МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ХАРКІВСКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ

ІМЕНИ СЕМЕНА КУЗНЕЦЯ

ЗВІТ

про виконання лабораторної роботи №6

з дисципліни «Бази Даних»

Варіант №17

Виконав:

Студент 2 курсу

групи 6.04.121.013.22.2

факультету ІТ

Максаков В. О.

(прізвище та ініціали)

Перевірив:

доц. Бредіхін В. М.

Харків – 2024

Частина 1

Встановлення обмежень цілісності даних

**Цілі роботи**

1. Реалізація надійної обробки даних, що зберігаються у базі.

2. Формулювання бізнес-правил, що забезпечують певний рівень обмежень цілісності даних для заданої бази даних.

3. Реалізація бізнес-правил цілісності даних засобами мови SQL.

4. Тестування обмежень цілісності даних.

**Хід роботи:**

Розробка програмного застосунку:

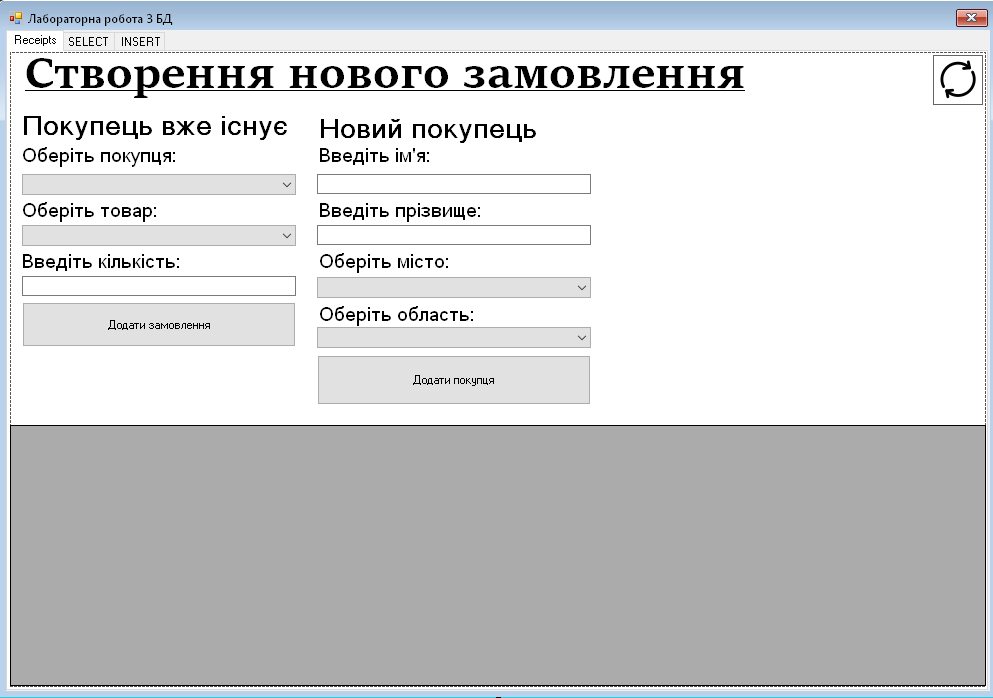
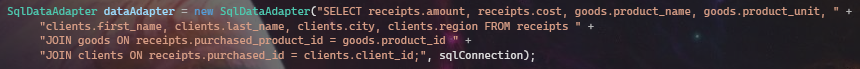


Рис 1 – TabPage1 (Receipts)

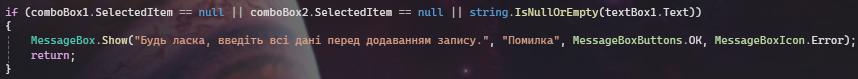
TabPage1 – На цій вкладці користувач може додати нове замовлення для вже існуючого користувача або додати нового користувача.

Також є DataGridView яка виводить дані за цим запитом:

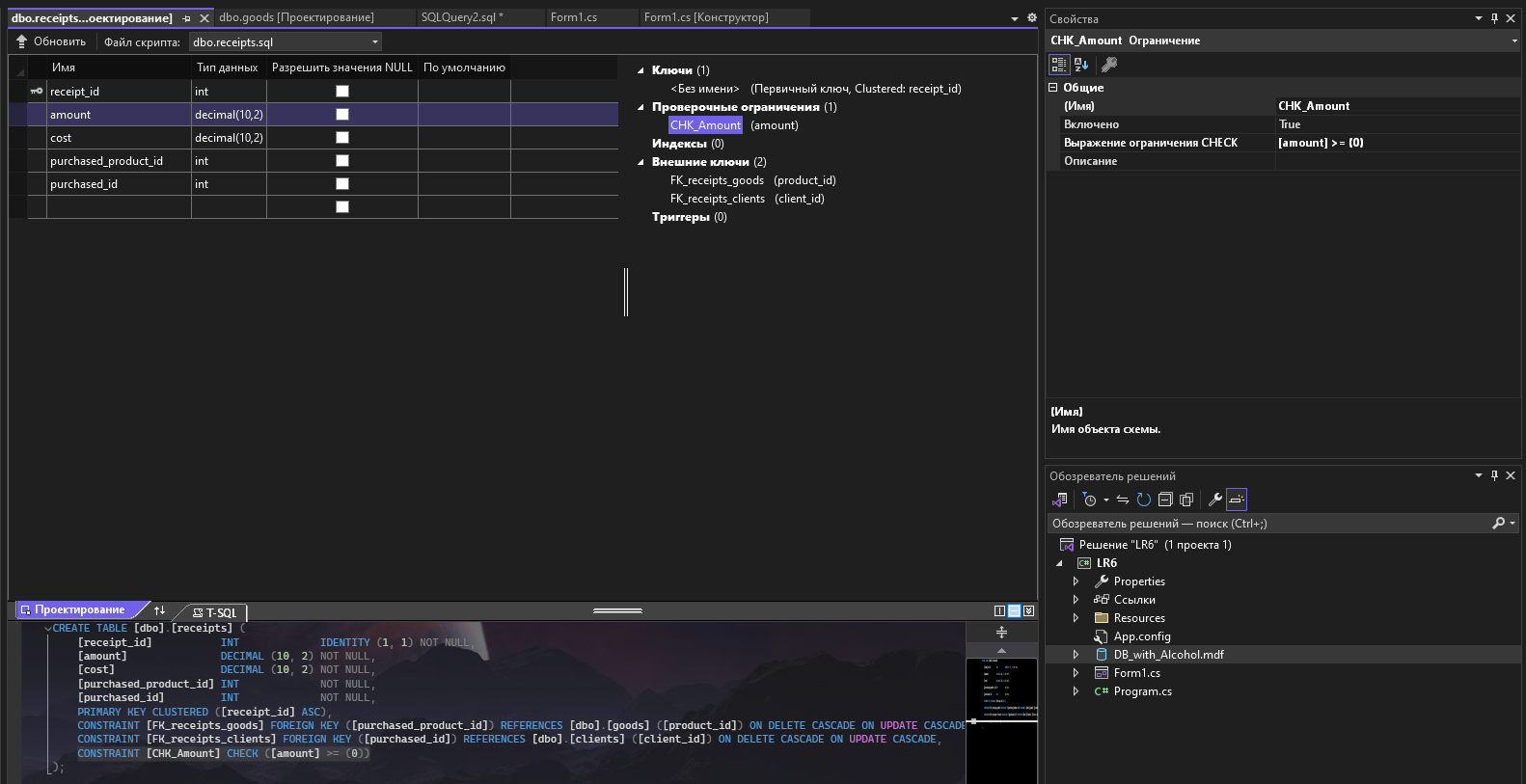


Обмеження:

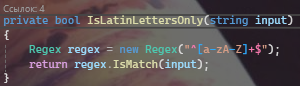
1. Користувач не може додати нове замовлення якщо якесь поле пусте.

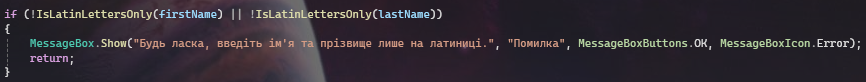


2. Користувач не може додати нове замовлення якщо данні введені не вірно.

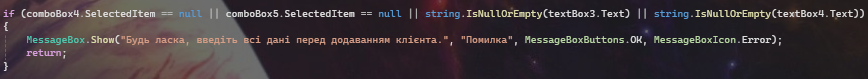


3. Користувач не може додати нового користувача якщо увів прізвище та ім’я кирилицею.





4. Користувач не може додати нового користувача якщо якщо якесь поле пусте.



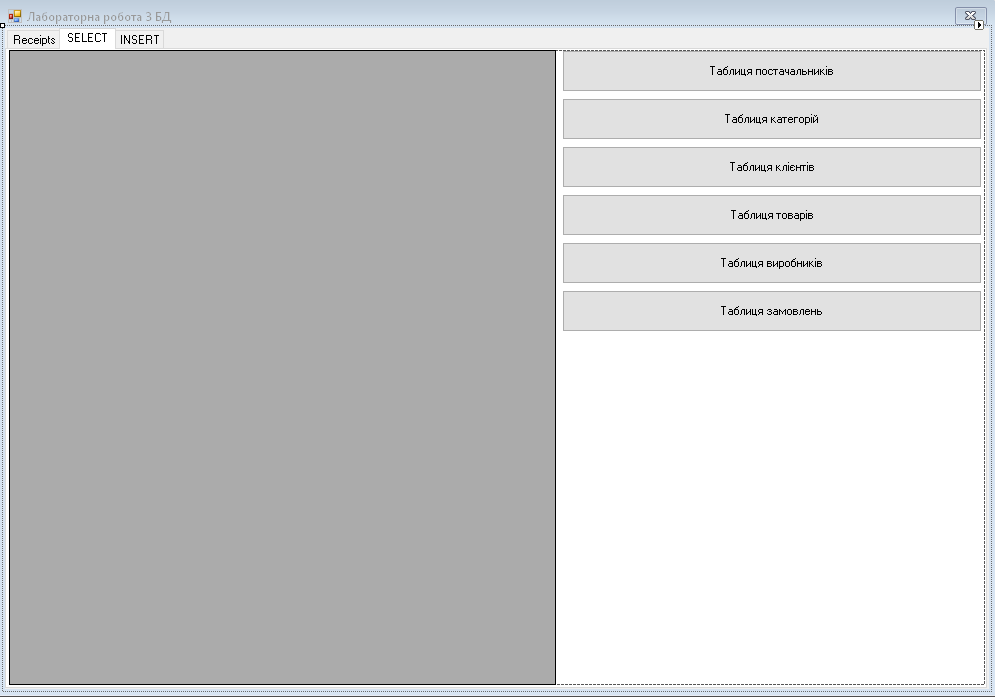


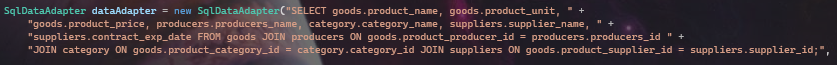
Рис 2 – TabPage2 (SELECT)

TabPage2 – На цій вкладці користувач може виконувати запити (SELECT) до БД та виводити данні у DataGridView.

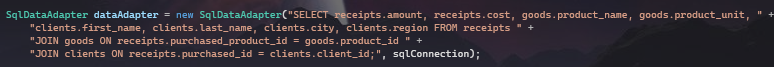












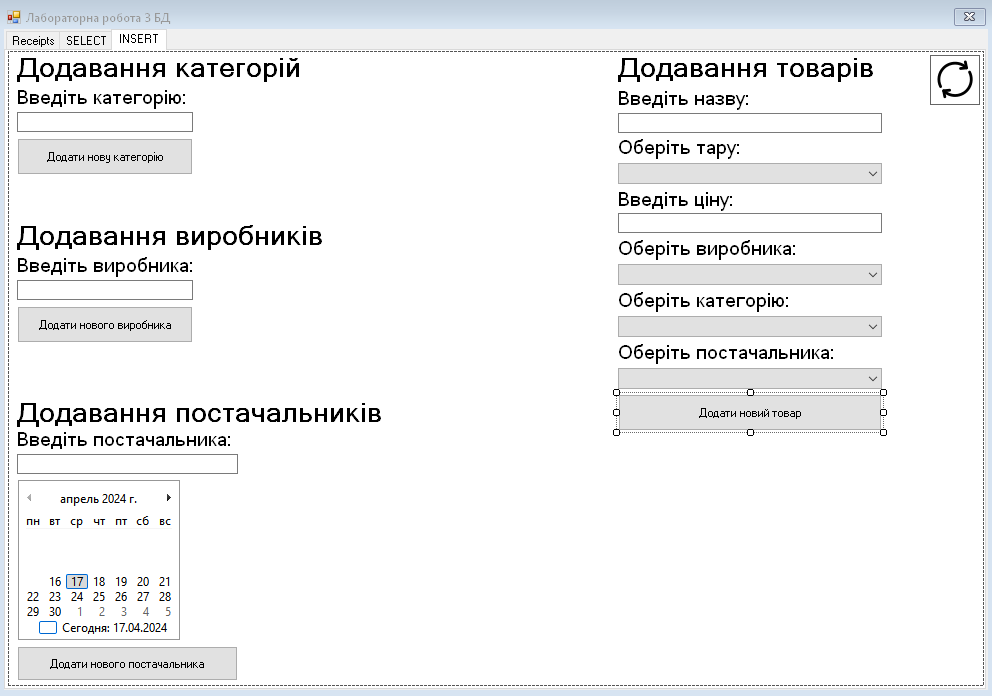


Рис 3 – TabPage3 (INSERT)

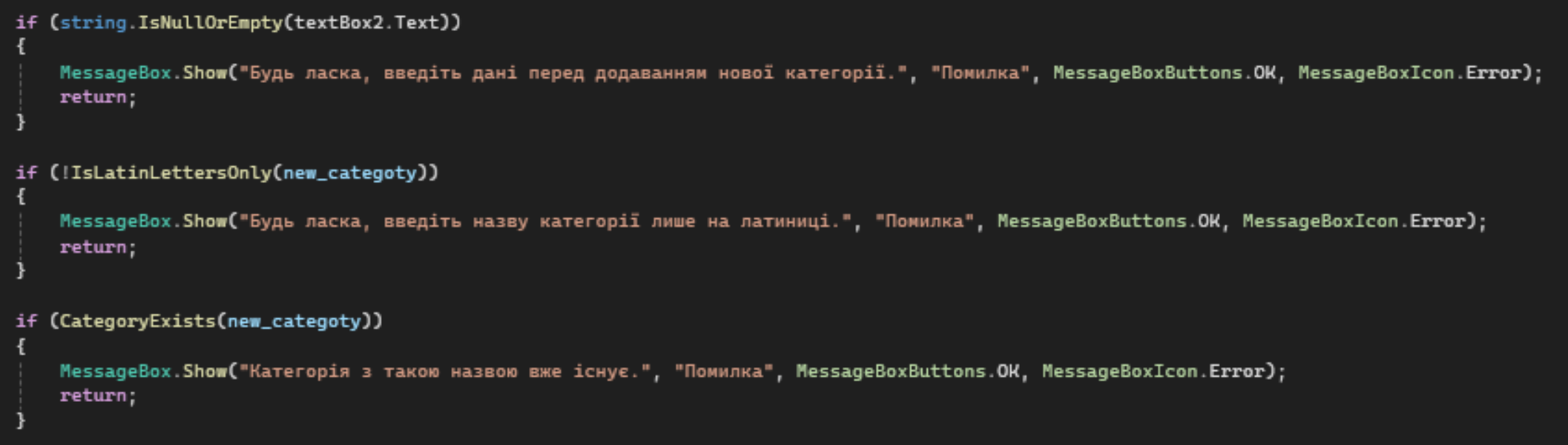
TabPage3 – На цій вкладці користувач додає данні в таблицю “category”, “producers”, “suppliers”, “goods”.

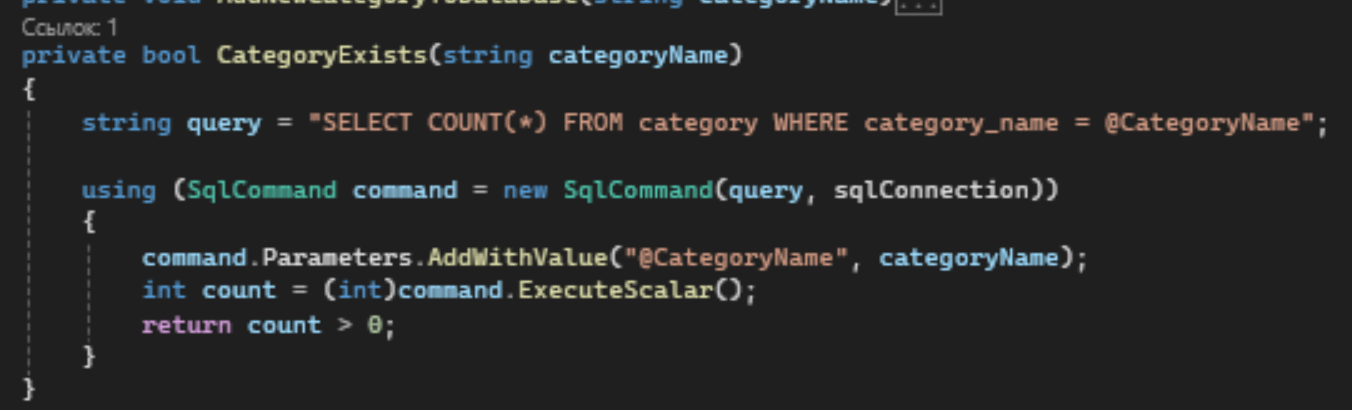
Обмеження:

1. Користувач не може додати нову категорію якщо поле для назви пусте.

2. Користувач не може додати нову категорію якщо назва написана кирилицею.

3. Користувач не може додати нову категорію якщо вона вже існує.

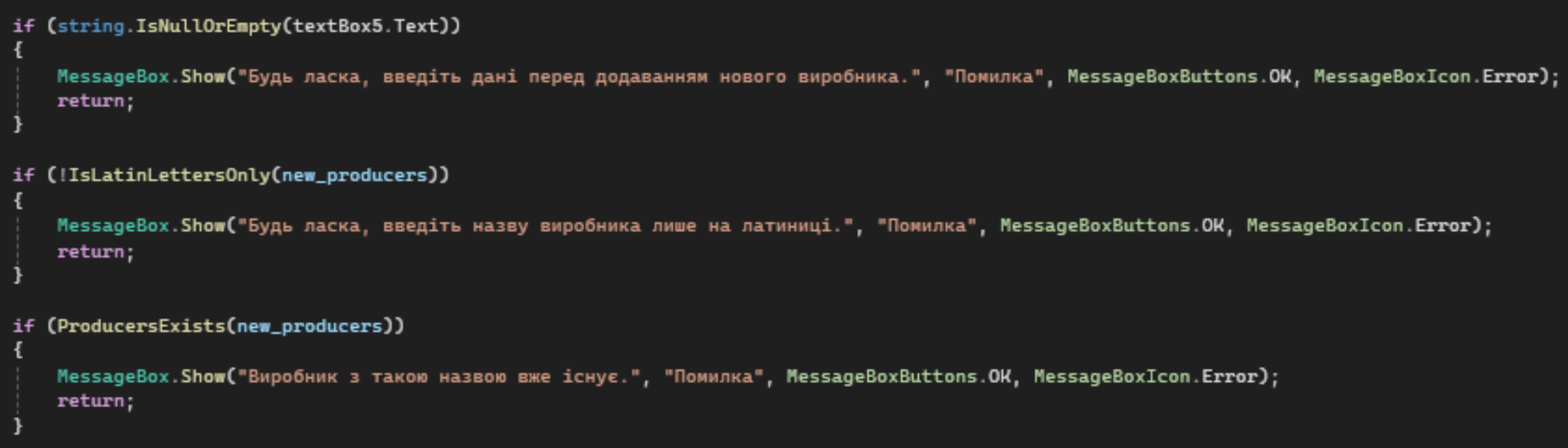


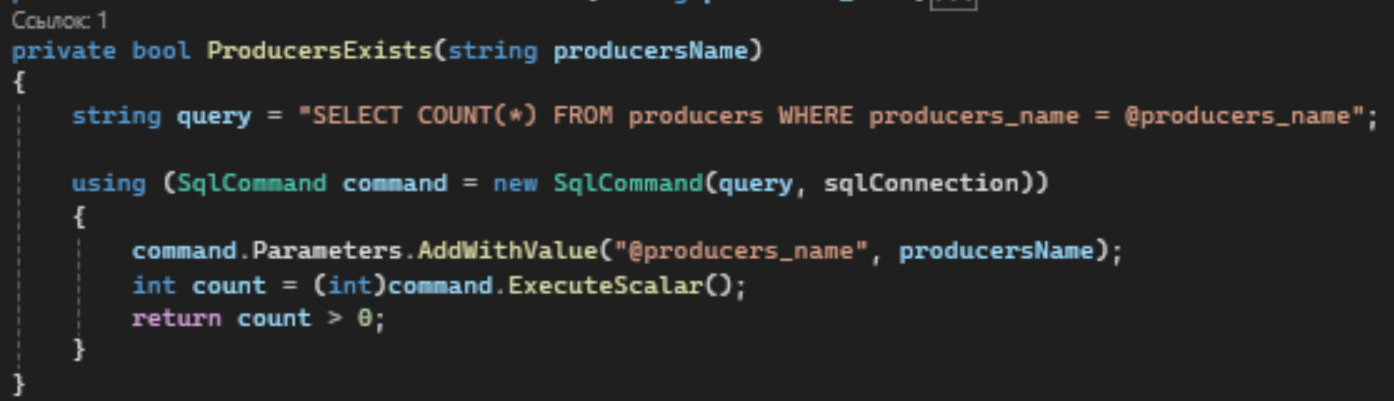


4. Користувач не може додати нового виробника якщо поле для назви пусте.

5. Користувач не може додати нового виробника якщо назва написана кирилицею.

6. Користувач не може додати нового виробника якщо він вже існує.

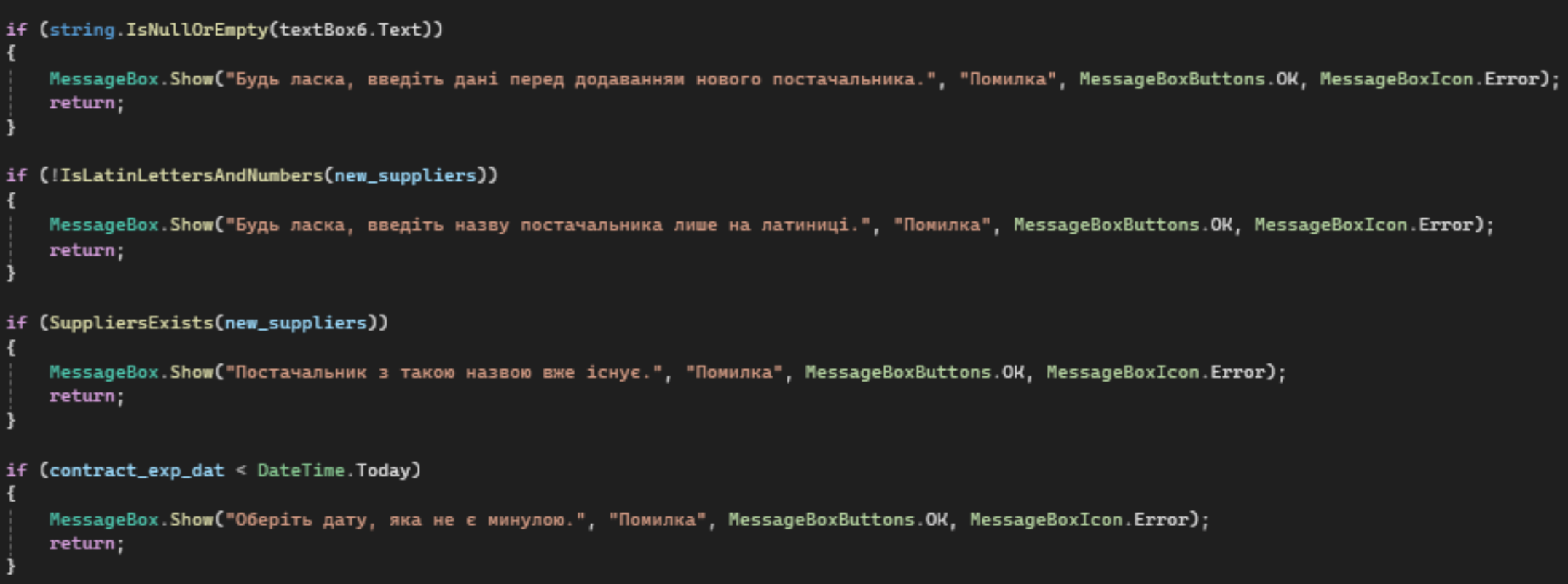


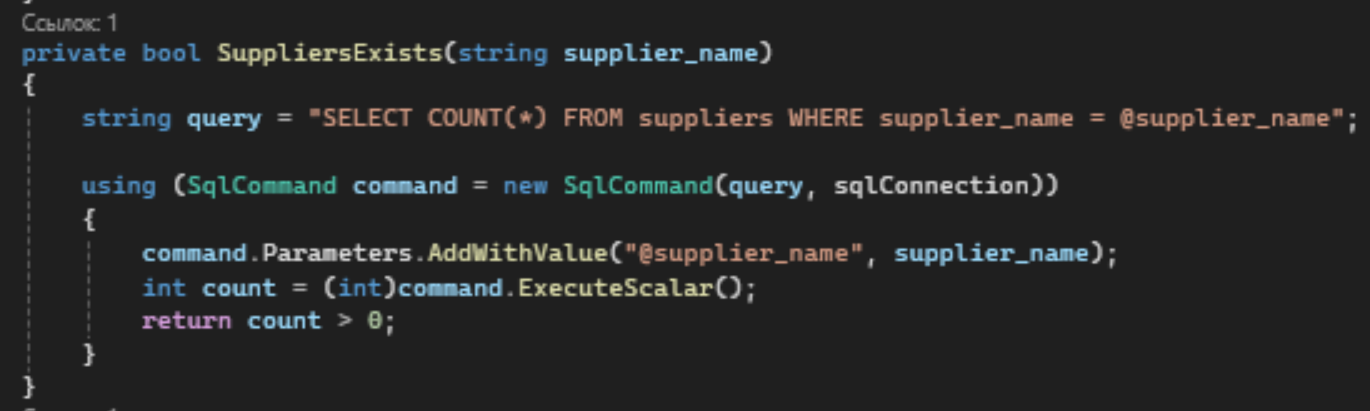


7. Користувач не може додати нового постачальника якщо поле для назви пусте а дата не обрана чи обрана не правильно.

8. Користувач не може додати нового виробника якщо назва написана кирилицею.

9. Користувач не може додати нового виробника якщо він вже існує.

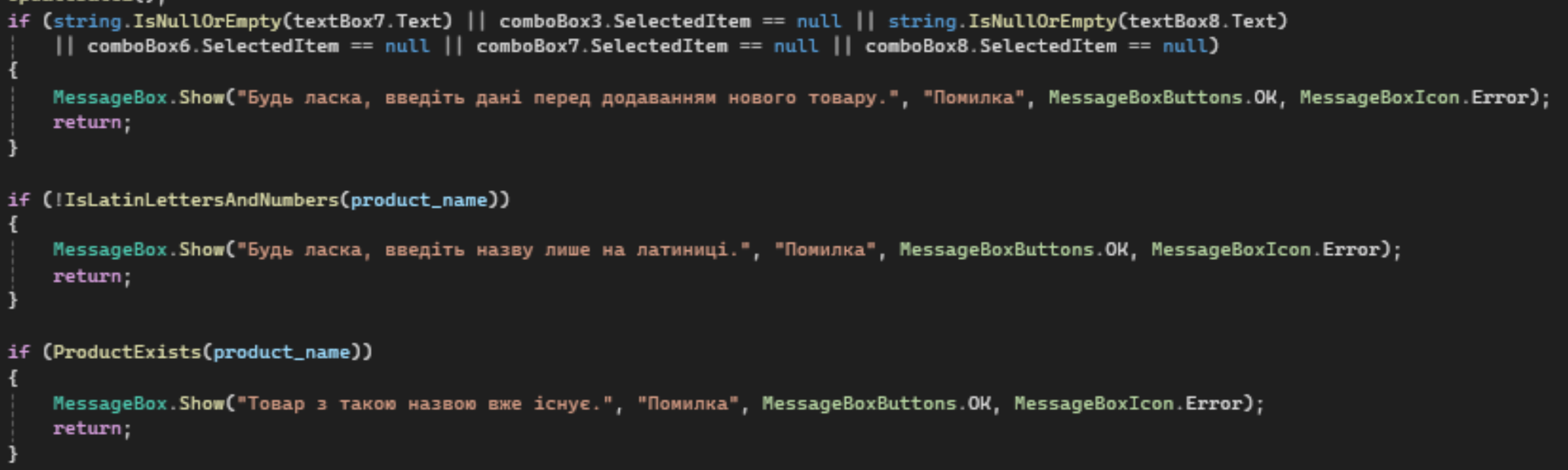


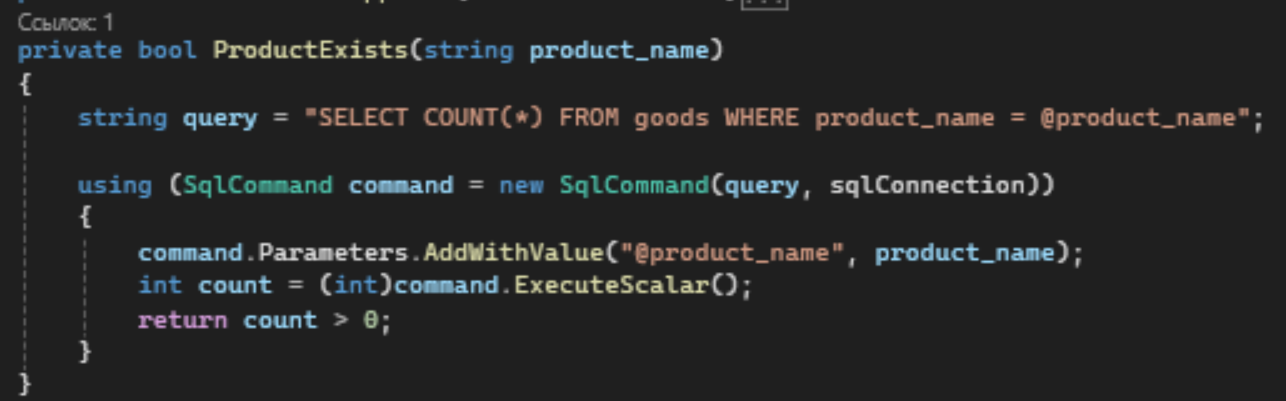


10. Користувач не може додати новий товар якщо поле для назви товару, ціна пусте а тара, категорія, постачальник та виробник чи обрана не правильно.

11. Користувач не може додати нового виробника якщо назва написана кирилицею.

12. Користувач не може додати нового виробника якщо він вже існує.

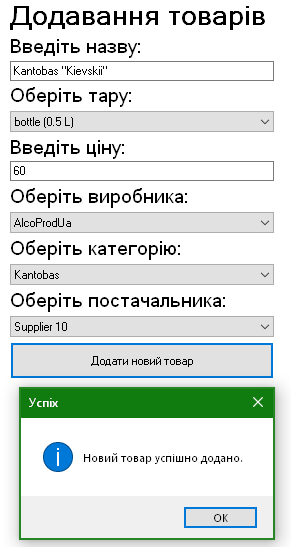
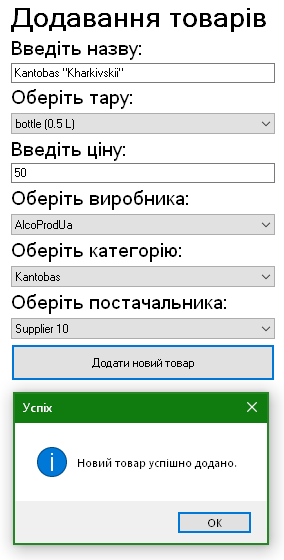




**Рівень атрибута.**

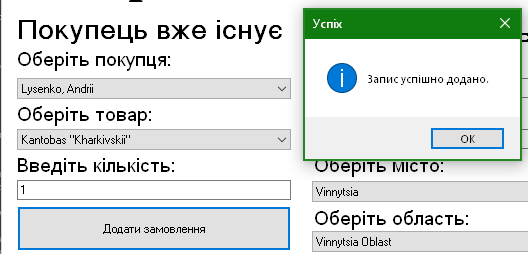
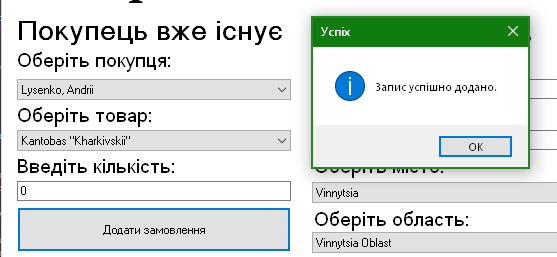
**CHECK ([product\_price]>=(50)):** Це обмеження перевіряє, що ціна кожного товару повинна бути не менше 50. Це забезпечує, що ціна товару завжди є позитивним числом.

****

****При додаванні товару ціна якого менша за 50 виконується виключення.

**CHK\_Amount (Check Constraint):** Кількість товару (amount) у кожному запису у таблиці receipts має бути більше або дорівнювати нулю. Це перевіряє, що кількість товару не може бути від'ємною.



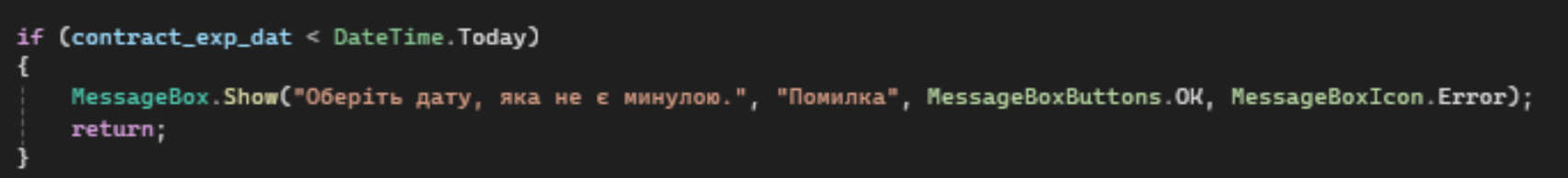
При додаванні замовлення де кількість товару < 0 виконується виключення.

**Рівень кортежа.**

**CHECK ([contract\_exp\_date] IS NOT NULL):** Кожен товар у базі даних повинен мати встановлену дату закінчення договору (contract\_exp\_date). Це обов'язкове поле, яке не може залишатися порожнім. Обмеження CHK\_DateAdded перевіряє, що для кожного запису у таблиці товарів дата закінчення договору обов'язково вказана. Якщо дата закінчення договору не вказана, це вважається порушенням цієї вимоги і заважатиме додаванню або оновленню запису.



Усі записи в таблиці постачальників мають дату постачання.

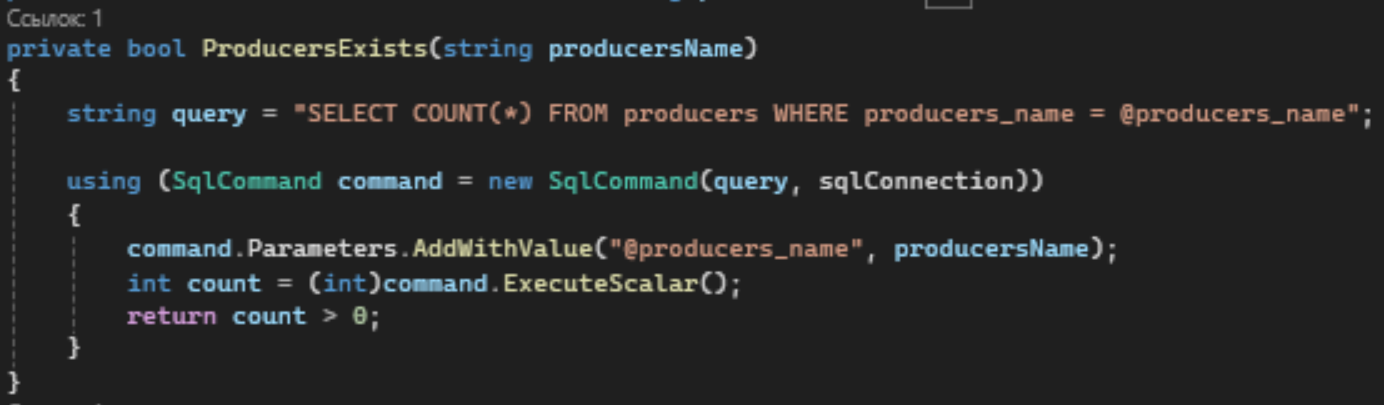


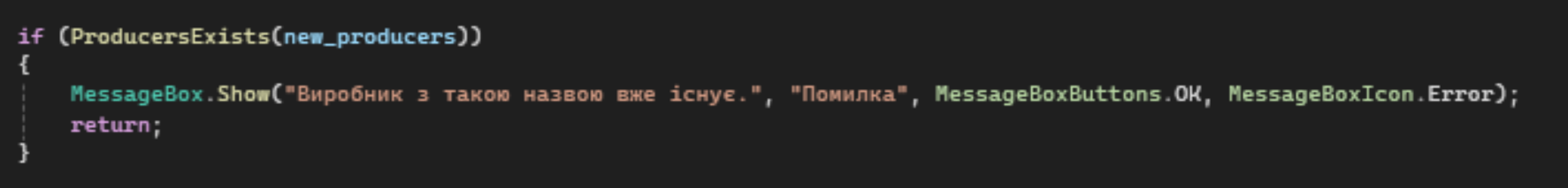
**Рівень відношення.**

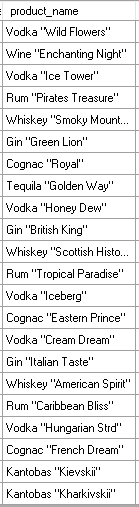
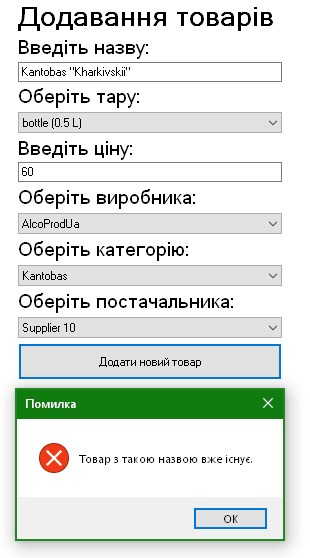
**1. UNIQUE NONCLUSTERED ([product\_name] ASC):** Ім'я кожного товару повинно бути унікальним. Це дозволяє уникнути конфліктів при додаванні товарів з однаковими назвами.



Усі данні в стовпчику product\_name є унікальними.





**Рівень бази даних.**

**1. FOREIGN KEY ([product\_producer\_id]) REFERENCES [dbo].[producers] ([producers\_id]) ON DELETE CASCADE ON UPDATE CASCADE:** Кожний товар повинен мати посилання на виробника, який поставляє його. Це забезпечує зв'язок між таблицею товарів та таблицею виробників.

**2. FOREIGN KEY ([product\_category\_id]) REFERENCES [dbo].[category] ([category\_id]) ON DELETE CASCADE ON UPDATE CASCADE:** Кожен товар повинен належати до певної категорії. Це забезпечує зв'язок між таблицею товарів та таблицею категорій.

**3. FOREIGN KEY ([product\_supplier\_id]) REFERENCES [dbo].[suppliers] ([supplier\_id]) ON DELETE CASCADE ON UPDATE CASCADE:** Кожний товар повинен бути постачаний певним постачальником. Це забезпечує зв'язок між таблицею товарів та таблицею постачальників.

