

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

**Кафедра прикладної математики**



**Звіт**

з лабораторної роботи № 1

з дисципліни «Організація баз даних та знань»

**Виконала:**

Студентка групи ПМ-22

Торська Марія

**Перевірив:**

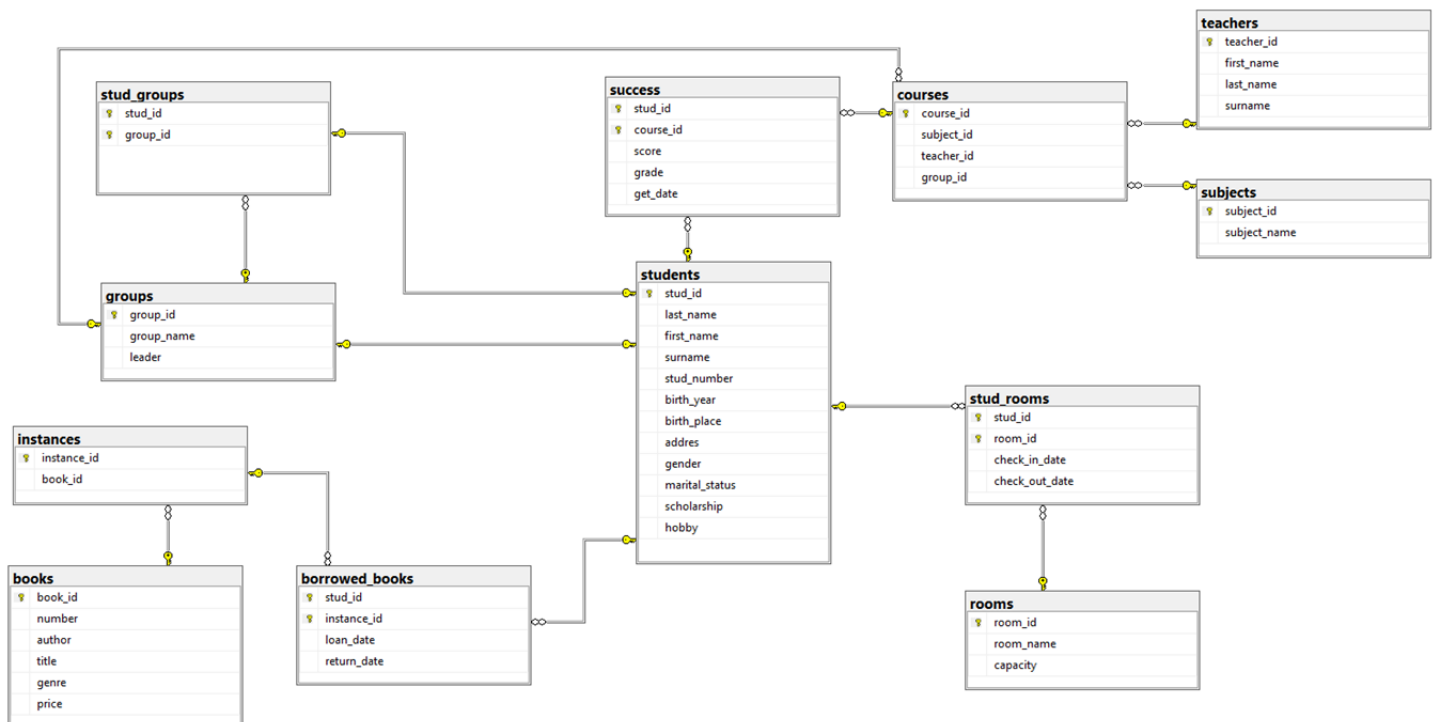
Алексєєв В.І.

Львів – 2023р.

# Тема: «Створення схеми БД»

## Варіант №1

### 1. Схема бази даних



### 2. SQL-запити для створення та наповнення таблиць

```
USE StudDB
GO
```

```
CREATE SEQUENCE stud_seq
START WITH 1
INCREMENT BY 1
NO CYCLE;
```

```
CREATE TABLE students (
    stud_id INT DEFAULT NEXT VALUE FOR stud_seq,
    last_name VARCHAR(50) NOT NULL,
    first_name VARCHAR(50) NOT NULL,
    surname VARCHAR(50) NOT NULL,
    stud_number VARCHAR(8) NOT NULL UNIQUE,
    birth_year INT NOT NULL,
    birth_place VARCHAR(200) NOT NULL,
    address VARCHAR(200) NOT NULL,
    gender VARCHAR(10) NOT NULL,
    marital_status VARCHAR(50) NOT NULL,
    scholarship DECIMAL(6,2) NOT NULL DEFAULT 0,
    room_number INT NOT NULL DEFAULT 0,
    hobby VARCHAR(200) NOT NULL DEFAULT '-',
    PRIMARY KEY (stud_id)
)
```

```
CREATE TABLE groups (
    group_id INT NOT NULL IDENTITY (1, 1),
    group_name VARCHAR(10) NOT NULL,
    leader INT NOT NULL UNIQUE,
    FOREIGN KEY (leader) REFERENCES students(stud_id),
    PRIMARY KEY (group_id)
)
```

```

CREATE TABLE stud_groups (
    stud_id INT NOT NULL UNIQUE,
    group_id INT NOT NULL,
    FOREIGN KEY (stud_id) REFERENCES students(stud_id),
    FOREIGN KEY (group_id) REFERENCES groups(group_id),
    PRIMARY KEY (stud_id, group_id)
)

CREATE TABLE books (
    book_id INT NOT NULL IDENTITY (1, 1),
    number VARCHAR(50) NOT NULL UNIQUE,
    author VARCHAR(100) NOT NULL,
    title VARCHAR(100) NOT NULL,
    genre VARCHAR(20) NOT NULL,
    price INT NOT NULL,
    PRIMARY KEY (book_id)
)

CREATE TABLE instances (
    instance_id INT NOT NULL IDENTITY (1, 1),
    book_id INT NOT NULL,
    FOREIGN KEY (book_id) REFERENCES books(book_id),
    PRIMARY KEY (instance_id)
)

CREATE TABLE borrowed_books (
    stud_id INT NOT NULL,
    instance_id INT NOT NULL UNIQUE,
    loan_date DATE NOT NULL,
    return_date DATE,
    FOREIGN KEY (stud_id) REFERENCES students(stud_id),
    FOREIGN KEY (instance_id) REFERENCES instances(instance_id),
    PRIMARY KEY (stud_id, instance_id)
)

CREATE TABLE subjects (
    subject_id INT NOT NULL IDENTITY (1, 1) PRIMARY KEY,
    subject_name VARCHAR(50) NOT NULL,
);

CREATE TABLE teachers (
    teacher_id INT NOT NULL IDENTITY (1, 1) PRIMARY KEY,
    first_name VARCHAR(50) NOT NULL,
    last_name VARCHAR(50) NOT NULL,
    surname VARCHAR(50) NOT NULL
);

CREATE TABLE courses (
    course_id INT NOT NULL IDENTITY (1, 1) PRIMARY KEY,
    subject_id INT NOT NULL,
    group_id INT NOT NULL,
    teacher_id INT NOT NULL,
    FOREIGN KEY (subject_id) REFERENCES subjects(subject_id),
    FOREIGN KEY (teacher_id) REFERENCES teachers(teacher_id),
    FOREIGN KEY (group_id) REFERENCES groups(group_id)
);

CREATE TABLE success (
    stud_id INT NOT NULL,
    course_id INT NOT NULL,
    score INT NOT NULL,
    grade VARCHAR(50) NOT NULL,
    get_date DATE NOT NULL,
    FOREIGN KEY (stud_id) REFERENCES students(stud_id),
    FOREIGN KEY (course_id) REFERENCES courses(course_id),
    PRIMARY KEY (stud_id, course_id),
    CONSTRAINT CHK_score CHECK (score <= 100)
);

```

```

CREATE TABLE rooms (
    room_id INT NOT NULL IDENTITY (1, 1) PRIMARY KEY,
    room_name INT NOT NULL,
    capacity INT DEFAULT 3,
)

CREATE TABLE stud_rooms (
    stud_id INT NOT NULL,
    room_id INT NOT NULL,
    check_in_date DATE NOT NULL,
    check_out_date DATE,
    FOREIGN KEY (stud_id) REFERENCES students(stud_id),
    FOREIGN KEY (room_id) REFERENCES rooms(room_id),
    PRIMARY KEY (stud_id, room_id),
)

INSERT INTO students(last_name, first_name, surname, stud_number, birth_year, birth_place, address, gender, marital_status, hobby)
VALUES ('Давидюк', 'Василь', 'Петрович', '00124586', 2002, 'м.Львів', 'вул.Любінська 24/15', 'ч', 'неодружений', 'Плавання'),
       ('Василишин', 'Анна', 'Антонівна', '00124545', 2002, 'м.Яворів', 'вул.І.Франка 12/3', 'ж', 'неодружена', 'Баскетбол'),
       ('Шульга', 'Анастасія', 'Кирилівна', '00568495', 2003, 'м.Львів', 'вул.Городоцька 145/34', 'ж', 'неодружена', 'Плавання'),
       ('Баворовський', 'Олександр', 'Борисович', '00485976', 2002, 'м.Рівне', 'вул.Т.Шевченка 85а/163', 'ч', 'неодружений', '-');

INSERT INTO groups(group_name, leader)
VALUES ('ПМ-32', 1);

INSERT INTO stud_groups(stud_id, group_id)
VALUES (1, 1), (2, 1)

INSERT INTO teachers(first_name, last_name, surname)
VALUES ('Наталя', 'Мельник', 'Петрівна'), ('Ігор', 'Вербовецький', 'Павлович'), ('Катерина', 'Мороз', 'Вікторівна')

INSERT INTO subjects(subject_name)
VALUES ('Математичний аналіз'), ('Чисельні методи'), ('Нейронні мережі'), ('Диференціальні рівняння'), ('Веб-програмування'), ('Іноземна мова');

INSERT INTO courses(subject_id, teacher_id)
VALUES (1, 1), (2, 1), (3, 2), (1, 3)

INSERT INTO teachers(last_name, first_name, surname)
VALUES ('Когут', 'Микола', 'Іванович'), ('Новак', 'Оксана', 'Володимирівна'), ('Жук', 'Ярослава', 'Миколаївна')

INSERT INTO courses(subject_id, teacher_id)
VALUES (4, 6), (5, 4), (6, 5)

INSERT INTO success(stud_id, course_id, score, grade, get_date)
VALUES (1, 1, 88, 'відмінно', '2022-12-05'), (2, 1, 94, 'відмінно', '2022-12-05'), (3, 1, 81, 'добре', '2022-12-05'), (4, 1, 95, 'відмінно', '2022-12-05'),
       (1, 2, 80, 'добре', '2022-12-09'), (2, 2, 83, 'добре', '2022-12-09'), (3, 2, 90, 'відмінно', '2022-12-09'), (4, 2, 100, 'відмінно', '2022-12-09'),
       (1, 3, 89, 'відмінно', '2022-12-02'), (2, 3, 91, 'відмінно', '2022-12-02'), (3, 3, 79, 'добре', '2022-12-02'), (4, 3, 88, 'відмінно', '2022-12-02');

INSERT INTO instances(book_id)
VALUES (1), (1), (1), (2), (2), (2), (3), (3), (3)

INSERT INTO borrowed_books(stud_id, instance_id, loan_date)
VALUES (1, 2, '2023-03-01'), (2, 2, '2023-03-01')

```

### 3. Роз'яснення до схеми даних:

Ця база даних призначена для зберігання інформації про студентів, навчальні курси, предмети, викладачів, успішність, групи, книги та їх екземпляри, кімнати в гуртожитку.

У таблиці 'students' зберігається інформація про студентів у полях ідентифікатора студента, прізвища, ім'я, по-батькові, номера студентського, року та місця народження, адреси, статі, сімейного стану, стипендії та хобі. Ідентифікатори студентів генеруються автоматично за допомогою послідовності stud\_seq.

Таблиця 'groups' зберігає інформацію про групи студентів, включаючи назву групи та ідентифікатор старости групи.

У таблиці 'stud\_groups' відображається зв'язок багато до багатьох між студентами та групами. Кожен запис відображає, який студент входить до якої групи.

Таблиця 'books' містить інформацію про книги, а саме поля ідентифікатора, серійного номера, автора, назви, жару та ціни.

У таблиці 'instances' зберігається інформація про конкретні екземпляри книг, які можуть бути позичені студентами.

Таблиця 'borrowed\_books' відображає зв'язок між студентами та екземплярами книг, які вони взяли в бібліотеці. Кожен запис містить ідентифікатори студента та екземпляра книги, дату видачі та дату повернення.

Таблиця 'rooms' зберігає інформацію про кімнати в гуртожитку, включаючи номер кімнати та її вмістимість.

У таблиці 'stud\_rooms' відображається зв'язок багато до багатьох між студентами та кімнатами. Кожен запис містить ідентифікатори студента та кімнати, дату заселення та дату виселення.

У таблиці 'subjects' зберігається інформація про предмети, які викладаються в університеті, їх ідентифікатор та назва.

Таблиця 'teachers' містить поля ідентифікатора, прізвища, ім'я, по-батькові викладачів, які працюють в університеті.

У таблиці 'courses' зберігається інформація про курси, які викладаються в університеті. Кожен запис містить ідентифікатор предмета, групи та викладача, які пов'язані з цим курсом.

У таблиці 'success' відображається взаємозв'язок між студентами та курсами, які вони вивчають, та містяться поля ідентифікаторів студента та курсу, балу, оцінки та дати отримання оцінки.