

Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра обчислювальної техніки

Пояснювальна записка

до бакалаврської дипломної роботи
бакалавра

на тему «Інформаційна система меблевого магазину»

Виконав: студент 2 курсу, групи
1КІ-22мс
спеціальності 123 — Комп'ютерна
інженерія

 Авраменко О.А.
Керівник к.т.н., доц. каф. ОТ

 Колесник І.С.
"10" 06 2024р.

Рецензент, к.т.н., доц. каф.
МБІС

 Карпінець В.В.
"11" 06 2024р.

Допущено до захисту
Завідувач кафедри ОТ
д.т.н., проф. Азаров О.Д.
«12» 06 2024р.

ВНТУ – 2024 року

Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра обчислювальної техніки
Освітньо-кваліфікаційний рівень бакалавр
Напрямок підготовки 123 – «Комп'ютерна інженерія»
(шифр і назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри
обчислювальної техніки
проф., д.т.н. О. Д. Азаров
« » КВІТНЯ 2024 р.



З А В Д А Н Н Я

НА ДИПЛОМНУ РОБОТУ СТУДЕНТУ

Авраменку Олександрю Андрійовичу

1 Тема проекту «Інформаційна система меблевого магазину», керівник проекту Колесник Ірина Сергіївна, к.т.н., доцент, затверджені наказом вищого навчального закладу від «11» березня 2024 р. №80

2 Строк подання студентом проекту 07.06.2024 р.

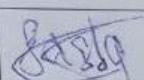
3 Вихідні дані до проекту: технічні параметри інформаційної система меблевого магазину.

4 Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити): вступ, аналіз основних понять, розробка бази даних, настанови користувачеві, висновки.

5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень): трирівнева архітектура СУБД, зв'язки між інформаційними об'єктами, схема бази даних.

6. Консультанти розділів роботи приведені в таблиці 1

Таблиця 6.1 — Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1- 3	Колесник І.С., к.т.н., доцент кафедри ОТ		

7 Дата видачі завдання 12.03.2024 р. Календарний план виконання роботи наведено в таблиці 2.

Таблиця 2 — Календарний план

з/п	Назва етапів виконання бакалаврського проекту	Строк виконання етапів роботи	Примітка
1	Постановка задачі для інформаційної системи	27.02.24	вик
2	Огляд інформаційних джерел	06.03-20.03.24	вик
3	Аналіз основних понять	27.03-7.04.24	вик
4	Розробка бази даних	10.04-21.04.24	вик
5	Настанови користувачеві	24.04-12.05.24	вик
6	Оформлення пояснювальної записки та ілюстративного матеріалу	15.05-26.05.24	вик
	Аналіз виконання проекту. Висновки. Додатки	29.05-09.06.24	вик
	Перевірка якості виконання бакалаврського проекту та усунення недоліків	06.06.24	вик

Студент

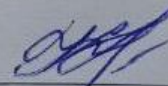


(підпис)

Авраменко О.А.

(прізвище та ініціали)

Керівник роботи



(підпис)

І.С. Колесник

(прізвище та ініціали)

АНОТАЦІЯ

УДК 004.9

Авраменко О.А.

Інформаційна система меблевого магазину. Бакалаврська дипломна робота зі спеціальності 123 — комп'ютерна інженерія, освітня програма — комп'ютерна інженерія. Вінниця: ВНТУ, 2024, 67 с.

На укр.мові. Бібліогр.: 9 назв, рис. 28, табл. 5.

Бакалаврська дипломна робота присвячена розробці інформаційної системи для меблевого магазину. Розробка стане інструментом автоматизованого та зручного засобу обліку роботи магазину.

Високий рівень виконання завдання досягнуто за рахунок використання сучасного засобу керування базами даних Access 2010.

Ключові слова: інформаційна система, база даних, реляційна модель, елемент, атрибут, нормалізація.

ANNOTATION

Avramenko O.A.

Information system of a furniture store. Bachelor thesis in the specialty 123 — computer engineering, educational program computer engineering. Vinnitsa, VTNU, 2024, 67 p.

In the Ukr.leng. Libr.name 9, figure 28, table 5

The bachelor thesis is devoted to the development of an information system for a furniture store. The development will become a tool of an automated and convenient means of accounting for the store's work.

A high level of task performance was achieved by using the modern Access 2010 database management tool.

Keywords: information system, database, relational model, element, attribute, normalization..

ЗМІСТ

1	АНАЛІЗ ОСНОВНИХ ПОНЯТЬ.....	10
1.1	Інформаційні системи.....	10
1.2	Основи проектування баз даних.....	13
1.3	Основні елементи бази даних.....	14
1.4	Системи управління базами даних.....	15
1.5	БД як автоматизована система.....	17
1.6	Моделі даних.....	19
1.7	Інфографічний метод для проектування баз даних.....	22
1.8	Реляційна модель даних.....	24
1.9	Цілісність баз даних.....	27
1.9.1	Структурна цілісність	27
1.9.2	Посилкова цілісність.....	28
1.10	Типи з'єднань.....	28
1.11	Презентація даних із використанням моделі особистих відносин....	29
1.12	Класифікація СУБД.....	31
2	РОЗРОБКА БАЗИ ДАНИХ.....	33
2.1	Проектування бази даних.....	33
2.2	Особливості проектування структурних БД	35
2.3	Нормалізація бази даних.....	36
2.4	Аналіз концептуальних вимог.....	37
2.5	Вибір конкретної СУБД.....	41
2.6	Таблиці.....	43
2.7	Форми.....	45
2.8	Звіти.....	48

					08-54.БДР.001.00.000 ПЗ							
Змн.	Арк.	№ докум.	Підпис	Дата								
Розроб.		Авраменко О.А.			Інформаційна система меблевого магазину Пояснювальна записка			Літ.		Арк.	Аркуші	
Перевір.		Колесник І.С.								6	67	
Реценз.		Карпінєць В.В.						ВНТУ, гр. ІКІ – 22мс				
Н. Контр.		Швець С.І.										
Затверд.		Азаров О.Д.										

2.9 Макроси.....	50
3 НАСТАНОВИ КОРИСТУВАЧЕВІ.....	51
3.1 Інструкція користувачеві для роботи із програмою ведення документації та звітності.	51
3.2 Тестування БД.....	53
ВИСНОВКИ.....	59
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	60
ДОДАТОК А Технічне завдання.....	61
ДОДАТОК В Трирівнева архітектура СУБД.....	64
ДОДАТОК Г Зв'язки між інформаційними об'єктами	65
ДОДАТОК Д Схема бази даних.....	66
ДОДАТОК Е Протокол перевірки кваліфікаційної роботи на наявність текстових запозичень.....	67

ВСТУП

Сучасне суспільство має тенденцію до збільшення обсягу інформаційного потоку навколо нас. Тому будь-яка організація має проблеми з управлінням даними, які забезпечують її безперебійне функціонування. Сьогодні більшість організацій створили та використовують інформаційні системи, які постійно оновлюють, детальну та систематизовану інформацію, наскільки це можливо. [1] .

Є вагомі причини комп'ютеризувати існуючу інформацію. Використання банків даних значно прискорює процес отримання інформації з першоджерел.

Використання інформаційних систем значно економить час, особливо час для отримання необхідних даних, і полегшує пошук інформації до сховища, оскільки такі технології покладаються на інтегровану обробку даних і центральне місце їх зберігання. Крім того, комп'ютер дозволяє переглядати всі формати даних, включаючи текст, зображення, рукописні дані, фотографії, аудіо-відеозаписи тощо.

Для використання такої кількості захищеної інформації, окрім розробки системних інструментів, засобів масової інформації, пам'яті, необхідні інструменти для забезпечення взаємодії людини з комп'ютером, що дозволяє користувачеві вводити запити, читати файли, дані збережених змін, додавати нові дані чи отримувати інформацію. Для забезпечення цих функцій були створені спеціальні інструменти – системи керування базами даних (СУБД), які спеціалізуються на управлінні масивом інформації, який одночасно використовується одним або кількома. , тому поставлена у роботі задача є актуальною [3] .

Мета дослідження — створення інформаційної системи для меблевого магазину.

Інформаційна система повинна виконувати такі задачі:

- надати інформацію про співробітників організації;
- надати інформацію про партнерів-постачальників;
- надавати інформацію про товари, які є в наявності;
- надавати інформацію про клієнтів.

Об'єкт дослідження — процеси, які відбуваються при розробці інформаційних систем.

Предмет дослідження — методи і засоби створення інформаційних систем.

Методи дослідження — аналіз інформаційних потоків, синтез отриманої інформації.

Практичне значення одержаних результатів: Створено інформаційну систему в середовищі СКБД MS Access.

1 АНАЛІЗ ОСНОВНИХ ПОНЯТЬ

1.1 Інформаційні системи

Інформаційна система — це система, яка організовує зберігання та маніпулювання інформацією, що стосується проблемної області.

Характерною рисою інформаційної системи є активна участь людини в інформаційному процесі. В автоматизованому робочому місці особа (кінцевий користувач) вводить дані в систему, підтримує їх актуальність, обробляє дані та використовує результати для управління. Вона служить способом опису взаємодії між джерелом даних і одержувачем. Одне і те саме повідомлення може передавати багато інформації одному одержувачу, але мало або взагалі нічого іншому.

Потенційні можливості інформаційної системи реалізуються за допомогою функцій, які включають:

- актуальну — своєчасно та якісно виконує інформацію за всіма напрямками, що цікавлять комп'ютерну систему управління;
- контрольна — контролює і формує всю необхідну для управління зовнішню і внутрішню інформацію;
- пам'ять — забезпечує безперервний збір, систематизацію, збереження та відновлення всієї необхідної інформації;
- регулююча — контролер даних впливає на об'єкт керування та його зв'язки при відхиленні робочих параметрів від заданих значень;
- оптимізація — забезпечує оптимальні розрахунки в міру зміни цілей, критеріїв і умов функціонування об'єкта управління;
- предиктор — визначає основні тенденції, послідовність і показники розвитку об'єкта управління;

— аналіз — визначає основні показники техніко-економічного рівня виробничо-господарської діяльності;

— документувальна — забезпечує обліково-звітну, планово-розпорядчу, конструкторсько-технологічну та інші види документів.

Одним із способів проектування інформаційної системи є поділ її на окремі частини, при необхідності комплектуються конкретною системою автоматичного керування (АСУ). Цей спосіб називається розсіюванням.

Будь-яка система складна по-своєму. Це означає, що загальність даних, що характеризують систему, і загальність зв'язків між її елементами не можна цілком і повністю припустити. Ця система відрізняється від будь-якої проблеми. Тому, відповідно до методу розподілу, вона повинна забезпечувати найбільш підходящу структуру для швидкого впровадження інформаційної системи.

Оптимальною структурованою системою є багаторівневі багатоцільові організовані елементи (модулі), які відповідають таким вимогам:

— кожен рівень ієрархії повинен бути повністю видимим і зрозумілим без детального знання нижчих рівнів;

— повинно бути мало зв'язків між елементами на одному рівні ієрархії;

— елементи не повинні бути пов'язані через однорангову ієрархію;

— елемент більш високого рівня повинен викликати елемент наступного рівня і передавати йому необхідні вхідні дані, утворюючи з ним єдине ціле;

— елемент більш високого рівня повинен викликати елемент наступного рівня і передавати йому необхідні вхідні дані, утворюючи з ним єдине ціле;

Коли елемент наступного рівня завершує свою роботу, він передає результати своєї роботи елементу, який його викликає.

Обмеження на структуру системи досить жорсткі, оскільки неможливо спроектувати ідеальну систему, яка б існувала без подальших змін. Крім того,

під час роботи інформаційної системи основною вимогою до неї є внесення змін у її структуру.

Внесення змін до добре структурованої системи не включає багато змін локалізованих елементів. В іншому випадку внесення невеликих змін призведе до перепроєктування та перепрограмування великих частин системи.

Збільшена декомпозиція функціональної частини автоматизованої інформаційної системи передбачає встановлення структури елементів (модулів) різних рівнів:

- комплекс першого рівня охоплює автоматизовані підсистеми, комплекси задач;
- комплекс другого рівня відповідає за автоматизовані функції (задачі) управління;
- комплекс наступних рівнів — машинні процедури, що реалізуються управлінським персоналом.

Функціональний поділ інформаційної системи рекомендується здійснювати за об'єктивним принципом за такою схемою: виробниче об'єднання - промислове підприємство - виробництво - цех - технологічний процес (ділянка) - робоче місце (вертикальний розподіл) з розподілом управління; Функції по кожному об'єкту згідно планувальної схеми: облік - контроль - аналіз - регулювання (відокремлення).

Метод декомпозиції використовується на кожному етапі проектування інформаційних систем, щоб зменшити невизначеність та виявити численні проблеми проекту, організувати процес їх вирішення та описати отримані структури у вигляді ієрархічно пов'язаних агрегатів даних.

Декомпозиція означає, що існує кілька способів розподілу розробленої системи. Його завершеність — це стан об'єкта, коли він формується на частини (частини системи), які вважаються неподільними.

Сучасна комп'ютерна інформаційна система — це сукупність апаратних, програмних, мережових та інформаційних ресурсів, які забезпечують вирішення низки завдань, пов'язаних із отриманням, збором, збереженням, передачею, зміною та відображенням інформації в процесі спільної роботи. в певній сфері людської діяльності. У складі інформаційних систем виділяють такі компоненти:

— апаратне або технічне забезпечення (технічне забезпечення) — сукупність технічних засобів, пристроїв та елементів, що складають фізичне середовище інформаційної системи та забезпечують технічне виконання завдань;

— програмне забезпечення - сукупність програмних засобів для реалізації методів, алгоритмів і сервісів, які підтримують функціонування інформаційних систем;

— мережеве забезпечення (мережеве програмне забезпечення) — особлива категорія обладнання, програмного забезпечення та інших ресурсів, що забезпечують роботу інформаційних систем у мережевому середовищі;

— забезпечення даних (база даних) - сукупність усіх значень, величин, показників, які використовує інформаційна система.

База даних (БД) — це будь-яка інформація, яка зберігається та організована в певному порядку, як правило, у формі таблиці, пов'язаної одна з одною певними характеристиками. Система керування базами даних (СУБД) — це програмне забезпечення, призначене для створення, керування та спільного використання баз даних для кількох користувачів. Основна функція СУБД — дозволити користувачам працювати з базою даних за допомогою зручного інтерфейсу без необхідності вникати в деталі апаратного рівня. СУБД створює можливість доступу до інтегрованих даних в наочній області.

1.2 Основи проектування баз даних

Життєвий цикл інформаційної системи бази даних можна розділити на три основні етапи;

- проектування бази даних;
- реалізація програмного забезпечення;
- операція.

На етапі проектування вам потрібно буде зібрати інформацію про предмет, структуру предмета, документообіг, дослідити проблеми автоматизації та визначити майбутні компоненти бази даних і зв'язки між ними. Для кожного елемента необхідно визначити атрибути, властивості, таблиці, зв'язки між таблицями, об'єкти бази даних. Для кожного об'єкта визначаються атрибути та атрибути, призначаються первинні ключі, виконується нормалізація, перевіряються вхідні та вихідні дані на точність.

Етап впровадження передбачає розробку бази даних у певному середовищі керування базами даних. Необхідно враховувати характеристики обраної бази даних, а також інтерфейс з базою даних, технологію запуску та підтримки бази даних у робочих умовах, тестування бази даних, інструкції з експлуатації та впровадження програмного забезпечення.

Етап роботи починається з заповнення бази даних реальними даними. Потім має бути досягнуто пряме впровадження, а також практичне впровадження, завершення операцій та окремих операцій. При розробці великих інтегрованих систем управління підприємством, які включають бази даних, моделювання та аналіз предмета розділені.

Крім того, коли ви створюєте сучасні моделі підприємства та аналізуєте програмне забезпечення для моделювання, ви можете створювати специфікації та проектувати майбутні інформаційні системи, які включають базу даних. [4] .

Створення ефективної бази даних є складним завданням, оскільки воно має багато суперечливих вимог. Програміст повинен враховувати не тільки функціональні вимоги програми, але також швидкість роботи та розмір бази даних.

1.3 Основні елементи бази даних

Об'єктом може бути особа, об'єкт, подія, місце чи концепція, яка містить записану інформацію, клієнтів, банківські рахунки тощо.

Властивість (елемент даних) — кожен об'єкт ідентифікується кількома атрибутами (будівля, розмір, колір, розмір ділянки, банк клієнта, повне ім'я, адреса, ідентифікаційний номер). Значення даних відображають фактичні дані, що містяться в кожному елементі даних. Залежно від того, як елементи даних описують об'єкт, їх значення можуть бути кількісними, якісними або описовими. Інформація про регіон може бути представлена кількома об'єктами, кожен з яких характеризується кількома елементами даних. це значення, отримане елементами даних. Наприклад, об'єкт — це єдиний набір, прийнятий елементами даних. Концептуальне моделювання — це моделювання об'єктів відповідно до інформаційних елементів та їхніх зв'язків. Речі взаємопов'язані, і концептуальна модель забезпечує огляд потоків даних у регіоні. Ключові елементи даних (ідентифікатори) — це елементи даних, які використовуються для ідентифікації інших елементів даних. Ключові елементи даних слід вибирати дуже ретельно, що сприяє створенню концептуальної моделі даних.

Запис даних — це набір значень пов'язаних елементів даних. Значення, отримані елементами даних, утворюють запис.

Недоліки традиційної організації файлів даних;

— надмірність даних (стан даних, що повторюються декількома програмами);

- неправильний спосіб відновлення;
- труднощі з імпортом довгих записів;
- труднощі адаптації;
- високі витрати на програмування;
- комплексні методи управління та використання ЕОМ.

Різниця між базою даних і файлом полягає в тому, що база даних може мати кілька цілей для даних, які зберігаються відповідно до різних параметрів.

СУБД - Hard & Soft - Програмна логіка та процедури, що забезпечують управління базою даних. СУБД забезпечує доступ до структурованої інформації, яка перетинає операційні та функціональні межі в полі зору організації.

Вимоги до бази даних:

- ефективність виконання різноманітних заходів регіону;
- зменшення надмірності;
- надання точної інформації для прийняття рішень;
- забезпечення безпеки;
- відсутність підвищених вимог до залучених працівників;
- розробка додатків;
- реорганізація бази даних;
- центральне управління;
- спрощення роботи з комп'ютером.

База даних повинна:

- задовольняти поточні потреби зовнішніх користувачів,
- надавати можливість зберігання та модифікації великих масивів даних;
- забезпечити певний рівень достовірності інформації та її зв'язності;
- доступ до конфіденційної інформації для конкретних користувачів;

- можливість пошуку інформації за ключем;
- задоволення потреб продуктивності праці;
- можливість реорганізації та розширення;
- різні види доставки інформації;
- легкість і зручність доступу до інформації;
- надавати послуги кільком користувачам одночасно

1.5 БД як автоматизована система

Для організації роботи з базою даних додатком необхідно забезпечити їх незалежність від даних. Це пов'язано з тим, що потрібно працювати над зміною методів зберігання даних, методів доступу до даних, форматування та форматів і зв'язків між ними в базі даних у міру зміни системи, а також над ефективним обслуговуванням клієнтів. , якщо ви не використовуєте спеціальні методи та включаєте опис програми під час написання програм.

Методи доступу, зберігання даних, формати даних і будь-які зміни в базі даних повинні бути скориговані з урахуванням цих проблем за допомогою письмової програми користувача, що є дуже дорогим.

Незалежність даних заявок забезпечується IMF. Цей підхід заснований на тому, що користувачі, які використовують базу даних, не знають внутрішнього представлення даних.

На рисунку 1.1 показана трирівнева архітектура SMD. Опис структури даних на всіх рівнях називається схемою. Існує три типи схем бази даних, визначених відповідно до рівнів абстракції бази даних. Є кілька зовнішніх датчиків, які відповідають різним показникам даних на верхньому рівні. Цей рівень визначає вигляд окремої бази даних. Кожна програма переглядає та обробляє лише інформацію, необхідну для цієї програми.

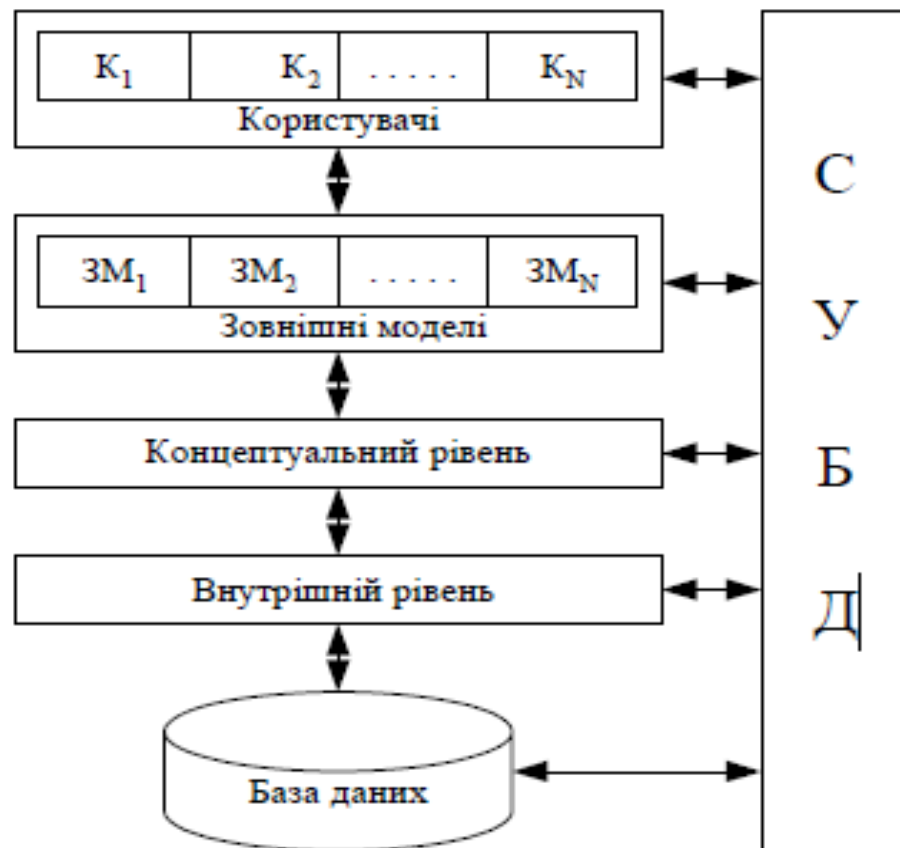


Рисунок 1.1 — Трирівнева архітектура СУБД

На концептуальному рівні опис бази даних називається концептуальним плануванням. Тут база даних представлена в дуже загальному вигляді, який поєднує дані, що використовуються всіма програмами, які обробляються базою даних. Насправді концептуальний рівень відображає модель предмета, для якого була побудована база даних.

На внутрішньому рівні опис бази даних називається внутрішньою схемою. Тут база даних зберігається у вигляді даних безпосередньо у файлах, що відповідає фізичній організації бази даних.

Трирівнева архітектура СУБД забезпечує незалежність даних. Це означає, що зміни на нижчих рівнях не впливають на вищі рівні. Є логічна і фізична свобода при роботі з даними.

Зміни концепцій плану бази даних не вимагають коригування існуючих зовнішніх планів і, відповідно, не змінюють програми, які працюють із цими схемами. Фізична незалежність від даних означає захист концепцій і зовнішніх схем від змін внутрішнього плану. Зміни у внутрішньому плані включають різні види використання файлової системи або структури даних, різні пристрої зберігання даних, модифікації структури пошуку тощо.

На додаток до цих трьох рівнів абстракції в даній базі даних існує ще один попередній рівень. На цьому рівні відображається інформація про об'єкт, а модель на цьому рівні називається інфографічною моделлю об'єкта. Таким чином, основними рівнями абстракції в базі даних є;

- інфологічний;
- зовнішній;
- концептуальний;
- внутрішній.

Перехід від одного рівня абстракції до іншого, як правило, є процесом розробки бази даних.

1.6 Моделі даних

Модель даних — це вид абстракції, яка відображає найважливіші аспекти діяльності певного регіону, а інші ігноруються.

Модель даних є частиною моделей цільової області. Модель даних складається з трьох основних компонентів;

- конструктивна частина, що описує правила виготовлення типів конструкцій;
- дані, дозволені для цієї бази даних;

- контрольна частина, яка визначає можливу роботу такої конструкції;
- розділи обмежень цілісності даних, які застосовуються цією системою.

Моделювання даних — це логічний процес створення та представлення структур бази даних.

На рисунку. 1.2 наведено класифікацію моделей даних.

Кожному рівню подання інформації відповідає певна модель.

Інфографічна модель — представляє інформацію про предмет у форматі, який не залежить від використовуваного SMB. Ця модель відображає інформаційно-логічний рівень абстракції, який пов'язаний з описом об'єктів у предметі, їх властивостей і взаємозв'язків.

Часто ці моделі виражаються в термінах концептуальних моделей предмета, які називаються концептуальними інфографічними моделями (концептуальними моделями внутрішньої та зовнішньої інфографіки).

Модель даних журналу — це модель логічного рівня, яка відображає логічні зв'язки між елементами даних незалежно від їх вмісту та середовища зберігання. Часто ці моделі описуються логічними моделями.

Фізична модель — описує, як дані зберігаються на комп'ютері, і надає інформацію про структуру записів, їх порядок і доступні методи доступу до даних.

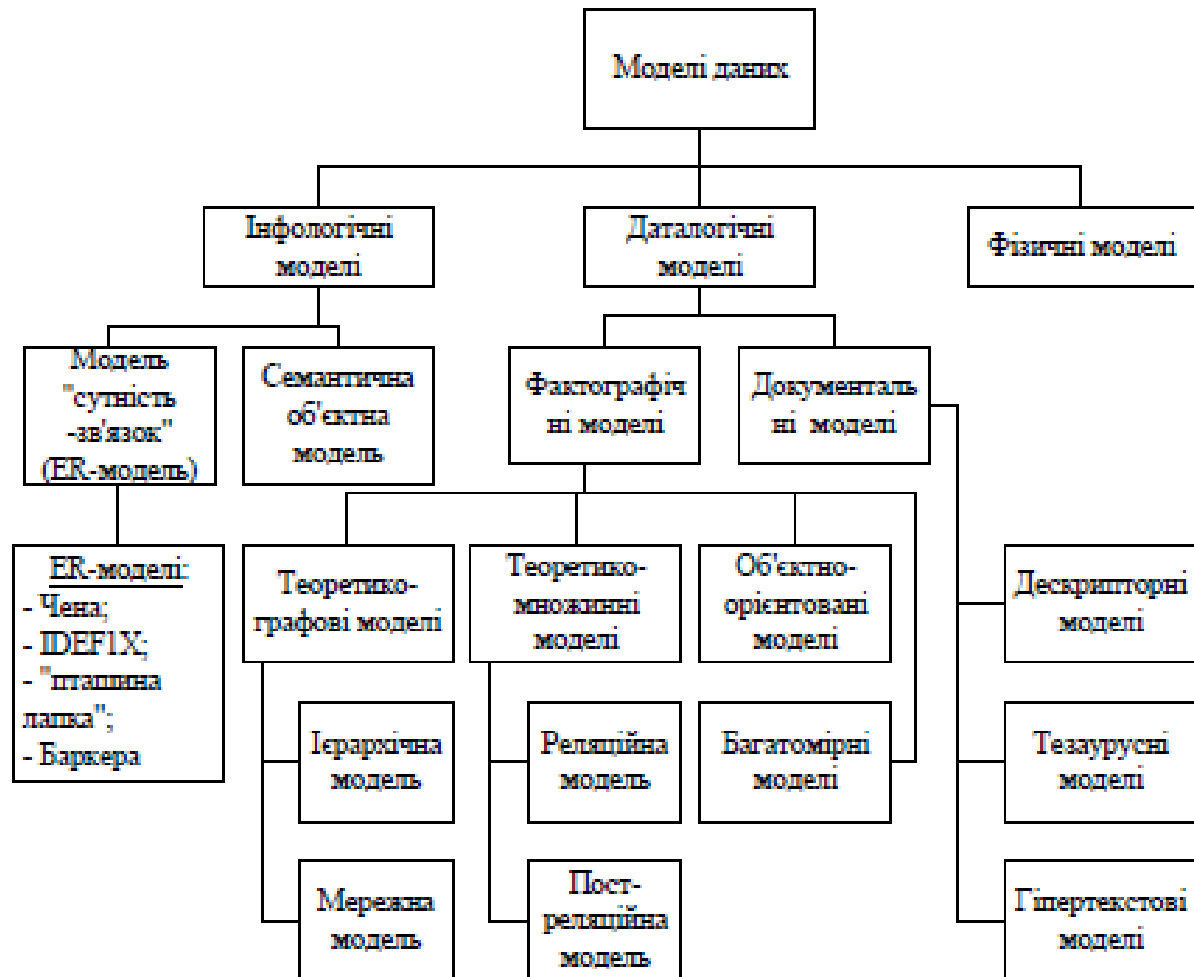


Рисунок 1.2 — Класифікація моделей даних

Модель людських стосунків (ER-модель) — це модель, що описує предметну область, що складається з набору сутностей, набору зв'язків між сутностями, а також атрибутів і зв'язків сутності. Модель передбачає обмеження цілісності даних, пов'язане з двома наборами об'єктів і називається залежністю присутності. ER-моделі дозволяють графічно відображати моделі предметних полів. Вони є невід'ємною частиною багатьох продуктів CASE.

Семантична модель суб'єкта — це модель, яка описує суб'єкт і є моделлю даних. Ця модель містить набір семантичних об'єктів, що містять атрибути.

Атрибути згруповані в розділи. Модель даних має більше інструментів семантичного представлення порівняно з моделями теорії складності та теорії графів.

Теоретико-графічна модель — інформаційна модель, у якій задані допустимі структури представлені у вигляді узагальненого або конкретно сформованого графа, наприклад дерева. Необхідною групою операцій на основі цієї моделі є дослідження операцій на мові керування даними. Операції з даними є ненормальними.

Теоретична модель — це набір даних, контрольованих за допомогою таблиць із математичного інструменту реляційної алгебри, реляційного числення та операцій з даними.

Справжні моделі містять інформацію, представлену у вигляді унікального набору формальних записів даних.

Моделі документів — як інформаційний елемент ми припускаємо, що документ не поділено на менші елементи, і що інформація про документ часто є неструктурованою або в обмеженій формі. Ці моделі в основному обробляють тексти природною мовою, формат документа.

Ієрархічна модель — це інформаційна модель, заснована на деревоподібній ієрархії. Вершинами цієї структури є записи, що складаються з різних типів простих елементів даних. Батьківський запис відповідає будь-якій кількості записів екземплярів кожного типу.

Мережна модель — модель даних, у якій допустима структура даних представлена у вигляді узагальненого графа. Вершинами такого графа можуть бути різні дані – від автономних елементів даних до записів.

Реляційна модель — це інформаційна модель, заснована на математичній концепції зв'язків і представленні зв'язків у табличній формі

Постреляційна модель — це розширена реляційна модель, яка долає

обмеження неподільних даних, що зберігаються в табличних записах. Ця модель дозволяє використовувати кілька полів, значення яких складається з підтипів. Набір значень різних полів значень в основній таблиці розглядається як вбудована незалежна таблиця. Часто ці моделі описуються об'єктивно-реляційними моделями.

Об'єктно-орієнтована модель - це інформаційна модель, яка базується на понятті об'єкта, людини зі станом і поведінкою. Стан об'єкта визначається його атрибутами та властивостями за допомогою набору операцій, визначених цим об'єктом. Також можна підтримувати зв'язки між типами об'єктів.

Багатовимірна модель у багатовимірному представленні даних (Модель даних, яка працює у формі гіперкуба та призначена для підтримки аналізу даних. Створюйте різні набори даних у гіперкубі, створюйте різні проекти (підкатегорії, списки та потоки даних Гіперкуба) та ряд інших функцій.

Описова модель описує кожен документ за допомогою опису. Опис має стійку структуру і складається з деякої лексики (слів, фраз, слів), необхідної для роботи з документами.

Модельний тезаурус – описує кожен документ за допомогою описів, а також змістовних словникових зв'язків (весь розділ, види статі, розділ-підрозділ тощо). Ці моделі дозволяють підвищити ефективність описових моделей шляхом відображення домену.

Гіпертекстова модель є прикладом, заснованим на компонуванні документа з використанням специфічних навігаційних структур, що відповідають семантичним зв'язкам між різними документами або окремими фрагментами документа. Такі конструкції утворюють семантичну мережу в документах бази даних.

1.7 Інфографічний метод для проектування баз даних

База даних показує деякі цільові моделі домену. При проектуванні будь-якої бази даних дизайнер повинен вміти ідентифікувати ті фактори, які чітко визначають предметні межі. Наступним кроком є відображення попередньо визначених факторів у структурі бази даних, визначених у візуальному середовищі цієї бази даних, якщо всі об'єкти мають свої властивості та зв'язки в них відомі. Проектування будь-якої бази даних починається з початкової структури суб'єкта, відбувається класифікація об'єктів у суб'єкті, повертаються їх зв'язки і так далі. Відповідно до інфографічного підходу, при проектуванні баз даних реальні явища, інформація про ці явища та подання цих даних повинні бути різноманітними. Системи об'єктів мають такі стани: об'єкт, властивість, відношення, час. Ці елементи є основними станами об'єкта системи.

Мета інфографічного підходу полягає в тому, як інформація повинна збиратися в інформаційній системі. Об'єкт вибирається відповідно до призначення інформаційних систем. Сам об'єкт можна розглядати як атом або сполуку. Той самий об'єкт, але в різних реалізаціях, можна розглядати як атом або як сполуку. Для кожного предмета необхідно провести декомпозицію шляхом виділення окремих елементів. Кожне відношення між об'єктами характеризується кількістю елементів, включених до нього на рівні n , причому об'єкти мають певні стани в певні моменти часу та в часові інтервали. Поняття часу дозволяє будувати динамічні моделі (моделі реального часу), в яких відображаються залежні від часу об'єкти, що складають систему. Основні складові системи можуть бути інтегровані в основні структури, які називаються первинними структурами. Об'єкти – це групи об'єктів, властивості – це атрибути, а початкові умови згруповані в початкові типи умов. Інформаційна область представлена поняттями, які можна використовувати для формального опису та аналізу систем об'єктів.

Атрибут, значення якого однозначно ідентифікує користувачів, називається ключем (або просто ключем). У нашому випадку атрибут «номер таблиці» є ключовим, оскільки його значення унікальне.

Зв'язок може містити кілька ключів. Один із ключів завжди оголошується першим, його значення не оновлюється. Усі інші ключі відношення називаються можливими ключами.

Функції, які є копіями інших ключів відносин, називаються зовнішніми ключами. [2] .

Будь-які відносини - це динамічна модель чогось реального. Тому в даний час введено поняття моделі відносин, що відображає стан даного об'єкта, і поняття планування відносин, що визначає структуру відносин.

Схема відношення R — список імен атрибутів даного відношення з символом відповідного поля;

$$SR = (A_1, A_2, A_n) ID$$

Управління реляційною базою даних базується на математичній моделі з використанням методів реляційної алгебри та реляційного числення.

Усі дані бази представлені у вигляді таблиці.

Відносини мають такі характеристики;

— відсутність дублікатів карток;

— відсутність карткових замовлень;

— неупорядкованість якостей;

— автономність значень атрибутів, тобто між значеннями домену, не може містити багато значень. [5]

1.8 Реляційна модель даних

Реляційна модель даних базується на математичній концепції зв'язків і поданні зв'язків у табличній формі. На початку 1970-х років американські вчені

eCode визначили, що в будь-якій реляційній базі даних передбачається, що користувач розуміє базу даних як набір таблиць.

Це стосується лише логічної структури бази даних, а саме концепцій і зовнішніх представлень. На фізичному рівні бази даних реалізуються за допомогою різних структур зберігання.

У таблиці 1.1 представлені елементи реляційної моделі.

Ключі використовуються для однорідної ідентифікації рядків, об'єднання таблиць і прискорення операцій з даними. Таблиця 1.2 показує можливі типи реляційних ключів. Сторонні та сумісні потенційні ключі мають бути визначені в одному полі.

Таблиця 1.1 — Елементи реляційної моделі

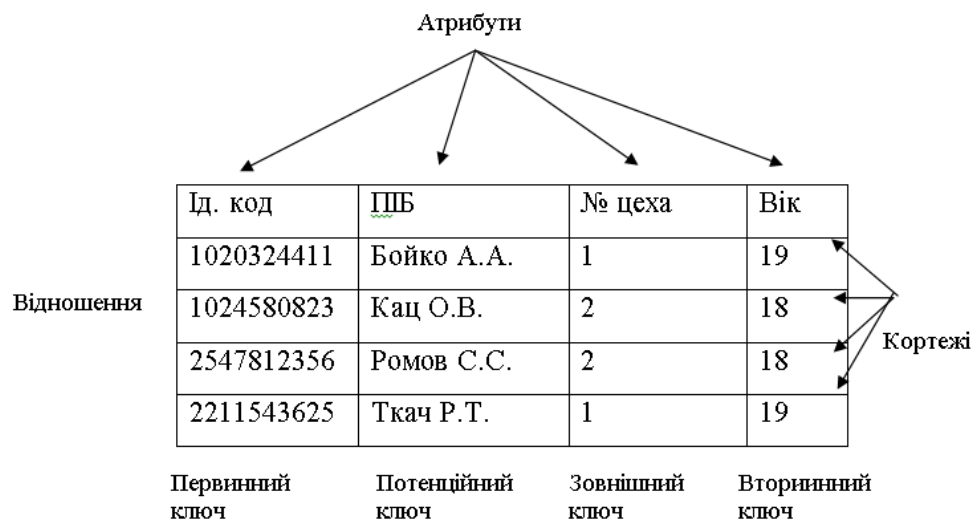
Елементи реляційної моделі	Форма представлення
Відношення	Таблиця
Кортеж	Рядок таблиці
Атрибут	Заголовок стовпця таблиці
Ключ	Сукупність атрибутів, які унікально визначають кожен рядок таблиці або виконують функції зв'язування таблиць або дозволяють прискорити операції над таблицями
Домен	Множина значень атрибута
Схема відношення	Рядок заголовків стовпців таблиці

Таблиця 1.2 — Реляційні ключі

Назва	Пояснення
(Candidate Key) Потенційний ключ	Мінімальна підмножина атрибутів відношення, що єдиним чином ідентифікують кортеж цього відношення
(Primary Key) Первинний	Потенційний ключ, який обраний для унікальної ідентифікації кортежів відношення

КЛЮЧ	
(Secondary Key) Вторинний ключ	Ключ, кожному значенню якого може відповідати більше ніж один екземпляр індексованих даних
(Foreign Key) Зовнішній ключ	Сукупність атрибутів відношення, значення яких є одночасно значеннями первинного чи потенційного ключа іншого відношення

Порядок вирівнювання карт є відносно невизначеним. У реляційній базі даних картки сортуються за ключем (основним або вторинним) для зручності. Первинний ключ — це код ідентифікації атрибута, який унікально ідентифікує кожного користувача. Атрибут віку вибрано як вторинний (необов'язковий) ключ для ранжирування та групування співробітників за віком. Функція Store# вибирається як зовнішній ключ для підключення до таблиці Staff. Поля являють собою набір усіх можливих значень відповідних атрибутів. Наприклад, для атрибута Age значення поля відноситься до цілого типу.



Реляційна модель складається з наступних частин:

- структурний (тут відносини встановлюються в цілому);
- маніпулятивний (тут використовуються два основні методи

управління реляційними базами даних - реляційна алгебра та реляційне числення;

— цілісність (тут використовується механізм втрати даних).

Реляційні моделі даних характеризуються простотою і природністю використовуваних структур даних і операцій управління даними, повною незалежністю від середовища сховища даних, а не підтримкою віртуальних зв'язків між даними замість фізичних (на основі значень даних, а не індикаторів).

Реляційна база даних включає наступні компоненти;

— масив інформації (таблиці, показники);

— системна інформація (структура бази даних, обмеження цілісності);

— програми (процедури, тригери).

Практичний навик відносин має дві еквівалентні форми — реляційну алгебру та реляційне числення. ECode запропонував відповідні тримовні принципи побудови для виконання запитів до бази даних.

Реляційні алгебраїчні мови запитів — це алгебраїчні мови, які дозволяють виражати запити за допомогою спеціальних операторів, що застосовуються до відносин.

Мови реляційних обчислень дозволяють вказувати запити, вказуючи префікс, якому повинні відповідати потрібні кортежі (домени).

Реальні мови запитів (SQL, QBE та ін.) реалізують не тільки функції відповідної теоретичної мови, а й деякі додаткові операції.

1.9 Цілісність баз даних

Цілісність бази даних — це властивість даних, яка визначає повноту та точність інформації в базі даних.

Підтримка цілісності даних включає наступні компоненти:

- структурна цілісність — обмеження реальних значень даних;
- посилкова цілісність.

1.9.1 Структурна цілісність

Структурна цілісність передбачає виконання наступних умов:

- існування тільки однотипних структур даних «реляційного» типу відносини»;
- відсутність дублікатів кортежу;
- обов'язкова наявність в кожному відношенні первинного ключа;
- обмеження доменів — передбачає визначення кожного атрибута на своєму домені;

—можливість використання невизначених значень NULL.

Невизначене значення NULL наразі розглядається як невідоме значення. Це значення можна змінити на певне значення, коли відображається додаткова інформація. Введення NULL призвело до необхідності використання тризначної логіки замість двозначної. У цьому випадку припускаються реляційні операції з невизначеними значеннями.

Обмеження на реальні значення даних вимагають, щоб значення полів належали до деякого діапазону значень або щоб вони задовольняли певне арифметичне співвідношення між значеннями різних полів. Значення обмежень також можуть включати визначення певних форматів для полів, задоволення певних статистичних умов значень полів, бізнес-правил суб'єкта, тощо.

1.9.2 Посилкова цілісність

Посилкова цілісність означає, що зміни в таблицях мають виконуватися паралельно, а вміст двох пов'язаних таблиць має відповідати наступним правилам:

- кожному запису основної таблиці відповідає нуль або більше записів підлеглої таблиці;
- у підтаблиці немає записів, які не мають батьківських записів. основної таблиці;
- кожен запис підтаблиці має лише один батьківський запис. основної таблиці.

Стан цілісності даних визначає, які дані можна записати в базу даних у результаті додавання або оновлення даних.

1.10 Типи з'єднань

Сучасні бази даних часто мають велику кількість взаємопов'язаних таблиць, що запобігає дублюванню. Наприклад, великі компанії можуть

зберігати інформацію про магазин в одній таблиці, запаси, отримані цими магазинами цього місяця в іншій таблиці, і інформацію про оптових покупців у третій таблиці. Access дозволяє працювати відразу з кількома таблицями, в кожній з яких повинні бути записи за певною темою. Зв'язок між ними встановлюється за допомогою загальних полів у кількох таблицях, наприклад кількість магазинів, де продається товар. Але в одному зі столів ключовим було загальне середовище. Якщо таблиці пов'язані, зміна одного запису таблиці може вплинути на інший запис таблиці.

Щоб зберегти цілісність і повноту даних, Access накладає певні обмеження на вставлення та редагування даних: наприклад, ви не можете видалити запис із таблиці, якщо в інших таблицях є пов'язані записи.

Усі інформаційні об'єкти взаємопов'язані. Існує кілька типів з'єднань, до яких вставляються наступні примітки;

- один до одного (1:1);
- один до багатьох (1: M);
- багато до багатьох (M: M).

Кореляція (1:1) припускає, що в кожен момент часу одна копія інформаційного об'єкта А відповідає не більш ніж одній копії інформаційного об'єкта В і навпаки. У відповідності один до багатьох (1:M) екземпляр інформаційного об'єкта А відповідає 0, 1 або більше екземплярам об'єкта В, але кожен екземпляр об'єкта В відповідає щонайбільше 1 екземпляру Фізичний об'єкт. об'єкт А. Графічно цей зв'язок має структуру. Відношення «багато до багатьох» (M:M) припускає, що в кожен момент часу один екземпляр інформаційного об'єкта А відповідає 0, 1 або більше, наприклад, об'єкт В, і навпаки. [5]

1.11 Презентація даних із використанням моделі особистих відносин

ER-модельовання — це поетапний підхід до проектування бази даних, який починається з визначення відповідних даних, які мають бути перераховані, і зв'язків між даними, представленими в моделі. Потім модель вводить інформацію про властивості та зв'язки об'єктів, так звані атрибути (атрибути), а також будь-які обмеження на об'єкти, зв'язки та властивості. Модель ER забезпечує графічне представлення логічних об'єктів та їх зв'язок у заданій структурі бази даних.

Зауважте, що модель розуму й свідомості не є автоматично моделлю даних, оскільки вона не описує операції з даними й обмежується описом їхньої логічної структури. Модель сутність-зв'язок була запропонована в 1976 році Пітером Піншен Ченом.

Організація дозволяє моделювати такий клас об'єктів. Організація має унікальну назву в системі моделі. Оскільки об'єкт відповідає певному класу подібних об'єктів, передбачається, що в масиві є кілька екземплярів. Об'єкт, до якого належить людина, має набір властивостей, що визначають її поведінку. У цьому випадку набір атрибутів повинен бути таким, щоб можна було розрізнити конкретні екземпляри організації. Сутність — це фактично набір атрибутів, які визначають атрибути всіх членів набору об'єктів.

Взаємодія між учнями створюється, щоб показати, як учні взаємодіють один з одним.

Хоча, строго кажучи, поняття «кореляція» і «кореляційна множина» різні (перше є другим елементом), їх часто плутають. Тому в майбутньому ми часто використовуватимемо слова «зв'язок» для позначення «набору відношень», а слова «сутність» — для позначення «набору сутностей». Коли $n = 2$ означає, що взаємодія поєднує дві одиниці, її називають бінарною. Було показано, що набір n -арних ($n > 2$) — кореляцій завжди можна замінити бінарним набором, але перший краще відображає опис візуального середовища.[2]

Кількість об'єктів, які можна підключити до іншого блоку за допомогою встановленого з'єднання, називається рівнем з'єднання. Розгляд ступенів особливо корисний для двосторонніх відносин. Є три етапи двосторонніх відносин.

Один до одного (1: — Це означає, що в таких взаємодіях присутність однієї ролі завжди відповідає не більше ніж одній одиниці ролі іншої.

Один до багатьох (1:n). У цьому випадку особа з однією роллю може відповідати будь-якій кількості інших носіїв ролі. Між двома об'єктами можна розрізнити кілька наборів зв'язків. У цьому розділі також слід розглянути людину.

від багатьох до багатьох (n: n). При цьому кожну відповідну одиницю можна представити кількома прикладами.

Якщо існування сутності *x* залежить від існування сутності *y*, *x* називається залежною сутністю (сутність *x* іноді називають «слабкою», а сутність «*y*» — сильною).

Зауважте, що здатність до прихильності на людину завжди буде високою (1: — клас членства та рівень прихильності можуть відрізнятися для штабів [5].

1.12 Класифікація СУБД

СУБД класифікують за певними ознаками:

— за рівнем прозподілення: централізовані чи локальні — усі частини бази даних зберігаються на одному комп'ютері; розподілені — компоненти бази даних можуть розташовуватися на двох і більше комп'ютерах;

— за вмістом: історичні, наукові, географічні, мультимедійні тощо;

— за архітектурою: файл-серверні база даних розміщується на комп'ютері (файл-сервері) у централізованій мережі, для роботи з цією базою на

комп'ютері кожного користувача встановлюється СУБД доступ до бази здійснюється через локальну мережу, прикладами є Microsoft Access, FoxPro, Paradox, dBase; клієнт-серверні: систему управління базами даних можна розглядати як систему, що складається з двох частин: набору серверів і клієнтів, сервера разом з базою даних, встановлених на центральному комп'ютері, самої СУБД; клієнти — це програми, які надають можливість працювати з базами даних, як приклад можна навести Firebird — IBExpert, Microsoft SQL Server;

— за моделлю даних: ієрархічні, мережеві, реляційні, об'єктно-орієнтовані;

— за сферою застосування: документографічні та документальні (органи влади та управління); бази даних про промислову, будівельну та сільськогосподарську продукцію; економічні бази даних (статистика, кредитно-фінансова, зовнішня торгівля); бази даних транспортних систем; наукові бази та банки даних; бази даних у сфері культури та мистецтв; лінгвістичні бази даних тощо.

2 РОЗРОБКА БАЗИ ДАНИХ

2.1 Проектування бази даних

Створення бази даних необхідно починати з її проектування (розробки). У результаті проектування має бути визначена структура бази даних, тобто структура таблиць, їх структура та логічні зв'язки. Структура реляційної таблиці визначається структурою стовпців, їх порядком, типом даних і розміром кожного стовпця, а також ключем таблиці. Процес проектування можна проводити двома способами. Згідно з першим підходом спочатку визначаються основні завдання, для вирішення яких створюється база даних, і потреби цих завдань у даних. За другим підходом визначається предмет (сфера), аналізуються його дані та встановлюються типові об'єкти предмета. Найбільш розумним підходом до проектування бази даних є поєднання обох підходів.

Зазвичай з базою даних працюють дві категорії користувачів. Перша категорія – дизайнери. Процес проектування бази даних розбивається на етапи, кожен з яких передбачає виконання певних дій. Перший етап — розробка інформаційно-логічної моделі даних досліджуваного, яка ґрунтується на описі досліджуваного, отриманому в результаті його дослідження. На цьому етапі визначається структура і структура предметних даних, які повинні спочатку існувати в базі даних і забезпечувати виконання запитів користувача, завдань і додатків. Ці дані у вигляді реквізитів містяться в різних документах — джерелах завантаження бази даних. Аналіз отриманих даних дозволяє визначити функціональні залежності факторів, які використовуються для відбору інформаційних елементів, що відповідають вимогам нормалізації даних. Подальше визначення структурних зв'язків між елементами дозволяє побудувати інформаційно-логічну модель.

Другим кроком є визначення логічної структури бази даних. Для реляційної бази даних цей етап значною мірою є формальним, оскільки інформаційно-логічна модель адекватно відображається в структурі реляційної бази даних.

Наступним етапом є побудова таблиці бази даних, яка здійснюється засобами СУБД, та їх інтеграція з клієнтом. Структура таблиць бази даних визначається методом опису (побудови) таблиць у СУБД з повною прив'язкою до інформаційних об'єктів. Крім таблиць, розробники готують інші об'єкти бази даних, які призначені для автоматизації роботи з базою даних, з одного боку, і для обмеження функції роботи з базою даних, з іншого (безпека бази даних). Після створення формату таблиці база даних може бути заповнена даними з вихідних документів. Розробники, як правило, не заповнюють базу конкретними даними (оскільки клієнт може вважати дані конфіденційними і не надавати їх третім особам). Винятком є експериментальне заповнення об'єктів бази даних даними моделі на етапі налагодження.

Друга категорія — користувачі, які працюють з базами даних. Вони отримують, заповнюють і обслуговують дані бази даних від дизайнерів. Користувачі не отримують контролю над структурою бази даних, вони мають доступ лише до даних, лише до тих, які призначені для роботи в їхній конкретній робочій області.

Розробка бази даних починається з вивчення завдання на розробку бази даних, яке буде подано клієнтом. Тому замовнику бажано володіти відповідною термінологією і хоча б узагальнено знати технічні можливості ядра СУБД. На жаль, на практиці ці бажання не завжди виконуються. Тому розробники часто використовують такі підходи: показують клієнту ту саму функціональність бази даних, потім узгоджують специфікації відмінностей; Якщо аналога немає, визначають рівень завдань і вимог замовника, а потім допомагають йому

підготуватися до технічного завдання. Під час підготовки технічного завдання збирається: внесення детальної інформації для замовника; необроблені дані, необхідні клієнту для управління структурою організації; Перелік вихідних даних, які не є необхідними для клієнта, але повинні бути надані іншим органам (вищестоящим структурам, органам статистики, іншим органам управління та контролю).

2.2 Особливості проектування структурних БД

На першому етапі проектування бази даних необхідно визначити мету створення бази даних, її основні функції та інформацію, яка буде міститися. Тобто потрібно визначити основні заголовки таблиць бази даних та інформацію, яку містять поля таблиці.

База даних повинна відповідати вимогам тих, хто з нею безпосередньо працює. Для цього вам потрібно визначити теми, які має охоплювати база даних, звіти, які вона має створювати, проаналізувати шаблони, які наразі використовуються для запису даних, і порівняти створювану базу даних із добре спроектованою подібною базою даних.

Визначте таблиці, які повинні містити базу даних. Одним із найскладніших етапів у процесі проектування бази даних є розробка таблиць, оскільки вихідні дані, які має видавати база даних (звіти, вихідні форми тощо), завжди стосуються структури... таблиці. Для розробки таблиць необов'язково використовувати СУБД. Краще спочатку розробити структуру на папері.

Визначте необхідні поля в таблиці. Кожна таблиця містить інформацію з іншої теми, а кожне поле в таблиці містить різні дані про тему таблиці. Наприклад, таблиця з даними про клієнта може мати поля з назвою компанії, адресою, містом, країною та номером телефону.

Призначте окреме значення кожному полю. Щоб СУБД могла співвідносити дані з різних таблиць, наприклад, про клієнта та його замовлення, у кожній таблиці повинно бути поле або набір полів, які задають індивідуальне значення кожного запису в таблиці. Таке поле або набір полів називається первинним ключем.

Визначте зв'язки між таблицями. Після розбиття даних на таблиці та визначення ключових полів необхідно вибрати схему зв'язку даних у різних таблицях. Для цього потрібно визначити зв'язки між таблицями. Рекомендується вивчати зв'язки між таблицями у вже існуючій базі даних. Щоб відобразити контакти у вибраній базі даних, відкрийте та виберіть відповідні команди.

Відновити структуру бази даних. Після розробки таблиць, полів і зв'язків структуру бази даних потрібно переоцінити та виявити потенційні недоліки. На цьому етапі рекомендується, щоб таблиці не заповнювалися даними. Щоб перевірити, вам потрібно створити кілька таблиць, визначити зв'язки між ними та вставити кілька записів у кожную таблицю, а потім перевірити, чи база даних відповідає критеріям. Також рекомендується створити проекти вихідних форм і звітів і перевірити, чи містять вони необхідну інформацію. Крім того, з таблиць необхідно виключити всі можливі дублікати даних.

Якщо структура таблиць відповідає вимогам, то всі дані можна імпортувати. Потім ви можете створювати будь-які запити, форми, звіти, макроси та модулі.

Використання засобів аналізу в СУБД. Наприклад, СУБД Microsoft Access має два інструменти для оптимізації структури бази даних. Майстер аналізу таблиць перевіряє таблицю, за потреби пропонує нову структуру та зв'язки та перемальовує її. Аналітик продуктивності переглядає всю базу даних, дає рекомендації щодо її покращення та реалізує їх.

2.3 Нормалізація бази даних

При розробці системи управління базами даних виникає проблема вибору кількості створюваних таблиць, визначення ключових полів, встановлення зв'язків між даними і як наслідок знаходження оптимальної бази даних. Однак між різними реалізаціями проектів баз даних можуть бути значні відмінності. Це залежатиме від успішного дослідження та організації даних. Існують стандартні методики, процедури, завдяки яким можна досягти сприятливих результатів. Найпомітніші з них – нормалізація та індексація.

Нормалізація — це процес приведення бази даних у нормальний формат. Нормальних форм кілька, редукція відбувається послідовно.

Зазвичай нормалізацію поділяють на п'ять форм або етапів – від першої нормальної форми до п'ятої нормальної форми, але, як правило, виконуються лише перші три. Щоб таблиця вважалася нормальною для першого нормального формату, кожне з її полів має бути неподільним і не мати дубльованих груп.

Поле вважається неподільним, якщо воно містить лише один елемент даних. Наприклад, адреси постачальників і оптових покупців краще розбити на 2-3 поля: місто, вулиця, будинок, тоді простіше вибрати всіх, хто доставляє або замовляє товари за однією адресою.

Неповторення означає, що у деякі поля не повторюють значення інших полів.

Дубльована група — це поле, яке дублюється для зберігання кількох значень для атрибута в середині визначення запису.

Щоб привести таблицю до другого нормального формату, всі неключові поля повинні повністю залежати від первинного ключа таблиці та від первинного ключа в кожному полі, якщо останнє складається з кількох полів. Щоб таблицю було приведено до третього нормального формату, усі неключові

поля повинні бути повністю залежними від первинного ключа таблиці та незалежними одне від одного. Таким чином, вимога, щоб кожне неключове поле таблиці отримало незалежність від інших неключових полів, додається до ефективності другого нормального формулювання. Якщо у, то будь-яка таблиця має такі поля.

2.4 Аналіз концептуальних вимог

На рівні аналізу концептуальних вимог та інформаційних потреб необхідно вирішити наступні завдання:

- аналіз вимог користувача до бази даних (концептуальні вимоги;
- виявлення проблем обробки даних, відображених у базі даних (аналіз програмного забезпечення);
- визначення модернізації бази даних;
- документування результатів аналізу.

Вимоги користувача до БД розробки являють собою перелік питань, при цьому визначаються вимоги щодо введення, оновлення та редагування інформації. Вимоги користувача визначаються та доповнюються під час аналізу існуючих і потенційних додатків.

Наприклад, при розробці бази даних зберігання електронних документів для меблевого магазину необхідно відповісти на такі питання;

- які комплектуючі є в магазині;
- які комплектуючі є на складі;
- до яких типів вони належать;
- яка кількість є в наявності;
- коли закінчується термін дії договору з певним постачальником;
- як часто оновлюється інформація в базі даних;

— які звіти, та запити доступні.

Необхідно розв'язати задачі;

— ведення звітності;

— статистичний аналіз продажу тих чи інших товарів;

— визначення найперспективніших напрямків для подальших заупок

.На основі інформації, що зберігається в базі даних, повинні бути створені такі звіти:

— дані про кількість і типи товарів;

— дані про клієнтів;

— дані про роботу з постачальниками;

— виявлення інформаційних об'єктів та зв'язки між ними;

Друга фаза аналізу предметної області — це виділення інформаційних об'єктів, присвоєння кожному об'єкту необхідних атрибутів, виявлення зв'язків між об'єктами, визначення обмежень на інформаційні об'єкти, типи зв'язків між ними, атрибути інформаційних об'єктів.

При відборі інформаційних елементів необхідно знайти відповіді на ряд питань:

— на які таблиці можна розбити дані, що зберігаються в базі даних?

— яку назву можна присвоїти кожній таблиці?

— які найважливіші характеристики можна виділити?

— які назви можна присвоїти виділеним атрибутам?

У нашому випадку передбачається створити наступні таблиці.

Таблиця 2. — Атрибути виділених інформаційних об'єктів

Таблиця 2.2 — Властивості вибраних інформаційних об'єктів

Співробітники	Облікові дані п-лу	Особисті дані п-лу
---------------	--------------------	--------------------

	Посада	Ідентифікаційний код
	Дата прийому на роботу	Паспортні дані
		Дата народження
		Домашня адреса

При виборі інформаційних об'єктів необхідно відповісти на ряд питань:

- на які таблиці можна розбити дані, що будуть зберігатися в БД;
- яке ім'я можна привласнити кожній таблиці;
- які найцікавіші характеристики можна виділити;
- які імена можна привласнити обраним характеристикам.

При проектуванні бази даних, між інформаційними об'єктами існують зв'язки: «один до одного», «один до багатьох» і «багато до багатьох» (рисунки 2.1).

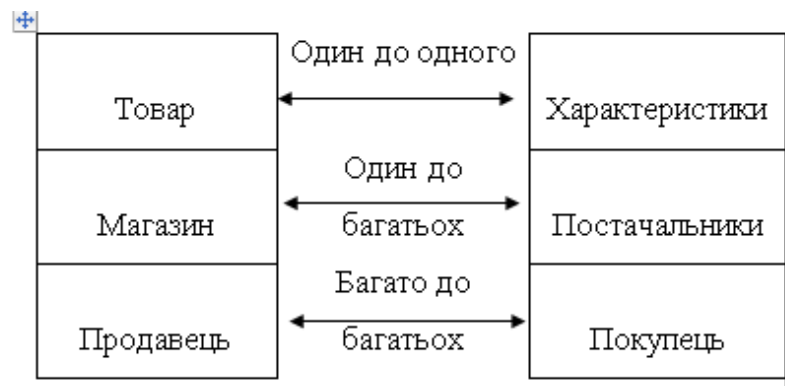


Рисунок 2.1— Типи зв'язків

Виділимо зв'язки між інформаційними об'єктами (рис.2.2)

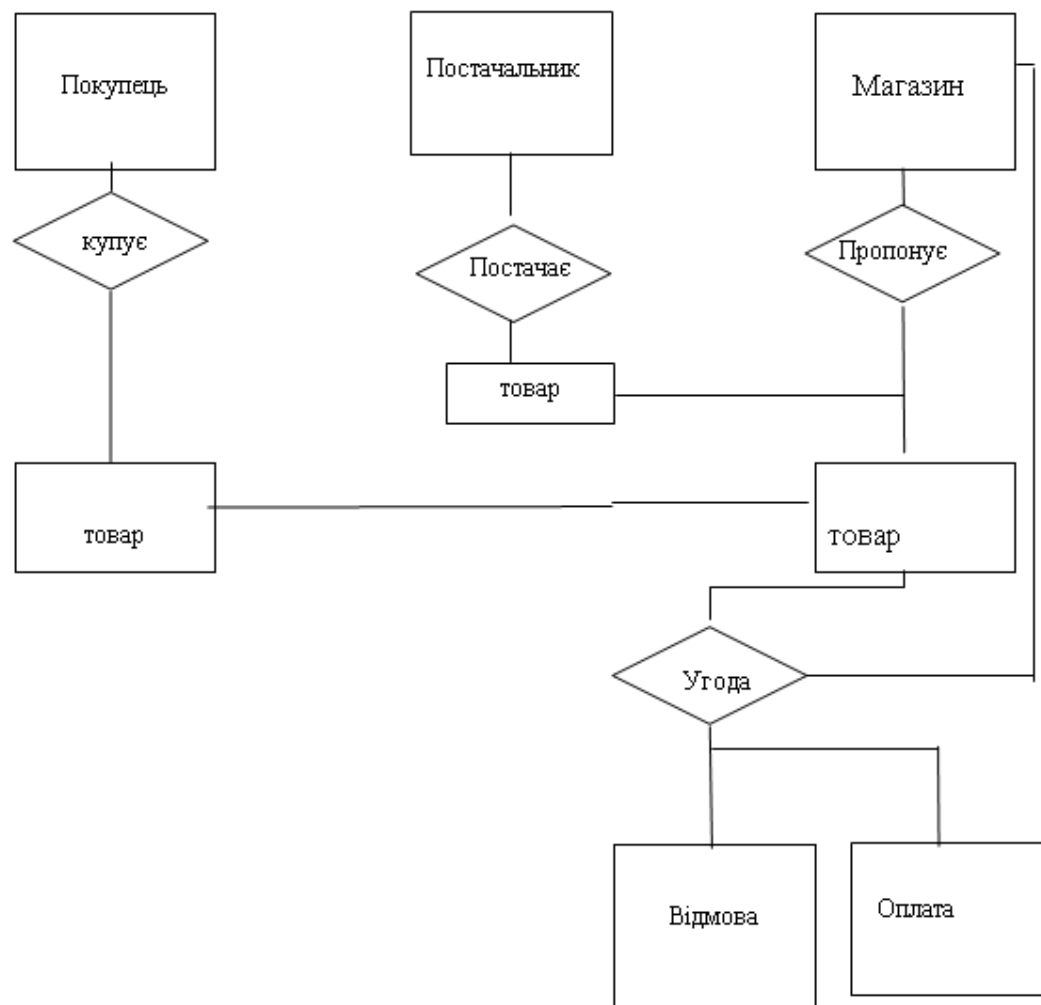


Рисунок 2.2 — Зв'язки між інформаційними об'єктами

Концептуальна діаграма ієрархічної моделі — це набір типів записів, пов'язаних з одним або кількома типами дерева. У цій моделі всі типи взаємодії є «один до багатьох» і відображаються стрілками.

Таким чином, зв'язок між об'єктами є генеалогічним деревом зв'язку, за одним винятком: тип кожного створеного (дочірнього) об'єкта може бути лише типом вхідного (батьківського).

Тобто ієрархічна модель даних допускає лише два типи зв'язків між об'єктами: «один до одного» та «багато до одного». Ієрархічні бази даних є

навігаційними, тобто доступ можливий лише через заздалегідь визначені з'єднання.

Під час моделювання подій часто потрібні такі зв'язки, як «більше одного». Дубльовані об'єкти можуть бути запропоновані як можливе рішення для подолання цього обмеження. Однак надмірність унеможлиблює синхронізацію даних.

Перевага ієрархічної бази даних полягає в тому, що її функції навігації роблять її швидко доступною під час проходження попередньо визначених зв'язків. Ще одним недоліком ієрархічної моделі даних є те, що пошук інформації з нижчих рівнів ієрархії не зосереджений на вищих вузлах.

2.5 Вибір конкретної СУБД

Одним із основних критеріїв вибору бази даних є оцінка того, чи може внутрішня модель даних, підтримувана системою, визначати концептуальні схеми. Системи керування базами даних, як правило, призначені для персональних комп'ютерів, часто підтримують модель з'єднання або мережі. Більшість сучасних баз даних взаємопов'язані.

Створення бази даних на основі моделей зв'язків має важливіші переваги перед іншими моделями. Незалежність фізичного та користувацького представлення логічної структури.

Важливою частиною бази даних є мова обробки даних, яка використовується різними програмами в роботі бази даних. Як правило, мова маніпулювання інформацією була введена в мову програмування. Також при виборі СУБД, що реалізує конкретну СУБД, необхідно оцінити технічні аспекти функціонування, які безпосередньо пов'язані з роботою системи. Загалом, щоб вибрати СУБД, потрібно оцінити сім основних характеристик, вони наведені нижче:

- характеристики ПК: виробник, тип, модель, наявність гарантії;
- керування файлами та пошук: типи посилань, оптимізація файлів, посилання на двійкові таблиці, вибір засобів обробки даних, типи пошуку;
- інструменти підтримки програмного забезпечення: інформація про каталог, генератор додатків, операційна мова, підпрограми, макроси, відладчик, виконання підтримки системи, програмне забезпечення та шифрування даних, обмеження доступу, графіка, текстовий редактор, статистика;
- сприяти та підтримувати цілісність: управління та контроль за замовленнями та контролювати списки, перевіряти цілісність даних у таблиці, перевірка ключа, перевірка дати, незалежність даних;
- Звітність — звіти з кількома файлами, збереження форматів звітів, перегляд звітів, які обчислюють екран, поля, медіа-групи, перевизначення формату дати, заголовку;
- операційне середовище: тип операційної системи, обсяг використовуваної оперативної пам'яті, необхідність використання енергонезалежної пам'яті, обсяг постійної пам'яті, мова підсистеми;
- інша інформація: наявність мережесхемних опцій, ціни, примітки, джерела.

Завжаючи на вищевикладене, для поставленого завдання було вибрано реляційну СУБД Microsoft Access, яка є частиною потужного набору програм Microsoft Office і тому доступна для користувачів і розробників, яким доводиться мати справу з використанням і створенням програм баз даних практично на кожному комп'ютері під керуванням операційної системи Windows. Microsoft Access поставляється з багатьма різними шаблонами, за допомогою яких можна миттєво створити потрібну програму, включаючи необхідні таблиці. База даних містить усі необхідні зв'язки, елементи, необхідні для створення запитів, форм, звітів та інші важливі елементи для додавання до електронної таблиці.

Можна погодитися, що програми, призначені для виконання хоч якоїсь корисної роботи, так чи інакше використовують структуровані дані, іншими словами, впорядковану інформацію.

Така інформація може включати, наприклад, деталі замовлень на певний продукт, деталі поданих та оплачених рахунків, деталі номерів телефонів.

2.6 Таблиці

В Access розробка бази даних починається з визначення реляційних таблиць і їх полів, які будуть містити дані.

Таблиці створюються трьома способами: в режимі конструктора, в режимі майстра, шляхом введення даних. У цьому випадку перевага віддається способу створення таблиці в режимі конструктора.

Основні етапи:

- вибір назви поля таблиці;
- завдання типу даних поля;
- заповнення стовпців таблиці.

Поля, які мають бути присутні в таблицях, базуються на принципах, які враховують усі релевантні в цьому дослідженні характеристики об'єкта після нормалізації. Потім описуються реляційні зв'язки між наведеними раніше таблицями. Таблиці можна створювати в режимі майстра, але створення в режимі конструктора зручніше.

В ім'я поля записується назва стовпця, а тип вмісту відповідного стовпця – до типу даних. Нижче наведено приклад створення таблиці бази даних «Особисті дані персоналу» в режимі конструктора (рис. 2.3 -2.5).

Имя поля	Тип данных	Описание
Код співробітника	Счетчик	
Прізвище	Текстовый	
Ім'я	Текстовый	
По батькові	Текстовый	
Сімейний стан	Текстовый	
Кількість дітей до 16 років	Числовой	
Адреса	Текстовый	
Телефон	Числовой	

Рисунок 2.3 — Створення таблиці

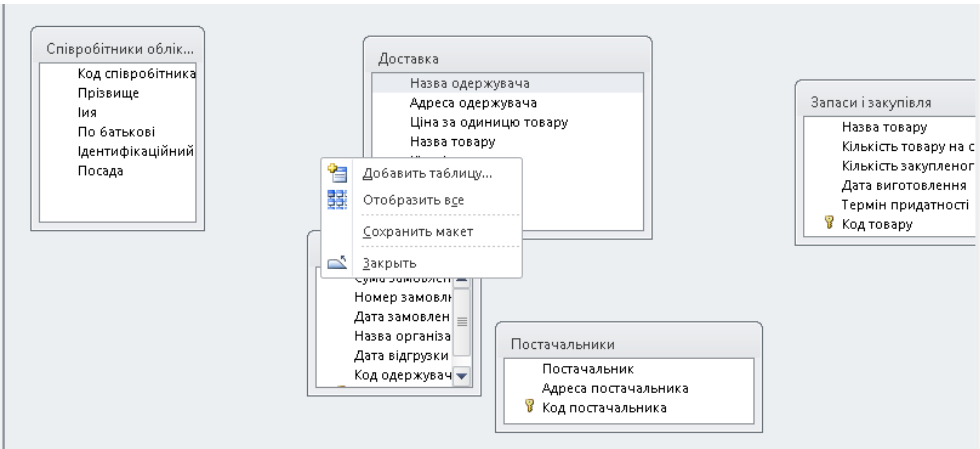


Рисунок 2.4 — Додавання таблиць

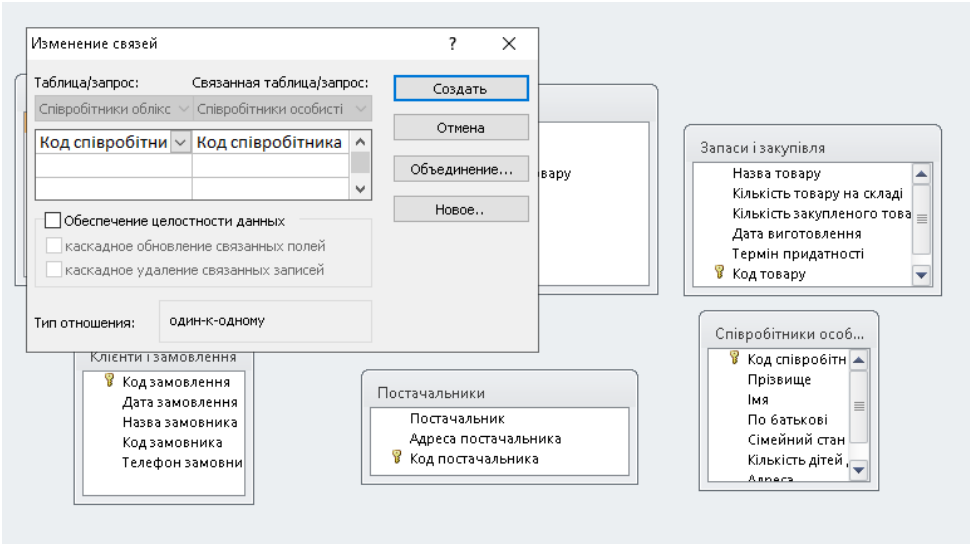


Рисунок 2.5 — Об'єднання таблиць

Далі таблиці необхідно об'єднати між собою, для чого створюється схема даних (рис. 2.6).

Під час виділення зв'язків між таблицями необхідно дотримуватися правил щодо забезпечення цілісності даних, їх каскадного оновлення та видалення. Нижче наведено схему зв'язків між таблицями.

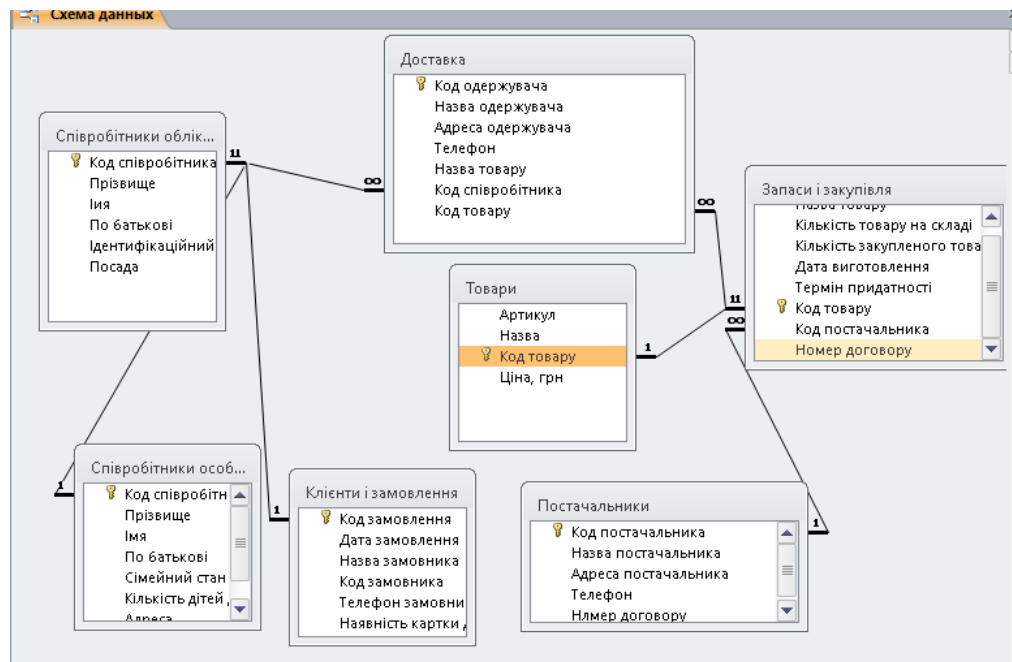


Рисунок 2.6 — Схема бази даних

2.7 Форми

Форми — це об'єкт бази даних, який використовується для створення інтерфейсу користувача для програми бази даних. «Зв'язана» форма — це форма, яка безпосередньо підключена до джерела даних, наприклад таблиці або запиту, і може використовуватися для вставлення, редагування або відображення даних із джерела даних.

У той же час також можна створити «головну» форму, яка не взаємодіє безпосередньо з джерелом даних, але містить кнопки, мітки або інші елементи

керування, необхідні для роботи програми. Пов'язані форми діють як вікна, які дозволяють користувачам переглядати дані та отримувати доступ до бази даних.

Ефективна форма дозволяє прискорити використання основної інформації, користувачам не доводиться довго шукати необхідну інформацію. Проектування форм під час створення бази даних — це діяльність, яка дозволяє користувачам працювати приємно та ефективно. Форма також може запобігти неправильному введенню даних. Звичайним підходом є збірка в режимах конструктора та майстра (рис. 2.7).

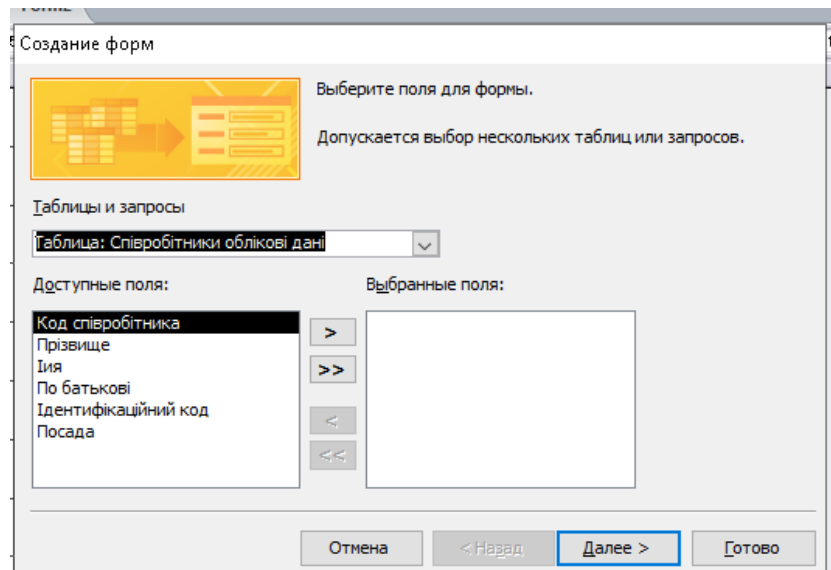


Рисунок 2.7 — Створення форм

Коли ви вибираєте команду в меню введення форми, на екрані відкривається вікно «Нова форма», у якому можна вибрати таблицю або запит, для якого була створена нова форма, і вказати спосіб її створення. Окрім «ручного» створення форми, у режимі майстра створення форми можна автоматизувати за допомогою конструктора форм. Елементами керування можуть бути текст, графіка та інші статичні об'єкти, які не змінюються під час перемикавання між записами. Це також має включати текстові поля, вміст яких

змінюється під час навігації між записами. Елементи керування можна використовувати для введення та відображення дат, виконання обчислень і виведення результатів. Елементи керування — це командні кнопки, що активують різні операції; Об'єкти типу є підкладками; об'єкти, що полегшують сприйняття даних, наприклад, календар або лічильник; та елементи користувача.

Часто для створення елемента керування достатньо перетягнути мишкою з панелі інструментів на форму. Кожен елемент розташовується в певній частині форми. Залежно від типу розділу (заголовок форми, область даних тощо) елемент керування з'являтиметься один раз, на сторінку, на групу записів або на запис.

Якщо ви вирішили створити форму за допомогою FormWizard, передбачається така послідовність дій;

Виберіть Налаштування / Створити. На екрані з'явиться діалогове вікно «Новий шаблон», у якому необхідно вибрати спосіб створення шаблону та вихідний об'єкт для створення шаблону. Таким об'єктом можна вибрати таблицю або запит (рис. 2.8).

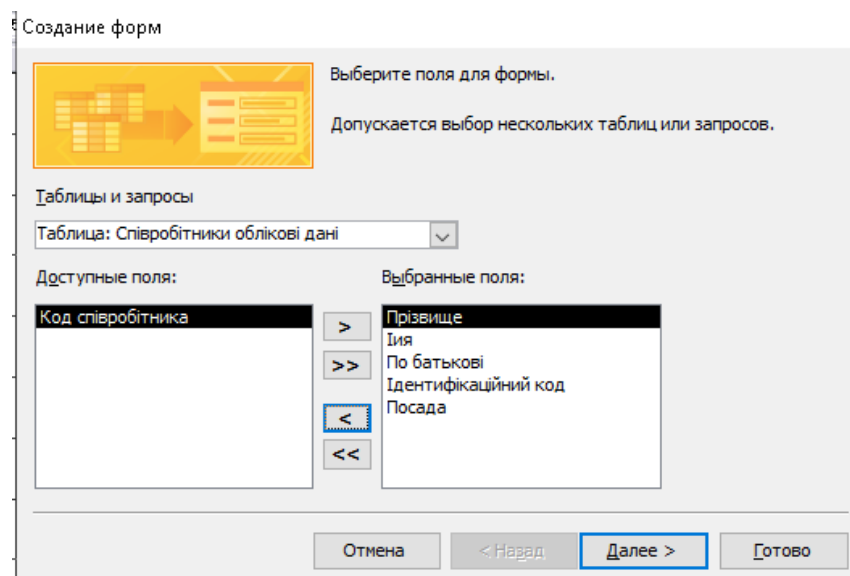


Рисунок 2.8 — Вибір полів для форми

Якщо форму потрібно створити, то як спосіб створення форми вибрали таблицю «Співробітники облікові дані» Майстер форм. Натисніть кнопку **Ок**, щоб перейти до наступного діалогового вікна (рис. 2.10).

Потім потрібно вибрати поля, які потрібно розмістити у формі. Кнопки знаходяться в нижній частині екрана і дозволяють вам зробити крок назад або продовжити процес створення шаблону далі.

Після цього майстер форм запропонує вибрати форму представлення форми (у стовпець, стрічка, таблиця або вирівняна) і стиль її оформлення.

Стилі — це набір різних фонових зображень із відповідним вибором форм і форм полів (рис. 2.10).

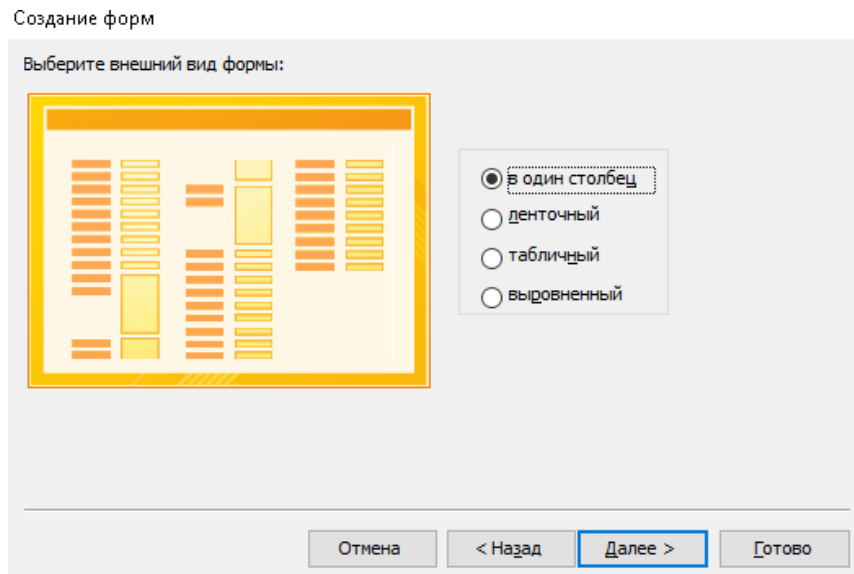


Рисунок 2.10 — Вибір полів для форми

Кінцевий вигляд створеної форми (рис. 2.11)

Рисунок 2.11 — Форма для заповнення даних про облікові дані

2.8 Звіти

Звіти, як і форми, є основними та важливими елементами інтерфейсу бази даних. Вони дозволяють експортувати інформацію про заповнення бази даних у зручному форматі. Більшість звітів генерується на етапі розробки офісної програми.

Звіт містить інформацію, отриману з таблиць або запитів, а також інформацію, що зберігається в конструкторі звітів, таку як мітки, заголовки та графіки. Таблиці та запити, які містять отримані дані, також називають джерелом звіту. Якщо в таблиці є необхідні поля, можна вказати цю таблицю як джерело записів. Якщо вам потрібні поля в кількох таблицях, вам слід створити один або кілька запитів для використання як вихідного запису. Ці запити можуть уже існувати в базі даних, але іноді потрібно створювати нові запити для задоволення потреб звіту.

Нижче наведено звіт, створений на основі таблиці «Товари»

Товари			
Артикул	Назва	Код товару	Ціна, грн
Ручки меблевi		1	
82842	Ручка врізна Citterio Line 4058-69-160 титан	2	571,03 ₴
82841	Ручка врізна Citterio Line 4058-71-160 білий матовий	3	562,87 ₴
118505	Ручка врізна Gamet MD04-202/128-G0004 хром	4	313,94 ₴
118506	Ручка врізна Gamet MD04-202/128-R31 чорний матовий	5	185,38 ₴
114594	Ручка врізна Gamet UN50-0096-G0004 хром	6	196,40 ₴
114596	Ручка врізна Gamet UN50-0096-G0008 матовий хром	7	200,92 ₴
114597	Ручка врізна Gamet UN50-0096-R31 чорний матовий	8	181,00 ₴
114593	Ручка врізна Gamet UN50-0128-G0004 хром	9	229,27 ₴
114595	Ручка врізна Gamet UN50-0128-G0008 матовий хром	10	263,66 ₴
114692	Ручка врізна Gamet UN50-0128-R31 чорний матовий	11	226,25 ₴
106121	Ручка врізна GIFF 9/105/128 сатин нікель	12	163,88 ₴
106120	Ручка врізна GIFF 9/105/128 титан матовий	13	183,05 ₴
122518	Ручка врізна GIFF 9/105/128 титан матовий/титан глянець	14	79,64 ₴
106119	Ручка врізна GIFF 9/105/96 сатин нікель	15	123,18 ₴
106118	Ручка врізна GIFF 9/105/96 титан матовий	16	154,41 ₴
122519	Ручка врізна GIFF 9/105/96 титан матовий/титан глянець	17	67,74 ₴

Рисунок 2.12 — Звіт «Товари»

2.9 Макроси

Макроси в Access можна розглядати як спрощену мову програмування, яку можна використовувати для додавання функцій до вашої бази даних. Наприклад, ви можете прикріпити макрос до кнопки форми, щоб запустити макрос, коли кнопку натиснуто. Макроси містять дії, які виконують такі завдання, як відкриття звіту, виконання запиту або закриття бази даних. Більшість ручних операцій з базою даних можна автоматизувати за допомогою макросів, тому вони можуть заощадити час.



Рисунок 2.13 — Макрос «Відкриття таблиці товари»

3 НАСТАНОВИ КОРИСТУВАЧЕВІ

3.1 Інструкція користувачеві для роботи із програмою ведення документації та звітності.

Програма призначена для збереження документів, автоматизації діловодства та надання різноманітних звітів і довідок.

У базу даних вносяться дані про співробітників, товари, партнерів і клієнтів.

Робота починається з головного вікна, де має з'явитися заголовок назви компанії (рис. 3.1).

На початку завдання з'являються тільки дві командні кнопки: кнопка «Зупинити» і кнопка «Вікно БД», за допомогою макросів можна завершити роботу цим вікном, а оскільки це вікно є основним, завершення означає робота з ним і програмою в цілому Це означає завершення роботи

Натисніть на іконку на вкладці «Постачальники», щоб заповнити дані про партнерів-постачальників деталей (рис. 3.1). У вікні, що відкриється, заповніть усі поля форми. Переходи між полями здійснюються клацанням миші на відповідних кнопках. Коли ви закінчите, перейдіть на іншу вкладку або натисніть кнопку «Зупинити».

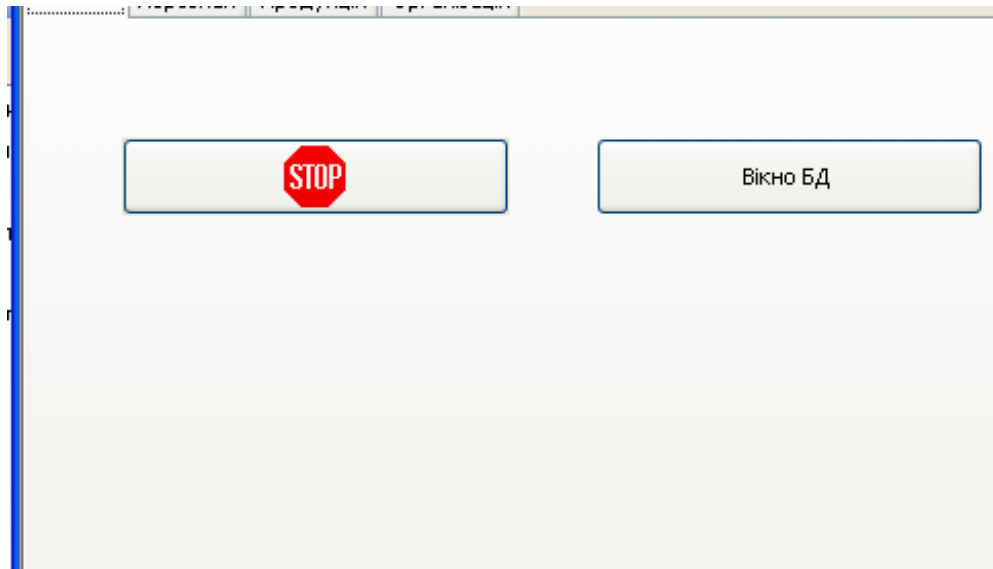


Рисунок 3.1 — Основне вікно

Щоб заповнити таблицю «Клієнти», натисніть вкладку «Клієнти». Заповніть усі поля та перейдіть на іншу вкладку, або натисніть кнопку «Зупинити». Перехід від одного поля до іншого здійснюється натисканням клавіші «Tab» або «ENTER» на клавіатурі. Коли всі поля заповнені, натисніть клавішу «ENTER» на клавіатурі.

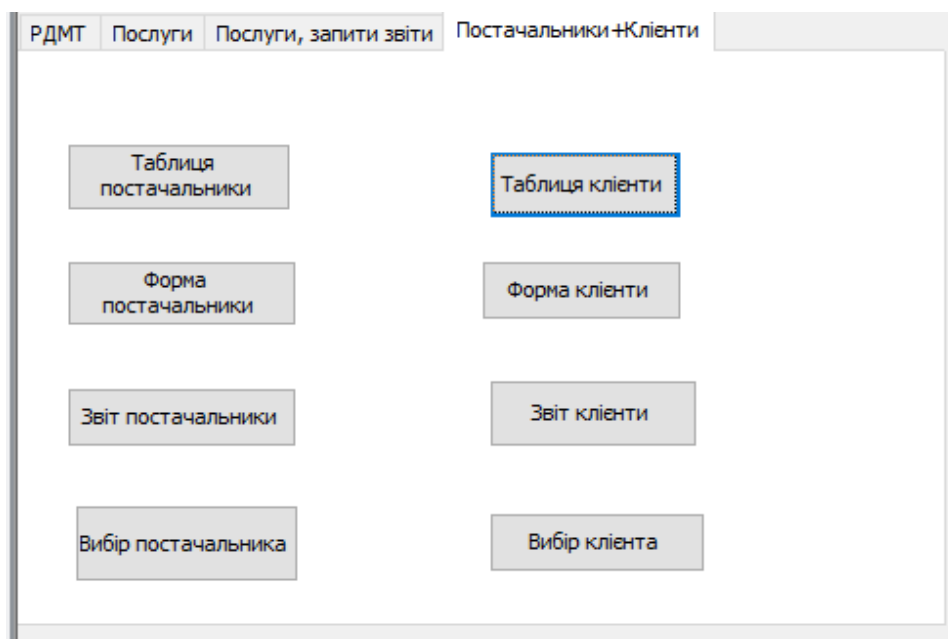


Рисунок 3.2 — Заповнення даних про клієтів

Для наступного запису натисніть клавішу з трикутником, спрямованим праворуч, і спочатку виконайте всю послідовність дій.

Двічі клацніть на прізвищі, щоб видалити. Натисніть на клавіатурі кнопку «Видалити».

Найкраще заборонити заповнення, редагування та видалення всім користувачам, крім системного адміністратора, оскільки наслідком невмілих дій може бути втрата інформації в базі.

Якщо видалити поле з цієї таблиці, буде видалено записи в усіх таблицях, пов'язаних із цим полем.

Ця таблиця служить класифікатором назв організацій-партнерів (рис. 3.3).

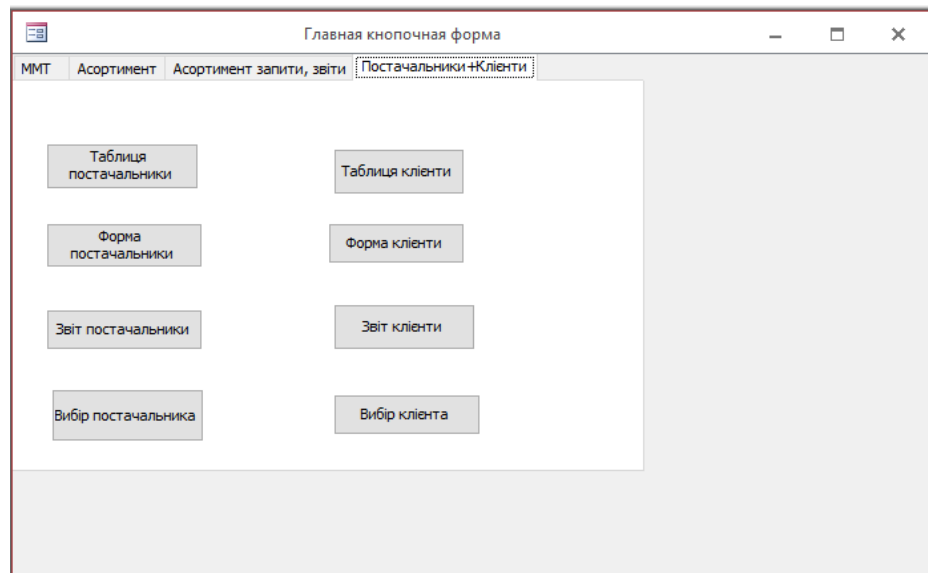


Рисунок 3.3 — Заповнення таблиці

Заповніть усі поля та натисніть кнопку «ENTER». Щоб змінити значення поля, виберіть те чи інше поле та введіть виправлення, помістивши його в потрібне поле, потім натисніть клавішу «ENTER» на клавіатурі.

3.3 Тестування БД

Можна використати запити для перевірки бази даних на результати різної інформації (рис. 3.4). Запити призначені для створення запитів, пов'язаних із даними в базі даних. На основі того, як формується запит, дані можуть бути отримані з різних таблиць бази даних і об'єднані, можуть бути обчислені нові значення.

Наприклад, ви можете створити питання про те, які товари зараз доступні та яка їх ціна або вибрати варіанти. За допомогою запитів також можна виконувати дії з групами даних, наприклад, оновлювати або виключати певні групи записів.

Вибирається відповідна вкладка, натискається кнопка, що відповідає за ту чи іншу інформацію. Приклади наведено нижче.

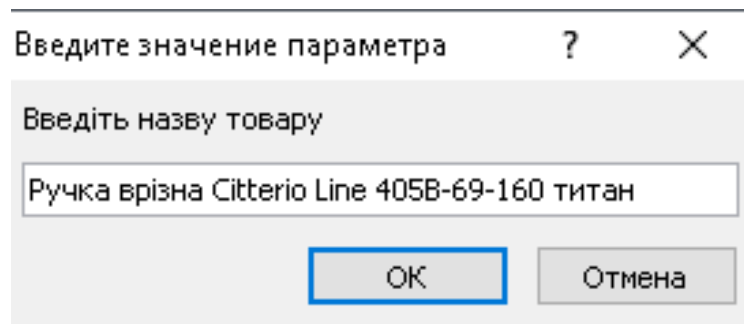


Рисунок 3.4 — Вікно запиту про товари

Розробка дає можливість переглядати і друкувати звіти, для чого на відповідній вкладці вибираються потрібні кнопки.

Товари			
Артикул	Назва	Код товару	Ціна, грн
	Ручки меблевi	1	
82842	Ручка врiзна Citterio Line 405B-69-160 титан	2	571,03 ₴
82841	Ручка врiзна Citterio Line 405B-71-160 бiлий матовий	3	562,87 ₴
118505	Ручка врiзна Gamet MD04-202/128-G0004 хром	4	313,94 ₴
118506	Ручка врiзна Gamet MD04-202/128-R31 чорний матовий	5	185,38 ₴
114594	Ручка врiзна Gamet UN50-0096-G0004 хром	6	196,40 ₴
114596	Ручка врiзна Gamet UN50-0096-G0008 матовий хром	7	200,92 ₴
114597	Ручка врiзна Gamet UN50-0096-R31 чорний матовий	8	181,00 ₴
114593	Ручка врiзна Gamet UN50-0128-G0004 хром	9	229,27 ₴
114595	Ручка врiзна Gamet UN50-0128-G0008 матовий хром	10	263,66 ₴
114692	Ручка врiзна Gamet UN50-0128-R31 чорний матовий	11	226,25 ₴
106121	Ручка врiзна GIFF 9/105/128 сатин нiкель	12	163,88 ₴
106120	Ручка врiзна GIFF 9/105/128 титан матовий	13	183,05 ₴
122518	Ручка врiзна GIFF 9/105/128 титан матовий/титан глянець	14	79,64 ₴
106119	Ручка врiзна GIFF 9/105/96 сатин нiкель	15	123,18 ₴
106118	Ручка врiзна GIFF 9/105/96 титан матовий	16	154,41 ₴
122519	Ручка врiзна GIFF 9/105/96 титан матовий/титан глянець	17	67,74 ₴

Рисунок 3.5 — Вікно звіту про товари

ВИСНОВКИ

У бакалаврській роботі проаналізовано вимоги користувачів, відібрано відповідні інформаційні елементи та їх характеристики, а на основі їх аналізу сконструйовано предмет.

Була створена база даних, де реалізовано основний інтерфейс для непрофесійного користувача, який міг отримати актуальну інформацію про стан роботи організації. Також була створена схема бази даних, що поєднує таблиці, форми та макроси, звіти бази даних.

Розроблений додаток забезпечує всі необхідні вимоги та потреби, в тому числі;

- надати інформацію про співробітників організації;
- надати інформацію про партнерів-постачальників;
- надавати інформацію про товари, які є в наявності;
- надавати інформацію про клієнтів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Історія розвитку баз даних. URL: <https://helpiks.org/2-9486.html>.
2. Гайна Г. А. Основи проектування баз даних: навч. посіб. Київ: КНУБА, 2005. 204 с.
3. Класифікація моделей даних. URL: <https://helpiks.org/2-9488.html>.
4. Реляційна алгебра. URL: https://rdb.dp.ua/uk/chapter_05.
5. Реляційна алгебра. URL: http://elenagavrile.narod.ru/dm/Lekccya_6.pdf.
6. Реляційна алгебра. URL: https://stud.com.ua/93791/informatika/relyatsiy_na_algebra#43.
7. Пасічник В. В., Реаніченко В. А. Організація баз даних та знань. Київ: Видавнича група ВНУ, 2006. 384 с.
8. Д. Мейер. Теорія реляційних баз даних. – Київ. 2000 р. – 322 с.
9. Романюк О.Н., Савчук Т.О. Організація баз даних і знань. Навчальний посібник. – Вінниця: УНІВЕРСУМ – Вінниця, 2003. – 217 с.

ДОДАТОК А

Міністерство освіти та науки України
Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії

ЗАТВЕРДЖУЮ

Завідувач кафедри ОТ ВНТУ

д.т.н., проф.

_____ О. Д. Азаров

“06” травня 2024 р.

ТЕХНІЧНЕ ЗАВДАННЯ

на виконання бакалаврської дипломної роботи

«Інформаційна система меблевого магазину»

08–54.БДР.001.00.000 ТЗ

Науковий керівник к.т.н., доц. каф. ОТ

_____ Колесник І.С.

Студент групи 1КІ–22мс

_____ Авраменко О.А.

Вінниця 2024

1 Найменування та область застосування

Інформаційна система меблевого магазину.

2 Основи для розробки

Необхідність проектування інформаційної системи меблевого магазину.

3 Мета та призначення розробки

Створення інформаційної системи для меблевого магазину.

4 Етапи БДР та очікувані результати

Робота виконується у вісім етапів, що наведені в таблиці 4.1.

Таблиця А.1 — Етапи виконання роботи

з/п	Назва етапів виконання бакалаврського проекту	Строк виконання етапів роботи	Примітка
1	Постановка задачі для інформаційної системи	27.02.24	
2	Огляд інформаційних джерел	06.03-20.03.24	
3	Аналіз основних понять	27.03-7.04.24	
4	Розробка бази даних	10.04-21.04.24	
5	Настанови користувачеві	24.04-12.05.24	
6	Оформлення пояснювальної записки та ілюстративного матеріалу	15.05-26.05.24	
7	Аналіз виконання проекту. Висновки. Додатки	29.05-09.06.24	
8	Перевірка якості виконання бакалаврського проекту та усунення недоліків	06.06.24	

5 Матеріали, що подаються до захисту БДР

Пояснювальна записка БДР, графічні і ілюстративні матеріали, протокол попереднього захисту БДР на кафедрі, відзив наукового керівника, рецензія опонента, протоколи складання державних екзаменів, анотації до БДР українською та іноземною мовами, довідка про відповідність оформлення БДР діючим вимогам.

6 Порядок контролю виконання та захисту БДР

Виконання етапів графічної та розрахункової документації БДР контролюється науковим керівником згідно зі встановленими термінами. Захист БДР відбувається на засіданні Державної екзаменаційної комісії, затвердженою наказом ректора.

7. Вимоги до оформлення БДР

Вимоги викладені в МЕТОДИЧНИХ ВКАЗІВКАХ до дипломного проектування, ДСТУ 3008–95, ДСТУ 3974–2000 «Правила виконання дослідно–конструкторських робіт. Загальні положення» та діючого ГОСТ 2.114–95 ЄСКД.

.

ДОДАТОК Б

Трирівнева архітектура СУБД

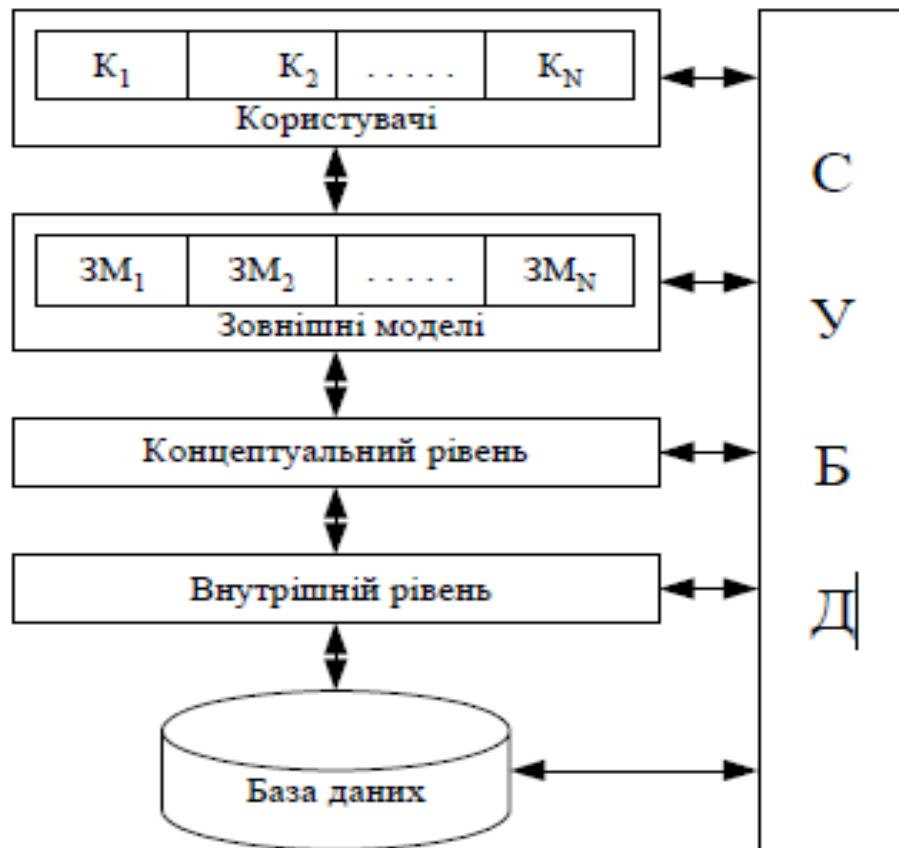


Рисунок Б1 — Трирівнева архітектура СУБД

ДОДАТОК В

Зв'язки між інформаційними об'єктами

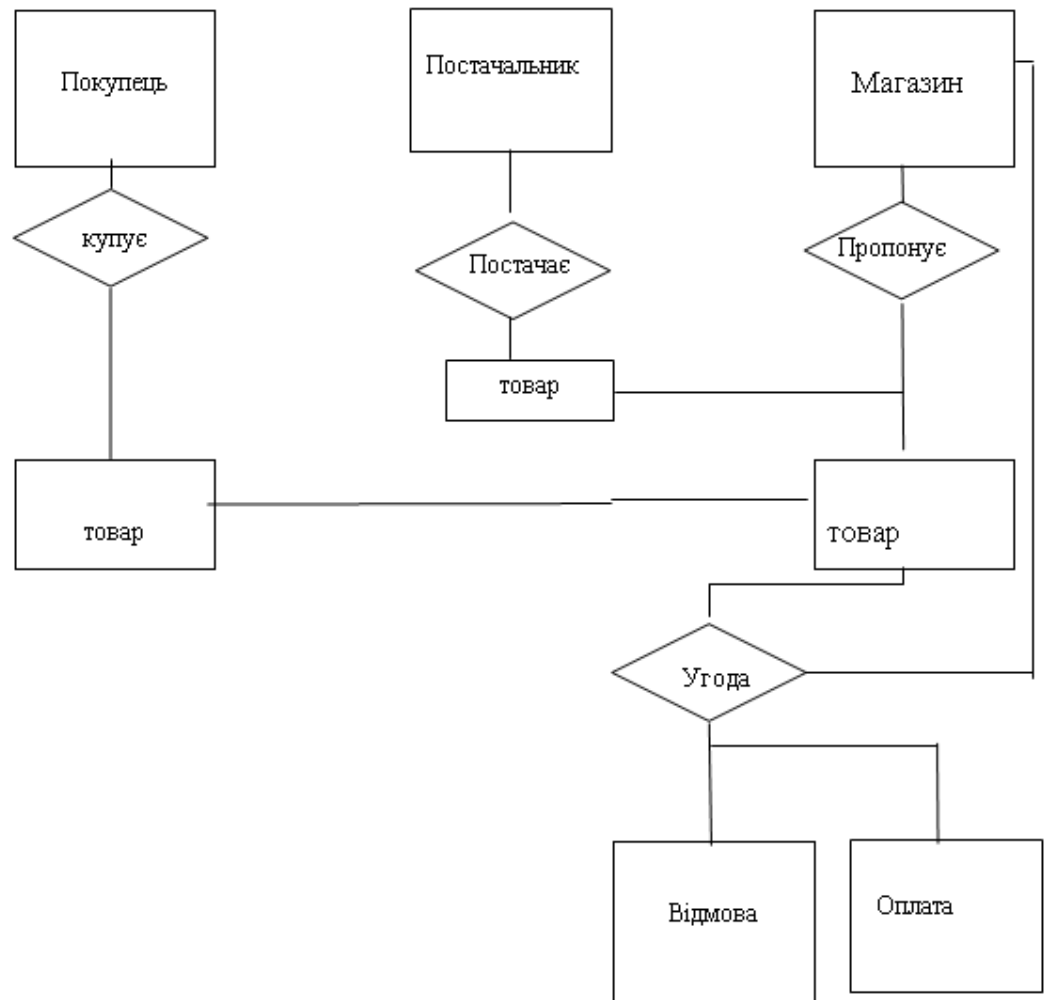


Рисунок В.1 — Зв'язки між інформаційними об'єктами

ДОДАТОК Г

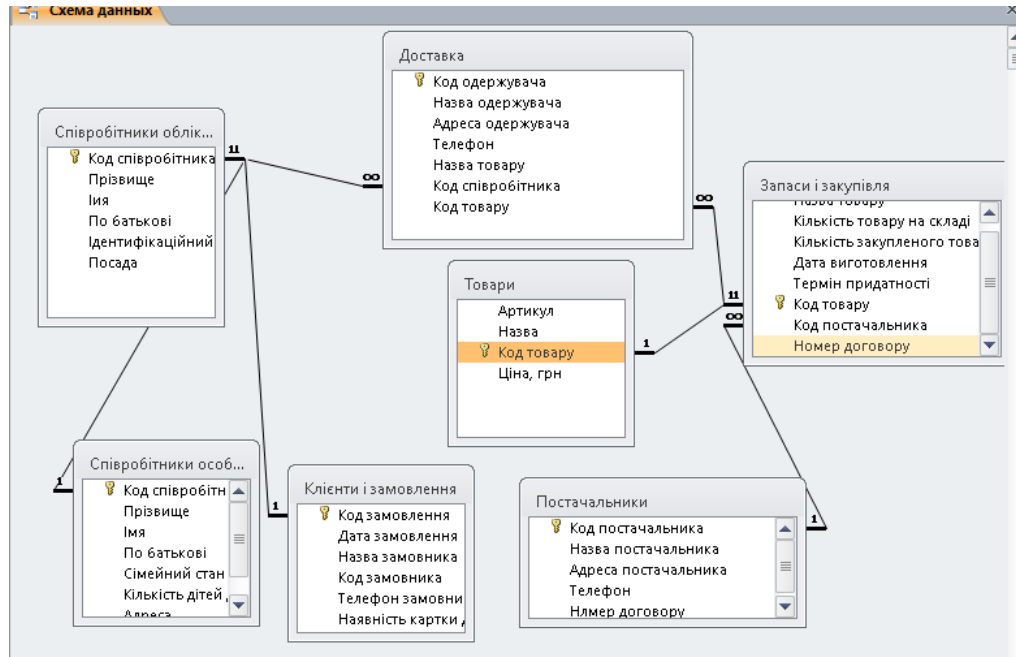


Рисунок Г.1 — Схема бази даних

ДОДАТОК Д

ПРОТОКОЛ ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ НА НАЯВНІСТЬ ТЕКСТОВИХ ЗАПОЗИЧЕНЬ

Назва роботи: Інформаційна система меблевого магазину

Тип работи: Бакалаврська дипломна робота
(БДР, МКР)

Підрозділ _____ кафедра обчислювальної техніки
(кафедра, факультет)

Показники звіту подібності Unichек

93.7%	Схожість	6.3%
-------	----------	------

Аналіз звіту подібності (відмітити потрібне):

- ✓ Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.
- Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її виконання автором. Роботу направити на розгляд експертної комісії кафедри.
- Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Особа, відповідальна за перевірку _____ Захарченко С.М.
(підпис) (прізвище, ініціали)

Ознайомлені з повним звітом подібності, який був згенерований системою Unichack щодо роботи.

Автор роботи		<u>Авраменко О.А.</u>
	(підпис)	(прізвище, ініціали)
Керівник роботи		<u>Колесник І.С.</u>
	(підпис)	(прізвище, ініціали)