**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ**

**КУРСОВА** **РОБОТА**

з дисципліни

**«Технології проектування та адміністрування баз даних та сховищ даних»**

на тему

**«Розробка стратегії, аналіз, концептуальне моделювання та проектування бази даних для інтернет-магазину з продажу комп’ютерних ігор»**

**Виконав**:

Студент групи ПД-34

Солов’ян Арсен

**Керівник курсової роботи:**

викладач кафедри ТЦР, Майборода М.В.

Київ – 2024

**ЗМІСТ**

**Вступ**

[**1. Стратегія автоматизації предметної області**](#_heading=h.tyjcwt)

[**1.1. Загальні положення**](#_heading=h.3dy6vkm)

[**1.2. Мета, цілі та задачі створення бази даних та додатку**](#_heading=h.1t3h5sf)

[**1.3. Вимоги до інформаційного забезпечення**](#_heading=h.4d34og8)

[**2. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ**](#_heading=h.2s8eyo1)

[**2.1. Загальні положення системного аналізу предметної області**](#_heading=h.17dp8vu)

[**2.2. Загальний опис предметної області**](#_heading=h.3rdcrjn)

[**2.3. Системний аналіз предметної області**](#_heading=h.26in1rg)

[**3. Концептуальне моделювання предметної області**](#_heading=h.2xcytpi)

[**3.1. Теоретичні положення концептуального моделювання**](#_heading=h.1ci93xb)

[**3.2. Даталогічна модель**](#_heading=h.3whwml4)

[**4. Логічне та фізичне проектування бази даних та інформаційної системи**](#_heading=h.qsh70q)

[**4.1. Логічне проектування**](#_heading=h.3as4poj)

[**4.2. Фізичне проектування**](#_heading=h.1pxezwc)

**4.3 Код фізичного проектування бази даних**

**4.4. Інформаційно-пошукові запити**

**5**[**. Висновки**](#_heading=h.3o7alnk)

**6. Список використаних джерел**

**ВСТУП**

Метою цієї курсової роботи є розробка бази даних предметної області, яка має відношення до інтернет-магазину цифрових ігор. Створення будь-якої програмної системи, зокрема бази даних, є складним процесом, який складається з численних етапів. Існує багато методологій по опису життєвого циклу проектування та розробки баз даних. У цій курсовій роботі буде використано методологію, згідно з якої життєвий цикл складається з наступних етапів:

* розробка стратегії автоматизації предметної області;
* проведення системного аналізу предметної області;
* концептуальне моделювання предметної області;
* логічне та фізичне проектування.

Як відомо, усяка предметна область складається з двох компонент: переліку задач, які повинні автоматизуватися, та інформації, на основі якої задачі вирішуються. Приймаючи до уваги, що курсова робота має відношення до проблематики баз даних, ми опишемо усі ці етапи не по відношенню до всієї предметної області, а тільки до її інформаційної моделі.

Головною ціллю курсового проекту є проектування бази даних для інтернет-магазину цифрових ігор.

**1. Стратегія автоматизації предметної області**

**1.1 Загальні положення**

Сучасний етап розвитку інформаційних технологій вимагає від підприємств і підприємців ефективних рішень для оптимізації та автоматизації бізнес-процесів. Однією з ключових складових цього процесу є створення та впровадження баз даних і інформаційних систем.

Метою даної курсової роботи є розробка бази даних та інформаційного додатку для інтернет-магазину, спеціалізованого на продажу комп'ютерних ігор. Проект передбачає ретельний аналіз предметної області, логічне та фізичне проектування бази даних, вибір технологій для створення інформаційної системи та розробку інформаційно-пошукових запитів.

**1.2 Мета, цілі та задачі створення бази даних та додатку**

Метою даного проекту є підвищення ефективності управління інтернет-магазином з продажу комп'ютерних ігор шляхом автоматизації облікових та торговельних операцій.

Основні цілі проекту включають:

* Створення зручного інтерфейсу для користувачів магазину;
* Впровадження системи обліку товарів, замовлень та клієнтів;
* Забезпечення надійності та цілісності даних через раціональне проектування бази даних;
* Підвищення швидкості та продуктивності роботи інтернет-магазину.

Для досягнення поставлених цілей, розглянемо основні задачі, які необхідно вирішити в рамках проекту:

* Розробка концептуальної моделі предметної області;
* Логічне та фізичне проектування бази даних;
* Вибір технологій для створення інформаційної системи;
* Розробка інформаційно-пошукових запитів для забезпечення продуктивної роботи системи.

**1.3 Вимоги до інформаційного забезпечення**

З метою ефективного функціонування інтернет-магазину, необхідно визначити ключові вимоги до

інформаційного забезпечення.

Основні вимоги включають:

* Масштабованість системи для обробки різноманітних обсягів даних;
* Зручний та інтуїтивно зрозумілий інтерфейс для користувачів;
* Безпеку даних та забезпечення конфіденційності особистої інформації клієнтів;
* Надійність та стійкість системи до витоків та атак.

Результатом успішної реалізації цих вимог буде створення ефективної та сучасної інформаційної системи для управління інтернет-магазином комп'ютерних ігор.

**2. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ**

**2.1 Загальні положення системного аналізу предметної області**

Системний аналіз є ключовим етапом у створенні інтернет-магазину з продажу комп'ютерних ігор. Він охоплює детальне вивчення функціональних та нефункціональних вимог до системи. На цьому етапі визначаються основні цілі та завдання магазину, а також взаємозв'язки між його складовими частинами.

Аналіз включає:

* проведення всіляких бесід з користувачами й узяття в них інтерв'ю;
* перегляд всіх циркулюючих в організації документів, бланків;
* аналіз потоку документів (документообіг);
* аналіз розв'язуваних в організації завдань і способів їхнього рішення;
* фіксація всіляких правил, обмежень, законів, що діють у ПО.

Факторами успіху проведення аналізу ПО є наступні:

* активна участь а проведенні аналізу не тільки системних аналітиків, а і всіх тих, хто буде використовувати розроблену систему;
* ретельна перевірка вірогідності, повноти, несуперечності отриманої інформації .
* виявлення всіх питань і припущень, що мають ключове значення для проектування й впровадження;
* точні об'ємно-частотні характеристики даних;
* твердий контроль за ходом робіт, повна концентрація зусиль на виконанні календарних планів і дотриманні запланованих строків.

**2.2. Загальний опис предметної області**

Інтернет-магазин з продажу комп'ютерних ігор є високотехнологічним та конкурентноздатним бізнесом. Основні складові системи включають в себе:

**Товари**

* Асортимент комп'ютерних ігор, поділений за категоріями, жанрами та іншими характеристиками.
* Інформація про кожну гру, включаючи назву, опис, ціну та наявність.

**Клієнти**

* Реєстрація та авторизація клієнтів.
* Профіль користувача з історією замовлень та особистими налаштуваннями.

**Замовлення**

* Система замовлення та його обробка.
* Інформація про статус та історію кожного замовлення.

**Система знижок**

* Реалізація різноманітних систем знижок для привертання клієнтів та збільшення продажів.

**Постачальники**

* Взаємодія з постачальниками комп'ютерних ігор.
* Облік та контроль постачання товарів.

**2.3. Системний аналіз предметної області**

Системний аналіз включає в себе вивчення взаємодії між основними бізнес-процесами:

**Управління асортиментом**

* Визначення нових видань та вилучення застарілих.
* Категоризація товарів для полегшення навігації покупців.

**Облік замовлень та клієнтів**

* Реєстрація нових клієнтів та управління їхніми особистими даними.
* Ефективна система обліку та виконання замовлень.

**Управління системою знижок**

* Розробка та впровадження різноманітних програм знижок для різних категорій товарів та клієнтів.

**Взаємодія з постачальниками**

* Забезпечення постачальників актуальною інформацією щодо попиту на товари.

**3. Концептуальне моделювання предметної області**

**3.1 Теоретичні положення концептуального моделювання**

Етап концептуального моделювання - це побудова строго опису ПО в термінах деякої формальної мови. На підставі змістовного опису ПО, побудованого в результаті виконання етапу аналізу, будується строгий формальний опис інформаційного забезпечення ПО, що автоматизується.

Концептуальне моделювання призначене для інтегрованого опису інформаційного забезпечення ПО, що автоматизується, не залежно від її сприйняття окремими користувачами й від способів її реалізації в комп'ютерній системі.

Концептуальне моделювання є ключовим етапом у розробці інтернет-магазину з продажу комп'ютерних ігор. Цей етап визначає основні концепції та взаємозв'язки між сутностями системи.

**Сутності предметної області**

1. **Товари (Product)**
   * Атрибути: ідентифікатор (ID), назва, опис, ціна, категорія, наявність.
   * Зв'язки: із замовленнями, постачальниками.
2. **Клієнт (Customer)**
   * Атрибути: ID, ім'я, прізвище, електронна пошта, адреса доставки.
   * Зв'язки: із замовленнями.
3. **Замовлення (Order)**
   * Атрибути: ID, дата замовлення, статус, загальна сума.
   * Зв'язки: із товарами, клієнтами, системою знижок.
4. **Система знижок (Discounts)**
   * Атрибути: ID, назва, опис, відсоток знижки.
   * Зв'язки: із замовленнями.
5. **Постачальник (Supplier)**
   * Атрибути: ID, назва, адреса, електронна пошта, телефон.
   * Зв'язки: із товарами.

**Зв'язки між сутностями**

* Кожен товар може бути частиною багатьох замовлень, а також мати зв'язки із системою знижок та постачальниками.
* Клієнти роблять замовлення та можуть користуватися системою знижок.
* Замовлення пов'язані з товарами, клієнтами та системою знижок.
* Система знижок впливає на замовлення та клієнтів.
* Постачальники постачають товари.

**Інфологічна модель**

Інфологічна модель створюється на основі досліджень предметної області і є описом майбутньої бази даних. Її основна мета - надати адекватний опис процесів, інформаційних потоків і функцій системи за допомогою природної мови, формул, графіків, діаграм, таблиць та інших інструментів, що зрозумілі як розробникам, так і користувачам бази даних.

Створення інфологічної моделі є продовженням досліджень предметної області, але вже з точки зору проектувальника. Модель включає в себе такі конструктивні елементи, як сутності, зв'язки та їх властивості (атрибути).

Сутність - це будь-який помітний об'єкт, інформацію про який необхідно зберігати в базі даних. Сутність може представляти різні об'єкти, такі як люди, місця, об'єкти, події і т.д. Тип сутності відноситься до групи однорідних об'єктів, а екземпляр сутності - до конкретного представника цієї групи.

Атрибут - це пойменована характеристика сутності. Вони використовуються для визначення інформації, яку необхідно зберігати про сутність. Атрибути можуть мати унікальні найменування в межах конкретного типу сутностей.

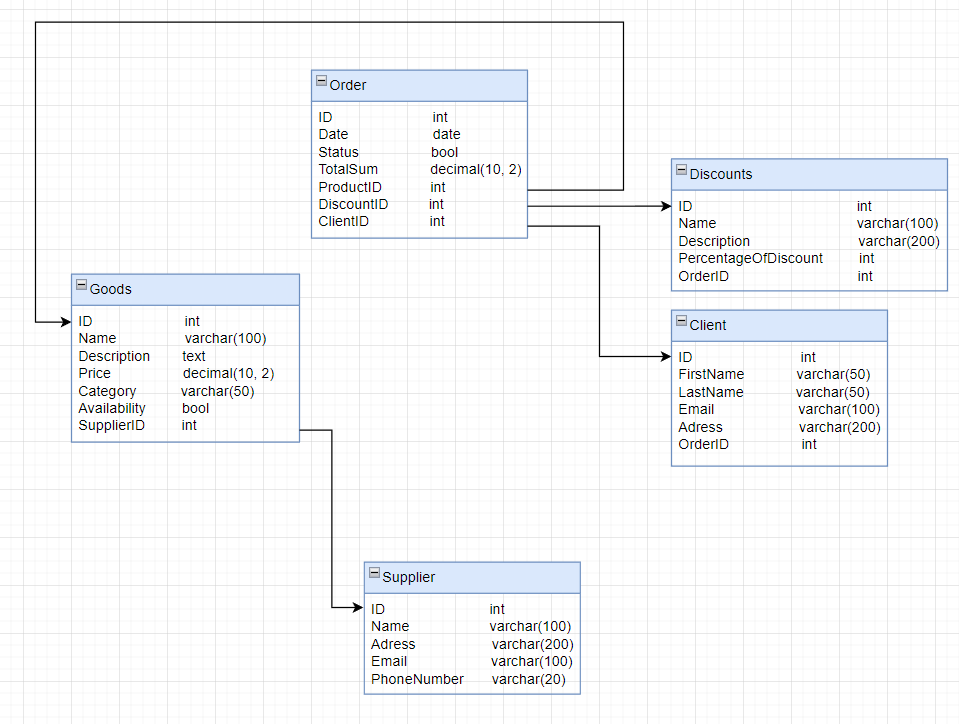
Ключ - це мінімальний набір атрибутів, за значеннями яких можна однозначно ідентифікувати екземпляр сутності. Ключ повинен бути мінімальним, тобто виключення будь-якого атрибуту з набору не дозволить ідентифікувати сутність.

Зв'язок - це асоціація між двома або більше сутностями. Відносини між сутностями створюються для забезпечення можливості відшукання однієї сутності за значеннями інших.

При побудові інфологічних моделей часто використовується мова ER-діаграм (Entity-Relationship). У цих діаграмах сутності представлені прямокутниками, а зв'язки - ромбами або шестикутниками. Атрибути зображуються овалами, а зв'язки між сутностями - ненаправленими ребрами з відповідною ступенем зв'язку (один або багато) і поясненнями.

Мова ER-моделювання базується на концепції представлення інформації про предметну область як сукупності взаємозалежних об'єктів. Процес моделювання включає виділення сутностей, визначення їхніх властивостей та встановлення існуючих між ними зв'язків.

Для конкретної предметної області була розроблена наступна інфологічна модель:

**

*Рисунок 3 Інфологічна модель інтернет-магазину цифрових ігор*

**3.2. Даталогічна модель**

Моделювання сутностей і зв'язків може використовуватися не тільки на етапі концептуального моделювання, але і на етапах розробки стратегії і аналізу. Воно ставить перед собою завдання створення точної та адекватної моделі інформаційних потреб організації.

Даталогічна модель є моделлю логічного рівня і відображає логічні зв'язки між елементами даних, незалежно від їхнього змісту та середовища зберігання. Ця модель будується в термінах інформаційних одиниць, припустимих у тій конкретній СУБД, у середовищі якої ми проектуємо базу даних. Етап створення даталогічної моделі називається даталогічним проектуванням. Опис логічної структури бази даних мовою СУБД називається схемою.

Розглянемо коротко основні властивості, формальні позначення та визначення сутностей, зв'язків і атрибутів.

**Сутність** — це реальний або уявлюваний об'єкт інтересу, інформація про який підлягає збору або зберіганню. Графічно сутність представляється пойменованим прямокутником із закругленими кутами. Ім'я сутності дається в однині і пишеться заголовними буквами. Ім'я сутності повинне представляти тип або клас об'єктів, а не окремий екземпляр. Будь-який предмет або об'єкт може бути представлений тільки однією сутністю. Сутності завжди є взаємовиключними.

**Зв'язок** — це деяка пойменована асоціація, що представляє інтерес двох сутностей. Зв'язок є бінарним, оскільки це завжди асоціація в точності двох сутностей або сутності із самої себе. Кожний зв'язок має два кінці зі своїми властивостями, такими як ім'я, ступінь/потужність та факультативність (обов'язковість).

Ці властивості використовуються для опису асоціації з кожної сторони. Щоб задати зв'язок, потрібно визначити його обидва кінці. На діаграмах зв'язки представляються лініями, що з'єднують два прямокутники сутностей.

**Атрибут** — це будь-яка деталь або аспект, що сприяють якісному або кількісному опису сутності, її ідентифікації, класифікації або відображенню стану. Атрибут може бути текстом, числом, зображенням, почуттям, запахом та іншими даними. Для подання атрибута використовується його ім'я малими літерами в однині, можливо, із прикладами значень.

Атрибути необов'язково відображаються на діаграмах сутностей і зв'язків, але додавання одного-двох атрибутів до сутності на етапі формування моделі може бути корисним.

Кожна сутність повинна однозначно ідентифікуватися за допомогою комбінації атрибутів і/або зв'язків. Тому серед можливих атрибутів сутності повинні бути ті, які дозволяють її ідентифікувати.

Важливо розуміти, що всі визначення сутності, зв'язку, атрибута та унікального ідентифікатора представляють собою визначення типу або класу, а не конкретного екземпляра. Екземпляри сутностей і зв'язків будуть представлені в самій базі даних.

Завдання цього етапу полягає у проведенні логічного та фізичного проектування бази даних.

**Логічне проектування** — це розробка логічної структури системи баз даних без прив'язки до конкретної СУБД, структур збереження, методів доступу і т.д..

**Фізичне проектування** — це проектування системи бази даних для конкретної СКБД. Під час виконання даного етапу модель сутностей і зв'язків перетворюється в схему бази даних і специфікації позамашинного збереження.

**4 Логічне та фізичне проектування бази даних та інформаційної системи**

**4.1. Логічне проектування**

Під час логічного проектування, враховуючи концептуальну модель у формі мови ER-моделювання, переходимо до створення детальної логічної моделі бази даних, використовуючи реляційну модель.

**Крок 1. Перетворення сутностей у таблиці:** Кожна сутність конвертується у таблицю, представлену семантично осмисленим іменем латинським алфавітом.

**Крок 2. Перетворення атрибутів у стовпці:** Кожен атрибут перетворюється в стовпець. Ім'я атрибуту представляє семантично осмислене ім'я у латинському алфавіті. Формат представлення значень стовпця уточнюється, а факультативні атрибути стають NULL-стовпцями, а обов'язкові - NOT NULL-стовпцями.

**Крок 3. Представлення унікальних ідентифікаторів ключами таблиць:** Унікальні ідентифікатори сутностей стають первинними ключами таблиць. Якщо сутність має більше одного унікального ідентифікатора, вибирається той, який найчастіше використовується, і інші стають обмеженнями цілісності UNIQUE NOT та NOT NULL.

**Крок 4. Перетворення зв'язків багато-до-одного й один-до-одного в зовнішні ключі:** Зв'язки багато-до-одного і один-до-одного породжують зовнішні ключі. Унікальні ідентифікатори сутностей, які з'єднані зв'язками типу багато-до-одного і один-до-одного, використовуються як стовпці зовнішніх ключів. Факультативним зв'язкам відповідають NULL-стовпці, а обов'язковим - NOT NULL-стовпці.

**Крок 5. Введення спеціальних первинних ключів:** Для адекватного відображення логічного проекту бази даних у фізичний, вводимо спеціальний стовпець з обмеженням цілісності первинного ключа у всі таблиці. Усі стовпці, які є первинними ключами згідно з концептуальною моделлю, отримують обмеження цілісності UNIQUE та NOT NULL.

Логічна база даних, розроблена згідно з концептуальною моделлю та урахуванням обмежень цілісності, детально представлена в наступних таблицях.

**Таблиця 1.**

**Відношення сутності Клієнт**

**Customer**

| **Ім’я стовпця** | **Тип** | **Довжина** | **Призначення** | **Обмеження цілісності стовпців** |
| --- | --- | --- | --- | --- |
| id | Integer |  | Унікальний ID | Первинний ключ |
| first\_name | CharField | 50 | Ім’я | Обов’язковий |
| last\_name | CharField | 50 | Прізвище |  |
| email | EmailField | 100 | Електронна пошта | Унікальний, обов’язковий |
| adress | CharField | 200 | Номер телефону | Унікальний |
| order | Integer |  | Зовнішній ключ | Зовнішній ключ |

**Таблиця 2.**

**Відношення сутності Товар**

**Product**

| **Ім’я стовпця** | **Тип** | **Довжина** | **Призначення** | **Обмеження цілісності стовпців** |
| --- | --- | --- | --- | --- |
| id | Integer |  | Унікальний ID | Первинний ключ |
| product\_name | CharField | 100 | Назва товару | Обов’язковий |
| product\_description | TextField |  | Опис товару |  |
| price | Decimal |  | Ціна | Обов’язковий |
| category | CharField | 50 | Категорія товару | Обов’язковий |
| availability | Boolean |  | Наявність | Обов’язковий |
| supplier | Integer |  | Зовнішній ключ | Зовнішній ключ |

**Таблиця 3.**

**Відношення сутності Знижки**

**Discount**

| **Ім’я стовпця** | **Тип** | **Довжина** | **Призначення** | **Обмеження цілісності стовпців** |
| --- | --- | --- | --- | --- |
| id | Integer |  | Унікальний ID | Первинний ключ |
| discount\_name | CharField | 100 | Назва знижки | Обов’язковий |
| discount\_description | TextField |  | Опис знижки | Обов’язковий |
| discount\_percentage | Decimal |  | Відсоток знижки | Обов’язковий |
| order | Integer |  | Зовнішній ключ | Зовнішній ключ |

**Таблиця 4.**

**Відношення сутності Постачальник**

**Supplier**

| **Ім’я стовпця** | **Тип** | **Довжина** | **Призначення** | **Обмеження цілісності стовпців** |
| --- | --- | --- | --- | --- |
| id | Integer |  | Унікальний ID | Первинний ключ |
| supplier\_name | CharField | 100 | Назва | Обов’язковий |
| supplier\_address | CharField | 200 | Адреса | Обов’язковий |
| email | EmailField | 100 | Електронна пошта | Унікальний, обов’язковий |
| phone | CharField | 20 | Номер телефону | Унікальний, обов’язковий |

**Таблиця 5.**

**Відношення сутності Замовлення**

**Orders**

| **Ім’я стовпця** | **Тип** | **Довжина** | **Призначення** | **Обмеження цілісності стовпців** |
| --- | --- | --- | --- | --- |
| id | Integer |  | Унікальний ID | Первинний ключ |
| date | DateTime |  | Дата замовлення | Обов’язковий |
| Status | Boolean |  | Статус замовлення | Обов’язковий |
| product\_total\_sum | Decimal |  | Загальна ціна | Обов’язковий |
| product | Integer |  | Зовнішній ключ | Зовнішній ключ |
| discount | Integer |  | Зовнішній ключ | Зовнішній ключ |
| client | Integer |  | Зовнішній ключ | Зовнішній ключ |

**4.2. Фізичне проектування**

Базу даних було спроектовано для зберігання в Системі Керування Базами Даних (СКБД) MySQL, яка підтримує реляційну модель даних та є об'єкто-реляційною СКБД. MySQL володіє розширеними можливостями щодо створення та обслуговування баз даних, має розвинену систему типів даних та індексації полів, що дозволяє швидко отримувати доступ до даних. Крім того, вона надає функції для забезпечення цілісності даних між реляційними таблицями, спрощуючи роботу розробників та знижуючи витрати на створення та обслуговування бази даних.

Робота з базою даних виконується за допомогою мови запитів SQL (Structured Query Language), яка є реляційною мовою запитів і використовується для взаємодії з реляційними базами даних.

Логічна модель бази даних легко відображається на фізичну модель, оскільки вона використовує реляційну структуру даних. Крім того, логічна модель приведена у третю нормальну форму, що дозволяє уникнути дублювання даних та забезпечити ефективність бази даних. Всі відношення представлені окремими таблицями у фізичній моделі. Злиття відношень в одну таблицю для оптимізації виконання запитів не застосовується, оскільки відповідні класи запитів не вимагають такого підходу.

**4.3 Вибір технологій для створення інформаційної системи**

1. Мова програмування: Python
   * Це скриптова мова програмування, яка використовується для розробки програмного забезпечення, та ідеально підходить під задачі цієї курсової роботи.
2. База даних: MySQL
   * Реляційна база даних для зберігання та управління даними.
   * MySQL обрано як систему управління базами даних (СУБД).
3. Бібліотека для роботи з MySQL: mysql-connector
   * Бібліотека для забезпечення взаємодії між Python та MySQL.
   * mysql-connector використовується для виконання SQL-запитів та інтеракції з базою даних MySQL.

**4.4 Код фізичного проектування бази даних**

CREATE TABLE Customer (

id INT PRIMARY KEY,

first\_name VARCHAR(50) NOT NULL,

last\_name VARCHAR(50),

email VARCHAR(100) UNIQUE NOT NULL,

address VARCHAR(200) UNIQUE,

order\_id INT

);

CREATE TABLE Product (

id INT PRIMARY KEY,

product\_name VARCHAR(100) NOT NULL,

product\_description TEXT,

price DECIMAL NOT NULL,

category VARCHAR(50) NOT NULL,

availability BOOLEAN NOT NULL,

supplier\_id INT

);

CREATE TABLE Discount (

id INT PRIMARY KEY,

discount\_name VARCHAR(100) NOT NULL,

discount\_description TEXT NOT NULL,

discount\_percentage DECIMAL NOT NULL,

order\_id INT

);

CREATE TABLE Supplier (

id INT PRIMARY KEY,

supplier\_name VARCHAR(100) NOT NULL,

supplier\_address VARCHAR(200) NOT NULL,

email VARCHAR(100) UNIQUE NOT NULL,

phone VARCHAR(20) UNIQUE NOT NULL

);

CREATE TABLE Orders (

id INT PRIMARY KEY,

date DATETIME NOT NULL,

status BOOLEAN NOT NULL,

product\_total\_sum DECIMAL NOT NULL,

product\_id INT,

discount\_id INT,

client\_id INT

);

Додавання ключів:

ALTER TABLE `Orders`

ADD FOREIGN KEY (product\_id) REFERENCES Product(id),

ADD FOREIGN KEY (discount\_id) REFERENCES Discount(id),

ADD FOREIGN KEY (client\_id) REFERENCES Customer(id);

ALTER TABLE Customer

ADD FOREIGN KEY (order\_id) REFERENCES `Orders`(id);

ALTER TABLE Product

ADD FOREIGN KEY (supplier\_id) REFERENCES Supplier(id);

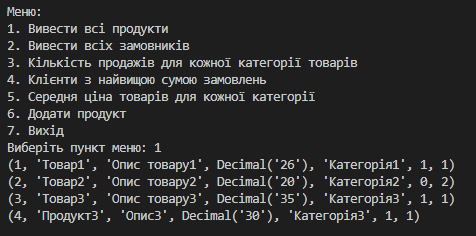
ALTER TABLE Discount

ADD FOREIGN KEY (order\_id) REFERENCES `Orders`(id);

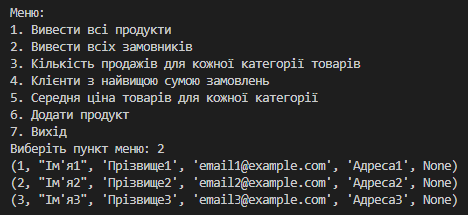
**4.4 Інформаційно–пошукові запити**

Приклади використання створеної БД:

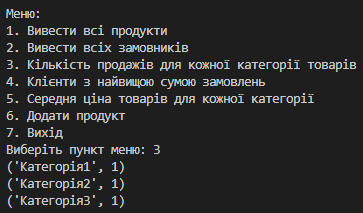
1. Вивести всі продукти



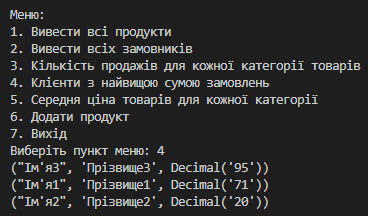
1. Вивести всіх замовників



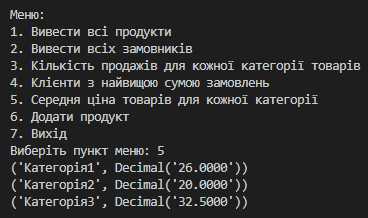
1. Вивести кількість продажів для кожної категорії товарів



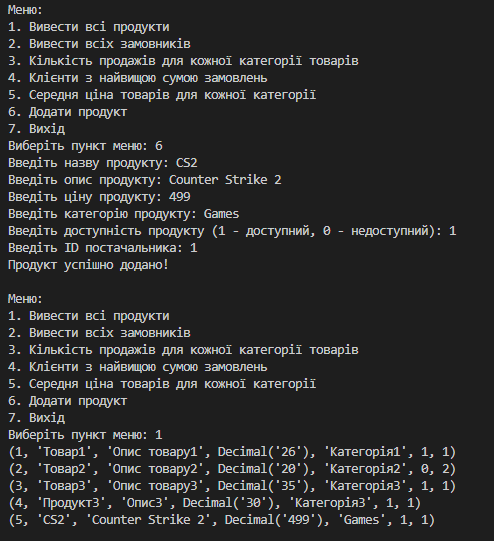
1. Клієнти з найвищою сумою замовлень



1. Середня ціна товарів для кожної категорії



1. Додати продукт



**Висновок**

В ході виконання курсового проекту на тему "Інтернет-магазин продажу ігор" було проведено комплексне проектування бази даних з використанням сучасних технологій, зокрема Python, mySQL та бібліотеки mysql-connector.

Етап стратегії автоматизації передбачав розробку оптимального плану впровадження автоматизованої системи для ефективного функціонування інтернет-магазину. Аналіз програмного забезпечення дозволив глибше зрозуміти можливості та обмеження використаного стеку технологій.

Побудова концептуальної моделі бази даних відбувалася з використанням мови ER-діаграм, що спростило візуалізацію структури та взаємозв'язків між об'єктами в межах предметної області. При цьому використано результати теорії нормалізації для забезпечення ефективності та оптимальності схеми бази даних.

Логічне та фізичне проектування включало в себе конвертацію концептуальної моделі у реляційну структуру, а також розробку команд створення таблиць та SQL-запитів за допомогою Python та бібліотеки mysql-connector. Застосування цих інструментів дозволило ефективно перетворити концепції проекту у функціональну базу даних.

Виконана курсова робота надала мені можливість ознайомитися з технологією проектування баз даних, та отримати практичний досвід у проектуванні бази даних з конкретної предметної області.

**Список використаної літератури**

1. <https://www.python.org/>
2. <https://www.mysql.com/>
3. <https://www.oracle.com/mysql/what-is-mysql/>
4. Ярцев В.П. Організація баз даних та знань: навчальний посібник.-К. ДУТ 2018.- 214с. <http://www.dut.edu.ua/ru/lib/1/category/96/view/1753>
5. <https://www.mysql.com/products/connector/>
6. <https://dev.mysql.com/doc/connector-python/en/>
7. <https://dev.mysql.com/doc/connector-python/en/connector-python-example-connecting.html>
8. Бардус І.О. Бази даних у схемах (на основі фундаменталізованого підходу): навч.посіб./ І.О.Бардус, М.І.Лазарев, А.О.Ніценко. – Харків: Вид-во «Діса плюс», 2017. – 133 с.
9. Мулеса О.Ю. Інформаційні системи та реляційні бази даних. Навч.посібник. – Електронне видання, 2018. – 118 с. Режим доступу до ресурсу: <https://dspace.uzhnu.edu.ua/jspui/bitstream/lib/19776/1/%D0%9C%D1%83%D0%BB%D0%B5%D1%81%D0%B0_%D0%91%D0%94.pdf>