

Лабораторна робота №2

Створення простої БД

Тема: Створення простої БД, що містить одну таблицю.

Мета роботи: отримати навички створення таблиці БД, ознайомитись з типами даних, що можуть зберігатись в БД.

Завдання до лабораторної роботи

1. Створити БД із використанням СУБД MySQL, MariaDB або іншої за узгодженням з викладачем.
2. Додати до цієї БД таблицю відповідно до варіанту (або індивідуального завдання).
3. Заповнити її тестовими даними (не менше 10 рядків)
Зауваження 1: пп.1-3 треба не потрібно виконувати програмно.
4. Реалізувати запити на перегляд всіх даних з таблиці.
5. Реалізувати запити на вибірку даних за вказаними критеріями
Зауваження 2: пп.4-5 пишемо програму на мові Java.
Зауваження 3: використання JPA необов'язково, але можливо.

Варіанти завдань

Варіант 1.

Student: id, Прізвище, Ім'я, По батькові, Дата народження, Адреса, Телефон, Факультет, Курс, Група.

Вивести:

- a) список студентів заданого факультету;
- b) список студентів, які народились після заданого року;
- c) список навчальної групи в порядку алфавіту;
- d) список студентів впорядкований за алфавітом назви факультету, а для студентів одного факультету – за датою народження
- e) список всіх факультетів, інформація про студентів яких є у програмі без повторів
- f) для всіх факультетів, визначених у пункті e) визначити кількість студентів, що там навчаються та вивести цю інформацію у зручному вигляді.

Варіант 2.

Customer: id, Прізвище, Ім'я, По батькові, Адреса, Номер кредитної картки, Баланс рахунку (кількість грошей).

Вивести:

- a) список покупців, із вказаним іменем;
- b) список покупців, у яких номер кредитної картки знаходиться в заданому інтервалі;
- c) кількість та список покупців, які мають заборгованість (від'ємний баланс на карті) в порядку зростання заборгованості
- d) список покупців, упорядкований за зростанням балансу рахунку, а при рівності балансів – за номером кредитної картки
- e) список років народження покупців, зареєстрованих у програмі без повторів
- f) для кожного року народження визначити покупця, що має найбільшу кількість грошей на картці

Варіант 3.

Patient: id, Прізвище, Ім'я, По батькові, Адреса, Телефон, Номер медичної карти, Діагноз.

Вивести:

- a) список пацієнтів, які мають вказаний діагноз в порядку зростання номерів медичної картки;
- b) список пацієнтів, номер медичної карти у яких знаходиться в заданому інтервалі;
- c) кількість та список пацієнтів, номер телефона яких починається з вказаної цифри;
- d) список діагнозів пацієнтів (без повторів) із вказанням кількості пацієнтів, що мають цей діагноз у порядку спадання цієї кількості;
- e) список діагнозів пацієнтів, зареєстрованих у системі без повторів;
- f) для кожного діагнозу визначити кількість пацієнтів, яким він поставлений.

Варіант 4.

Abiturient: id, Прізвище, Ім'я, По батькові, Адреса, Телефон, Середній бал.

Вивести:

- a) список абітурієнтів із вказаним іменем, в порядку спадання середнього балу;
- b) список абітурієнтів, середній бал у яких вище заданого;
- c) вибрати задане число n абітурієнтів, що мають найвищий середній бал.
- d) список абітурієнтів в порядку алфавіту за прізвищем, при збігу прізвищ – за іменами;
- e) список років народження абітурієнтів в порядку зростання без повторів;
- f) визначити кількість абітурієнтів кожного року народження.

Варіант 5.

Book: id, Назва, Автор, Видавництво, Рік видання, Кількість сторінок, Ціна.

Вивести:

- a) список книг заданого автора в порядку зростання року видання;
- b) список книг, що видані заданим видавництвом;
- c) список книг, що випущені після заданого року;
- d) список авторів в алфавітному порядку;
- e) список видавництв, книги яких зареєстровані в системі без повторів;
- f) для кожного видавництва визначити список книг, виданих ним.

Варіант 6.

House: id, Номер квартири, Площа, Поверх, Кількість кімнат, Вулиця.

Вивести:

- a) список квартир, які мають задане число кімнат;
- b) список квартир, які мають задане число кімнат та розташовані на поверсі, який знаходиться в заданому проміжку;
- c) список квартир, які мають площу, що перевищує задану в порядку спадання площі. Якщо площа однакова – то в порядку зростання поверху;
- d) список всіх квартир, в порядку зростання площі ;
- e) список поверхів на яких розташовані квартири, у порядку спадання;
- f) для кожного поверху визначити список квартир

Варіант 7.

Phone: id, Прізвище, Ім'я, По батькові, Номер рахунку, Час міських розмов, Час міжміських розмов.

Вивести:

- a) відомості про абонентів, у яких час міських розмов перевищує заданий;
- b) відомості про абонентів, які користувались міжміським зв'язком в порядку алфавіту за прізвищем, при однакових прізвищах – за іменами, потім по-батькові;
- c) відомості про абонентів чий номер рахунку знаходиться у вказаному діапазоні;
- d) відомості про всіх абонентів в порядку зростання сумарного часу розмов;
- e) список міст, телефони з яких зареєстровані в системі;
- f) для кожного міста визначити список телефонів з нього.

Варіант 8.

Car: id, Модель, Рік випуску, Ціна, Реєстраційний номер.

Вивести:

- a) список автомобілів заданої моделі в порядку зростання року випуску;
- b) список автомобілів заданої моделі, які експлуатуються більше n років;
- c) список автомобілів заданого року випуску, ціна яких більше вказаної;
- d) список автомобілів в порядку спадання ціни. Якщо ціна однакова, то в порядку зростання року випуску;
- e) список моделей автомобілів, зареєстрованих у програмі;
- f) для кожної моделі вивести список автомобілів.

Варіант 9.

Product: id, Найменування, Виробник, Ціна, Термін зберігання, Кількість.

Вивести:

- a) список товарів для заданого найменування в порядку спадання терміну зберігання;
- b) список товарів для заданого найменування, ціна яких не перевищує задану;
- c) список товарів, термін зберігання яких більше заданого;
- d) список товарів, впорядкований за зростанням вартості (кількість * ціна), якщо вартість однакова, то за спаданням ціни;
- e) список виробників продуктів, зареєстрованих в програмі;
- f) для кожного виробника вивести список продуктів, які він виробляє.

Варіант 10.

Train: Пункт призначення, Номер поїзду, Час відправки, Число місць (загальних, купе, плацкарт, люкс).

Вивести:

- a) список поїздів, які прямують до заданого пункту призначення в порядку зростання часу відправки, якщо час однаковий – за зростанням номеру поїзда;
- b) список поїздів, які прямують до заданого пункту призначення та відправляються після заданої години;
- c) список поїздів, які відправляються до заданого пункту призначення та мають загальні місця;
- d) список поїздів, які відправляються до заданого пункту призначення в порядку зростання кількості всіх місць;
- e) список пунктів призначення (без повторів) в порядку зростання кількості поїздів, що до них прямують, якщо кількість однакова – в порядку алфавіту;
- f) для кожного пункту призначення вивести список поїздів, які до нього прямують.