МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

КРЕМЕНЧУЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ІМЕНІ МИХАЙЛА ОСТРОГРАДСЬКОГО

КАФЕДРА АВТОМАТИЗАЦІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ

ЗВІТ З ЛАБОРАТОРНИХ РОБІТ

З НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

**«СУЧАСНІ ПАРАДИГМИ ЗБЕРЕЖЕННЯ ДАНИХ»**

Виконав: студент групи КН-22-1(а)

Панцюк К.В.  
Перевірив: Андреєв П.І.

КРЕМЕНЧУК 2025

**Лабораторна робота № 2**

**Тема. Розробка моделей даних в MongoDB**

**Мета:** навчитися створювати та використовувати моделі даних у MongoDB для зберігання й обробки інформації.

**Порядок виконання роботи**

1. Отримати індивідуальний варіант завдання.

2. Спроєктувати схему БД.

3. Реалізувати схему у MongoDB.

4. Створити представлення (view), яке об'єднує створені колекції.

**Виконання роботи**

1.Індивідуальне завдання:

Індивідуальне завдання

Розробити схему бази даних для інтернет-магазину електроніки у бізнес-домені Електронна комерція (E-commerce). Система має задовольняти наступні вимоги:

1. Кожен товар належить до певної категорії та підкатегорії
2. Товари мають різні технічні характеристики залежно від категорії
3. Користувачі можуть додавати товари до кошика та створювати замовлення
4. Користувачі можуть залишати відгуки та рейтинги для товарів
5. Система має зберігати історію переглядів товарів
6. Адміністратори можуть керувати складом та цінами товарів
7. Система знижок та промокодів
8. Управління доставкою та відділеннями

**Бізнес-домен:** Електронна комерція (E-commerce)

Ця предметна область охоплює онлайн-продаж електронних товарів, управління каталогом продуктів, обробку замовлень та взаємодію з клієнтами.

2.Проектуємо схему бази даних:



Рисунок 2.1–Схема бази даних відповідно до поставленої задачі

3.Реалізуємо схему у MongoDB:

Для цього напишемо скрипт для створення бази даних та відповідних колецій зі схеми зазначеної вище.

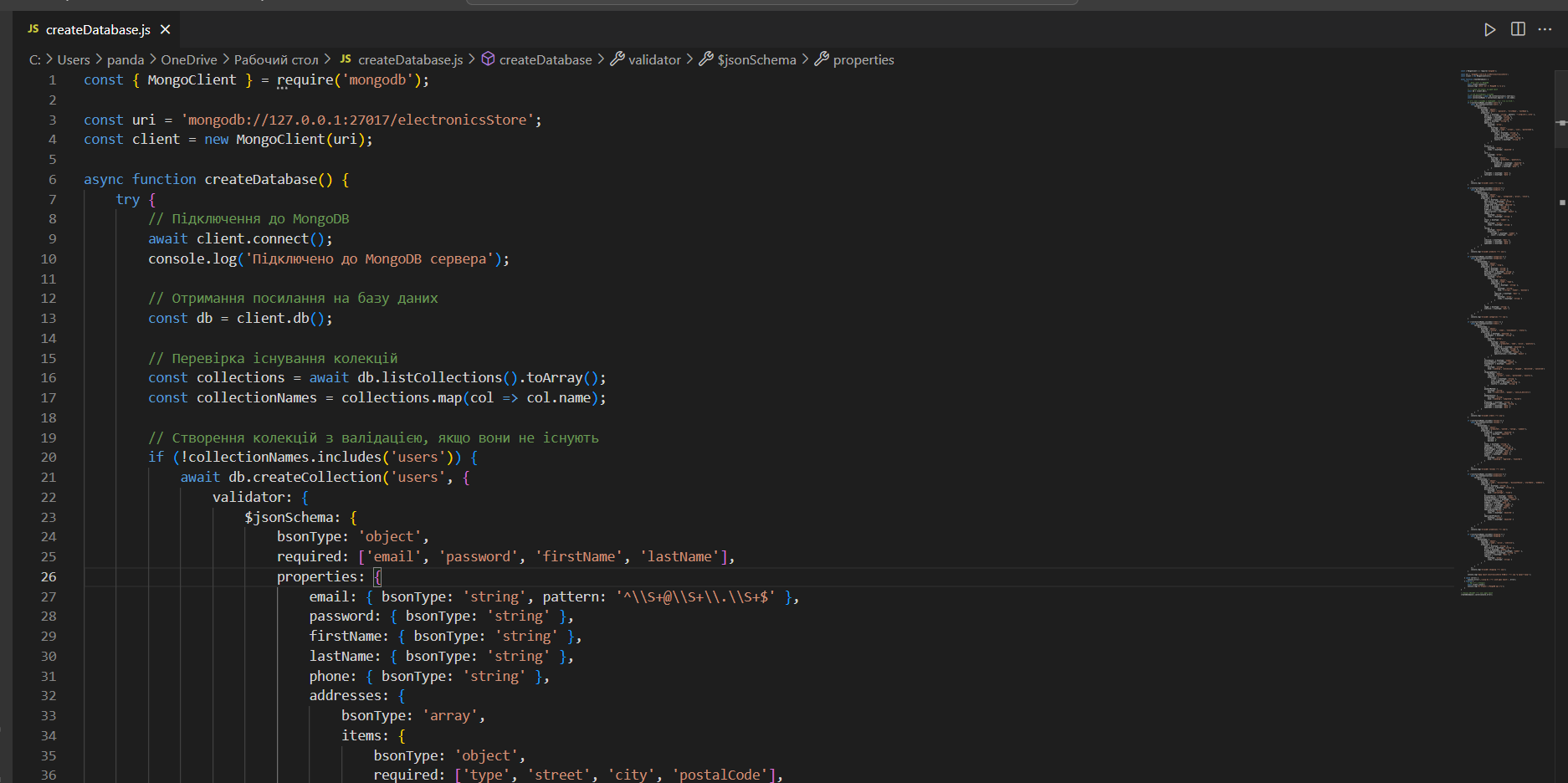


Рисунок 2.2–Частина скрипту для створення БД

У терміналі отримуємо наступний результат :

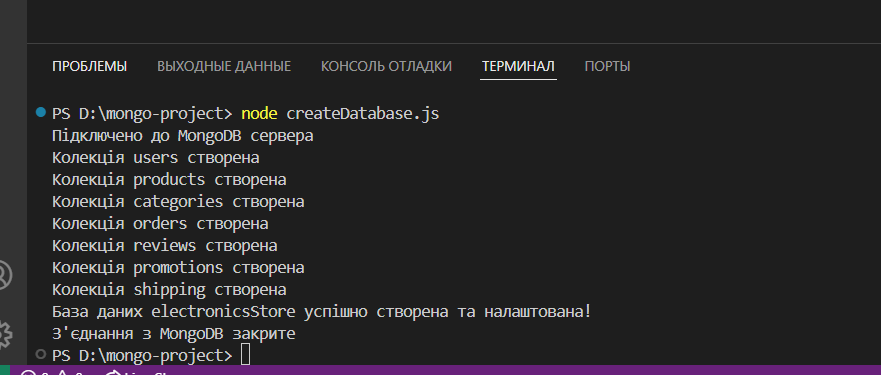


Рисунок 2.3–Результат виконання скрипта для створення БД

Заповнимо БД інформацією:

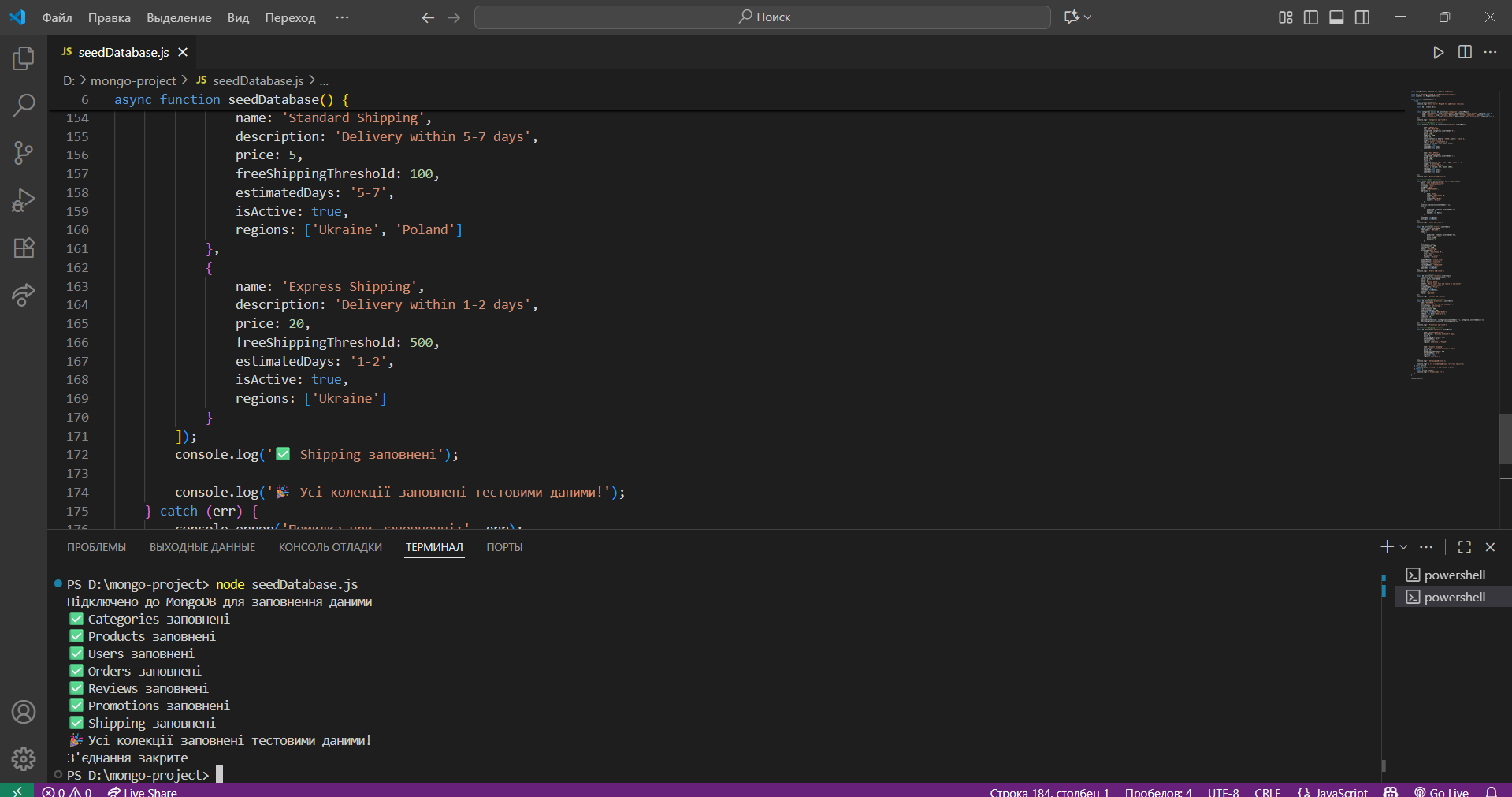


Рисунок 2.4–Результат виконання скрипта для заповнення БД інформацією

4. Створюємо представлення (view), яке об'єднує створені колекції.

Представлення orderDetails показує всі замовлення разом з інформацією про користувача та товари в замовленні. Воно містить номер замовлення, користувача, список товарів із їх ціною та кількістю, фінансові підсумки замовлення, статус, адресу доставки, метод оплати та промокод. Фактично об’єднує колекції orders, users та products, щоб мати всі потрібні дані для перегляду замовлення в одному місці.

Представлення productDetails показує всі продукти разом із категоріями та рейтингом. Воно містить назву, опис, артикул, бренд, ціну, характеристики, зображення, кількість на складі, теги та статус активності. Об’єднує інформацію з колекцій products, categories та reviews, щоб можна було бачити повний профіль продукту в одному документі.

Представлення userActivity показує активність користувачів. Воно містить ім’я, прізвище, email користувача, кількість замовлень, останній вхід, кількість товарів у корзині та в списку бажань. Об’єднує колекції users та orders, щоб швидко бачити, хто і що замовляв та наскільки активний.



Рисунок 2.5 –Код для створення представлень

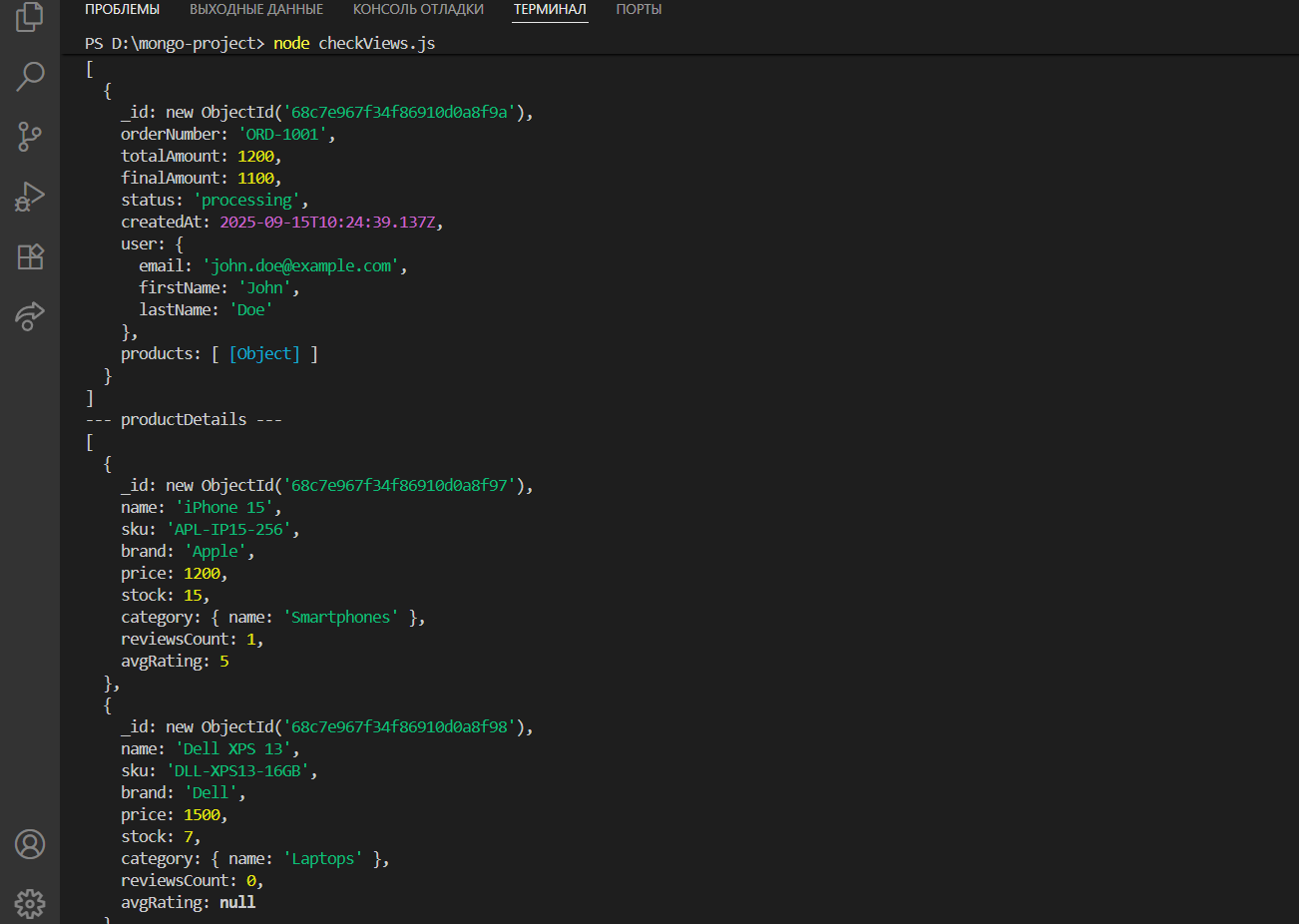


Рисунок 2.6 –Вивід представлень

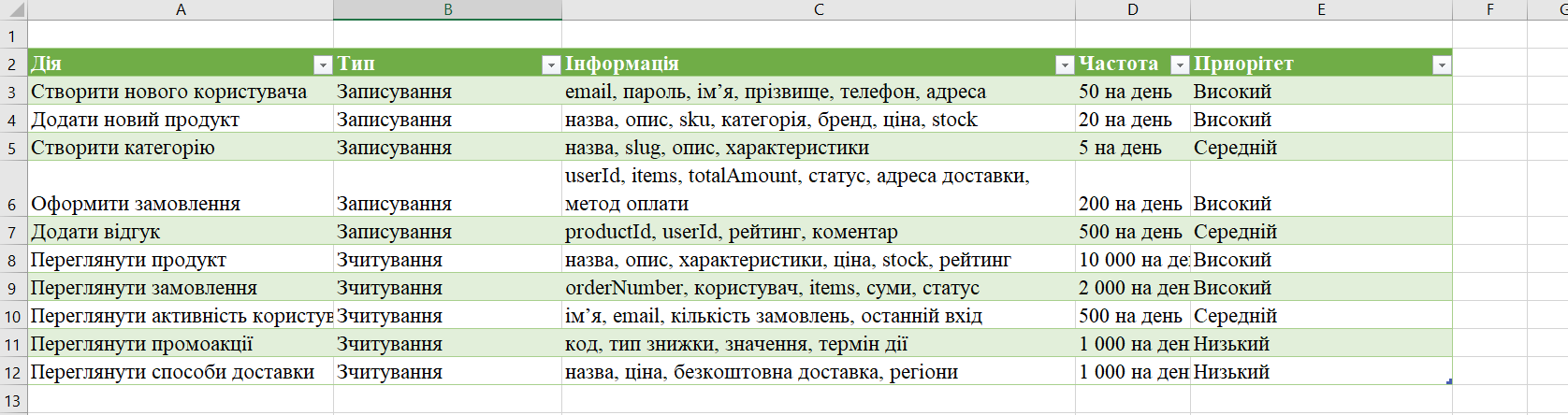


Рисунок 2.7–Таблиця дій їх типів та частоти виконання

**Відповіді на контрольні питання**

**Як обрати між вбудованими документами та референсами при моделюванні даних у MongoDB?**

Вибір залежить від шаблонів доступу до даних. Вбудовані документи кращі для даних, які часто зчитуються разом, мають невеликий розмір і рідко оновлюються. Референси кращі для окремих сутностей, які часто оновлюються, мають великий розмір або входять у зв'язки "багато-до-багатьох".

**Що таке денормалізація в контексті MongoDB, і в яких випадках її варто застосовувати при проєктуванні моделі даних?**

Денормалізація - це навмисне дублювання даних для покращення продуктивності читання. Її варто застосовувати для даних, які рідко оновлюються, але часто зчитуються разом, або коли потрібно уникнути дорогих операцій об'єднання.

**Що таке View у MongoDB, як вона створюється, і які її обмеження?**

View - це віртуальна колекція, результат агрегаційного запиту. Створюється командою db.createView(). Обмеження: лише для читання, не можна створювати індекси, продуктивність залежить від складності базового пайплайну.

**Як змінити або видалити представлення?**

Для видалення використовують db.viewName.drop(). Для зміни - видаляють і створюють нове представлення або використовують команду collMod.

**Що таке обмежені колекції?**

Це колекції фіксованого розміру, які автоматично видаляють найстаріші документи при заповненні. Використовуються для логування, кешування та черг повідомлень.

**Як забезпечується узгодженість даних при дублюванні даних?**

Через транзакції для критичних даних, паттерн "Versioned Documents", фонові процеси синхронізації, оптимістичне блокування та використання Change Streams для відстеження змін.