для роботи з базою даних 27](#_Toc9568888)

[3.3 Інструкція користувача 37](#_Toc9568899)

[3.4 Вимоги до технічних засобів 39](#_Toc9568900)

[ВИСНОВОК 41](#_Toc9568901)

[Перелік посилань 44](#_Toc9568902)

[Додаток А 44](#_Toc9568903)

Додаток Б 45

Вступ

Одним із активів будь-якого навчального закладу є дані. Те, як вводяться, витягуються та використовуються ці дані, може вимагати значних витрат часу та вплинути на ефективність установи загалом. Електронна база даних забезпечує високоефективний спосіб зберігання даних. Є переваги очевидні: економія часу, правильно спроектована база даних може забезпечити доступ до інформації в найкоротші терміни; бази даних забезпечують велику безпеку.

Йдучи в ногу з часом, навчальні заклади стали активно впроваджувати обчислювальну техніку та інформаційні технології для поліпшення якості освіти. Престиж навчального закладу сьогодні залежить не тільки від рівня викладання і матеріально-технічної бази, але й від ефективності і якості системи контролю знань учнів.

Існує безліч форм контролю знань – усне та письмове опитування, різноманітні контрольні та самостійні роботи. Але вони мають великий недолік: людський фактор та суб’єктивна точка зору викладача може вплинути на оцінку студента. Тим більше потрібно кожному студенту приділити увагу при перевірці його знань незалежно від того чи це усне, чи письмове опитування. Використання ж автоматизованої системи усуває зазначені недоліки. Тому найбільш оперативною, сучасною і об'єктивною формою оцінювання знань є комп'ютерне тестування. В наші роки, все більш актуальними стають комп'ютерні системи тестування, здатні доповнити або замінити традиційні методи контролю і методики викладання. Наприклад, зараз все більшої популярності набуває дистанційна освіта, в якій мало не єдиним способом перевірки знань є тестування.

Даний курсовий проєкт є реалізацією реалізацїєю бази даних, що може бути в подальшому використана для системи оцінювання знань студентів. Остання дозволить зменшити час на перевірку рівня засвоєної інформації студентами та усунути суб’єктивне ставлення викладачів до певних студентів, тим самим зробивши отриману оцінку більш об’єктивною та обгрунтованою.

1. Дослідження предметної області
   1. Характеристика предметної області. Постановка задачі

Для контролю знань використовуються як традиційні, так і сучасні методи контролю. Найширше розповсюдження знаходять методи контролю знань шляхом тестування. Зокрема, впровадження модульно-рейтингової системи в навчальний процес вищих закладів освіти вимагає застосування тестового контролю для оцінки знань студентів, що забезпечує високу технологічність проведення контролю та об’єктивність його результатів. Для підвищення ефективності організації тестового контролю його доцільно проводити з використанням комп’ютерних тестових програм, що дозволяє автоматизувати процес проведення контролю та обробку результатів тестування.

Перевірка якості знань за допомогою автоматизованого тестового опитування є актуальною для переважної більшості закладів освіти. Сюди входять такі заклади як школи, коледжі та університети. Головною вимогою для використання є вміння використання комп’ютеру та програмного забезпечення хоча б на базовому рівні.

Задача даного курсового проекту розробити базу даних для навчального закладу, що має на меті автоматизувати процес оцінювання знань. В процесі розробки бази, потрібно передбачити ввід даних про викладачів, їхні ролі в навчальному закладі, студентів, групи студентів, предмети тестування, та тести. На даному етапі передбачається лише простий тип тестових питань, тобто, перемикачі або прапорці.

Основними завданнями бази даних є:

* Ввід даних про кожну сутність предметної області;
* Перегляд і коригування даних;
* Вилучення непотрібної або застарілої інформації;
* Фільтрація, відбір даних за певними критеріями в якості звітності.
  1. Виділення сутностей предметної області (неформальна модель ПО)

Вхідна інформація передбачає певні дані, які необхідні для опрацювання у проєкті. У цьому проєкті є два актора, які вносять вхідну інформацію – адміністратор та студент.

Вхідною інформацією адміністратора будуть виступати дані про предмети тестування, про тести, а також інформація про студентів та інших адміністраторів, якщо ми розглядаємо їх реєстрацію. Оскільки кожен студент повинен відноситись до якоїсь групи, то дані про групи студентів також відносяться до вхідної інформації. Кожна група, в свою чергу, відноситься до конкретного відділення. Тому вимальовується деревнопоідбна структура із відділень, груп студентів, та студентів.

Зі сторони ж студента, він проходить тести по певним предметам, тому вхідною інформацією виступає набір відповідей на тест.

Результуючу інформацію ми отримуємо після проходження студентами тестів. Після відправлення форми, студент бачить рівень своїх знань, відображений оцінкою п’ятибальної системи, а також відсоток правильних відповідей. Викладачі отримують більше інформації, аніж студенти. По-перше, вони можуть переглядати інформацію про проходження тестів усіма студентами. По-друге, є можливість генерування відомості по окремому результату проходження студента (переглянути які відповіді він давав, які помилки допустив), а також по групі (щоб побачити успішність, якість, середній бал та інші показники). Викладачі та можуть переглядати всі результати здачі тестів студентами, а для старости доступні лише результати його групи.

* 1. Приведення до третьої нормальної форми

Нормальна форма — властивість відношення в реляційній моделі даних, що характеризує його з точки зору надмірності, яка потенційно може призвести до логічно помилкових результатів вибірки або зміни даних. Нормальна форма визначається як сукупність вимог, яким має задовольняти відношення.

Перша нормальна форма характеризується тим, що кожна таблиця повинна мати основний ключ, тобто мінімальний набір колонок, які ідентифікують запис, запобігає повторенню груп (категорії даних, що можуть зустрічатись різну кількість разів в різних записах), а також передбачає атомарність кожного атрибута, що означає можливість мати лише одне значення, а не множину значень.

Друга нормальна форма розширює обмеження першої тим, що дані, які повторно з'являються в декількох рядках, повинні бути винесені в окремі таблиці, де вони стануть унікальними.

Третя нормальна форма доповнює другу нормальну форму тим, що будь-яке поле, залежне не лише від первинного ключа, а також від будь-якого іншого поля, має виноситись в окрему таблицю.

Інколи проектування бази даних може розпочинатись з уже ісунючої двовимірної таблиці, яка вже містить дані про усі сутності в предметній області, і проектуванням в даному випадку буде розбиття цієї таблиці до тих пір, поки усі схеми таблиць даних не будуть задовільняти умовам другої або третьої нормальної форми.

В рамках даної предметної області системи оцінювання знань студентів, ми ще не маємо жодної наявної структури даних, яку можна було б нормалізувати, тому ніщо не перешкоджає одразу проектувати таблиці таким чином, щоб вони задовільняли нормальним формам.

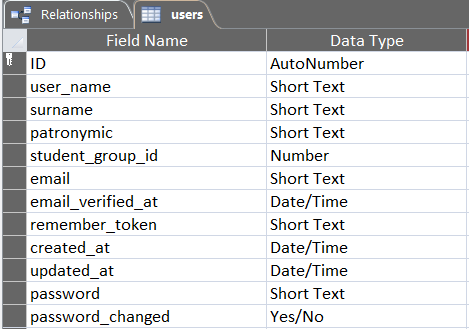


Рисунок 1 – Таблиця студентів

На рисунку 1 зображено приклад таблиці, що відповідає третій нормальній формі. У цій таблиці не міститься надлишкових даних (таких як назва групи, до прикладу, чи інформація про пройдені тести), а лише необхідний мінімум інформації, яка потрібна для користувача. Така інформація як ПІБ також міститься в окремих полях, що робить кожне з них окремо атомарним.

* 1. Побудова інформаційно-логічної моделі даних (концептуальна ER-діаграма БД)

В результаті проектування таблиць бази даних із врахуванням критеріїв трьох нормальних форм, отримуємо наступні схеми таблиць з відповідними полями.

Інформація про групи студентів зображена на рисунку 2.

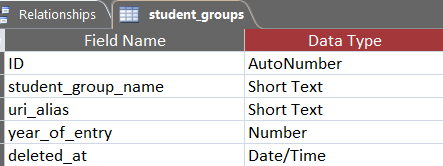


Рисунок 2 – Таблиця «student\_groups»

Інформація про студентів зображена на рисунку 3.

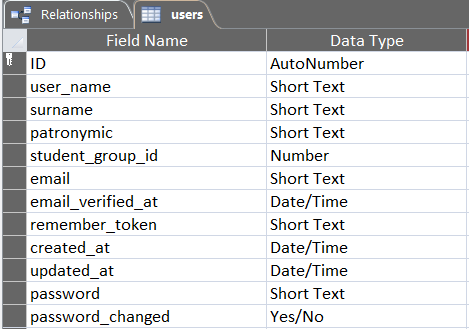


Рисунок 3 – Таблиця «users»

Інформація про предмети тестування зображена на рисунку 4.

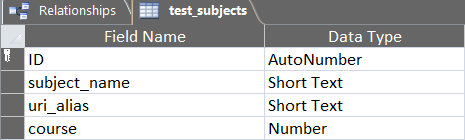


Рисунок 4 – Таблиця «test\_subjects»

Інформація про тести зображена на рисунку 5.

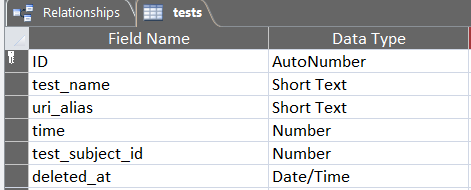


Рисунок 5 – Таблиця «tests»

Інформація про композитність тестів зображена на рисунку 6.

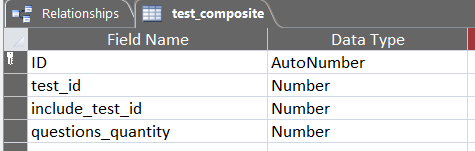


Рисунок 6 – Таблиця «test\_composite»

Інформація про питання тестів зображена на рисунку 7.

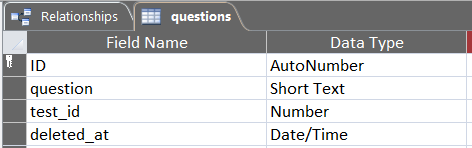


Рисунок 7 – Таблиця «questions»

Інформація про варіанти відповідей питань зображена на рисунку 8.

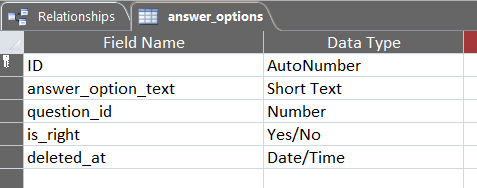


Рисунок 8 - Таблиця «answer\_options»

Інформація про результати тестування студентів зображена на рисунку 9.

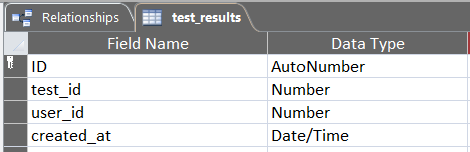


Рисунок 9 - Таблиця «test\_results»

Інформація про задані студентам питання зображена на рисунку 10.

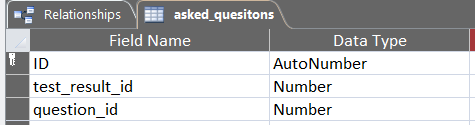


Рисунок 10 - Таблиця «asked\_questions»

Інформація про обрані студентами відповіді зображена на рисунку 11.

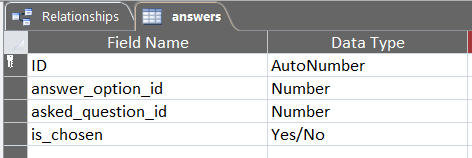


Рисунок 11 - Таблиця «answers»

Таким чином, об’єднавши всі таблиці зв’язками зовнішніх ключів в одну схему, можна отримати ER-діаграму, що зображена на рисунку 12, що покаже загальний вигляд структури таблиць бази даних, а також їх зв’язків між собою.

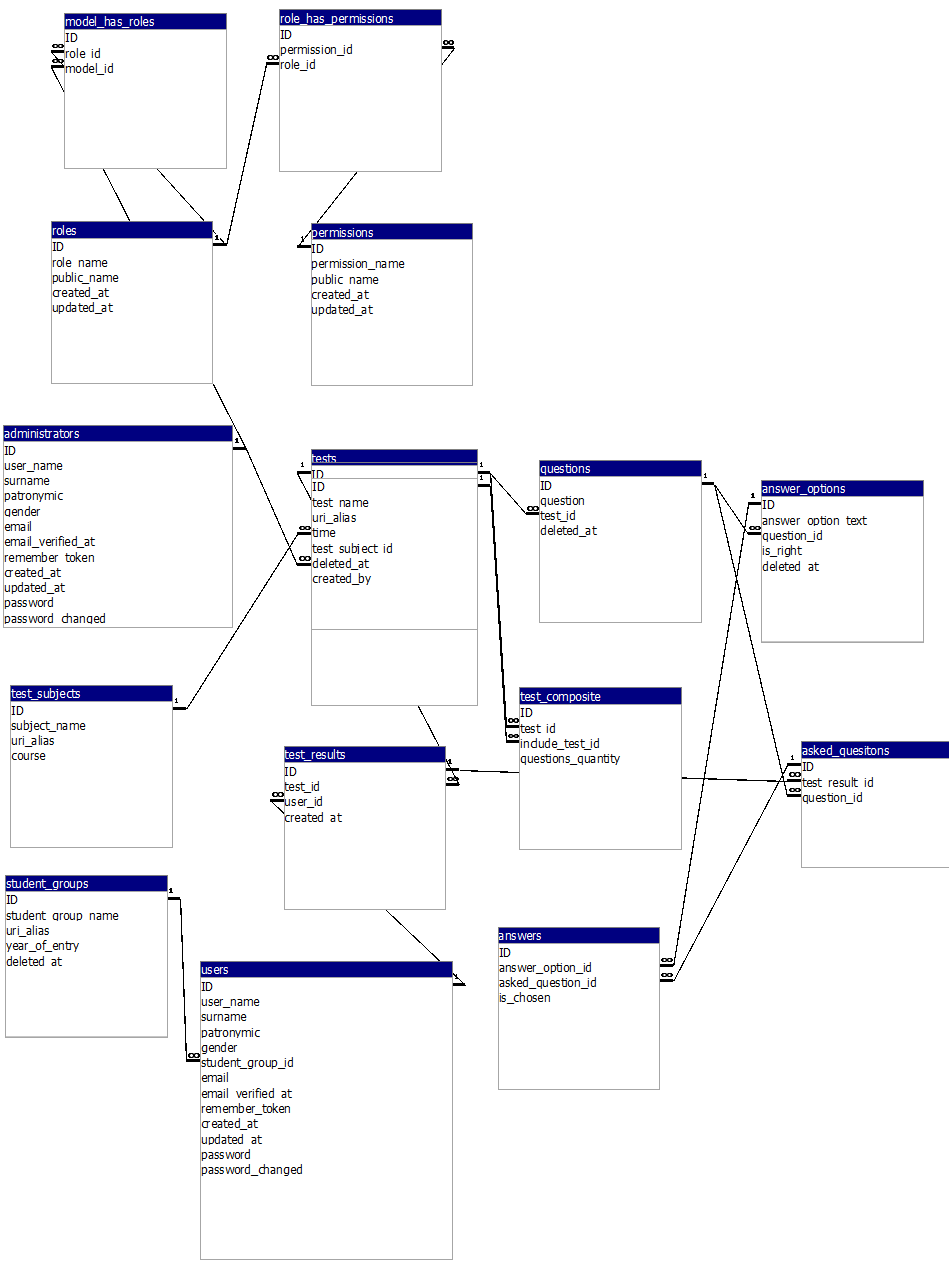


Рисунок 12 – ER-діаграма

1. Проектування та реалізація бази даних на фізичному рівні
   1. Вибір та обґрунтування СКБД

База даних – сукупність даних, організованих відповідно до концепції, яка описує характеристику цих даних і взаємозв'язки між їх елементами; ця сукупність підтримує щонайменше одну з областей застосування.

Для того, щоб використовувати її та зберігати великий обсяг інформації, ділитися нею з іншими користувачами потрібне спеціальне програмне забезпечення. Цим ПЗ являється система керування базами даних, так зване СКБД.

Microsoft Access (MS Access) – реляційна клієнт-серверна система управління базами даних (СУБД) від компанії Microsoft. Реляційна означає, що вона заснована на таблицях. Розглянемо цю систему детальніше.

MS Access відноситься до файл-серверних СУБД. Це означає, що вся обробка даних відбувається безпосередньо на комп'ютері клієнта, користувача. Серверна частина лише витягує дані і передає їх на вимогу. Це призводить до того, що по локальній мережі курсують великі потоки даних, і при великій кількості користувачів, що працюють з базою даних, що може трапитися перевантаження мережі. Класична файл-серверна архітектура завжди тягне за собою низьку продуктивність при великій кількості користувачів. Однак у нових версіях MS Access це питання було вирішено. Багатокористувацький режим можливий тільки в одноранговій мережі або в мережі з файловим сервером. Для розробки нестандартного інтерфейсу потрібно залучати програмістів. MS Access заточений під операційні системи Windows.

Access зберігає всі таблиці, запити, форми, звіти, макроси та модулі в БД у вигляді одного файлу. Однією з переваг є те, що в Access можна імпортувати та експортувати в інші програмні продукти від Microsoft та Google такі як Excel, Outlook, Paradox, SQL Server, Oracle та ін. Дана СКБД є простим у використанні інструментом для створення бізнес-додатків із шаблонів або з нуля. Завдяки багатим та інтуїтивно зрозумілим інструментам проектування Access може допомогти вам створити привабливі та високофункціональні програми за мінімальний час.

Приступити до роботи з даною СУБД можна, не маючи ніяких особливих навичок. Крім того, на допомогу новачкові приходять численні довідкові посібники, онлайн - та офлайн-курси. СУБД MS Access загальновідома і дуже популярна в настільних ПК. Це означає, що ви завжди можете звернутися за допомогою до знайомого, вже давно використовує дану СУБД, за радою, а також разом працювати з однією базою даних по мережі.

Так як MS Access чудово підходить для початківців, було обрано саме цю СКБД як основу для збереження даних. Microsoft Access володіє усіма необхідними функціями для створення та заповнення БД, написання запитів, побудови форм та звітів, тобто усе що мінімально потрібне для повноцінної роботи.

* 1. Створення та заповнення бази даних. Фізична ER-діаграма бази даних

Вибравши систему керування базами даних, можна розпочати створення таблиць бази даних. Для кожної сутності, що була описана раніше, потрібно буде створити одну або більше таблиць.

Для створення та заповнення таблиць можна використовувати стандартний графічний інтерфейс, що надається MS Access, або використовувати SQL-консоль. В другому випадку, ми маємо повний контроль над тим, що буде виконано в СКБД, і це дає нам гнучкість в об’явлені структури даних. В рамках розробки бази даних нашої предметної області, будемо використовувати SQL-скрипти.

Розпочнемо створення таблиць на прикладі таблиці administrators. Зробимо поле id первинним ключом та задамо йому авто інкремент на 1 починаючи зі значення 1, як це зображено на рисунку13.

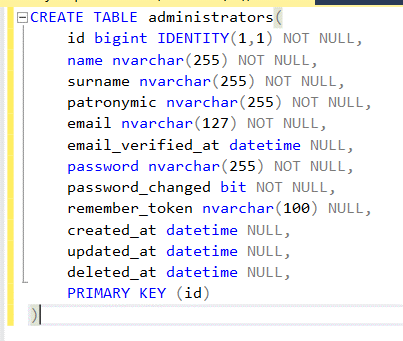


Рисунок 13 – створення таблиці адміністраторів

Можемо перейти в базу даних та побачити в списку новостворену таблицю, і перейшовши в схему цієї таблиці, побачимо ті параметри, які ми задали в запиті SQL як це показано на рисунку 14.

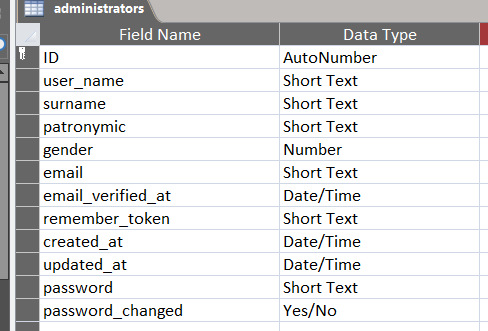


Рисунок 14 – схема таблиці адміністраторів

Аналогічним чином створюємо інші таблиці.

Створення таблиці tests зображено на рисунку 15.

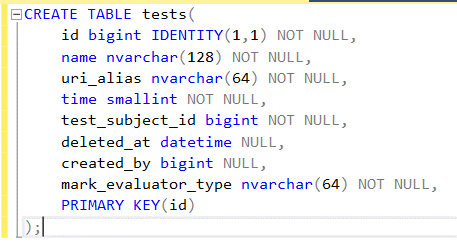


Рисунок 15 – створення таблиці tests

Створення таблиці test\_composite зображено на рисунку 16.

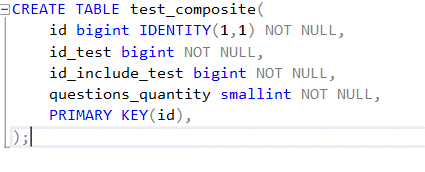


Рисунок 16 – створення таблиці test\_composite

Створення таблиці test\_subjects зображено на рисунку 17.

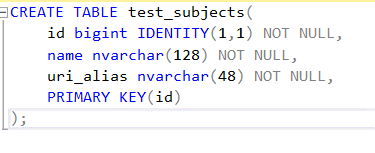


Рисунок 17 – створення таблиці test\_subjects

Створення таблиці courses зображено на рисунку 18.

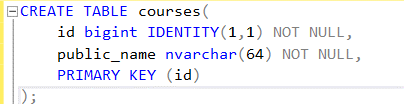


Рисунок 18 – створення таблиці courses

Створюємо таблицю course\_test\_subject для того, щоб зв’язати таблиці courses та test\_subjects. Тут немає поля id, але є складений первинний ключ (course\_id та test\_subject\_id унікально ідентифікують запис). Створення таблиці зображено на рисунку 19.

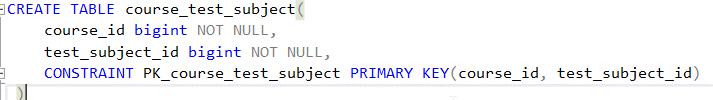


Рисунок 19 – створення таблиці course\_test\_subject

Створення таблиці questions зображено на рисунку 20.

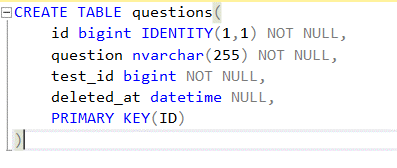


Рисунок 20 – створення таблиці questions

Створення таблиці answer\_options зображено на рисунку 21.

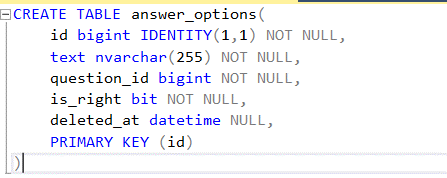


Рисунок 21 – створення таблиці answer\_options

Створення таблиці departments зображено на рисунку 22.

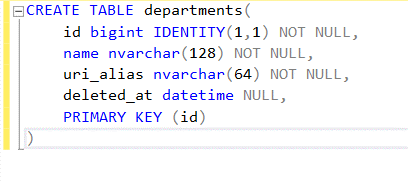


Рисунок 22 – створення таблиці departments

Створюємо таблицю department\_test\_subject для того, щоб зв’язати таблиці departments та test\_subjects. Тут немає поля id, але є складений первинний ключ (department\_id та test\_subject\_id унікально ідентифікують запис). Створення таблиці зображено на рисунку 23.

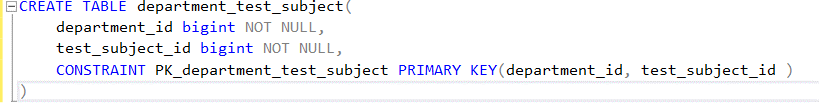


Рисунок 23 – створення таблиці department\_test\_subject

Створення таблиці student\_groups зображено на рисунку 24.

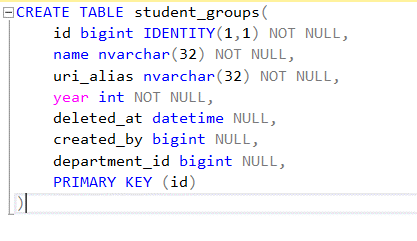


Рисунок 24 – створення таблиці адміністраторів

Створення таблиці користувачів (студентів) зображено на рисунку 25.

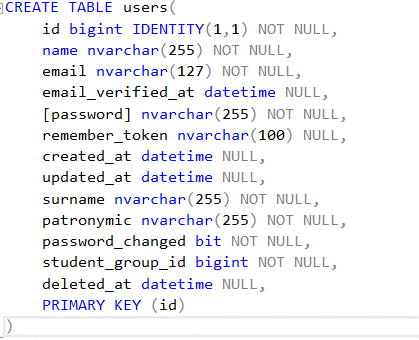


Рисунок 25 – створення таблиці users

Створення таблиці test\_results зображено на рисунку 26.

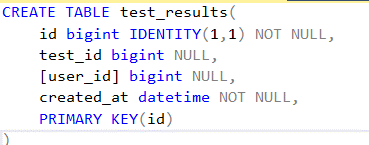


Рисунок 26 – створення таблиці test\_results

Створення таблиці asked\_questions зображено на рисунку 27.

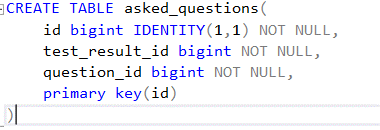


Рисунок 27 – створення таблиці asked\_questions

Створення таблиці answers зображено на рисунку 28.

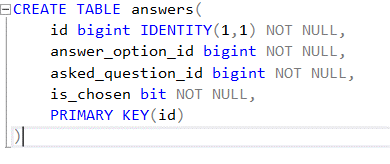


Рисунок 28 – створення таблиці answers

Далі, коли вже створено таблиці бази даних, можемо перейти до оголошення зовнішніх ключів. В таблиці tests поле created\_by посилається на запис з таблиці administrators. Опис foreign key для цього зв’язку зображено на рисунку 29.

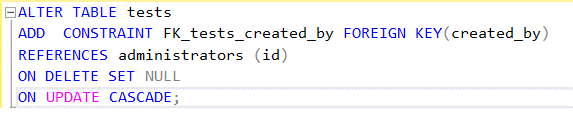
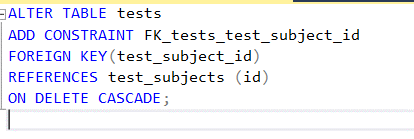


Рисунок 29 – опис foreign key для created\_by

В таблиці tests поле test\_subject\_id посилається на запис з таблиці test\_subjects. Опис foreign key для цього зв’язку зображено на рисунку 30.

  
Рисунок 30 – опис foreign key для test\_subject\_id

Таблиця test\_composite має 2 посилання на tests. Обидва зовнішніх ключа містять no action. Це через те, що при виставленні параметра cascade, неможливе додання зовнішнього ключа. Оголошення ключів зображено на рисунку 31.

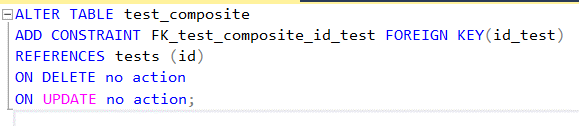
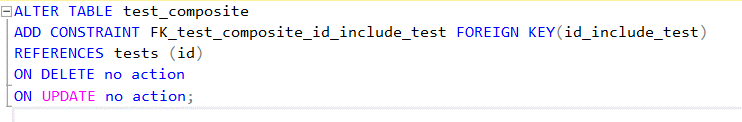


Рисунок 31 – опис foreign key для таблиці test\_composite

Аналогічно робимо для проміжної таблиці course\_test\_subject, зображено на рисунку 32.

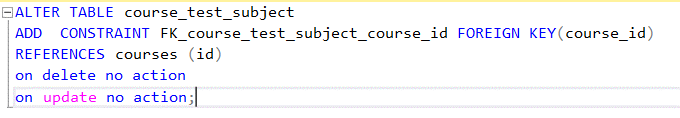
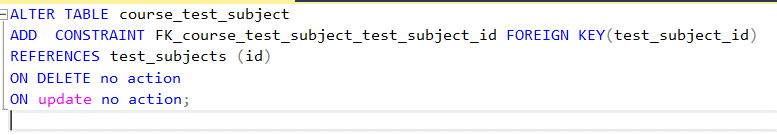


Рисунок 32 – опис foreign key для таблиці course\_test\_subject

А також, для department\_test\_subject, зображено на рисунку 33.

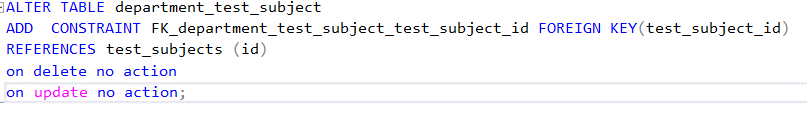
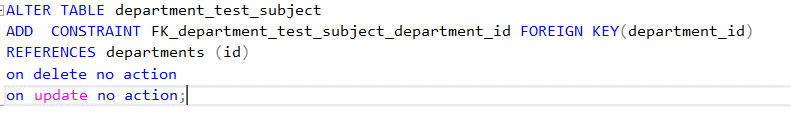


Рисунок 33 – опис foreign key для таблиці department\_test\_subject

Далі, оголосимо зовнішні ключі для відношень між departments, student\_groups, users. Це зображено на рисунку 34.

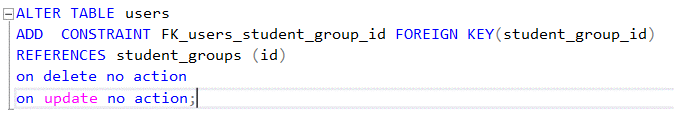
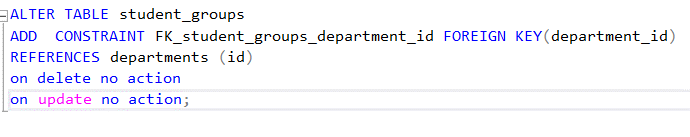


Рисунок 34 – опис foreign keys між departments, student\_groups, users

Пов’яжемо таблиці, що призначені для збереження всієї інформації про тести. Це зображено на рисунку 35.

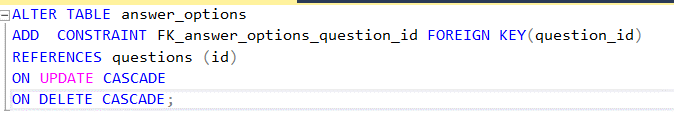
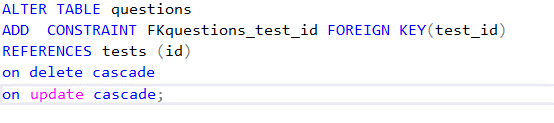


Рисунок 35 – опис foreign keys між tests, quesitons, answer\_options

А також, зв’язки між таблицями збереження результатів. Це зображено на рисунку 36.

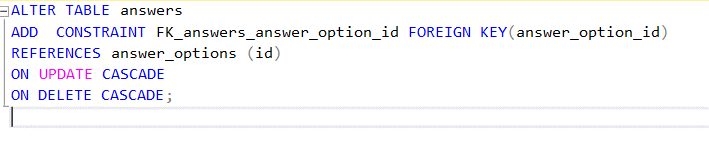
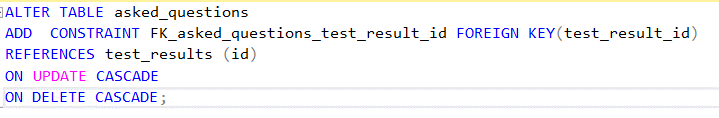
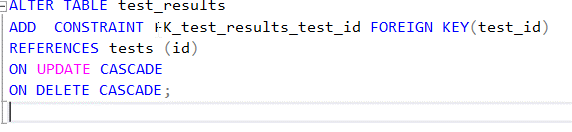


Рисунок 36 – опис foreign keys між таблицями збереження результатів

Тепер, коли створено таблиці та встановлено між ними зв’язки, можна перейти до заповнення їх даними.

Вставка записів в таблицю адміністраторів зображена на рисунку 37.

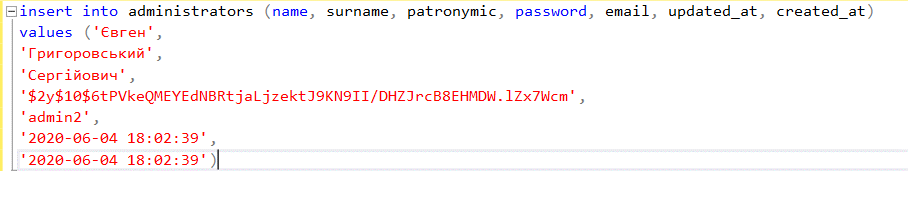


Рисунок 37 – вставка записів в таблицю адміністраторів

Вставка записів в таблицю відділень зображена на рисунку 38.

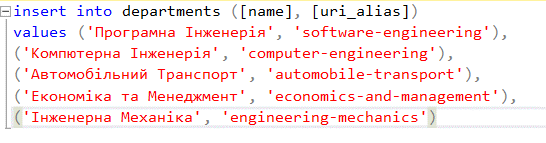


Рисунок 38 – вставка записів в таблицю відділень

Вставка записів в таблицю груп зображена на рисунку 39.

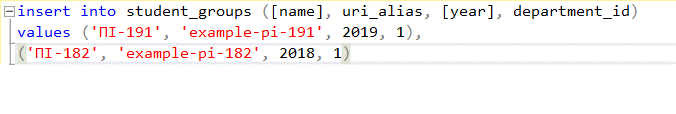


Рисунок 39 – вставка записів в таблицю груп

В інші таблиці бази даних, записи додаємо аналогічним чином.

Таким чином, об’єднавши всі таблиці зв’язками зовнішніх ключів в одну схему, можна отримати ER-діаграму, що зображена на рисунку 40, яка покаже загальний вигляд структури таблиць бази даних, а також їх зв’язків між собою.

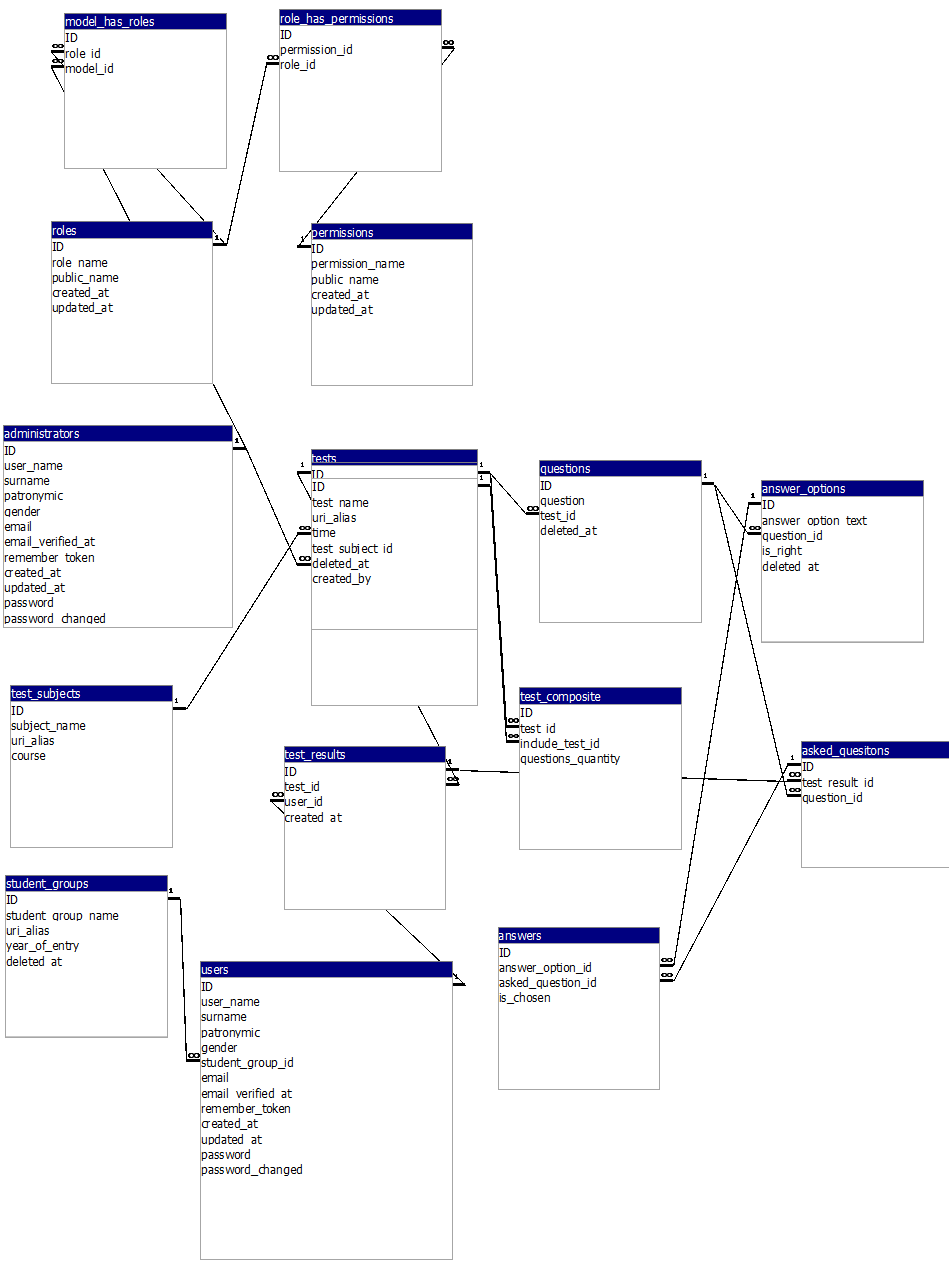


Рисунок 40 – ER-діаграма

1. Реалізація БД та розробка програмного забезпечення з її ведення
   1. Вибір засобів розробки інтерфейсу програмного забезпечення

Створивши та заповнивши таблиці бази даних, можна приступити до створення графічного інтерфейсу для зручного перегляду та заповнення даних. Однак перш, ніж це зробити, потрібно оприділитись з засобами, за допомогою яких буде розроблятись інтерфейс. Найпростішим варіантом в контексті MS Acces будуть вбудований функціонал форм.

Форма Access – це об’єкт бази даних, за допомогою якого можна створити інтерфейс користувача для програми бази даних. Форми полегшують перегляд або отримання потрібних елементів. Оскільки форми  це об’єкти, за допомогою яких користувачі можуть додавати, редагувати або відображати дані, що зберігаються в базі даних настільного комп’ютера Access, дизайн форми є важливим аспектом. Якщо база даних Access буде використовуватися кількома користувачами, добре розроблені форми мають важливе значення для ефективності та точності введення даних.

СКБД Access підтримує прості, композитні та зв’язані форми. Це такі форми, що безпосередньо підключені до певного джерела даних, як-от таблиці чи запиту, за допомогою якої можна вводити, редагувати або відображати дані з цього джерела даних. Вільні форми не ведуть прямо до джерела даних, але все одно містять кнопки, підписи або інші елементи керування, що потрібні для роботи з програмою.

* 1. Створення інтерфейсу для роботи з базою даних

Форми в Access – це як вітрини в магазинах, які полегшують перегляд або отримання потрібних елементів. Створення графічного інтерфейсу бази даних відбувається за допомогою головної кнопкової форми та підлеглих форм, на які буде вести головна. Головна форма представляє собою щось схоже на меню. Кожен пункт перенаправляє на інші графічні елементи такі як форми, звіти, чи навіть прямий доступ до таблиці.

Розпочнемо з створення простих форм, які потім будуть використані для компонування головної кнопкової форми. Створення простих форм може відбуватись різними способами. Однак всі дії зі створення форм виконуються за допомогою інструментів, що знаходяться на вкладці меню «Create», що зображене на рисунку 41.

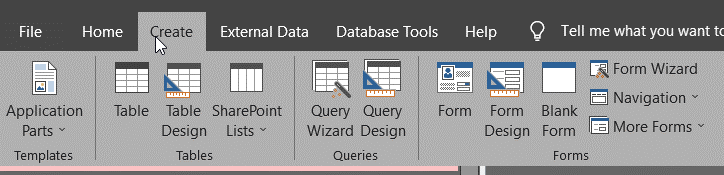


Рисунок 41 – Вкладка меню Create

Створимо просту форму для таблиці предметів тестування, як зображено на рисунку 42.

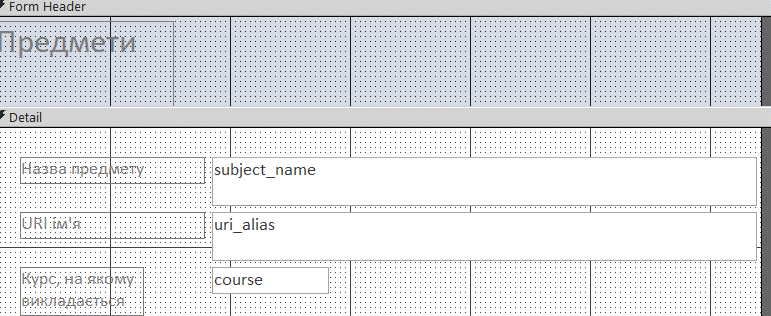


Рисунок 42 – Форма для таблиці предметів тестування

В режимі заповнення форма має вигляд, як зображено на рисунку 43.

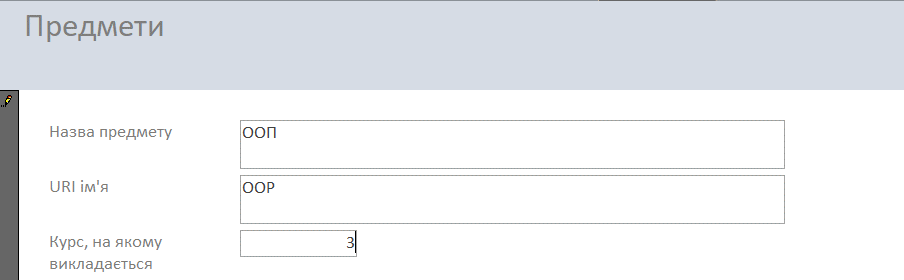


Рисунок 43 – Форма для таблиці предметів тестування

Створимо просту форму для таблиці груп студентів, як зображено на рисунку 44.

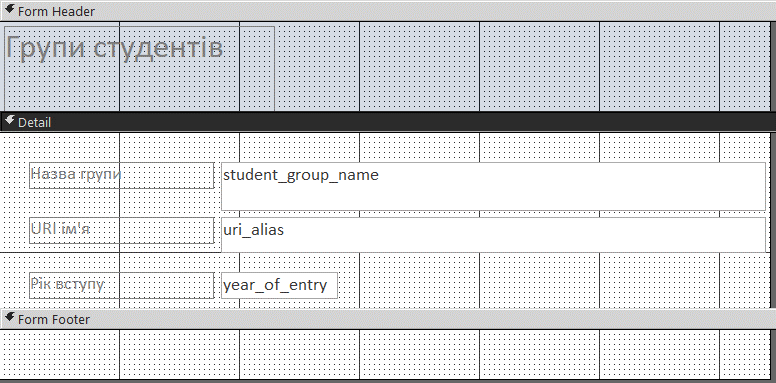


Рисунок 44 – Форма для таблиці груп студентів

В режимі заповнення форма має вигляд, як зображено на рисунку 45.

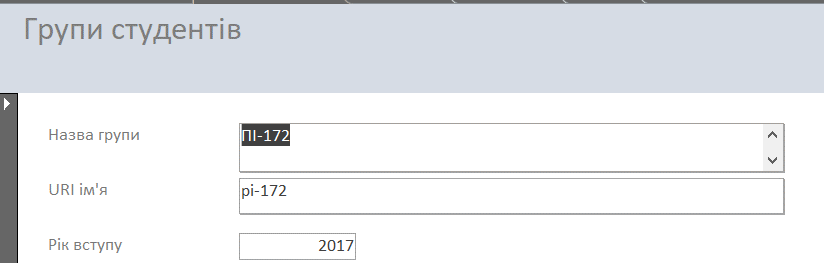


Рисунок 45 – Форма для таблиці груп студентів

Створимо просту форму для таблиці варіантів відповідей, як зображено на рисунку 46.

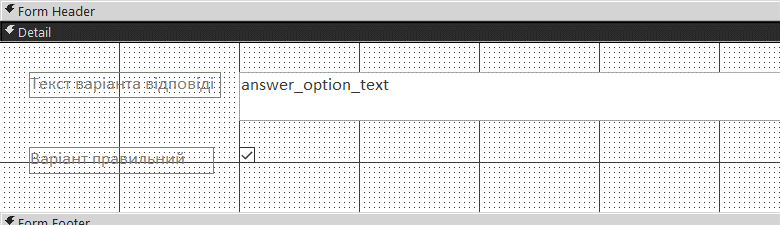


Рисунок 46 – Створення форми для варіантів відповідей

В режимі заповнення форма має вигляд, як зображено на рисунку 47.

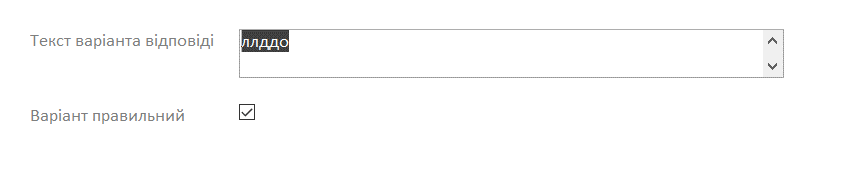


Рисунок 47 – Форма для варіантів відповідей

Альтернативний вигляд форми заповнення варіантів відповідей зображено на рисунку 48.

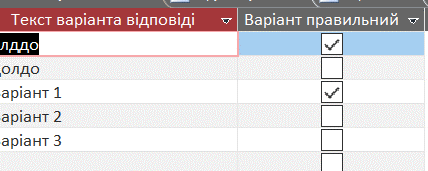


Рисунок 48 – Форма варіантів відповідей, колонковий вигляд

Створимо складну форму для таблиці запитань, яка буде містити підлеглу форму варіантів відповідей. Створення зображено на рисунку 49.

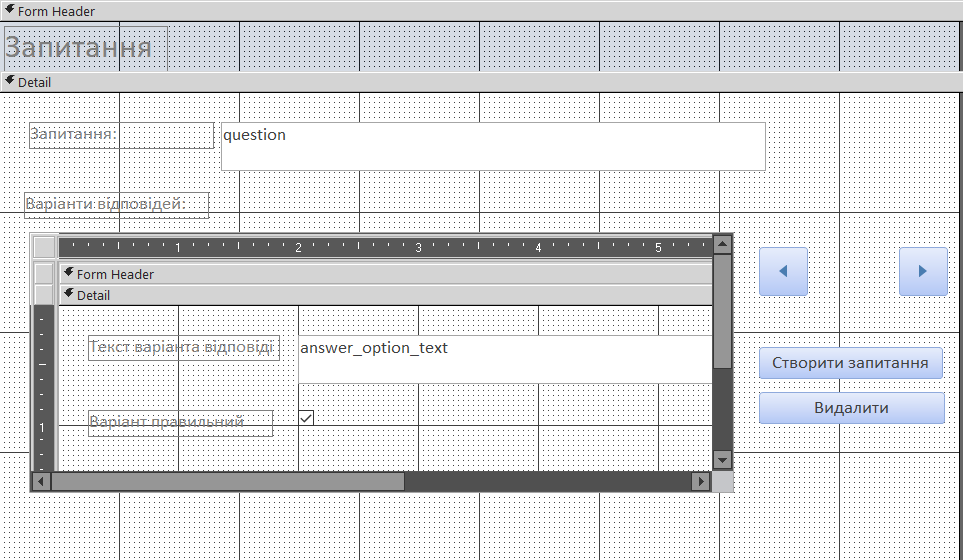


Рисунок 49 – Форма для створення запитань

В режимі заповнення форма має вигляд, як зображено на рисунку 50. За допомогою елементів навігації є можливість переміщуватись по існуючих записах, одразу перейти до створення нового, або видалити поточний запис, який зараз відображається.

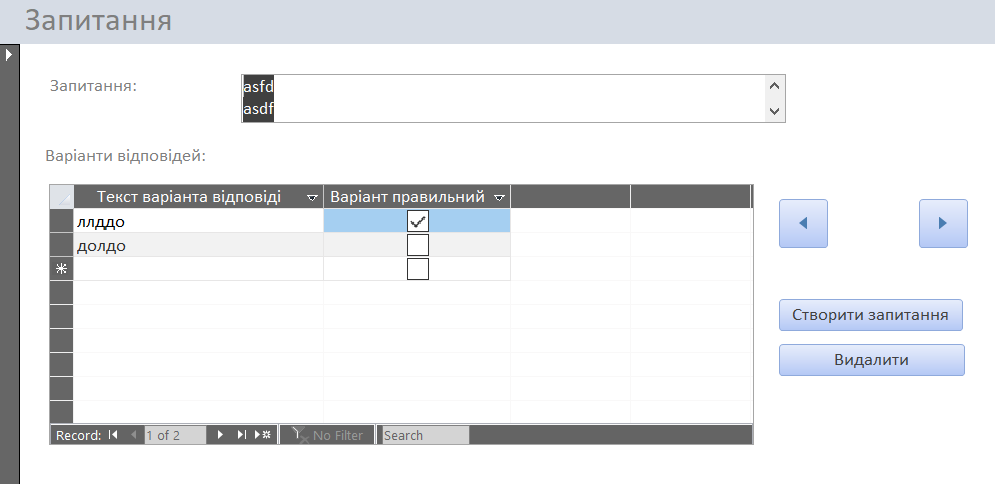


Рисунок 50 – Форма для заповнення запитань

Що ж стосується варіантів відповідей, то вони редагуються одразу із запитанням, тобто можна створити всі варіанти одразу, а не по окремості.

Створимо складну форму для таблиці Тестів, яка буде містити дві вкладені одна в одну форми, як це зображено на рисунку 51.

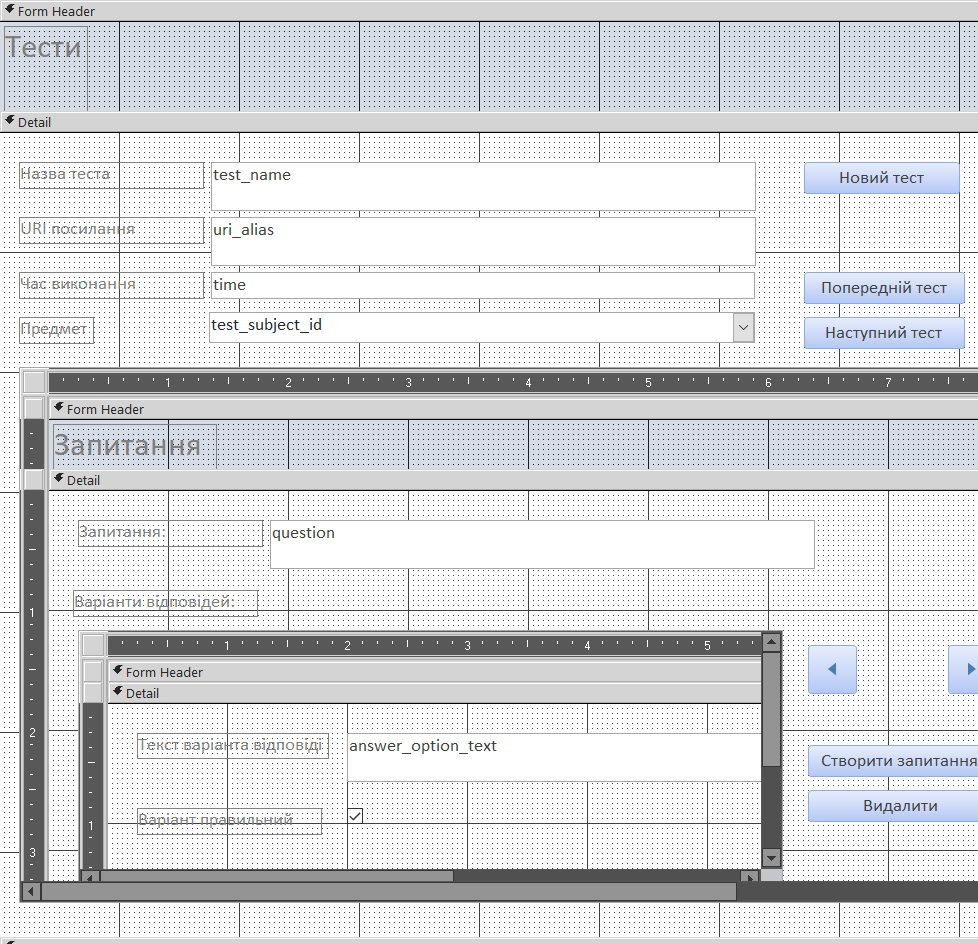


Рисунок 51 – Форма для управління тестами

Створення подібних вкладених форм дає можливість компонувати форми таким чином, як потрібно. Що ж стосується даних, що відображаються в підформах, то для кожної з них потрібно налаштувати, щоб вибірка йшла по зв’язку один до багатьох зі сторони форми, що містить вкладену. Якщо цього не зробити, тоді вкладена форма відобразить всі записи, що існують в таблиці, а це не те, чого ми хочемо.

В режимі заповнення форма має вигляд як показано на рисунку 52.

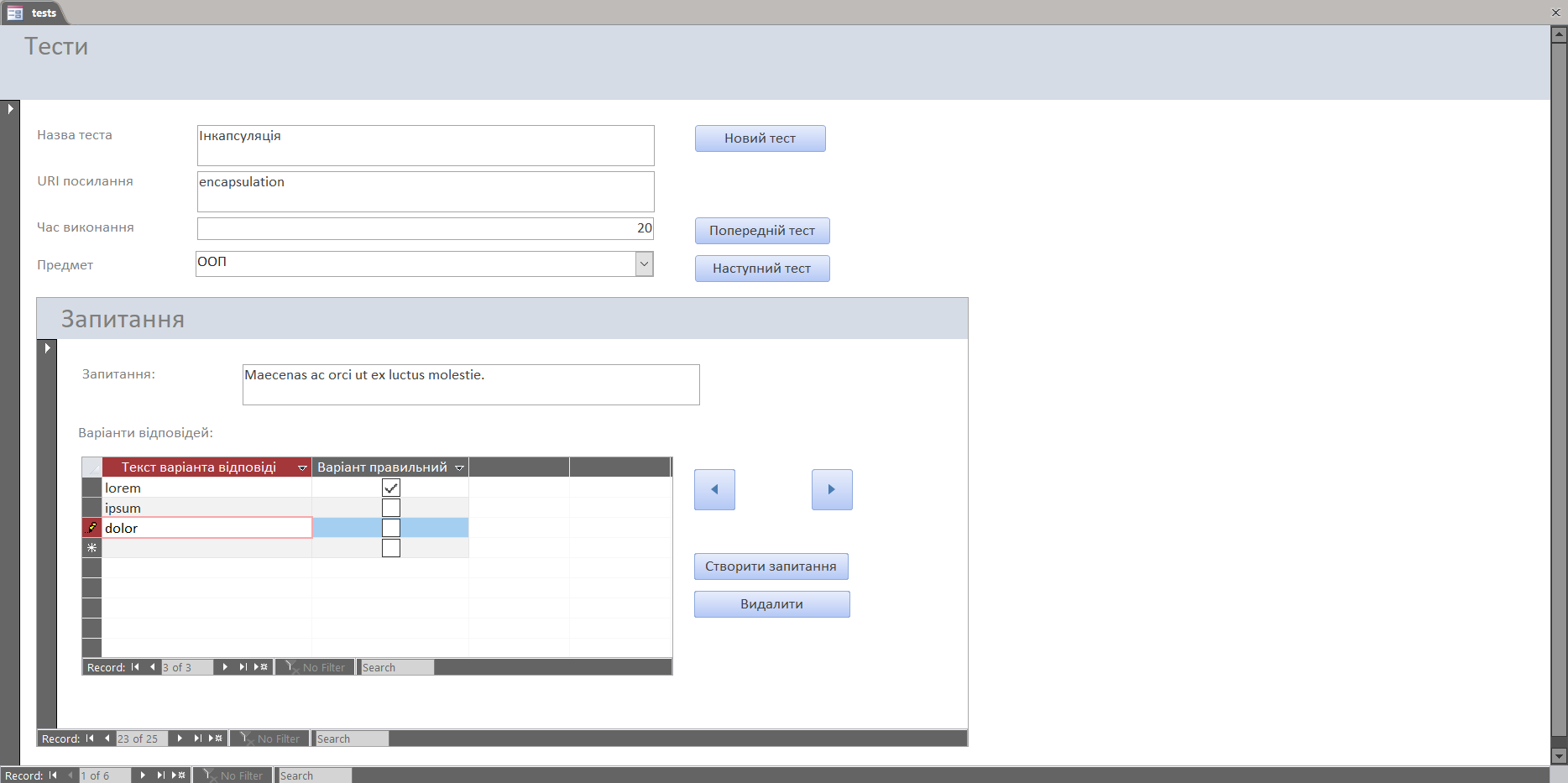


Рисунок 52 – Форма для управління тестами

Також зробимо головну вкладочну форму, кожна вкладка якої буде містити одну форму для заповнення (групи студентів, студенти, тести). Режим конструктора форми зображено на рисунку 53.

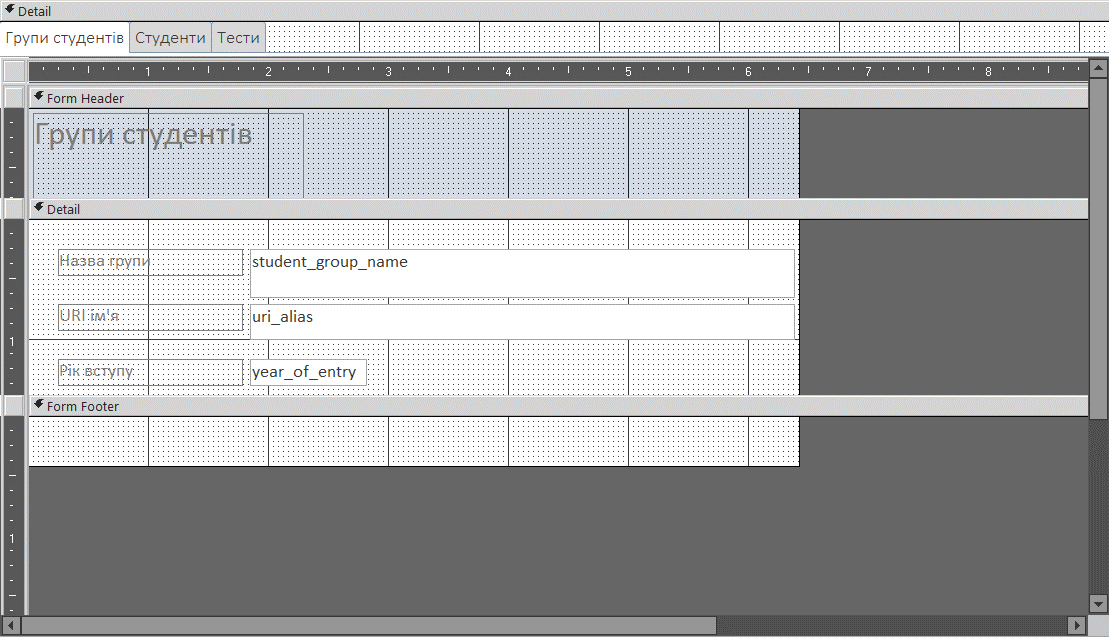


Рисунок 53 – Вкладочна форма для груп, студентів, тестів

Вкладка для заповнення інформації про студентів зображена на рисунку 54.

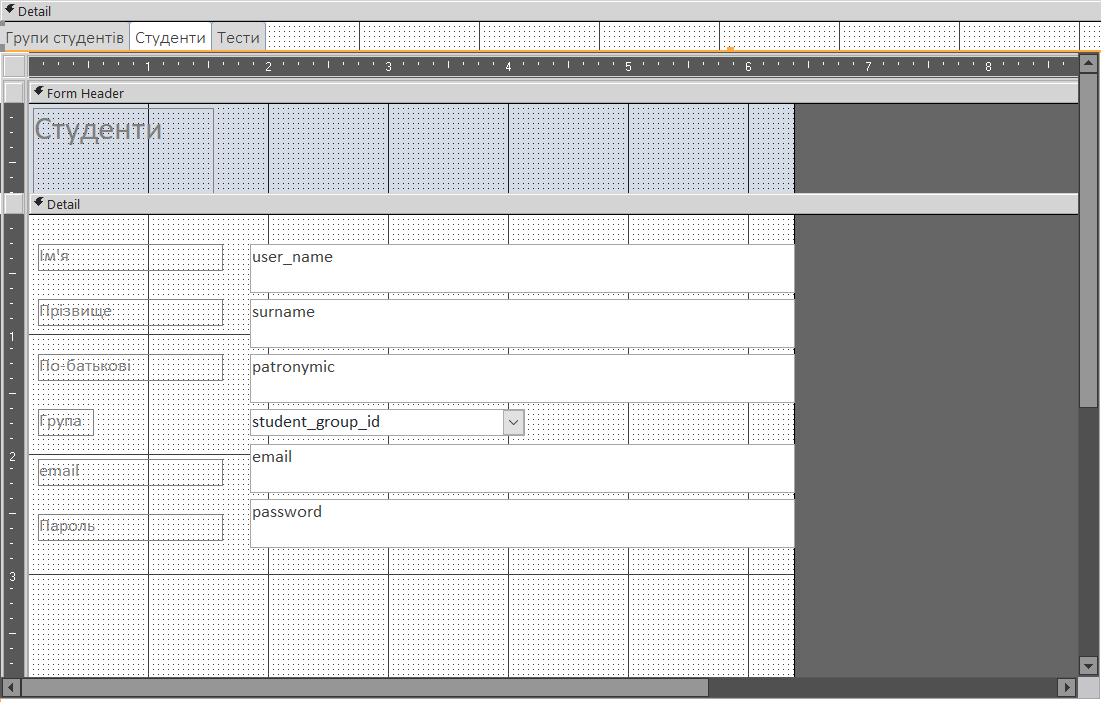


Рисунок 54 – Вкладочна форма, вкладка «студенти»

Вкладка тестів зображена на рисунку 55. Сюди підключена форма для заповнення тестів, що була створена раніше.

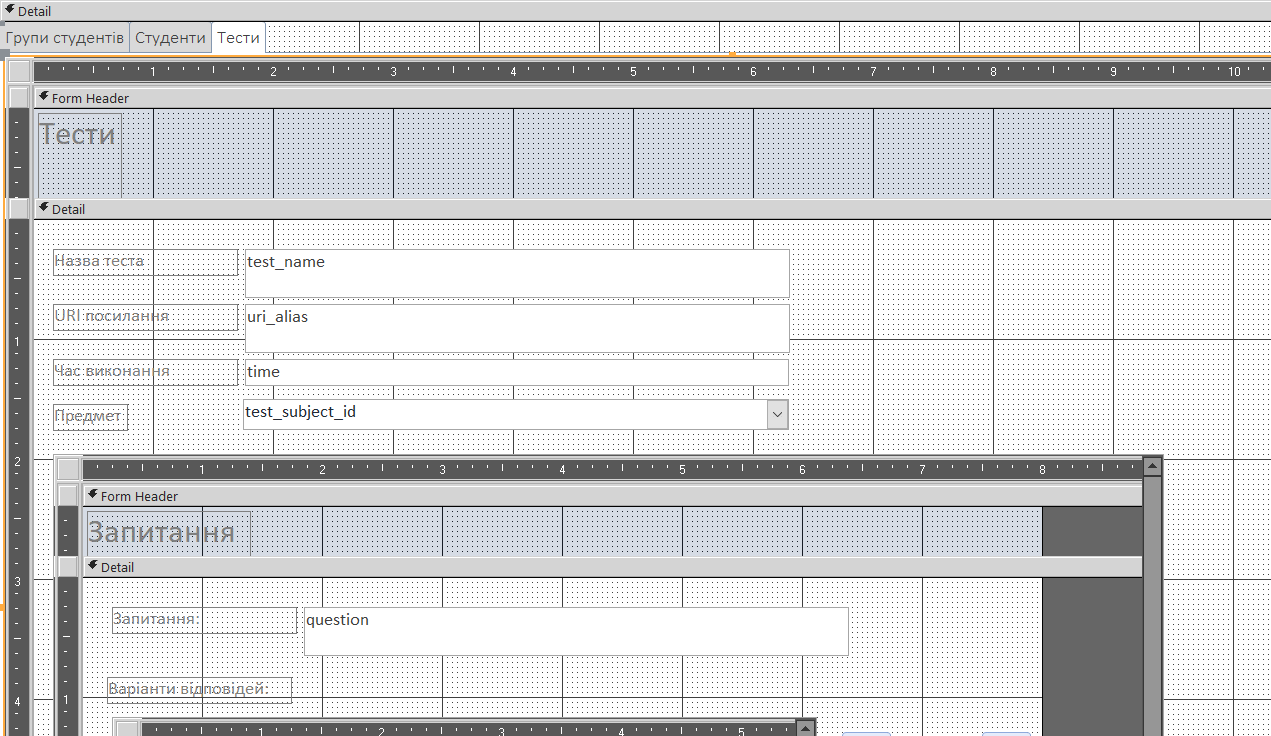


Рисунок 55 – Вкладочна форма, вкладка «тести»

На вкладці «студенти» для вибору групи було реалізовано випадаючий списко із всіма доступними групами, який зображено на рисунку 56.

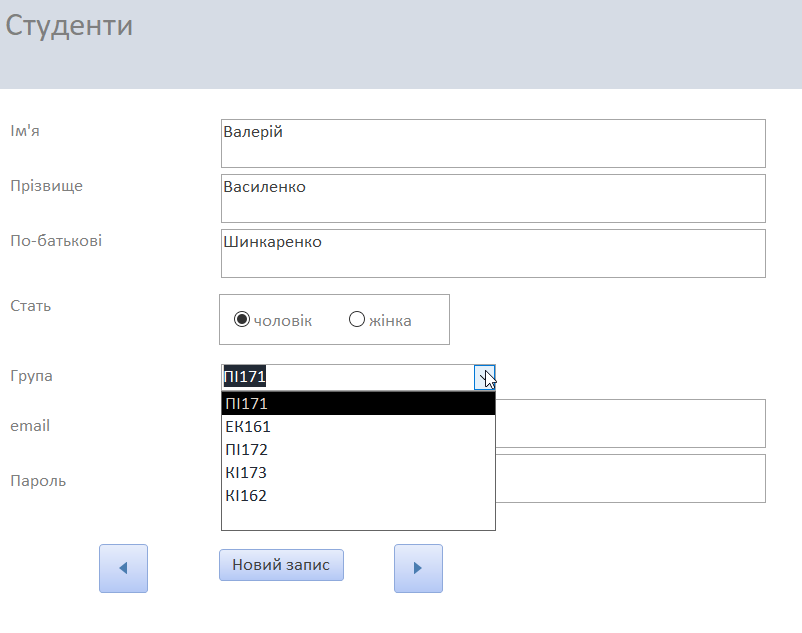


Рисунок 56 – Випадаючий список із групами

На вкладці «групи студентів» було реалізовано обрахункове поле для відображенн курсу на основі року вступу. Це зображено на рисунку 57.

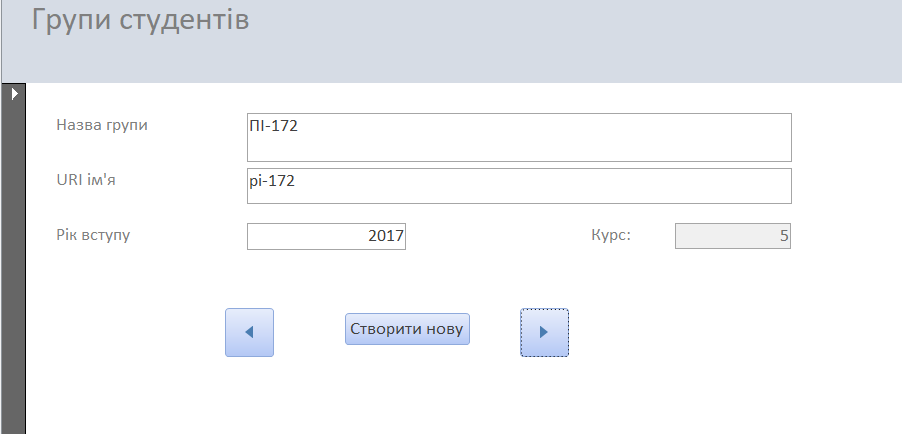


Рисунок 57 – Обрахункове поле для курсу

Для заповнення інформації для administrators - roles, створимо 3 форми: одна для заповнення інформації про адміністраторів, інша – про ролі, а третя – для встановлення зв’язку між ними.

Форма викладачів виглядає наступним чином як на рисунку 58.

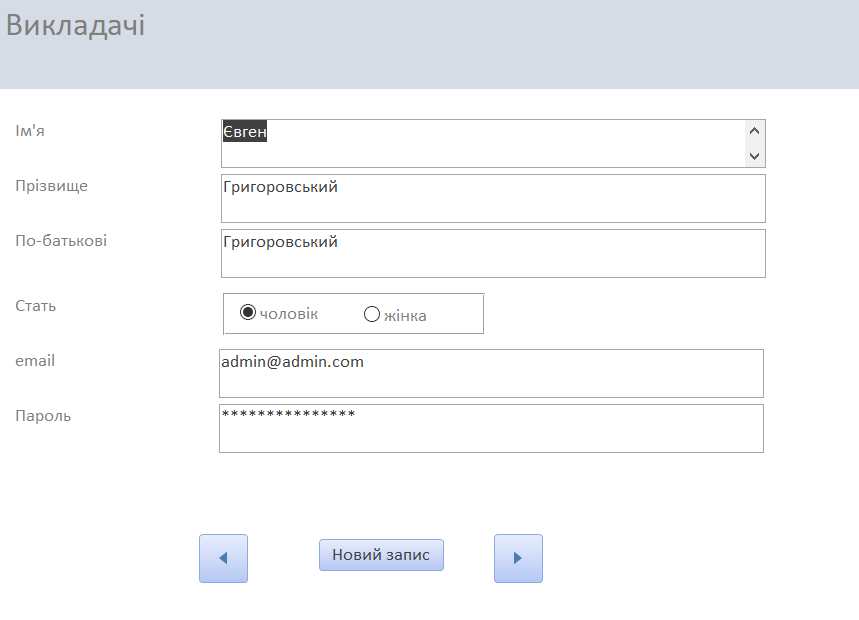


Рисунок 58 – Форма викладачів

Форма ролей виглядає наступним чином як на рисунку 59.

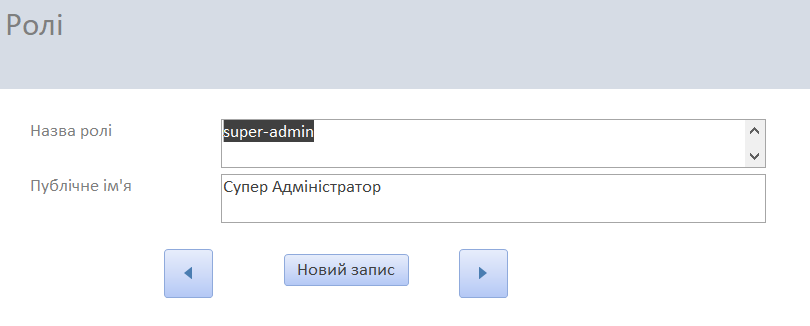


Рисунок 59 – Форма ролей

Далі створимо форму для заповнення проміжної таблиці між ними як прості поля з ідентифікаторами, це зображено на рисунку 60.

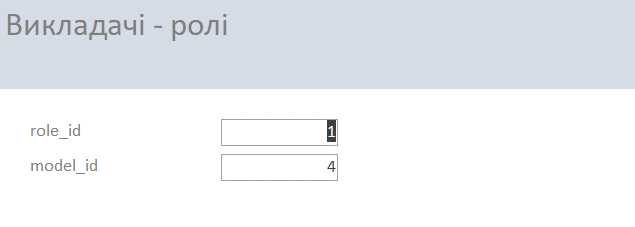


Рисунок 60 – Форма вилкадачі-ролі

Однак незручно завжди пам’ятати індекси ролей та викладачів, тому перейдемо в режим конструктора та виправимо це. Обираємо поля по черзі, та за допомогою елемента керування «Combo box» робимо їх більш зручними для заповнення. Але тут портібен запит в іншу таблицю, тому вказуємо всю цю інформацію про те які данні повинні бути відображені, а які просто збережені. Послідовність дій зображена на рисунках 61, 62, 63.

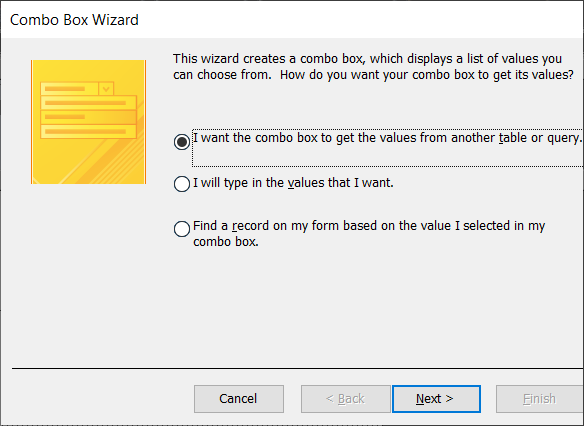


Рисунок 61 – Вибір даних з іншої таблиці

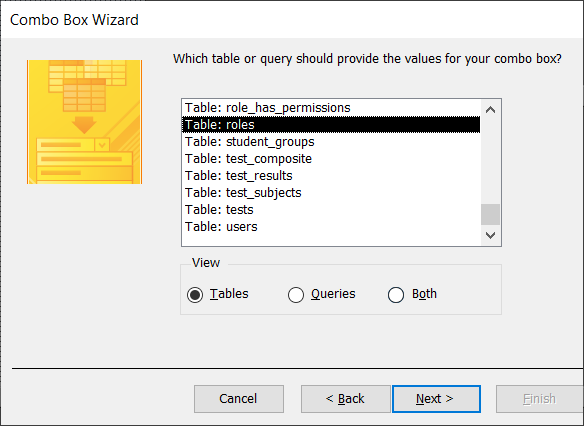


Рисунок 62 – Налаштовуємо з якої таблиці брати дані

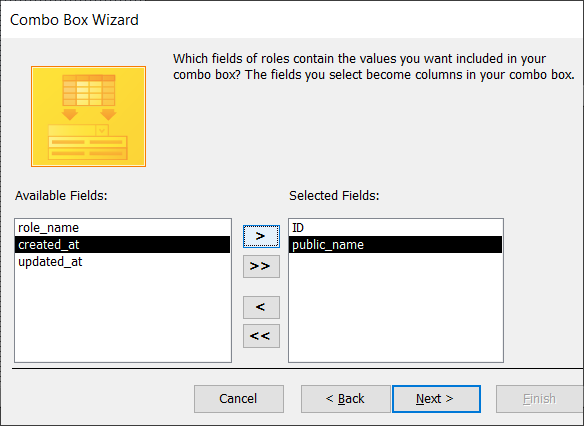


Рисунок 63 – Обираємо потрібні поля

Провівши ці маніпуляції, можемо пересвідчитись, що все працює та зробити аналогічно для другого поля. Результат зображено на рисунку 64.

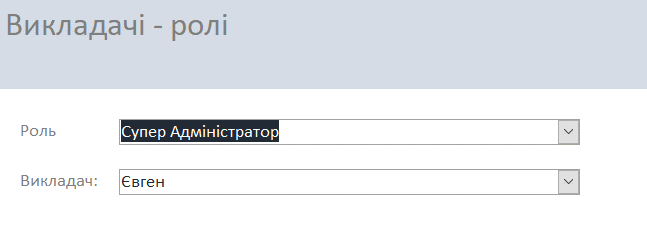


Рисунок 64 – Кінцева форма викладачі-ролі

Для того, щоб переглядати і редагувати усю інформацію з предметної області, створимо головну форму, що буде мати вкладки, які будуть містити всі потрібні форми. Для цього, створюємо нову форму, натиснувши кнопку «Form Design», як показано на рисунку 65.

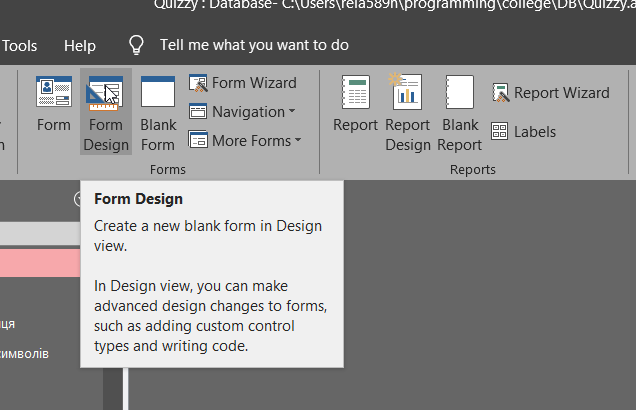


Рисунок 65 – Дизайнер форм

За допомогою компоненту “Tab control”, створюємо вкладки, кожна з яких буде підключати форму для редагування якоїсь конкретної частини сутностей предметної області. Кінцевий вигляд в конструкторі зображено на рисунку 66.

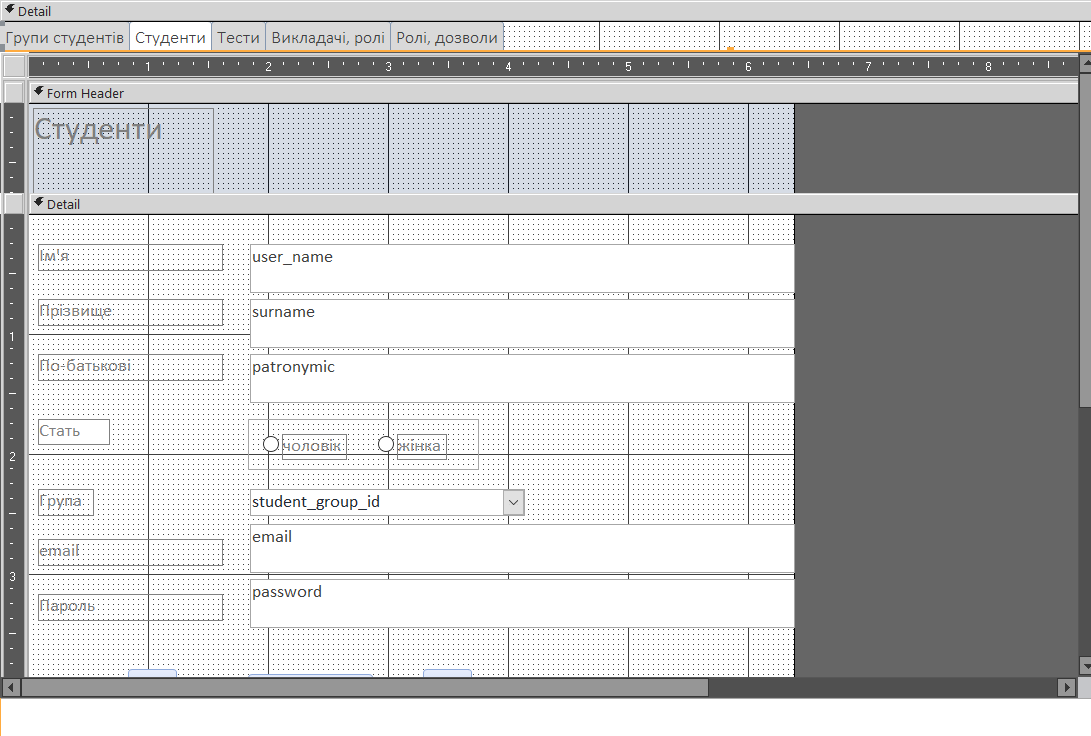


Рисунок 66 – Головна вкладкова форма

Вигляд форми в режимі заповення має вигляд як зображено на рисунку 67.

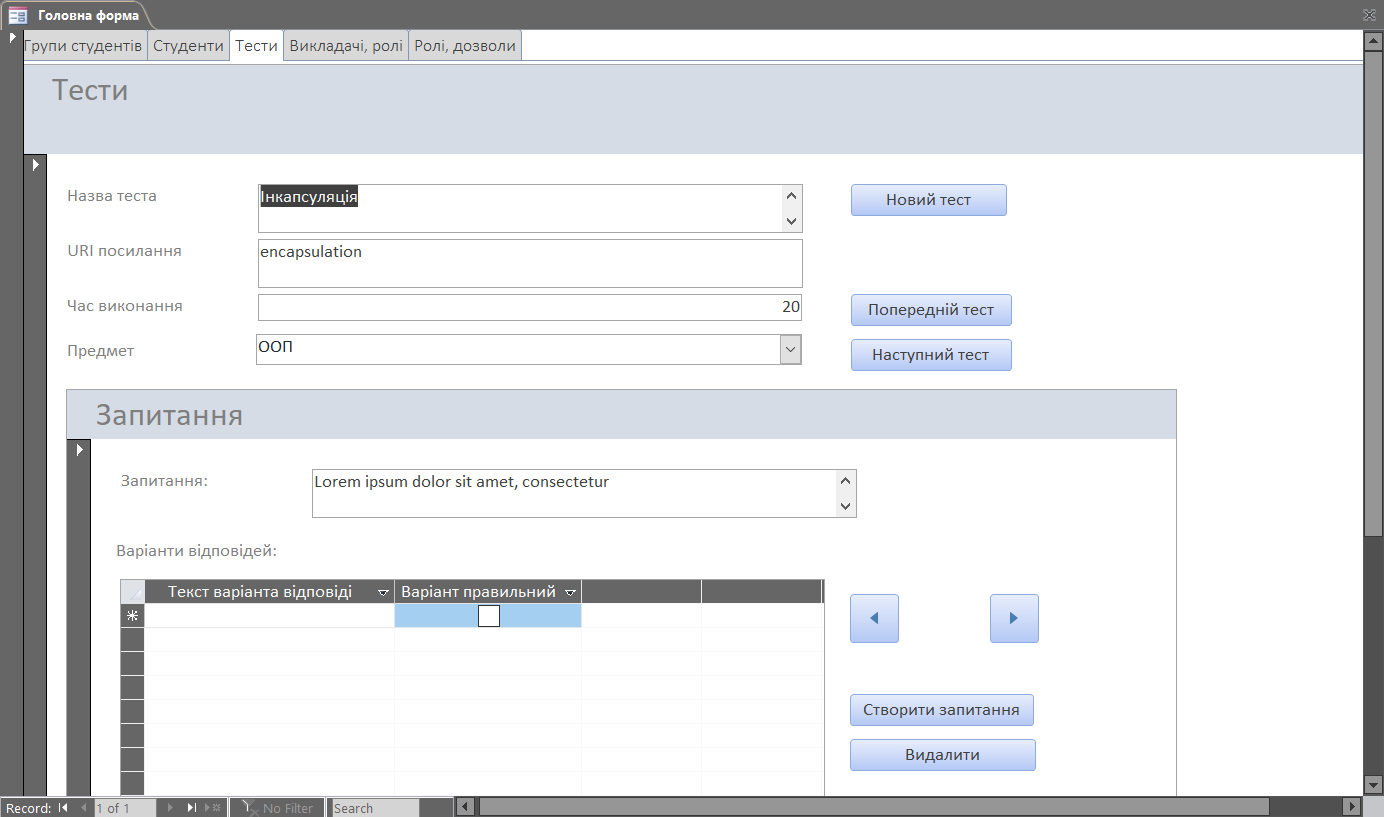


Рисунок 67 – Заповнення з гловної кнопкової форми

* 1. Інструкція користувача

Спочатку нам потрібно запустити базу даних. Після запуску ми можемо побачити головне меню, яке зображене на рисунку 68. Це меню міститься 5 кнопок керування, за допомогою яких можна перейти в різні розділи, такі як до головної форми, до таблиць, необхідних запитів та звітів а токож вийти з бази даних.

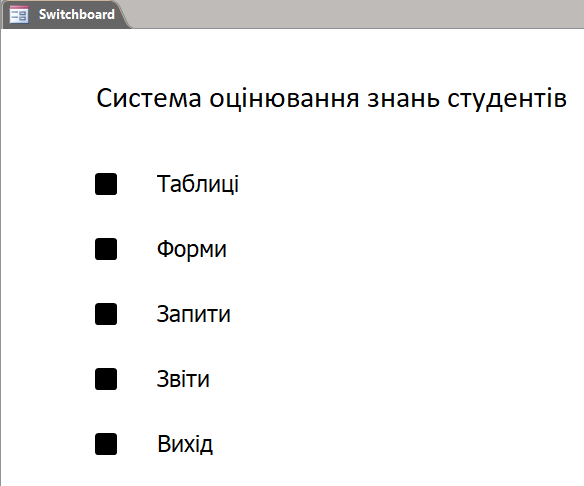


Рисунок 68 – Головне меню користувача

Натиснувши на пункт меню «Таблиці», відкривається вікно зі списком таблиць, доступних для переходу та для редагування даних в них. Це зображено на рисунку 69.

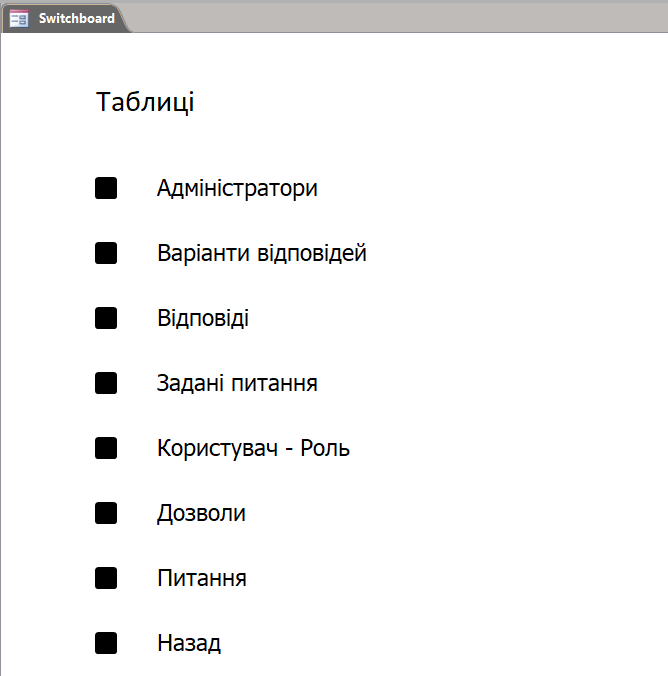


Рисунок 69 – Меню з таблицями БД

Перейшовши на конкретний пункт меню, відкривається вікно з можливістю прямого редагування даних таблиці як це зображено на рисунку 70.

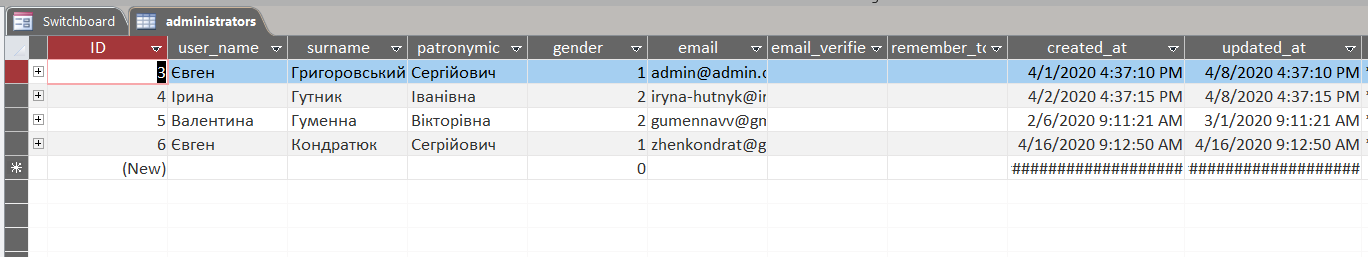


Рисунок 70 – Пряме редагування даних таблиці

Інші пункти меню з посиланням на таблиці ведуть на аналогічні форми прямого редагування таблиць. Вони самі по собі не є дуже зручними з точки зору користувацького інтерфейсу, тому було створено окремі форми для всіх потрібних функціональних можливостей.

Перейшовши на пункт меню «Форми», можемо побачити список форм, які користувач може використати для заповнення інфомації. Меню зі списком форм зображено на рисунку 71.

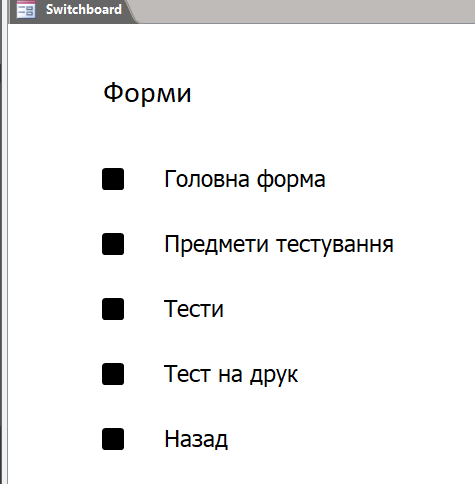


Рисунок 71 – Пряме редагування даних таблиці

Перейшовши на головну форму, користувач бачить функціональне вікно, в якому у вигляді вкладок організовано можливість заповнення інформації про групи студентів, студентів, тести, викладачів, ролі, та дозволи. Ця форма зображена на рисунку 72.

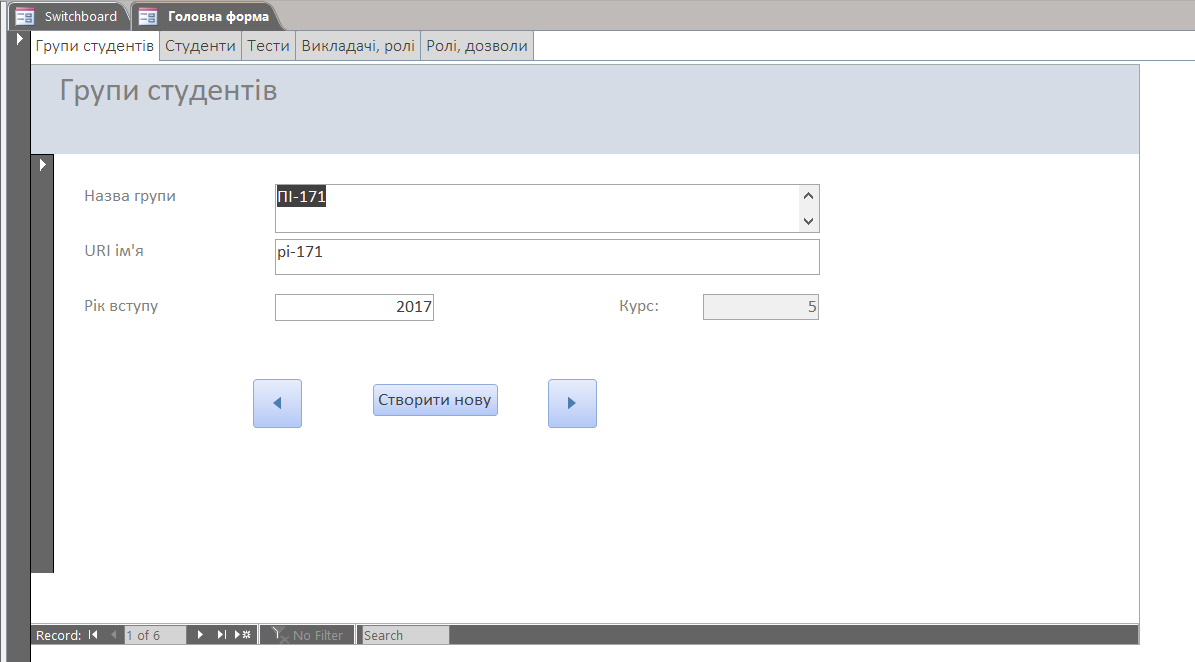


Рисунок 72 – Основна форма для адміністраторів

Для реєстрації студентів, викладачі можуть використовувати вкладку «Студенти», вигляд якої зображено на рисунку 73.

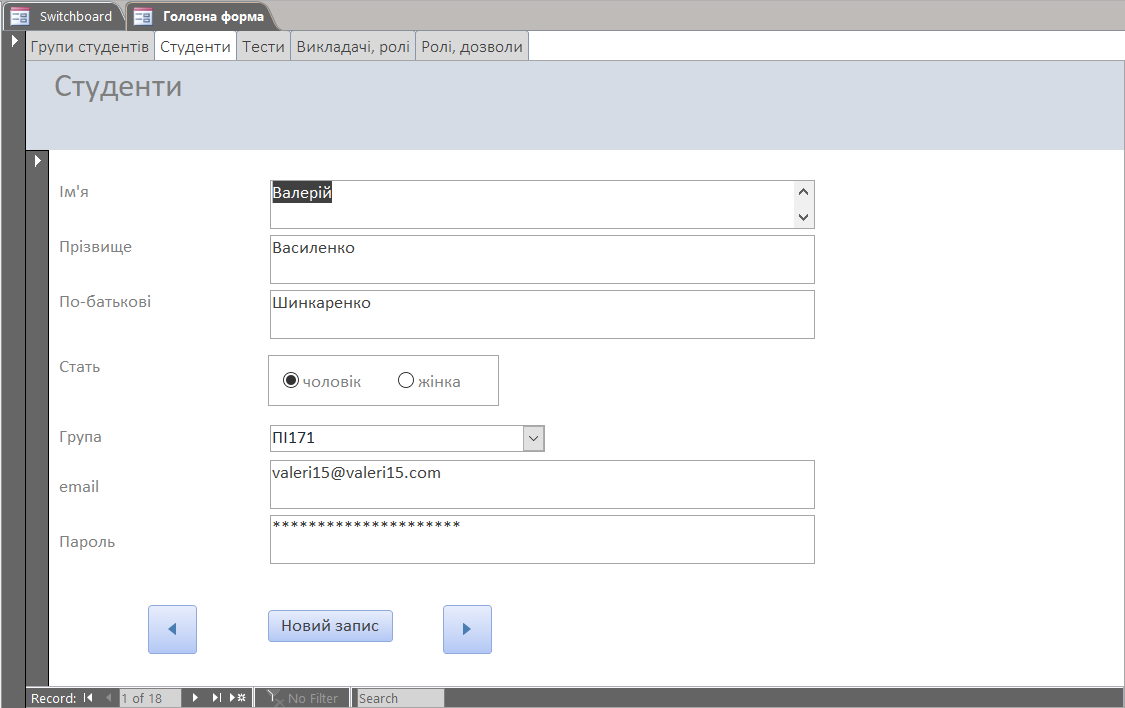


Рисунок 73 – Форма редагування інформації про студентів

Вкладка «Тести» може бути використана для створення, перегляду, редагування, видалення тестів та всіх підлеглих до них сутностей. Форма зображена на рисунку 74.

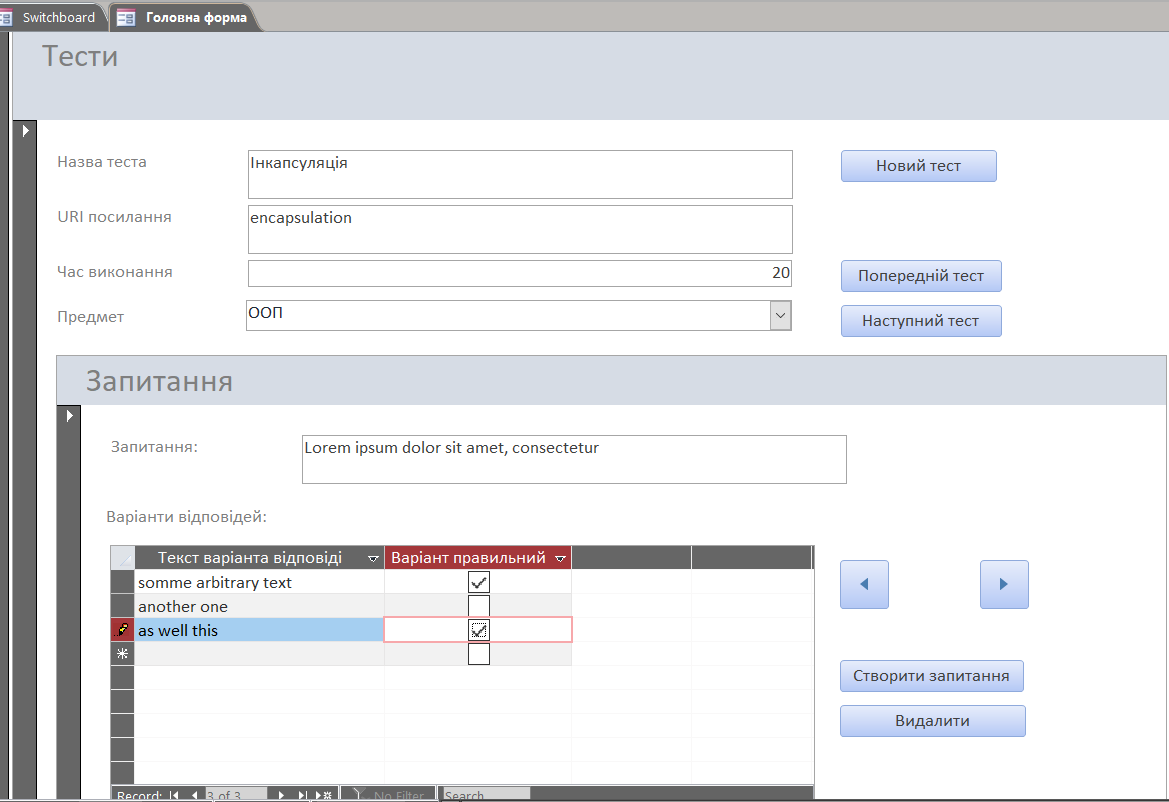


Рисунок 74 – Форма для управління тестами

Управління аккаунтами викладацького персоналу та їхніх ролей знаходиться на вкладці «Викладачі, ролі», як це зображено на рисунку 75.

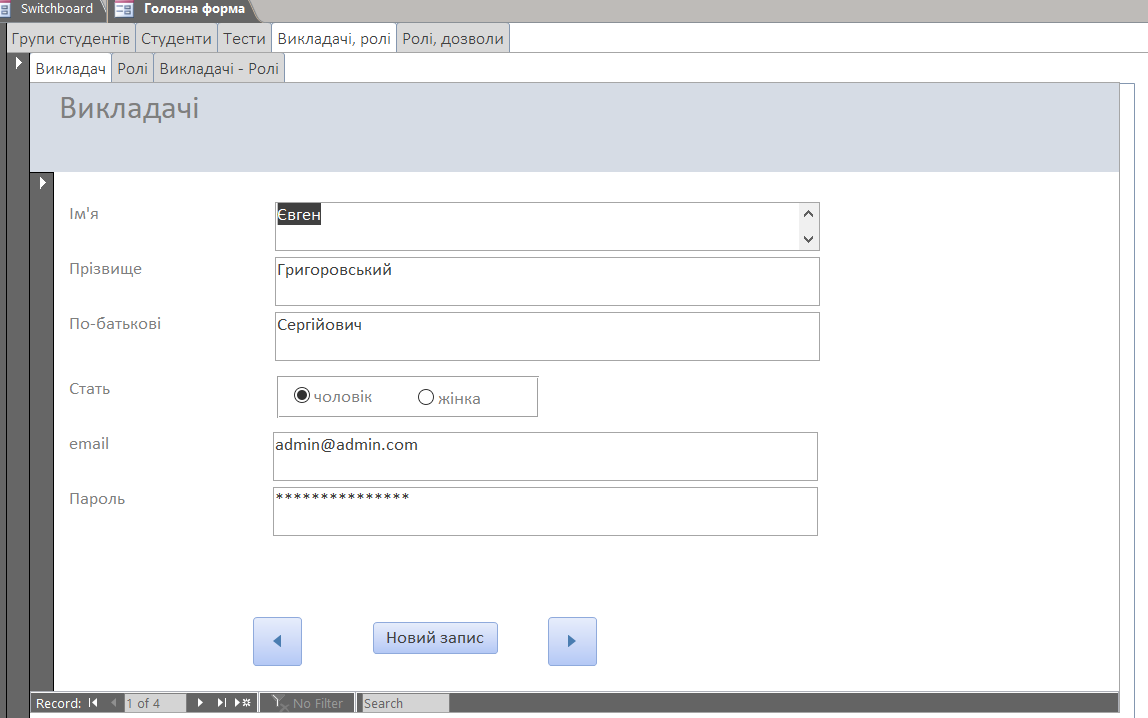


Рисунок 75 – Форма управління аккаунтами викладачів

Контроль доступу реалізований на базі ролей та дозволів. Редагування цих параметрів знаходиться на відповідній вкладці «Ролі, дозволи». Тут можна задати хто має які права в системі тестування, це зображено на рисунку 76.

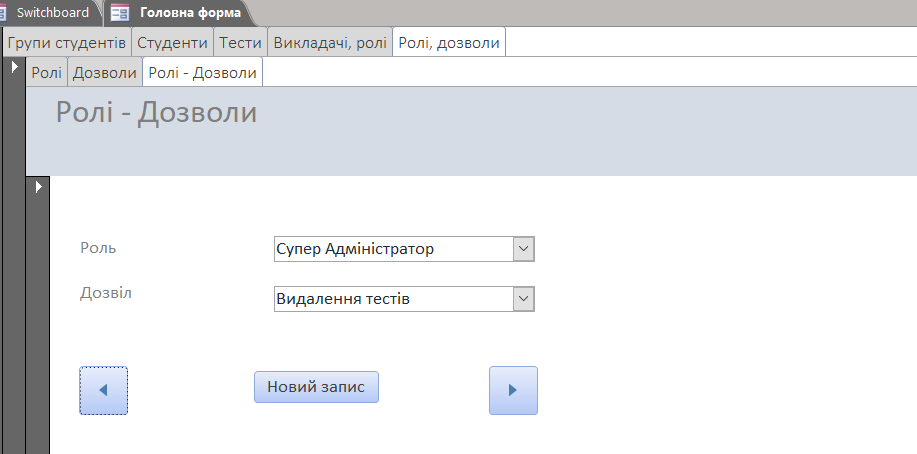


Рисунок 76 – Форма управління аккаунтами викладачів

Повертаючись до меню форм, наступним елементом в списку є «Тест на друк». Тут знаходиться функціонал генерації кінцевої форми тесту на друк. Це може бути корисно, наприклад, коли все ж таки було прийнято рішення проводити офлайн тестування традиційним методом на листочках. Форма для генерації документу зображена на рисунку 77.

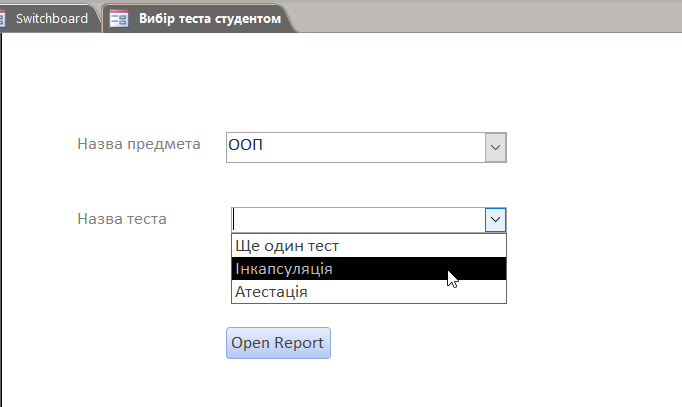


Рисунок 77 – Форма для вибору тесту на друк

В результаті відправки форми, можемо побачити попередній перегляд згенерованого документу як це зображено на рисунку 78.

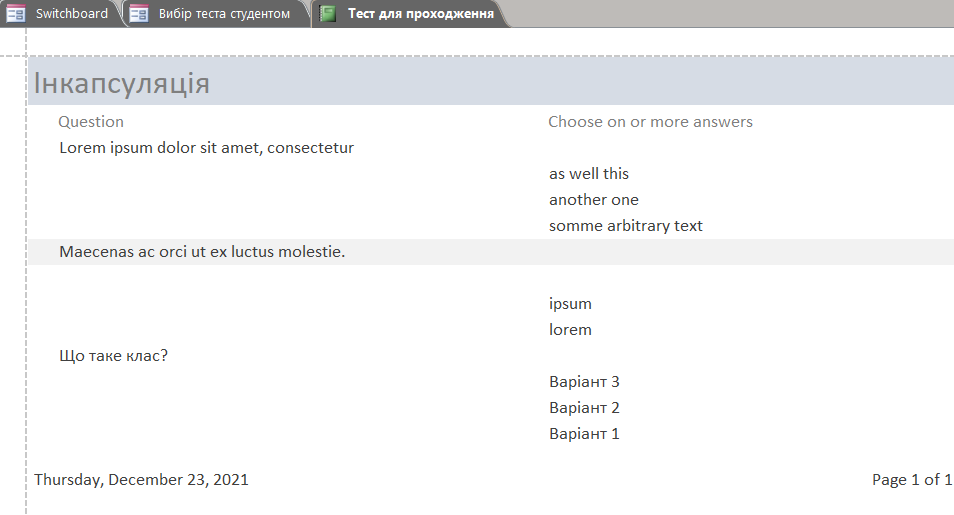


Рисунок 78 – Форма для вибору тесту на друк

Наразі це всі форми, що є доступні для користування. Далі розглянемо пункт головного меню «Звіти». Тут містяться функціонал для перегляду певної корисної інформації предметної області. Всього є доступно 5 звітів, меню яких зображено на рисунку 79.

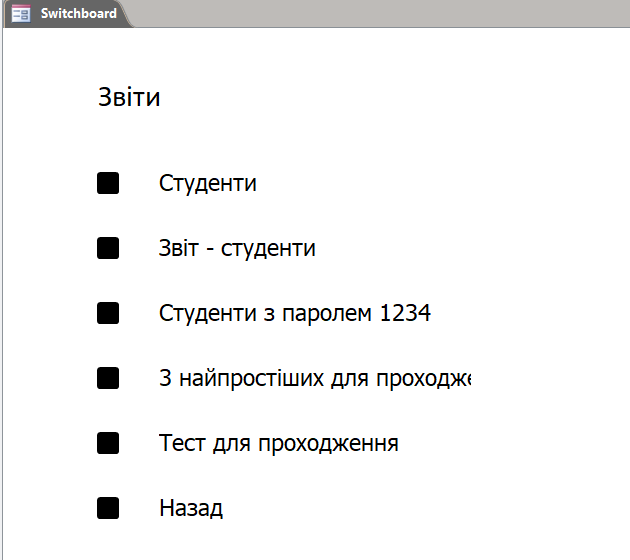


Рисунок 79 – Доступні звіти

Перший із звітів надає можливість переглянути студентів конкретної групи, ім’я якої потрібно ввести параметром для запиту. Звіт зображено на рисунку 80.

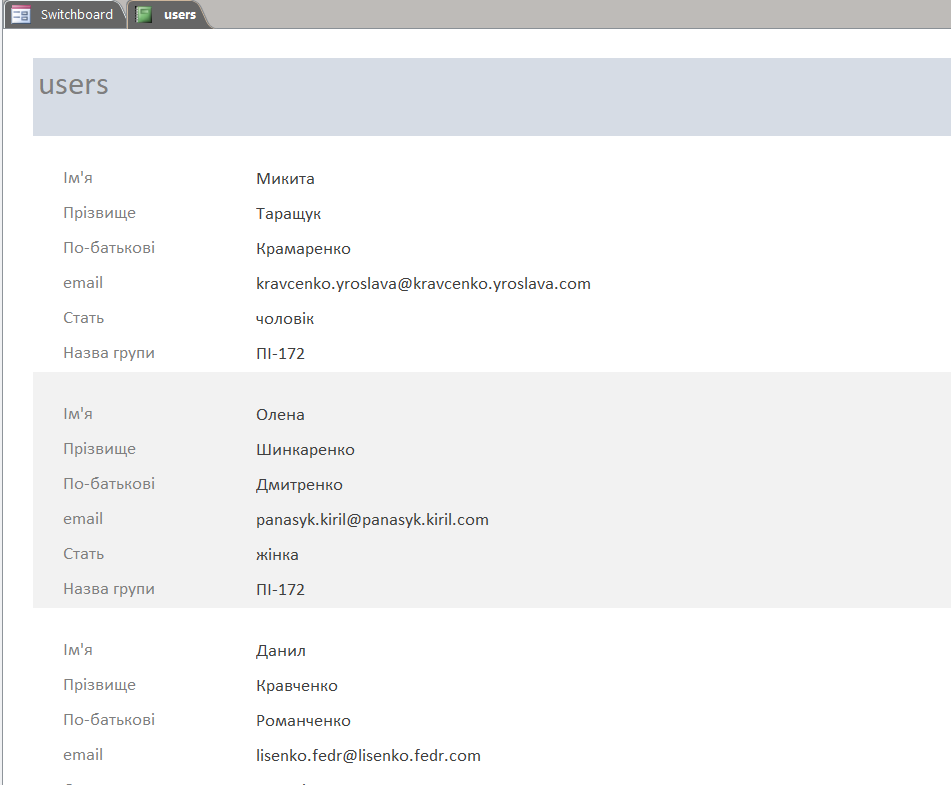


Рисунок 80 – Звіт студентів групи ПІ-172

Наступний із звітів дає можливість переглянути список усіх студентів, записи про яких є в базі даних. Вигляд звіту зображено на рисунку 81.

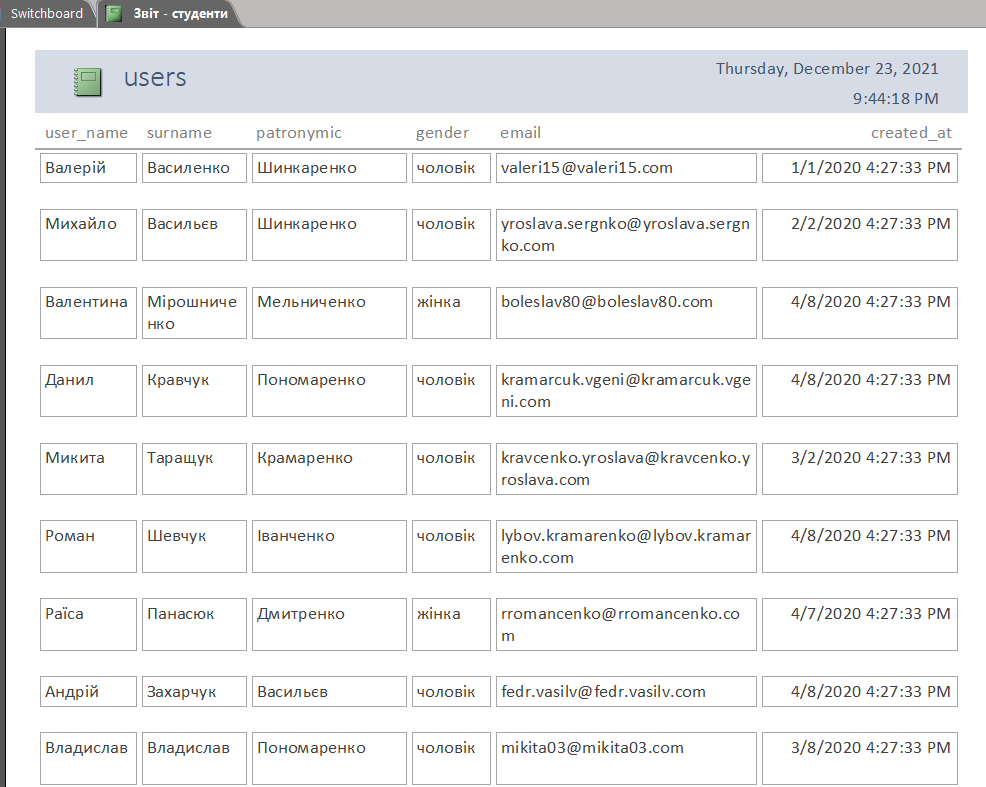


Рисунок 81 – Звіт всіх студентів

Іншим звітом є звіт, що показує список студентів, які містять пароль 1234. Так як цей пароль є досить ненадійним, можна буде повідомити студентів про необхідність зміни паролю. Звіт зображено на рисунку 82.

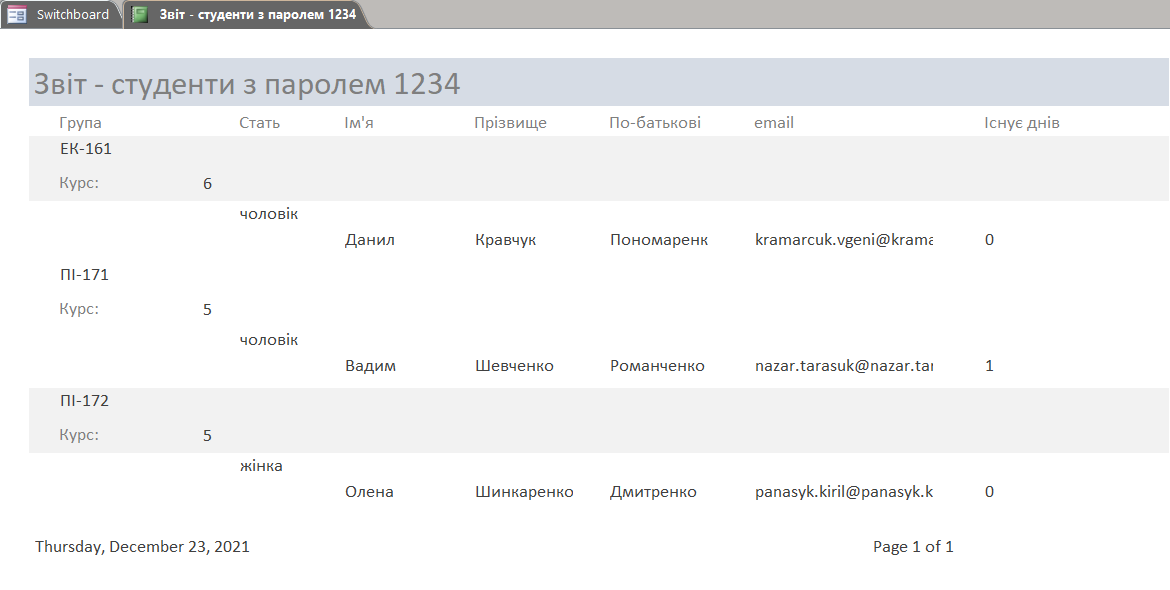


Рисунок 82 – Звіт студентів із простим паролем

Одним із найкорисніших звітів для викладачів може бути звіт, що показує найпростішя для проходження тести. Вигляд такого звіта зображено на рисунку 83.

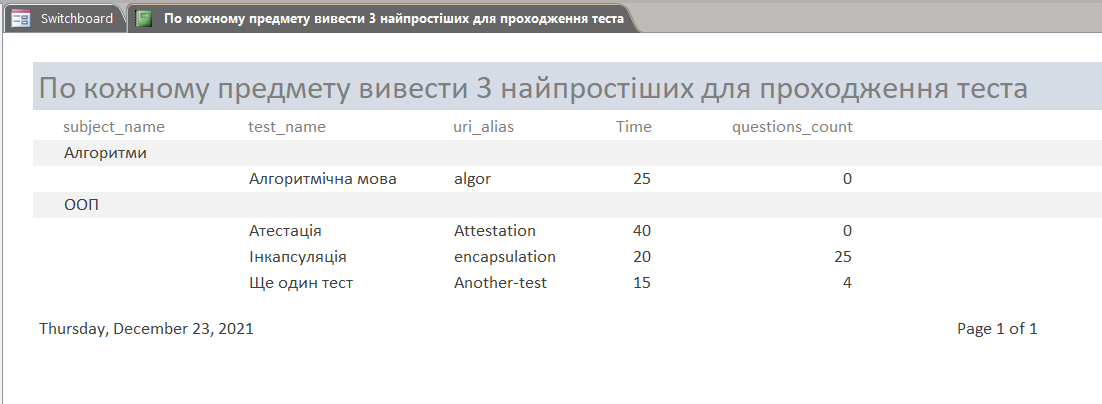


Рисунок 83 – Звіт найпростіших тесів

Інший схожий по функціоналу до вже показаної форми звіт дає можливість надрукувати тест для проходження тесту студентами офлайн. В ньому виводиться список усіх питань конкретного тесту, який задається вхідним параметром, разом із варіантами відповідей Звіт зображено на рисунку 84.

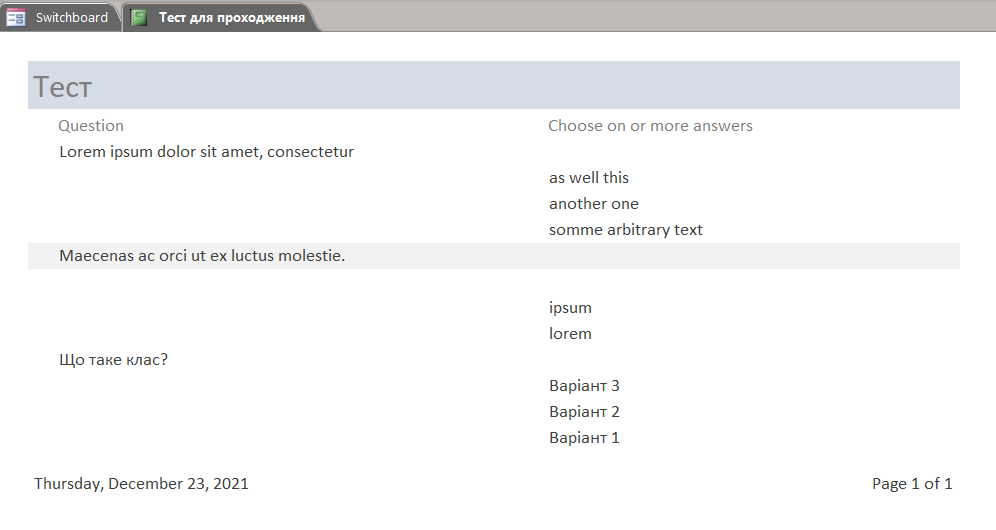


Рисунок 84 – Звіт тесту для проходження студентами

Окрім форм та звітів, користувачу також надається змога запускати виконувати певні запити для перегляду актуальної інформації прямо в програмному додатку Microsoft Access. Список доступних запитів зображено на рисунку 85.

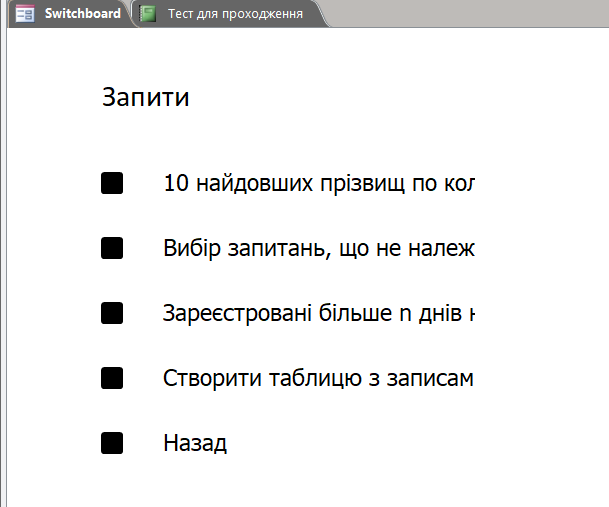


Рисунок 85 – Доступні запити

Запустивши перший звіт, можемо побачити вибірку із записами в порядку спадання довжини прізвища, як це зображено на рисунку 86.

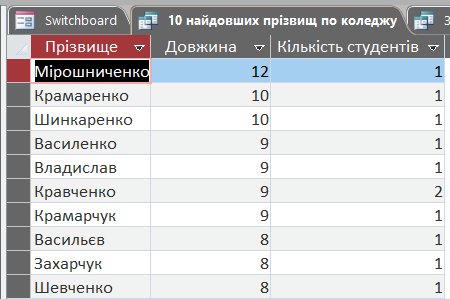


Рисунок 86 – Звіт найдовших прізвищ по коледжу

Наступний звіт приймає параметром кількість днів, і виводить список студентів, що були зареєстровані раніше, аніж передана кількість днів тому. Вигляд вибірки зображено на рисунку 87.

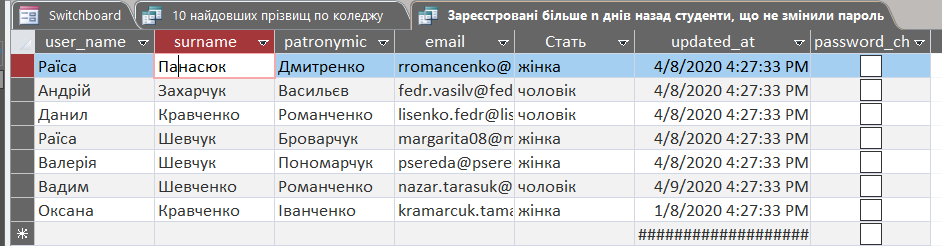


Рисунок 87 – Студенти, зареєстровані раніше 200 днів тому

Звіт «Вибір запитань, що не належать до жодного теста» дає можливість перевірити базу даних на неконсистентність даних. Такі можуть виникнути при видаленні пов’язаних сутностей. Результат запиту зображено на рисунку 88.

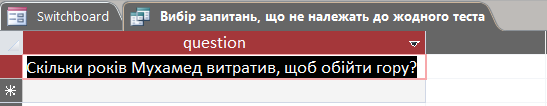


Рисунок 88 – Відв’язані запитання

* 1. Вимоги до технічних засобів

Для запуску програми СКБД Microsoft Access необхідні такі мінімальні системні вимоги до комп’ютера:

* операційна система Windows 7 та вище;
* Microsoft Access 2016 або вище;
* тактова частота процесора не менше 1,6 ГГц;
* об’єм оперативної пам’яті не менше 4 ГБ;
* місце на жосткому диску: 2 ГБ.

Більше ніяких спеціальних пристроїв чи додаткових програмних засобів для роботи за базою даних не потрібно.

ВИСНОВОК

Під час виконання курсового проекту було розроблено базу даних для приватного підприємця, який займається створенням меблів під індивідуальне замовлення. Така база даних передбачає ввід даних про існуючі матеріали, додаткові деталі (ручки, ніжки), створення власних класифікацій меблів, створення галереї зображень до матеріалів, додаткових деталей ти меблів, ввід даних про замовників та замовлення, передбачає створення власної системи знижок для постійних клієнтів, а для замовлення передбачає ввід дати створення та дати виконання для його вчасного виконання. Після тривалої роботи з’являється інформація, яка стає непотрібною чи неправильною, тому вона знищується.

Основні реалізовані завдання бази даних:

* ввід даних про матеріали, ручки, ніжки, класифікації, замовників, складові меблів, знижки;
* перегляд та коригування даних;
* вилучення непотрібної інформації;
* відбір даних за деякими критеріями.

Для повноцінної роботи автоматизованої системи також передбачено відбір даних по певних критеріях та вивід необхідних даних на друк.

Перелік посилань

1. Пісічник В.В.-Організація баз даних. – М.: И.Д. Вильямс, 2006 – 348 с.
2. Документація Access - https://www.microsoft.com/uk-ua/microsoft-365/access
3. Створення таблиць Access - https://support.microsoft.com/

## Додаток Б

(обов’язковий)

Презентація

Рисунок Б.1 – слайд «Створення інтерфейсу»