НАЦIОНАЛЬНИЙ ТЕХНIЧНИЙ УНIВЕРСИТЕТ УКРАЇНИ «КИЇВСЬКИЙ ПОЛIТЕХНIЧНИЙ IНСТИТУТ IМЕНI IГОРЯ СIКОРСЬКОГО» ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

**Кафедра системного програмування і спеціалізованих комп'ютерних систем**

**Лабораторна робота №2**

з дисципліни «[Бази даних і засоби управління](http://wiki.kpi.ua/index.php/%D0%91%D0%B0%D0%B7%D0%B8%20%D0%B4%D0%B0%D0%BD%D0%B8%D1%85%20%D1%96%20%D0%B7%D0%B0%D1%81%D0%BE%D0%B1%D0%B8%20%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D1%96%D0%BD%D0%BD%D1%8F_(19301930))»

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

Виконав: студент ІІI курсу

ФПМ групи КВ-81

Кравчук В.В.

Викладач: [Петрашенко А. В.](http://rozklad.kpi.ua/Schedules/ViewSchedule.aspx?v=dc2ccd54-2021-4dbe-a8b3-d354f484e595)

Київ 2020

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

*Загальне завдання* роботи полягає у наступному:

1. Реалізувати функції 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).

*Деталізоване завдання*:

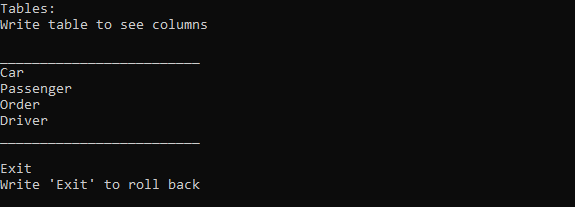
1. Забезпечити можливість уведення/редагування/вилучення даних у таблицях бази даних з можливістю контролю відповідності типів даних атрибутів таблиць (рядків, чисел, дати/часу). Для контролю пропонується два варіанти: контроль при введенні (валідація даних) та перехоплення помилок (try..except) від сервера PostgreSQL при виконанні відповідної команди SQL. Особливу увагу варто звернути на дані таблиць, що мають зв’язок 1:N. При цьому з боку батьківської таблиці необхідно контролювати **вилучення** рядків за умови наявності даних у підлеглій таблиці. З точки зору підлеглої таблиці варто контролювати наявність відповідного рядка у батьківській таблиці при виконанні **внесення** нових даних. Унеможливити виведення програмою системних помилок на екрані шляхом їх перехоплення і адекватної обробки. Внесення даних виконується користувачем у консольному вікні програми.
2. Забезпечити можливість автоматичної генерації великої кількості даних у таблицях за допомогою вбудованих у PostgreSQL функцій роботи з псевдовипадковими числами. Дані мають бути згенерованими **не мовою програмування, а відповідним SQL-запитом**! Кількість даних для генерування має вводити користувач з клавіатури. Для тесту взяти 100 000 записів для однієї-двох таблиць. Особливу увагу слід звернути на відповідність даних вимогам зовнішніх ключів з метою уникнення помилок порушення обмежень цілісності (foreign key).
3. Для реалізації пошуку необхідно підготувати 3 запити, що включають дані з декількох таблиць і фільтрують рядки за 3-4 атрибутами цих таблиць. Забезпечити можливість уведення конкретних значень констант для фільтрації з клавіатури користувачем. Крім того, після виведення даних необхідно вивести час виконання запиту у мілісекундах. Перевірити швидкодію роботи запитів на попередньо згенерованих даних.
4. Програмний код організувати згідно шаблону Model-View-Controller(MVC). При цьому модель, подання та контролер мають бути реалізовані у окремих файлах. Для доступу до бази даних використовувати **лише мову SQL** (без ORM).

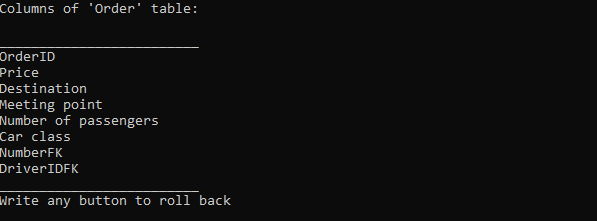
Посилання на репозиторій:

**Пункт №1:**

Перегляд таблиць і стовпців:

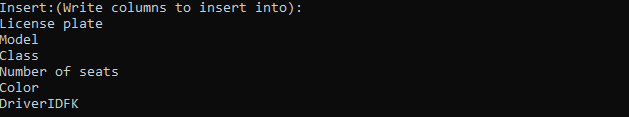


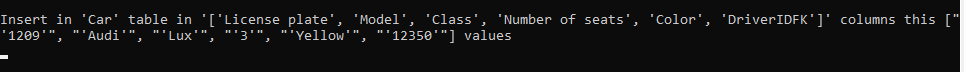




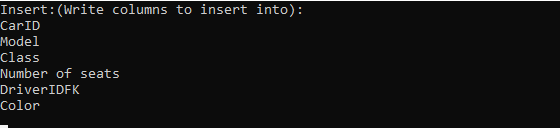
Insert:

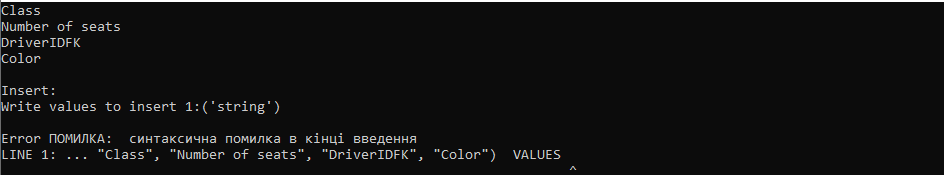






При введені таблиці якої не існує чи значень неправильних типів, виведеться відповідні повідомлення про помилки:





Update:







Delete:

Виводить помилку тільки у випадку неправильної назви таблиці, або неіснуючої колонки. “Delete” дозволяє користувачу видалити поле із таблиць на які посилаються поля інших таблиць. “Delete” видаляє всі значення в усіх таблицях які були зв’язані з полем яке безпосередньо видаляє користувач.





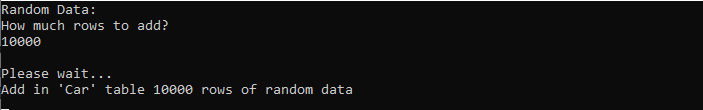


**Пункт №2:**

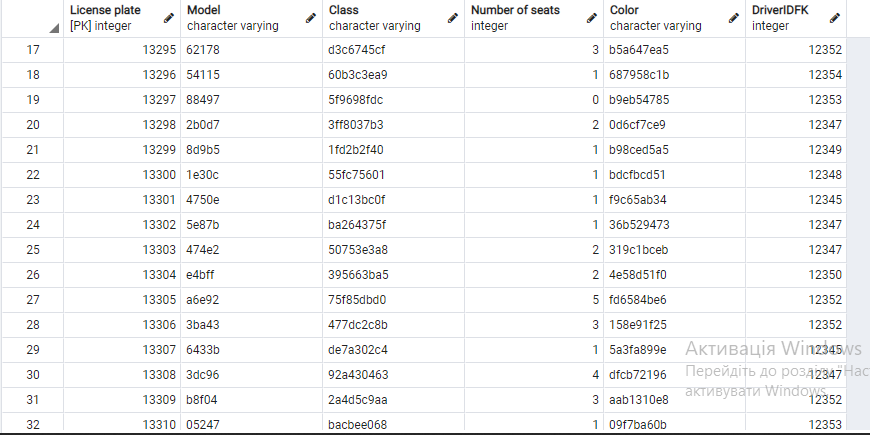
Генерація даних автоматично реалізована для таблиць “Car ” та “Driver”. При введені користувачем іншої таблиці буде виведено повідомлення про не можливість виконати дану команду. У випадку “Сar ” значення для “DriverIDFK” беруться рандомно з “DriverID”.









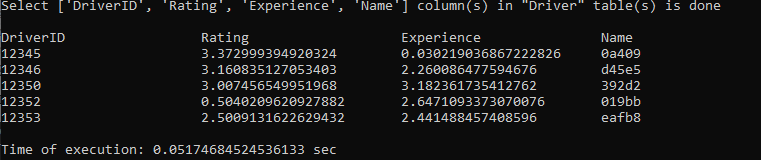


**Пункт №3**:







В PgAdmin: 