

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені Ігоря Сікорського»

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

# Кафедра системного програмування та спеціалізованих комп’ютерних систем

**Розрахункова робота**

з дисципліни

**«Бази даних і засоби управління»**

**на тему** «Створення додатку бази даних, орієнтованого на взаємодію з СУБД PostgreSQL»

Виконав: студент ІII курсу

ФПМ групи КВ-11

Лабазов Володимир Володимирович

Перевірив: Петрашенко А.В.

Київ – 2023

**Створення додатку бази даних, орієнтованого на взаємодію з СУБД PostgreSQL**

*Метою роботи* є здобуття вмінь програмування прикладних додатків баз даних PostgreSQL.

*Загальне завдання* роботи полягає у наступному:

1. Реалізувати функції перегляду, внесення, редагування та вилучення даних у таблицях бази даних, створених у лабораторній роботі №1, засобами консольного інтерфейсу.
2. Передбачити автоматичне пакетне генерування «рандомізованих» даних у базі.
3. Забезпечити реалізацію пошуку за декількома атрибутами з двох та більше сутностей одночасно: для числових атрибутів – у рамках діапазону, для рядкових – як шаблон функції LIKE оператора SELECT SQL, для логічного типу – значення True/False, для дат – у рамках діапазону дат.
4. Програмний код виконати згідно шаблону MVC (модель-подання-контролер).

# Опис предметної галузі з лабораторної роботи №1

При проектуванні бази даних «Система обліку екзаменаційних балів студентів» я виділив наступні сутності: Студент (Student), Група (Group), Предмет (Discipline), Оцінка (Mark).

Група може містити багато студентів , але студент може знаходитись в одній групі (зв'язок 1:N).

Кожен студент має багато оцінок (зв'язок 1:N).

Кожен студент ходить на не одну дисципліну , і на не одну дисципліну ходять не один студент (зв'язок N:M).

Кожна оцінка має один предмет з якого вона була отримана , і кожен предмет має одну оцінку за екзамен (зв'язок 1:1).

# Таблиця сутностей з описом їх призначення

|  |  |  |
| --- | --- | --- |
| **Сутність** | **Атрибут** | **Тип (розмір)** |
| Сутність «Student» містить інформацію про студента | id (PK) – унікальний id студента  name – ім’я студента  group\_id(FK) – унікальний id власника | Числовий  Текстовий (50)  Числовий |
| Сутність «Group» містить інформацію про групу | id (PK) – унікальний id групи  name – назва групи | Числовий  Текстовий (5) |
| Сутність «Discipline» містить інформацію про предмет | id (PK) – унікальний id дисципліни  name – назва дисципліни  teacher\_name – ім’я викладача дисципліни | Числовий  Текстовий (50)  Текстовий (50) |
| Сутність «Mark» містить інформацію про екзаменаційну оцінку | value – оцінка в балах  dicipline\_id(FK) – id дисципліни з якої була отримана оцінка  student\_id(FK) – id студента , що отримав оцінку  when\_received – унікальний id головного тренера | Числовий  Числовий  Числовий  Дата |

# Концептуальна модель предметної області “Система обліку екзаменаційних балів студентів”

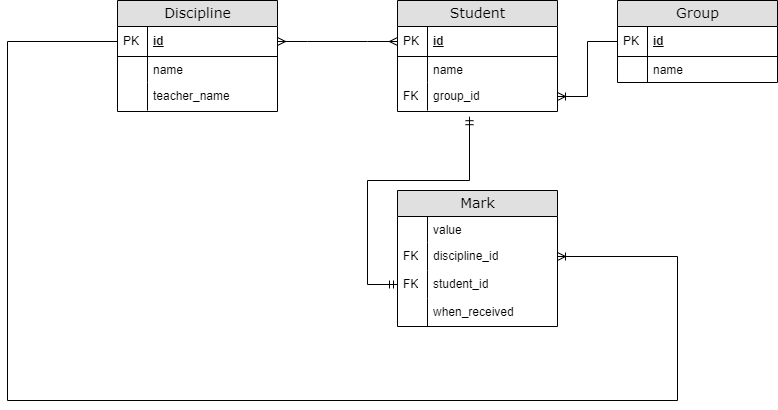


Рисунок 1 – Концептуальна модель предметної області «Система обліку екзаменаційних балів студентів».

**Нотація:** «UML». Модель побудована засобами програми draw.io

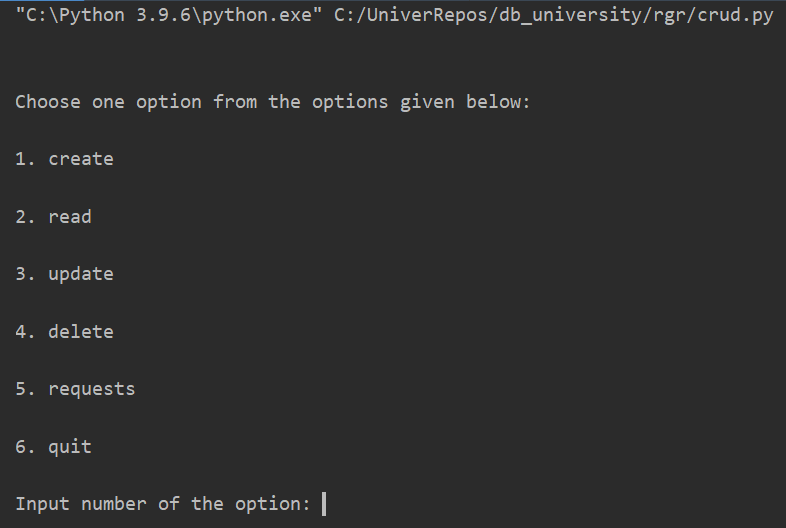
# Опис процесу перетворення

Сутності «Student», «Group», «Discipline», «Mark» було перетворено у таблиці. Зв'язок між студентом та предметом (зв'язок багато до багатьох) зумовив появі додаткової таблиці «student\_disciplines», яка містить унікальні id студента та дисципліни.

# 

**Демонстрація роботи програми**

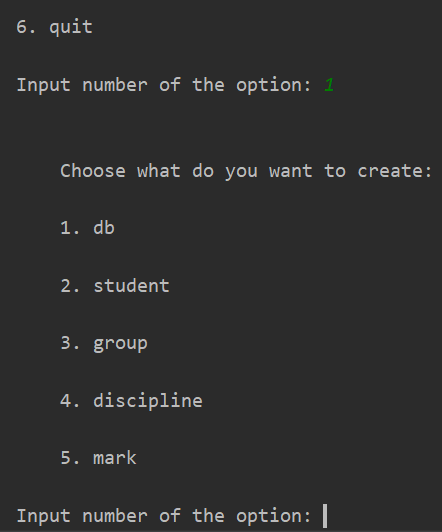
Головне меню користувача



Меню створення

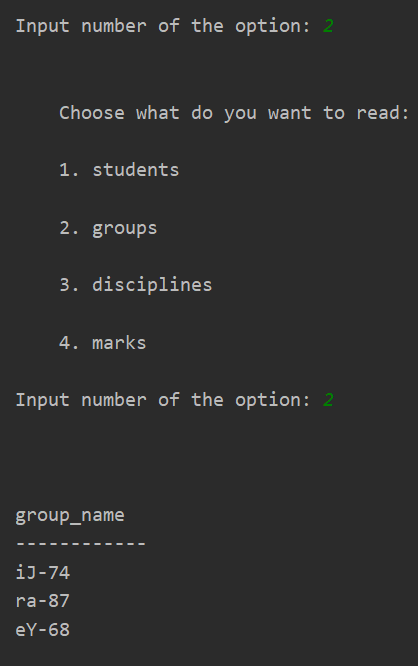
Через першу опцію можна видалити всі записи з бд, або перезаповнити її випадково згенерованими даними

Для створення запису в певній таблиці треба вибрати відповідну опцію і ввести необхідні дані



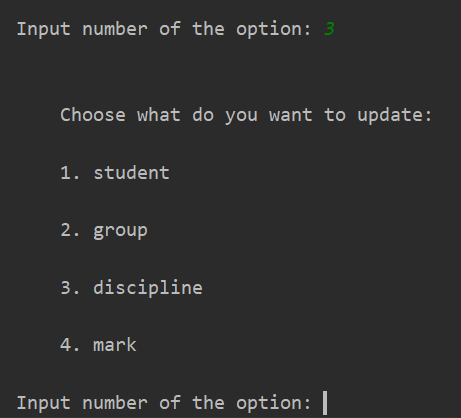
Меню читання

Тут потрібно просто вибрати необхідну опцію для показу вмісту певної таблиці



Меню модифікації

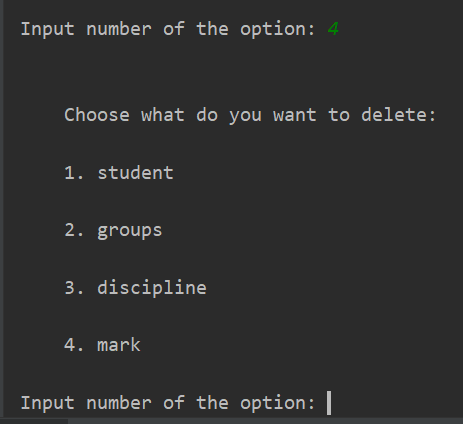
Для модифікації необхідно вибрати таблицю для модифікації і ввести необхідні дані



Меню видалення

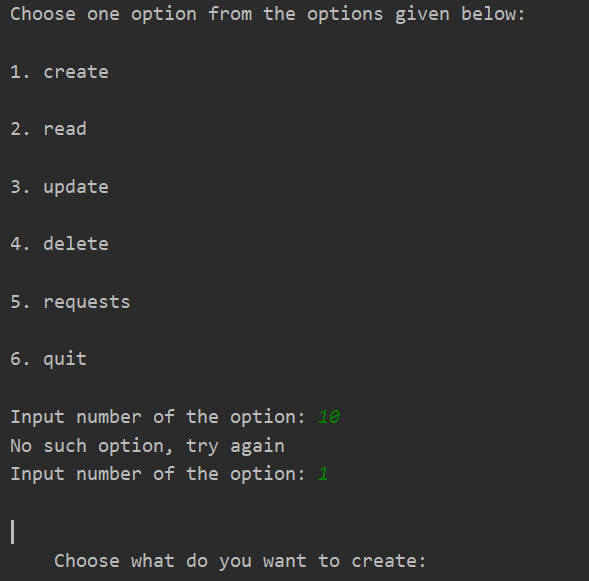
Для видалення необхідно вибрати таблицю, з якої будуть видалятись дані, і ввести необхідні дані

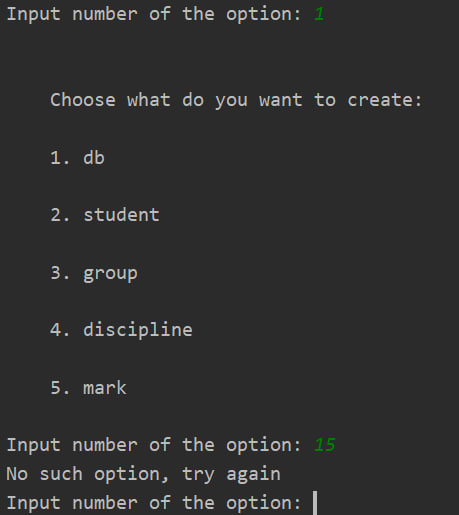
Записи в дочірніх таблицях видаляються каскадно



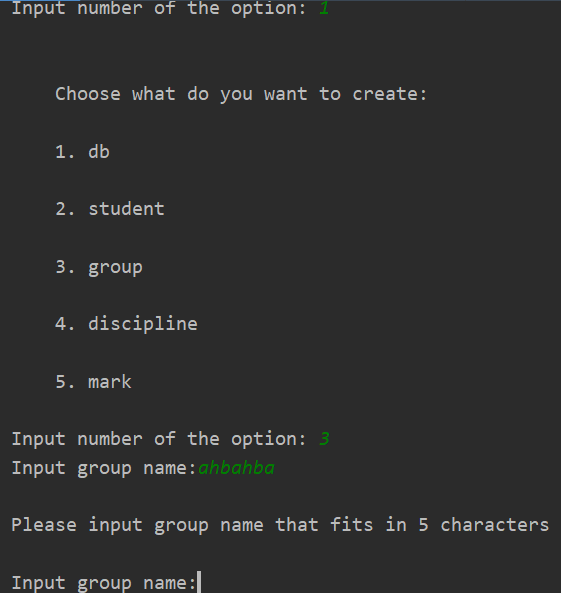
Помилки у всьому додатку відловлюються на різних етапах і виводяться відповідні помилкові повідомлення

Неправильний ввід опції:

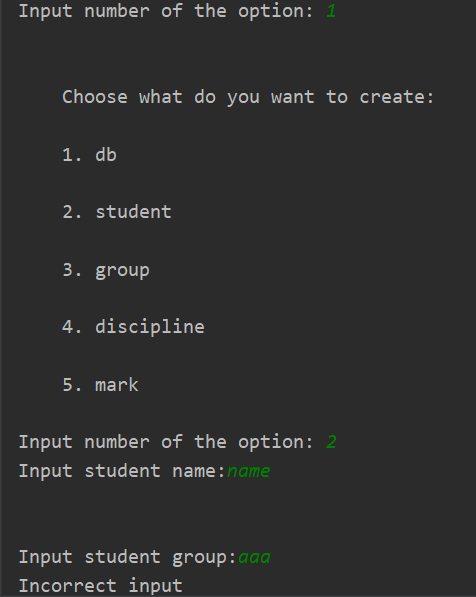




Ввід неправильної інформації в поле при додаванні/зміні/видаленні:

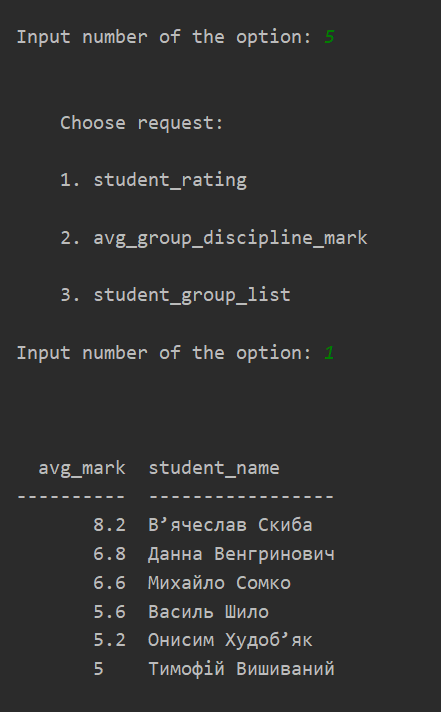


Ввід неіснуючого запису в таблиці(при додаванні студента до неіснуючої групи наприклад):



У додатку можливі наступні запити: рейтинг студентів, середні бали групи по кожному предмету, та список заданної групи

Рейтинг студентів по середньому балу з усіх предметів



Запит в SQL:

SELECT ROUND(AVG(m.value), 2) as avg\_mark, s.name

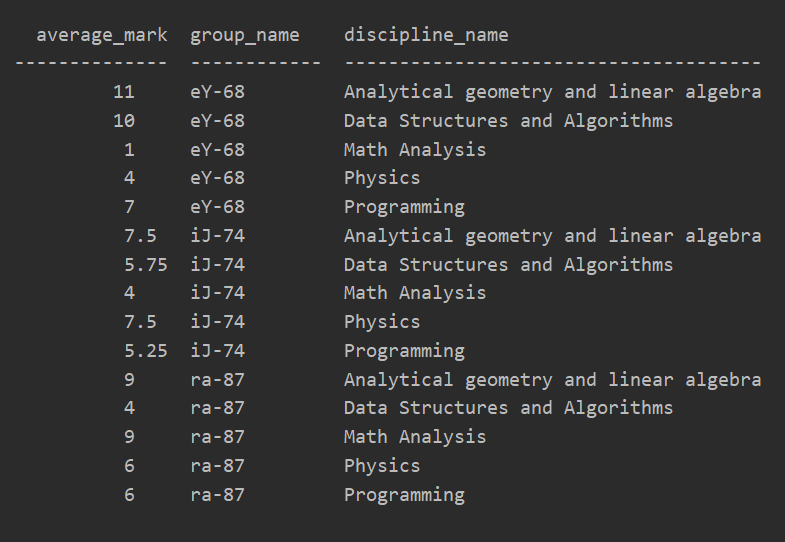
FROM marks as m

INNER JOIN students as s ON m.student\_id = s.id

GROUP BY s.id

ORDER BY avg\_mark DESC

Середні бали групи по кожному предмету:



Запит в SQL:

SELECT ROUND(AVG(m.value), 2) as average\_mark, g.name as group\_name, d.name as discipline\_name

FROM marks as m

INNER JOIN students as s ON m.student\_id = s.id

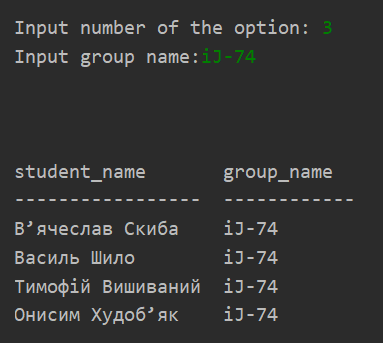
INNER JOIN disciplines as d ON m.discipline\_id = d.id

INNER JOIN groups as g ON s.group\_id = g.id

GROUP BY d.name, g.name

ORDER BY g.name

Список певної групи:



Запит в SQL:

SELECT s.name as student\_name, g.name as group\_name

FROM students as s

INNER JOIN groups as g ON s.group\_id = g.id

WHERE g.id = {}

У фігурні дужки за допомогою засобів роботи з рядками у мові Python вставляється id введеної групи(якщо програма знайшла групу за введеним ім’ям у базі, інакше виводиться помилкове повідомлення)

