

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

СУМСЬКИЙ НАЦІОНАЛЬНИЙ АГРАРНИЙ УНІВЕРСИТЕТ

Факультет економіки та менеджменту

Кафедра кібернетики та інформатики

БАЗИ ДАНИХ ТА СУБД

**Методичні вказівки щодо
виконання курсової роботи**

**для студентів 2-го курсу
за спеціальністю: 126 “Інформаційні системи та технології”
денної форми навчання
освітнього ступеня «бакалавр»**

Суми 2022

УДК 621.382
М 54

Укладач: Пасько Н.Б., к.т.н., доцент кафедри кібернетики та інформатики

Пасько Н.Б.

М 54 **Бази даних та СУБД:** методичні вказівки щодо виконання курсової роботи /Уклад.: Н.Б. Пасько . – Суми, 2022. – 111 с.

Методичні вказівки спрямовані на надання методичної допомоги під час вивчення курсу «Бази даних та СУБД» та виконання курсових робіт з дисципліни. Містять загальні методичні рекомендації, теоретичні питання, список рекомендованої літератури, варіанти тем курсових робіт та приклад виконання курсової роботи щодо однієї теми. Призначені для студентів 2-го курсу освітнього ступеня бакалавр за спеціальністю 126 «Інформаційні системи та технології» денної форми навчання.

Рецензенти:

Барченко Н.Л., к.т.н., доцент кафедри комп'ютерних наук СумДУ;

Толбатов А.В., к.т.н., доцент кафедри кібернетики та інформатики СНАУ

Відповідальний за випуск:

Толбатов А.В., к.т.н., доцент кафедри кібернетики та інформатики СНАУ

Рекомендовано до видання навчально-методичною радою факультету економіки та менеджменту СНАУ. Протокол № 3 від «10» 02 2022 року

© Сумський національний
аграрний університет, 2021

ЗМІСТ

Вступ	4
РОЗДІЛ 1. Основні теоретичні положення проектування баз даних	9
1.1. Цілі проектування бази даних, етапи проектування	9
1.2. Формулювання та аналіз вимог до бази даних	17
1.3. Передпроектний аналіз проблемної області	19
1.4. Концептуальне інфологічне проектування бази даних	22
1.5. Проектування глобальної інфологічної моделі проблемної області	29
1.6. Даталогічне проектування бази даних	32
1.7. Фізичне проектування бази даних	45
1.8. Вибір системи управління базами даних	46
РОЗДІЛ 2. Курсова робота з дисципліни "Бази даних та СУБД"	49
2.1 Завдання та варіанти курсових робіт	50
2.2 Варіанти курсових робіт	50
Список використаних джерел	66
ДОДАТОК Приклад виконання курсової роботи	

ВСТУП

Мета дисципліни «Бази даних та СУБД»: є отримання майбутніми спеціалістами ІТ-галузі відповідного рівня фахово спрямованих теоретичних знань, формування та розвиток спеціальних умінь і практичних навичок з розробки та використання баз даних сучасних інформаційних систем для ефективного здійснення управлінської діяльності відповідно до спеціальності 126 «Інформаційні системи та технології». В процесі вивчення дисципліни студент повинен також набути практичних навичок аналізу предметної області та моделювання бази даних системи засобами уніфікованої мови моделювання UML в інструментальній середовищі (MS Office Visio, StarUml або аналогічний Case-засіб).

Основними завданнями навчальної дисципліни є підготовка студентів з наступних питань:

Завданням навчальної дисципліни є

- моделі структур даних;
- способи зберігання даних на фізичному рівні, типів і способів організації файлових систем;
- реляційна модель даних і СУБД, яка реалізує цю модель, мова запитів SQL;
- можливості СУБД, що підтримують різні моделі організації даних, переваги і недоліки цих СУБД при реалізації різних структур даних, засобами цих СУБД;
- розуміння способів класифікації СУБД залежно від реалізованих моделей даних і способів їх використання;
- проблеми і основні способи їх вирішення при колективному доступі до даних;
- етапи життєвого циклу бази даних, підтримка та супровід;
- уявлення про спеціалізовані апаратні й програмні засоби, що орієнтовані на побудову баз даних великих обсягів зберігання.

Вивчення навчальної дисципліни «Бази даних та СУБД» передбачає формування у студентів компетентностей (*витікають із освітньо-професійної програми*):

№	Вид програмних компетентностей	Програмна компетентність	Шифр
Загальні			
1		Здатність до абстрактного мислення, аналізу та синтезу.	КЗ 1
3		Здатність до розуміння предметної області та професійної діяльності.	КЗ 3
4		Здатність вчитися і оволодівати сучасними знаннями.	КЗ 5
Спеціальні (фахові, предметні) компетентності			
7		Здатність аналізувати об'єкт проектування або функціонування та його предметну область.	КС 1
8		Здатність застосовувати стандарти в області інформаційних систем та технологій при розробці функціональних профілів, побудові та інтеграції систем, продуктів, сервісів і елементів інфраструктури організації.	КС 2
9		Здатність до проектування, розробки, налагодження та вдосконалення системного, комунікаційного та програмно-апаратного забезпечення інформаційних систем та технологій, комп'ютерно-інтегрованих систем та системної мережної структури, управління ними.	КС 3
10		Здатність проектувати, розробляти та використовувати засоби реалізації інформаційних систем, технологій та інфокомунікацій (методичні, інформаційні, алгоритмічні, технічні, програмні та інші).	КС 4
11		Здатність використовувати сучасні інформаційні системи та технології (виробничі, підтримки прийняття рішень, інтелектуального аналізу даних та інші), методики й техніки кібербезпеки під час виконання функціональних завдань та обов'язків.	КС 6
12		Здатність застосовувати інформаційні технології у ході створення, впровадження та експлуатації системи менеджменту якості та оцінювати витрати на її розроблення та забезпечення.	КС 7
13		Здатність управляти та користуватися сучасними інформаційно-комунікаційними системами та технологіями (у тому числі такими, що базуються на використанні Інтернет).	КС 15

У результаті вивчення навчальної дисципліни «Бази даних та СУБД» студент повинен бути здатним продемонструвати такі результати навчання:

№	Програмні результати навчання	Шифр
1	Використовувати базові знання інформатики й сучасних ІС та технологій, навички програмування, технології безпечної роботи в комп'ютерних мережах, методи створення БД та інтернет-ресурсів, технології розроблення алгоритмів і комп'ютерних програм мовами високого рівня із застосуванням об'єктно-орієнтованого програмування для розв'язання задач проектування і використання інформаційних систем та технологій.	ПР 3
2	Демонструвати знання сучасного рівня технологій інформаційних систем, практичні навички програмування та використання прикладних і спеціалізованих комп'ютерних систем та середовищ з метою їх запровадження у професійній діяльності.	ПР 6
3	Обґрунтовувати вибір технічної структури та розробляти відповідне програмне забезпечення, що входить до складу інформаційних систем та технологій.	ПР 7
4	Здійснювати системний аналіз архітектури підприємства та його ІТ-інфраструктури, проводити розроблення та вдосконалення її елементної бази і структури.	ПР 9
5	Розуміти і враховувати соціальні, екологічні, етичні, економічні аспекти, вимоги охорони праці, виробничої санітарії, пожежної безпеки та існуючих державних і закордонних стандартів під час формування технічних завдань та рішень.	ПР10
	<i>Додаткові програмні результати навчання, визначені за освітньою програмою:</i>	
6	Демонструвати навички самостійно приймати рішення, лідерські навички та вміння мотивувати людей.	ПРН 12
7	Проводити дослідження , генерувати нові ідеї, здійснювати інноваційну діяльність.	ПРН 15.
8	Аналізувати на основі створених моделей економічні об'єкти та процеси, інтерпретувати отримані результати і на підставі зроблених висновків, виробляти управлінські рішення на всіх рівнях господарської ієрархії управління.	ПРН 17
9	Застосовувати сучасні інформаційні технології у соціально-економічних дослідженнях.	ПРН 19.

В процесі виконання та захисту лабораторних, самостійних робіт, курсової роботи студенти повинні проявити знання теоретичних та методологічних основ проектування баз даних, демонструвати уміння працювати з літературними джерелами, здатність аналізувати, узагальнювати інформацію; приймати рішення на основі зроблених висновків; логічно і чітко викладати матеріал.

Потоки інформації, що циркулюють у світі, який нас оточує, величезні. До того ж сьогодні ці потоки мають тенденцію до збільшення. Тому в будь-якій установі, на будь-якому підприємстві виникає проблема такої організації керування даними, що забезпечила б найбільш ефективну роботу.

Успішна діяльність підприємства не можлива без повноти та ефективної організації інформаційного забезпечення. Своєчасна та достовірна інформація дозволяє добре орієнтуватися в ринковому середовищі, зменшувати фінансовий ризик, слідкувати за змінами в зовнішньому середовищі, оцінювати свою діяльність, розробляти і коригувати стратегію підприємства. Компетентність керівника залежить не стільки від минулого досвіду, скільки від володіння достатньою кількістю інформації про ситуацію, яка постійно змінюється та уміння скористатися цією інформацією.

Головним напрямком перебудови менеджменту і його радикального удосконалення, пристосування до сучасних умов стало масове використання новітньої комп'ютерної і телекомунікаційної техніки, формування на її основі високоефективних інформаційно-управлінських технологій. Засоби і методи прикладної інформатики використовуються в менеджменті й маркетингу. Нові технології, засновані на комп'ютерній техніці, вимагають радикальних змін організаційних структур менеджменту, його регламенту, кадрового потенціалу, системи документації, фіксування і передачі інформації.

Сучасний світ інформаційних технологій важко уявити собі без використання баз даних. Практично всі системи в тій чи іншій мірі пов'язані з функціями довготермінового збереження та обробки інформації. Фактично інформація становиться фактором, що визначає ефективність будь-якої сфери діяльності. Збільшились інформаційні потоки й підвищились вимоги щодо швидкості обробки даних, і тепер уже більшість операцій не може бути виконана вручну, вони вимагають застосування найбільш перспективних комп'ютерних технологій. Всякі адміністративні рішення вимагають чіткої і точної оцінки поточної ситуації та можливих перспектив її зміни. І якщо раніше оцінка ситуації досягала декількома десятками факторів, то тепер таких факторів сотні і сотні

тисяч, і ситуація змінюється не протягом року, а через декілька хвилин, а обґрунтованість рішень, що приймаються, вимагається велика, так як і реакція на неправильні рішення більш серйозна, набагато швидша та потужніша, ніж раніше. Тому, звичайно, обійтись без інформаційної моделі виробництва, що зберігається в базі даних, в такому випадку неможливо. Крім цього, необхідні засоби забезпечення діалогу людина - ЕОМ, що дозволяють користувачеві вводити запити, читати файли, модифікувати збережені дані, додавати нові дані або приймати рішення на підставі збережених даних.

Для забезпечення цих функцій створені спеціалізовані засоби – системи керування базами даних (СУБД). Сучасні СУБД - багатокористувацькі системи керування базою даних, що спеціалізується на керуванні масивом інформації одним або безліччю одночасно працюючих користувачів.

Сучасні СУБД забезпечують: набір засобів для підтримки таблиць і відносин між зв'язаними таблицями; - розвинутий інтерфейс користувача, що дозволяє вводити і модифікувати інформацію, виконувати пошук і представляти інформацію в графічному або текстовому режимі; - засоби програмування високого рівня, за допомогою яких можна створювати власні додатки.

Для реалізації бази даних цілком підходить СУБД MS Access, як одна з найпопулярніших при розв'язуванні задач подібного класу, завдяки високим швидкодії, рівню автоматизації програмування та спадкоємності версій.

РОЗДІЛ 1. Основні теоретичні положення проектування баз даних

Сучасний світ інформаційних технологій важко уявити без використання баз даних (БД). Практично всі системи в тій чи іншій мірі пов'язані з функціями довготермінового збереження та обробки інформації. Фактично інформація становиться фактором, що визначає ефективність будь-якої сфери діяльності. Збільшилися інформаційні потоки й підвищилися вимоги щодо швидкості обробки даних, і тепер уже більшість операцій не може бути виконана вручну, вони вимагають застосування найбільш перспективних комп'ютерних технологій. Всякі адміністративні рішення вимагають чіткої й точної оцінки поточної ситуації та можливих перспектив її зміни. І якщо раніше оцінка ситуації досягалась декількома десятками факторів, то тепер таких факторів сотні й сотні тисяч, і ситуація змінюється не протягом року, а через декілька хвилин, обґрунтованість рішень, що приймаються, вимагається велика, так як реакція на неправильні рішення більш серйозна, набагато швидша та потужніша, ніж раніше. Тому, звичайно, обійтись без інформаційної моделі виробництва, що зберігається в базі даних, в такому випадку неможливо. Крім цього, необхідні засоби забезпечення діалогу людина - ЕОМ, що дозволяють користувачеві вводити запити, читати файли, модифікувати збережені дані, додавати нові дані або приймати рішення на підставі збережених даних.

Для забезпечення цих функцій використовують спеціалізовані засоби – системи керування базами даних (СУБД). Сучасні СУБД - багатокористувацькі системи керування базою даних, що спеціалізується на керуванні масивом інформації одним або безліччю одночасно працюючих користувачів.

Сучасні СУБД забезпечують: набір засобів для підтримки таблиць і відносин між зв'язаними таблицями; - розвинутий інтерфейс користувача, що дозволяє вводити й модифікувати інформацію, виконувати пошук і представляти інформацію в графічному або текстовому режимі; - засоби програмування високого рівня, за допомогою яких можна створювати власні додатки.

Для реалізації бази даних цілком підходить СУБД MS Access, як одна з найпопулярніших при розв'язуванні задач подібного класу, завдяки високим швидкодії, рівню автоматизації програмування та спадкоємності версій.

1.9. Цілі проектування бази даних, етапи проектування

Проект реляційної бази даних – це набір взаємозв'язаних відношень, в яких визначені всі атрибути, задані первинні ключі відношень та задані ще деякі додаткові властивості відношень, які відносяться до принципів цілісності. Проект бази даних повинен бути точним і вивіреном. Фактично проект бази даних – це фундамент майбутнього програмного комплексу, який буде використовуватися досить довго та багатьма користувачами. Етапи життєвого циклу баз даних мають вигляд, зображений на рис.1.1:



Рисунок 1.1 - Етапи життєвого циклу баз даних

Процес проектування БД являє собою послідовність переходів від неформального словесного опису інформаційної структури предметної області до формалізованого опису об'єктів предметної області в термінах деякої моделі та фізичної реалізації бази даних в середовищі вибраної СУБД. В загальному випадку можна виділити такі етапи проектування:

1. Системний аналіз і словесний опис інформаційних об'єктів предметної області.
2. Проектування інфологічної моделі предметної області.

3. Даталогічне або логічне проектування БД, тобто опис БД в термінах прийнятої даталогічної моделі даних (реляційна, ієрархічна, сіткова).

4. Фізичне проектування БД, тобто вибір ефективного розміщення БД на зовнішніх носіях для забезпечення найбільш ефективної роботи додатку.

Між другим та третім етапом вирішують, з використання якої стандартної СУБД буде реалізовуватися проект. Тоді умовно процес проектування БД можна уявити послідовністю виконання п'яти відповідних етапів (рис.1.2). Кожен з етапів життєвого циклу бази даних (рис.1.1) складається з окремих процедур, або робіт, що реалізують цей етап. Етапи життєвого циклу БД та процедури, що виконуються на цих етапах, зображено на рис.3. На всіх етапах життєвого циклу БД складається з двох компонентів: структури та даних. На різних рівнях абстракції опису БД її структура зображується по-різному, як показано на рис. 1.4.

На інфологічному рівні структура БД ефективно відображається у вигляді ER-діаграми "Сутність — зв'язок"; на даталогічному рівні вона є або ієрархічною, або мережною, або реляційною, або об'єктно-орієнтованою, або об'єктно-реляційною, або багатовимірною, або гібридною моделлю даних. На фізичному рівні структура БД — це структура файлів даних і допоміжних файлів. Самі файли даних також складаються з двох компонентів: структури та даних. Структура файла в реляційній моделі— це ім'я, тип поля, його довжина, точність (для числових полів).

Структура ідеальної БД має бути чіткою, ясною та прозорою. БД має бути вільною від баластної інформації: дублювання даних, синонімів, омонімів, груп даних, що повторюються, обчислювальних елементів.

БД повинна містити мінімальну кількість елементів даних, які несуть максимум інформації про стан предметної області.

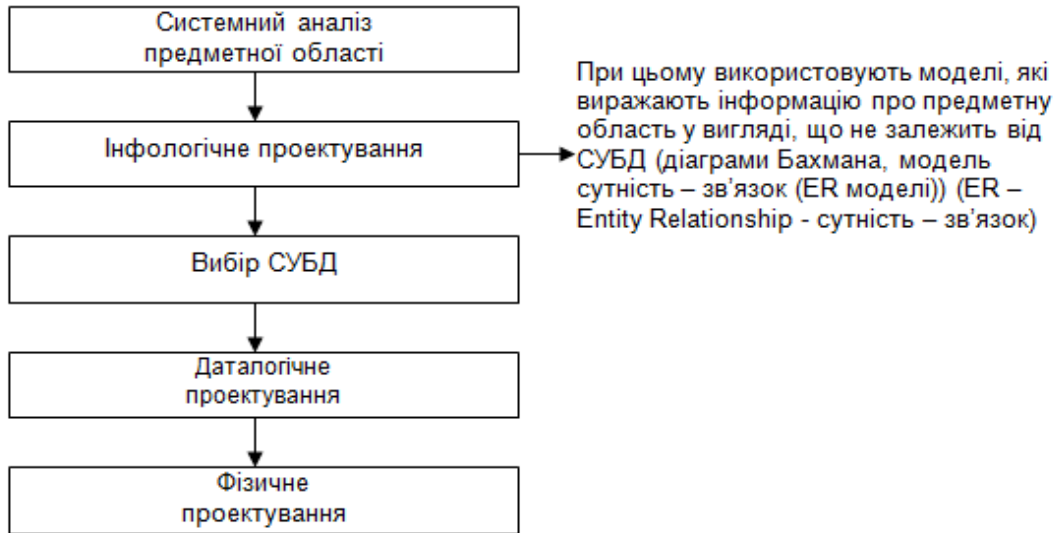


Рисунок 1.2 - Послідовність етапів проектування БД

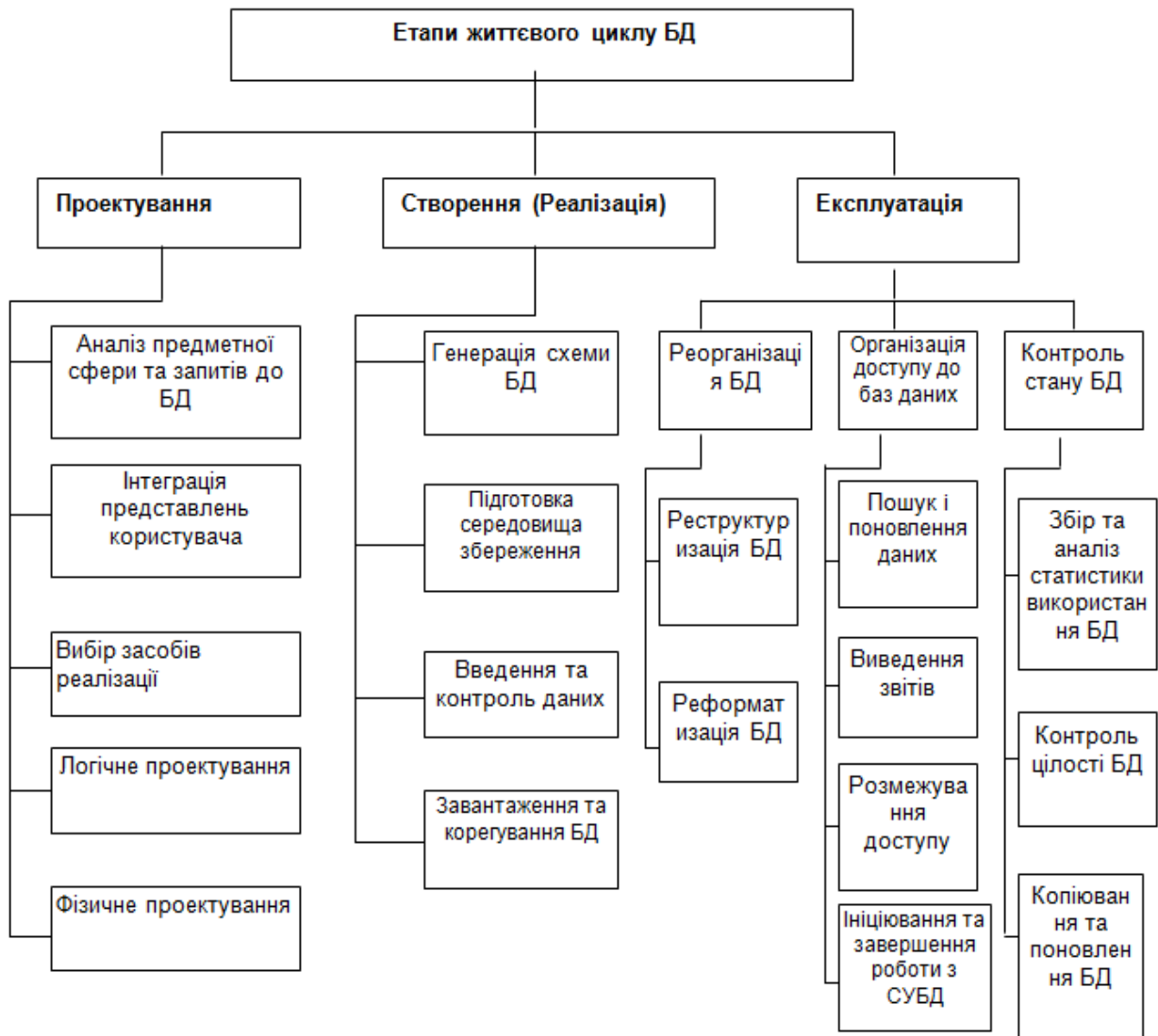


Рисунок 1.3 - Етапи та процедури життєвого циклу бази даних

Елементи БД мають бути організовані оптимальним способом, тобто таким чином, щоб забезпечити сталість БД у розумінні семантичної цілісності опису стану предметної області при маніпулюванні даними або несуперечливості даних, прийнятної швидкості пошуку елементів даних, надійності зберігання, узгодженості, безпеки, відкритості структури для поповнення та модифікації.

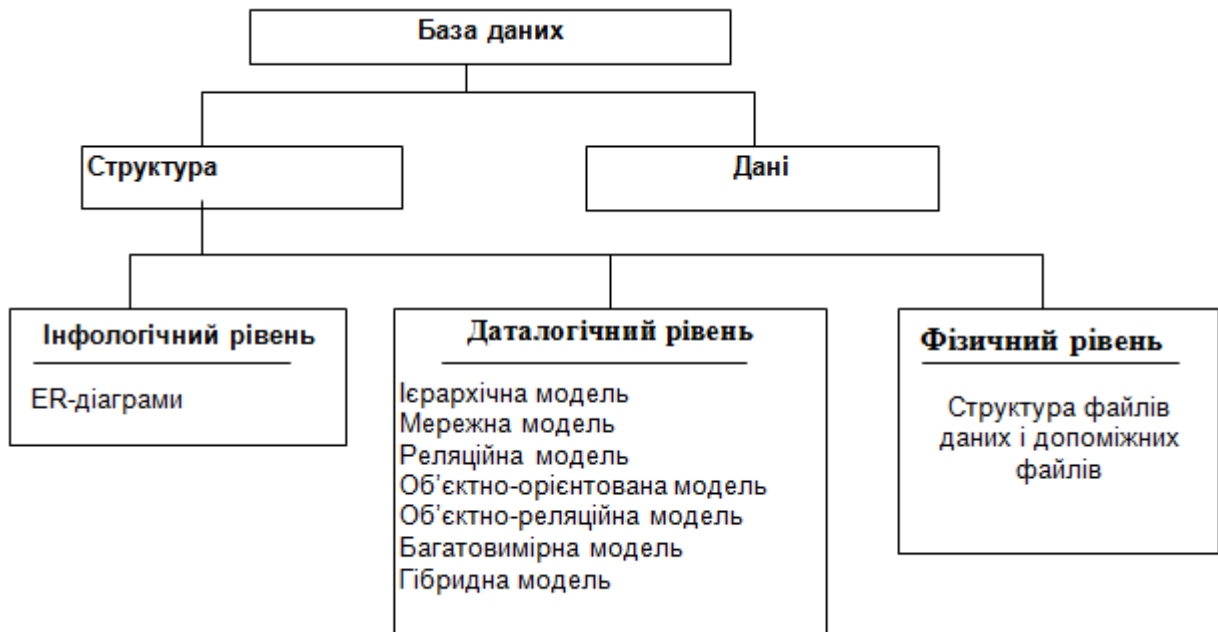


Рисунок 1.4 - Складові бази даних на етапах життєвого циклу

Розробленням структури БД на всіх етапах займається системний аналітик, який забезпечує БД усіма необхідними властивостями.

Дані - це компонент БД, яким маніпулює користувач інформаційної системи, актуалізуючи БД.

Дані сучасних БД поділяються на:

- алфавітно-цифрові (використовуються практично в усіх інформаційних системах);
- графічні образи (застосовуються в системах автоматизованого проектування);
- гіпертекстові, гіпермедійні, мультимедійні (використовуються в інформаційних системах на основі технологій Internet/Intranet/Extranet);
- просторові або картографічні (застосовуються в гео-інформаційних

системах);

— ряди даних (як правило, хронологічні ряди, що використовуються у багатовимірних БД, які становлять основу сховищ даних).

Дамо коротку характеристику основних етапів проектування бази даних.

Системний аналіз предметної області. На цьому етапі необхідно провести детальний опис (словесний) об'єктів предметної області та реальних зв'язків, які присутні між об'єктами. Існує два підходи до вибору складу та структури предметної області - функціональний підхід та предметний підхід.

Функціональний підхід реалізує принцип руху “від задач” і застосовується тоді, коли заздалегідь відомі функції деякої групи осіб і комплексів задач, для обслуговування інформаційних потреб яких створюється БД. В цьому випадку можна чітко виділити мінімально необхідний набір об'єктів предметної області, які повинні бути описані (товари, постачальники, банківські реквізити і т.д.).

Предметний підхід застосовується тоді, коли інформаційні потреби користувачів повністю не фіксовані. До опису предметної сфери при цьому включаються такі об'єкти та взаємозв'язки, які найбільш характерні та суттєві для неї. В такому випадку БД називається предметною, тобто вона може бути використана для розв'язання різноманітних, заздалегідь не визначених задач. Предметний підхід може призвести до збитково складної схеми БД, яка для конкретних задач буде неефективною.

Системний аналіз повинен закінчуватись детальним описом інформації про об'єкти предметної сфери, яка необхідна для вирішення конкретних задач і яка повинна зберігатись в БД. Також формулюванням конкретних задач, які будуть вирішуватись з використанням даної БД, з конкретним описом алгоритмів їх рішення, описом вихідних документів, які повинні генеруватись в системі, описом вхідних документів, які є основою для заповнення даними БД.

Інфологічне моделювання. Інфологічна модель застосовується на другому етапі проектування БД, тобто після словесного описування предметної сфери. Навіщо потрібна інфологічна модель? Проектування БД – це складний процес.

Він вимагає обговорень із замовником, зі спеціалістами в предметній сфері. При розробці складних корпоративних інформаційних систем проект бази даних є тим фундаментом, на якому будується вся система в цілому, і питання про можливе кредитування часто вирішується експертами банку на основі грамотно складеного інфологічного проекту БД. Тобто, інфологічна модель повинна включати такий формалізований опис предметної сфери, який буде читатись не тільки спеціалістами з баз даних. Цей опис не повинен бути прив'язаним до конкретної СУБД. Інфологічне проектування, перш за все, пов'язане зі спробою представлення семантики предметної області в моделях БД.

Проблема представлення семантики давно цікавила розроблювачів, і в 70-х роках було запропоновано декілька моделей даних, названих семантичними моделями. Найбільш відома, випробувана часом, модель “сутність – зв’язок”, запропонована П.Ченом в 1976 р., (інша назва – Entity Relationship - , або **ER-модель**). Сучасні CASE-технології містять засоби автоматичного перетворення проекту БД із ER-моделі в реляційну. При цьому перетворення виконується в даталогічну модель, що відповідає конкретній СУБД. Всі CASE-системи мають розвинуті засоби документування процесу розробки БД, автоматичні генератори звітів дозволяють підготувати звіти про поточний стан проекту БД з детальним описом об’єктів БД та їх відношень як в графічному вигляді, так і готових стандартних друкованих звітів.

В основі ER-моделі лежать такі базові поняття як **сутність** та **зв’язки**.

Сутність, за допомогою якої моделюється клас однотипних об’єктів. Сутність має ім’я унікальне в межах системи, що моделюється. В системі існує множина екземплярів даної сутності. Об’єкт, якому відповідає поняття сутності, має свій набір атрибутів, - характеристик, що визначають властивості даного представника класу. Загальноприйняте графічне позначення сутності – прямокутник, у верхній частині якого записане ім’я сутності, а нижче вказані атрибути (рис.1.5):

Співробітник
Таб. номер
Прізвище
Ім'я
По-батькові
Дата народження

Рисунок 1.5 - Позначення сутності

Набір атрибутів повинен бути таким, щоб можна було розрізнити конкретні екземпляри сутностей. Набір атрибутів, що однозначно ідентифікує екземпляр сутності є ключовим.

Між сутностями можуть бути встановлені **зв'язки**, що показують, яким чином сутності співвідносяться, або взаємодіють між собою. Зв'язок може існувати між двома різними сутностями, або між сутністю і нею ж самою (рекурсивний зв'язок). Цей (останній) зв'язок показує, як зв'язані між собою екземпляри сутності.

Зв'язок між сутностями визначає взаємозв'язок між екземплярами одної та іншої сутності. Наприклад, якщо ми маємо зв'язок між сутністю “студент” та сутністю “викладач” і цей зв'язок – керівництво дипломними проектами, то кожний студент має тільки одного керівника, але один і той же викладач може керувати декількома студентами – дипломниками (рис.1.6). Це буде зв'язок “один до багатьох” (1:Б).

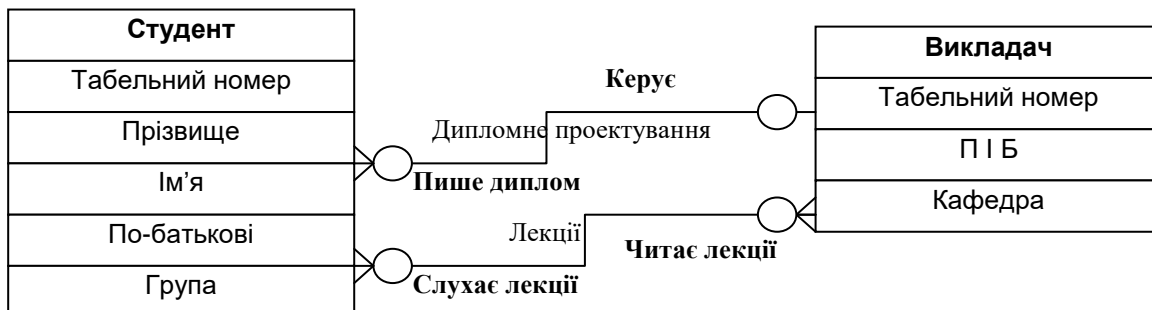


Рисунок 1.6 - Приклад зв'язків між сутностями

Зв'язок має ім'я “Дипломне проектування”, а також має імена ролей з боку обох сутностей:

- Керує;
- Пише диплом.

Між сутностями може бути задано декілька зв'язків з різним змістовним навантаженням. Приклад: ще один зв'язок: **лекції**. Цей зв'язок має дві ролі: - **читає лекції**; - **слухає лекції**. Зв'язок має тип “багато-до-багатьох” (Б:Б).

Тобто, інфологічна модель предметної області містить сутності і відображає зв'язки між сутностями.

Даталогічне проектування. Даталогічне або логічне проектування – це розробка схем БД, (для реляційної моделі це сукупність схем відношень), які моделюють об'єкти предметної сфери і зв'язки між цими об'єктами в термінах вибраної СУБД. На даталогічному рівні структура бази даних є або ієрархічною, або мережною, або реляційною, або об'єктно-орієнтованою, або об'єктно-реляційною, або багатовимірною, або гібридною моделлю даних. В результаті виконання цього етапу повинні бути отримані такі результатні документи:

- Опис концептуальної схеми БД в термінах вибраної СУБД;
- Опис зовнішніх моделей в термінах вибраної СУБД;
- Опис правил підтримки цілості БД;
- Розробка процедур підтримки цілості бази даних.

Проектування БД на цьому етапі пов'язане з теорією нормалізації, що оснований на аналізі функціональних залежностей між атрибутами відношень.

Процес проектування БД залежить від досвіду та інтуїції розроблювача, тобто є творчим, але деякі його моменти можна формалізувати. Одною із таких формалізацій є вимога, відповідно до якої реляційна база даних повинна бути нормалізована. Процес нормалізації має за мету усунення збитковості даних і полягає в приведенні до третьої нормальної форми (пункт 1.).

1.2. Формулювання та аналіз вимог до бази даних

При розробці автоматизованої інформаційної системи використовується досвід оброблення інформації вручну або за допомогою комплексу інструментальних засобів попереднього покоління. Тому надзвичайно важливим і відповідальним етапом проектування як ІС, так і БД, є передпроектна стадія, на

якій проводяться співбесіди з виконавцями робіт, що підлягають автоматизації. Власне вони в перспективі й будуть користувачами ІС та БД. У процесі проектування БД (як і при проектуванні ІС) дотримуються принципів *системної єдності, розвитку, сумісності, стандартизації*. Сфера застосування БД має визначатися незалежно від будь-якої прикладної задачі в межах певної ПС, а при зміні меж ПС повинна відбивати інформаційні потреби всіх підрозділів підприємства чи установи. Іншими словами, БД має бути універсальною, придатною для розв'язання багатьох прикладних задач і відкритою для подальшого нарощування її структури. Для цього вона має бути логічно та фізично незалежною.

Вимоги до БД, що розроблюється, або властивості, які вона повинна мати, на сьогодні є класичними. До них належать:

- **функціональна повнота** — властивість БД, що забезпечує повноту відповідей на інформаційні запити користувачів ІС;

- **мінімальна надмірність** — властивість БД, що дає змогу мати в ній понад необхідний мінімум даних тільки ті дані, які використовуються для зв'язування запитів, що описують різноманітні об'єкти ПС;

- **цілісність БД** — властивість БД, що дає змогу зберігати певні обмеження значень даних при всіх модифікаціях БД. У БД розрізняють такі види цілісності: цілісність домену; цілісність таблиці; цілісність посилання; цілісність, обумовлену правилами бізнесу;

- **узгодженість БД** — властивість, що дає змогу забезпечити видачу однакової відповіді на один і той самий запит усім користувачам ІС;

- **відновлюваність БД** — властивість, що дає змогу забезпечити відновлення даних після будь-якої нештатної ситуації в системі;

- **безпе́чність БД** — властивість, що дає змогу забезпечити захист даних від навмисного чи випадкового доступу до даних, їх модифікації або руйнування;

- **логічна і фізична незалежність** — властивість, що забезпечує можливість зміни загальної логічної структури даних без зміни подання

прикладних програм про дані та зміни фізичної структури даних без зміни їхньої логічної структури;

— *розширюваність (відкритість) БД* — властивість, що забезпечує зміну логічної та фізичної моделей даних при зміні меж ПС тільки завдяки доданню нових структурних компонентів БД. Забезпечується оптимальною організацією даних на логічному та фізичному рівнях;

- *дружність інтерфейсу користувача* — властивість БД, що забезпечує користувачеві комфортний доступ до даних в інтерактивному режимі.

Головним засобом забезпечення таких вимог до БД, як мінімальна надмірність, цілісність, несуперечливість, логічна та фізична незалежність, є **нормалізація логічного подання даних**.

Відповідність властивостей БД та етапів її проектування наведено в таблиці 1.

Таблиця 1 - Відповідність властивостей БД та етапів її проектування

№ п/п	Етап проектування БД	Властивості БД
1	Формулювання та аналіз вимог до БД	Функціональна повнота, мінімальна надмірність, несуперечливість, ефективність, розширюваність
2	Інфологічне проектування (Проектування СУБД-незалежної моделі ПС)	Функціональна повнота, мінімальна надмірність, несуперечливість, безпечність, ефективність, розширюваність
3	Проектування реалізації (Проектування даталогічної моделі або СУБД-орієнтованої моделі ПС)	Функціональна повнота, мінімальна надмірність, несуперечливість, безпечність, логічна та фізична незалежність, ефективність, розширюваність, <i>дружність інтерфейсу користувача</i>
4	Фізичне проектування	Несуперечливість, цілісність, узгодженість, відновлюваність, безпечність, логічна та фізична незалежність, ефективність, розширюваність, <i>дружність інтерфейсу користувача</i>

1.3. Передпроектний аналіз проблемної області

При проектуванні підсистеми інформаційної системи цей аналіз зводиться до обстеження підприємства, під час якого складають функціональну схему підприємства і визначають місце розроблюваної підсистеми в інформаційній системі всього підприємства.

Далі збирають й аналізують вхідні та вихідні документи. Аналіз вихідних документів дає можливість установити основні функції підсистеми і джерела

формування реквізитів цих документів. Відповідно визначають перелік реквізитів, джерела надходження, способи та шляхи одержання вхідних документів.

На стадії проектування БД складають за певною формою інформаційний список вхідних і вихідних документів, які фігурують в межах відокремленої підсистеми (таблиця 2), розробляють схему документообігу між її підрозділами у вигляді схеми технологічного процесу або алгоритму оброблення інформації чи у вигляді схеми бізнес-процесів.

Таблиця 2 - Інформаційний список документів

№ п/п	Назва документа	Характеристика документа (вхідний/вихідний)
-------	-----------------	--

Якщо оброблення інформації не було автоматизованим і логіка цього процесу або предметна технологія не повинна змінюватися, тобто не передбачається докорінна зміна бізнес-процесів на підприємстві, то схема оброблення інформації відображає наявну предметну технологію.

За докорінної реконструкції бізнес-процесів або розроблення оригінального проекту уточнюють форми документів, їхній реквізитний склад, а також склад кінцевих користувачів підсистем, документообіг, організаційні та правові моменти оброблення інформації. У цьому разі схема оброблення інформації відповідає новій предметній технології або новому бізнес-процесу.

Вхідні та вихідні документи аналізують на наявність реквізитів, що перетинаються. З цією метою складають родо-видові списки елементів даних окремо для вихідних і вхідних документів за певною формою (таблиця 3).

Таблиця 3 - Родо-видовий список реквізитів вихідних (вхідних) документів

№ п/п.	Назва реквізиту	Характеристика реквізиту (фактичний/ обчислюється)	Призначення реквізиту
--------	-----------------	---	-----------------------

Зі списків вилучають елементи, що дублюються, синоніми, омоніми, елементи, які обчислюються. Родо-видовий список включає згруповані за видом

реквізити документів. Наприклад, перелічуються разом усі дати, потім усі коди та ін. Це спрощує процедуру вилучення елементів, що дублюються, синонімів, омонімів, елементів, які обчислюються.

У межах конкретної задачі часто, згідно з умовами оброблення інформації, постає потреба тривалого зберігання даних, що обчислюються, оскільки процедура повторних обчислень виявляється менш ефективною, ніж зберігання таких даних. Для цього, як правило, створюють допоміжний файл. Оскільки такий файл актуальний лише в межах конкретної задачі, а всі наявні в ньому дані можна відтворити, його не вважають файлом БД. Такі елементи даних слід унести в словник даних прикладних програм. Проте бувають випадки, коли доцільніше виділити кілька полів файла БД для збереження обчислюваних даних, ніж створювати для цього окремий файл. У цьому разі дані, що обчислюються, не вилучають із родо-видових списків.

Після цього родо-видові списки вхідних і вихідних документів порівнюють з метою вилучення з розгляду елементів даних, які не є актуальними для підсистеми. У словник даних, складений за формою (таблиця 4), необхідно внести реквізити, спільні для обох списків, з урахуванням тих реквізитів вхідних документів, які використовуються для формування реквізитів вихідних документів.

Таблиця 4 - Словник даних

№ п/п	Назва елемента даних	Ідентифікатор	Тип і довжина елемента даних	Призначення елемента даних
-------	----------------------	---------------	------------------------------	----------------------------

За результатами попередніх етапів проектування БД відокремлюють локальні задачі виконання окремих функцій у розроблюваній підсистемі. З цією метою схему оброблення інформації розбивають на локальні задачі. Головна умова при відокремленні задач полягає в тому, щоб у межах однієї задачі оброблявся один набір даних з однією метою.

Після цього складають таблицю зв'язків "Задача—дані" за формою (таблиця 5), яку потім використовують при побудові локальних ER-діаграм.

Таблиця 5 - Таблиця зв'язків "Задача—дані"

№ п/п	Назва задачі	Частота виконання задачі	Обсяг даних, що обробляються в задачі	Відділ, де виконується оброблення даних	Елементи даних (номери елементів із словника)
-------	--------------	--------------------------	---------------------------------------	---	---

Після цього розробляють ескізи вхідних і вихідних документів. У такий спосіб забезпечується доповнення основних функцій розробленої підсистеми.

1.4 Концептуальне інфологічне проектування бази даних

Концептуальна модель (схема БД) є формальним поданням ПС на понятійному рівні, тобто загальною логічною структурою БД. Завдання концептуального інфологічного проектування полягає в одержанні логічної моделі БД у термінах об'єктів ПС та зв'язків між ними, що не залежить від конкретної СУБД й узагальнює інформаційні вимоги потенційних користувачів ІС.

Розрізняють два основних методи концептуального інфологічного проектування: низхідне проектування (метод формулювання та аналізу сутностей) і висхідне проектування (метод синтезу атрибутів). Ці методи недостатньо формалізовані, єдиних правил використання їх не існує.

Найпридатнішим для практичного застосування є перший метод. Він складається з двох етапів проектування БД: ідентифікації та моделювання локальних інформаційних структур БД у вигляді локальних ER-діаграм і побудови глобальної інформаційної моделі — глобальної ER-діаграми.

Локальні інформаційні структури відповідають локальним задачам, процес відокремлення яких описано вище. У процесі проектування **ER-діаграми** для локальної задачі доцільно керуватися кількома правилами.

Правило 1. Визначити кінцевих користувачів підсистеми ІС, які будуть користуватися БД. На підставі анкетування та регулярних співбесід сформулювати запити до БД і тим самим визначити характер зв'язків між об'єктами ПС.

Правило 2. Відокремлюючи типи сутностей з множини даних, що є в словнику даних і належать до конкретної задачі (ці дані фігурують у таблиці "Задача—дані"), слід зважити на те, що тип сутності — це набір об'єктів ПС, однотипних у тому розумінні, що вони описуються однаковими атрибутами. У постановці задачі такі об'єкти ПС фігурують у множині. Множина таких об'єктів — скінченна. Графічно тип сутності в нотації П. Чена зображується у вигляді поймаючого прямокутника, назви атрибутів відокремлюються від назви сутності прямою лінією (пункт 5, Рис.1.6). Найменування заноситься у називному відмінку однини.

Правило 3. Кожен тип сутності повинен мати певний ідентифікатор: первинний ключ (один чи кілька атрибутів, що однозначно ідентифікують конкретний об'єкт ПС), або первинний атрибут чи атрибути — складові первинного ключа та атрибути опису. Первинний ключ має бути унікальним для всієї БД і коротким (якщо його вибирають із можливих ключів). За відсутності такого ключа його розробляють і потім вводять у словник даних. У разі складної схеми, коли не вдається відразу встановити первинний ключ об'єкта зберігання, виділяють первинний атрибут (або атрибути), а первинний ключ об'єкта уточнюють на етапі даталогічного проектування БД. Графічно атрибути типів сутностей зображують також в овалах, які зв'язуються з прямокутниками неспрямованими зв'язками. Ключ або первинні атрибути на діаграмі підкреслюють.

Наприклад, тип сутності "Виріб", як показано на рис.1.7, характеризується набором таких атрибутів: "Назва", "Параметр 1", "Параметр 2", "Параметр 3", "Параметр 4", "Кількість", "Ціна". Якщо ці атрибути, крім атрибутів "Кількість" та "Ціна", незалежні, то їх сукупність єдиним способом визначає конкретний екземпляр сутності "Виріб" і тому є можливим складним ключем. Для розв'язання задачі обліку виробів такий ключ незручний, через що його замінюють (навіть при ручному обробленні інформації) коротким еквівалентом "**Код виробу**".

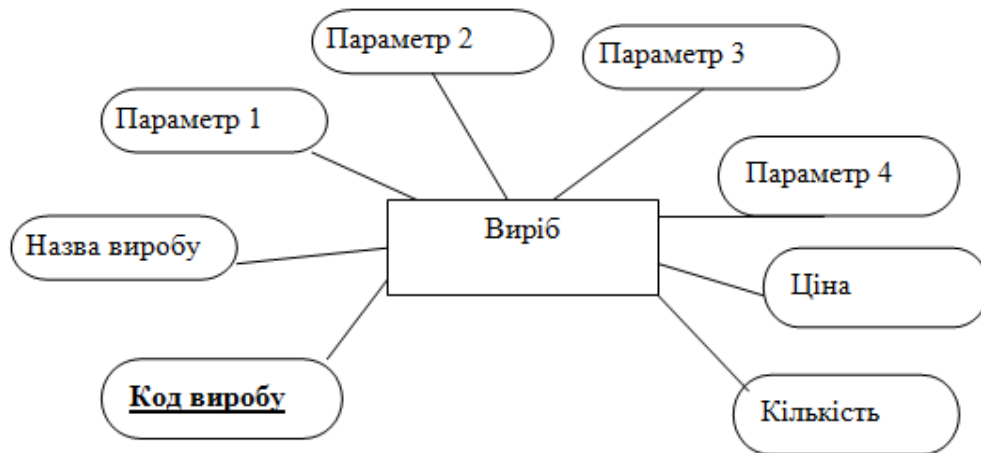


Рисунок 1.7 - Відображення типу сутності "Виріб" на ER-діаграмі

Правило 4. Зв'язок між типами сутностей відбиває фактичну або можливу взаємодію між ними, а також динаміку взаємодії між екземплярами сутностей. Графічно зв'язок зображують у вигляді пойменованого ромба з обов'язковим позначенням типу зв'язку (1:1, 1:Б, Б:Б). Найменування зв'язку має відображати його зміст і бути коротким. Наприклад, при накопиченні у БД інформації про сукупність складів і сукупність виробів, що зберігаються, зв'язки між вищезазначеними сукупностями даних визначають різні постановки задачі зберігання та оброблення даних.

Зв'язок типу 1:1. Він передбачає, що на кожному складі можуть зберігатися вироби одного типу (ідентифікуються власним кодом) у певній кількості. Кожен із виробів може зберігатися лише на одному складі, тобто склад визначає виріб, який на ньому зберігається, і навпаки, виріб характеризує склад, на якому зберігаються вироби одного типу з однаковими параметрами. ER-діаграму цього фрагмента БД зображено на рис.1.8

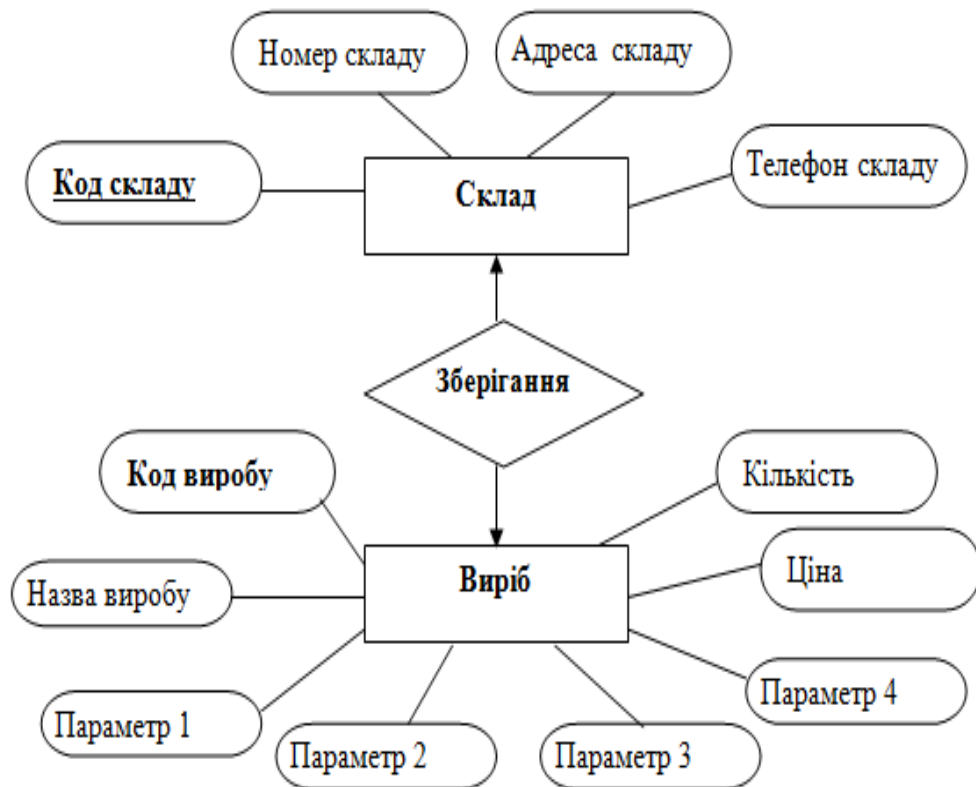


Рисунок 1.8 - ER-діаграма БД "Склад—Виріб" з типом зв'язку 1:1

Зв'язок типу 1:Б. Він означає, що на кожному складі, зберігається своя підмножина виробів або може зберігатися багато різних виробів, але вироби кожного типу зберігаються лише на одному складі. В цьому разі склад визначає тип виробу. Наприклад, склад процесорів, склад модулів пам'яті, склад устаткування для мереж тощо. При цьому всі вироби можуть мати різні параметри. ER-діаграму цього фрагмента БД показано на рис.1.9

Зв'язок типу Б:Б. Він свідчить, що на кожному складі може зберігатися багато різних виробів, причому кожен виріб може зберігатися на багатьох складах.

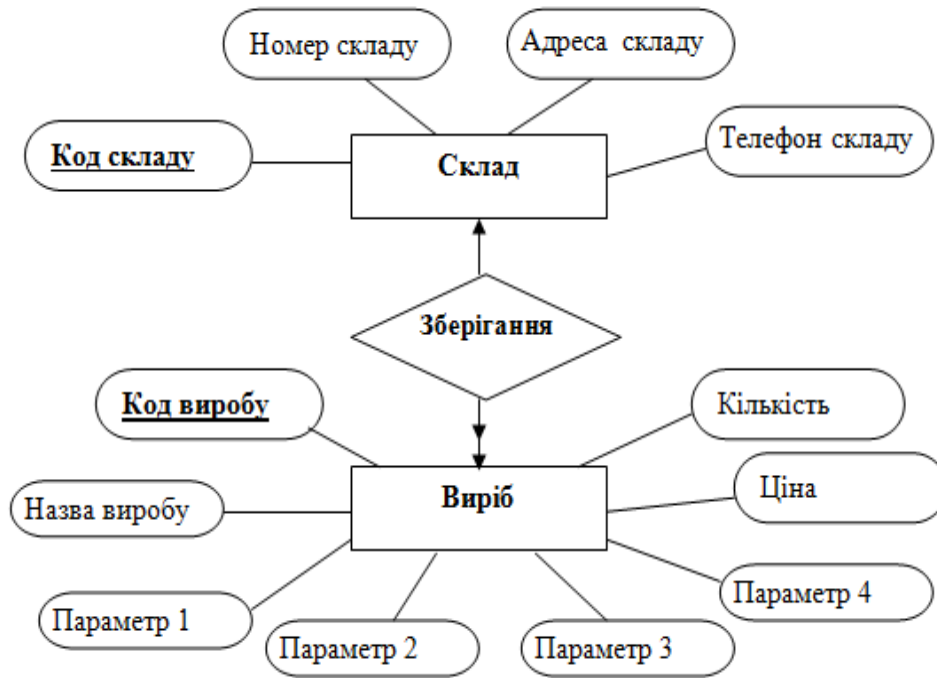


Рисунок 1.9 - ER-діаграма БД "Склад—Виріб" з типом зв'язку 1:Б

Наприклад, склади комерційних організацій зберігають різноманітні вироби, які можуть бути розміщені на багатьох складах в різних кількостях. Крім того, ціна одного й того самого виробу може бути різною (залежати, скажімо, від відстані від складу до місця доставки). ER-діаграму цього фрагмента БД зображено на рис. 1.10.

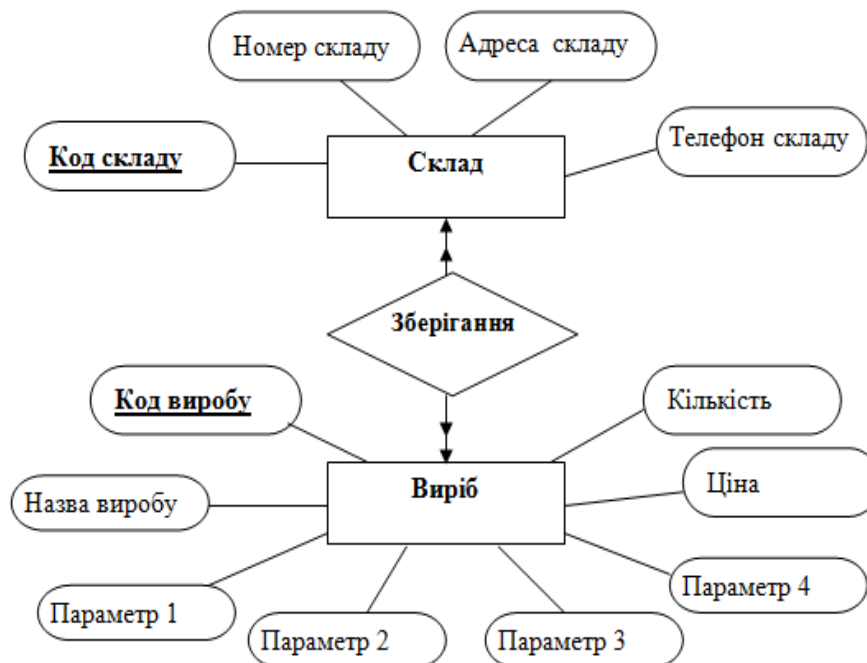


Рисунок 1.10 - ER-діаграма БД "Склад—Виріб" з типом зв'язку Б:Б

Правило 5. Тільки при зв'язку типу Б:Б можуть бути дані перетину, тобто дані, що одночасно належать з'єднуваним типам сутностей. Такі дані є атрибутами зв'язку.

У зв'язку типу Б:Б (див. рис.1.10) даними перетину є атрибути зв'язку "Кількість" та "Ціна".

Правило 6. Розрізняють унікальні сутності, що не залежать від жодних сутностей в межах ПС конкретної задачі, та залежні (породжені) сутності. Це важливо враховувати при встановленні зв'язку між типами сутностей.

У зв'язку типу 1:1 (див. рис.1.8) сутності "Склад" та "Виріб" залежать одна від одної. Яку з них вважати породжувальною, а яку породженою, можна зробити висновок тільки після уточнення семантики задачі. Точкою входу в таку модель даних може бути будь-яка з сутностей. У даному разі породжувальною можна вважати сутність "Склад", а породженою — "Виріб".

У зв'язку типу 1:Б (див. рис.1.9) сутність "Склад" є породжувальною, а сутність "Виріб" — породженою.

У зв'язку типу Б:Б сутності "Склад" та "Виріб" є незалежними (автономними). Зв'язок між ними встановлюється тоді, коли конкретний екземпляр виробу потрапляє на конкретний склад.

Правило 7. За наявності відсутних труднощів щодо відокремлення типів сутностей з множини даних таблиць 3 та 4 використовують метод синтезу атрибутів. Для цього доцільно скласти таблицю бінарних зв'язків між елементами даних за формою таблиці 6, указавши в ній тип зв'язку між ними.

Таблиця 6

Таблиця 6 - Таблиця бінарних зв'язків між елементами даних

Елемент 1	Елемент 2	Тип зв'язку

Якщо типом зв'язку між елементами даних є 1:1, то ці елементи скоріше за все належать до однієї сутності, а якщо типом зв'язку є 1:Б чи Б:Б, то елементи

належать до різних типів сутностей. Елементи, які неможливо віднести до жодної з сутностей, є даними перетину.

Правило 8. Окреме місце у схемах БД займають однорідні моделі, в яких усі типи сутностей мають однакову структуру, тобто описуються однаковим набором атрибутів.

Бази даних часто мають фрагменти з однорідними структурами і певними зв'язками між екземплярами об'єктів ПС. Залежність екземплярів об'єктів може бути різною (типу 1:Б або типу Б:Б). Наприклад, залежність виробів один від одного, коли в кожного більш складного виробу є своя підмножина виробів, які входять до його складу, ілюструє рис.1.11.

У цьому випадку існує рекурсивна залежність екземплярів об'єктів між собою, тобто залежність типу сутності самої від себе, що відображається як петля зв'язку. Така петля є тільки в схемі даних, у зв'язках між конкретними екземплярами об'єктів схеми її немає.

Залежність Б:Б (рис.12) інтерпретується так. Кожний з великих виробів складається з певної кількості малих, причому кожний з малих виробів може надходити до складу будь-якого з великих.

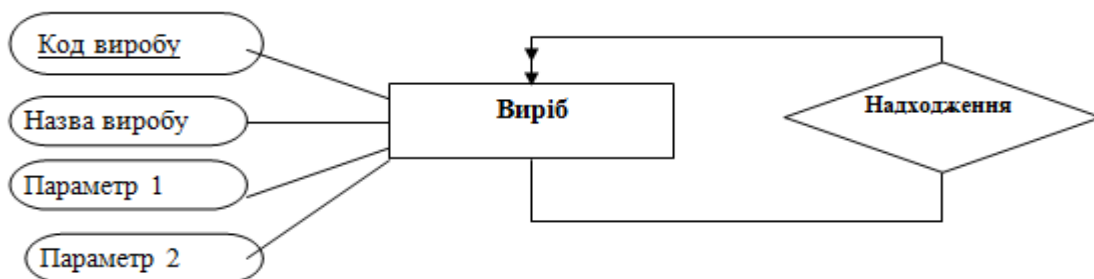


Рисунок 1.11 - ER-діаграма фрагмента БД з однорідною структурою та зв'язком 1:Б

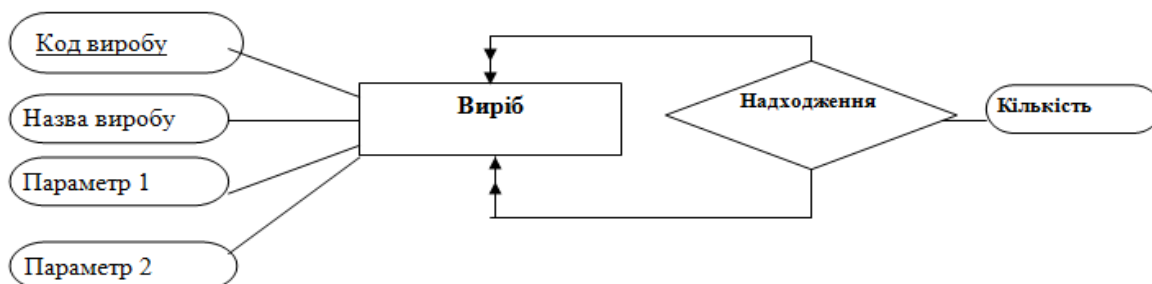


Рисунок 1.12 - ER-діаграма БД з однорідною структурою та зв'язком Б:Б

1.5. Проектування глобальної інфологічної моделі проблемної області

Проектування глобальної інфологічної моделі даних полягає в інтеграції локальних інформаційних структур, здобутих на попередньому етапі (рис.1.13).

При об'єднанні локальних інформаційних структур у глобальну використовують поняття: *ідентичність*, *агрегація*, *узагальнення*. Усі вони однаковою мірою належать до типів сутностей або об'єктів ПС та їхніх атрибутів, зв'язків між об'єктами ПС та їхніх атрибутів.

Ідентичність — однакове семантичне значення двох або більше об'єктів моделі.

Наприклад, об'єкти "Фірма-виробник" і "Фірма-постачальник" у межах певної ПС можуть належати до однієї категорії, тобто бути ідентичними.



Рисунок 1.13 - Об'єднання локальних інформаційних структур у глобальну

Наступний приклад, пов'язаний із фрагментом БД ІС обліку випуску на ринок продукції фірм-виробників, ілюструє рис.1.14. Зв'язки "Пропозиція на продаж" і "Випуск на ринок" та їхні атрибути ідентичні й мають бути об'єднані в один зв'язок з новими атрибутами наприклад, "Випуск на ринок", як це показано на рис.1.15. Це приклад того, що при складанні родо-видових списків реквізитів документів, а також словника даних були виявлені не всі синоніми.

Графічне відображення даних є більш наочним і дає змогу під іншим кутом зору виконати аналіз семантики предметної області.

Це приклад того, що при складанні родо-видових списків реквізитів документів, а також словника даних були виявлені не всі синоніми.

Графічне відображення даних є більш наочним і дає змогу під іншим кутом зору виконати аналіз семантики предметної області.

Після корекції ER-діаграми необхідно повернутися до словника даних й уточнити склад його елементів.

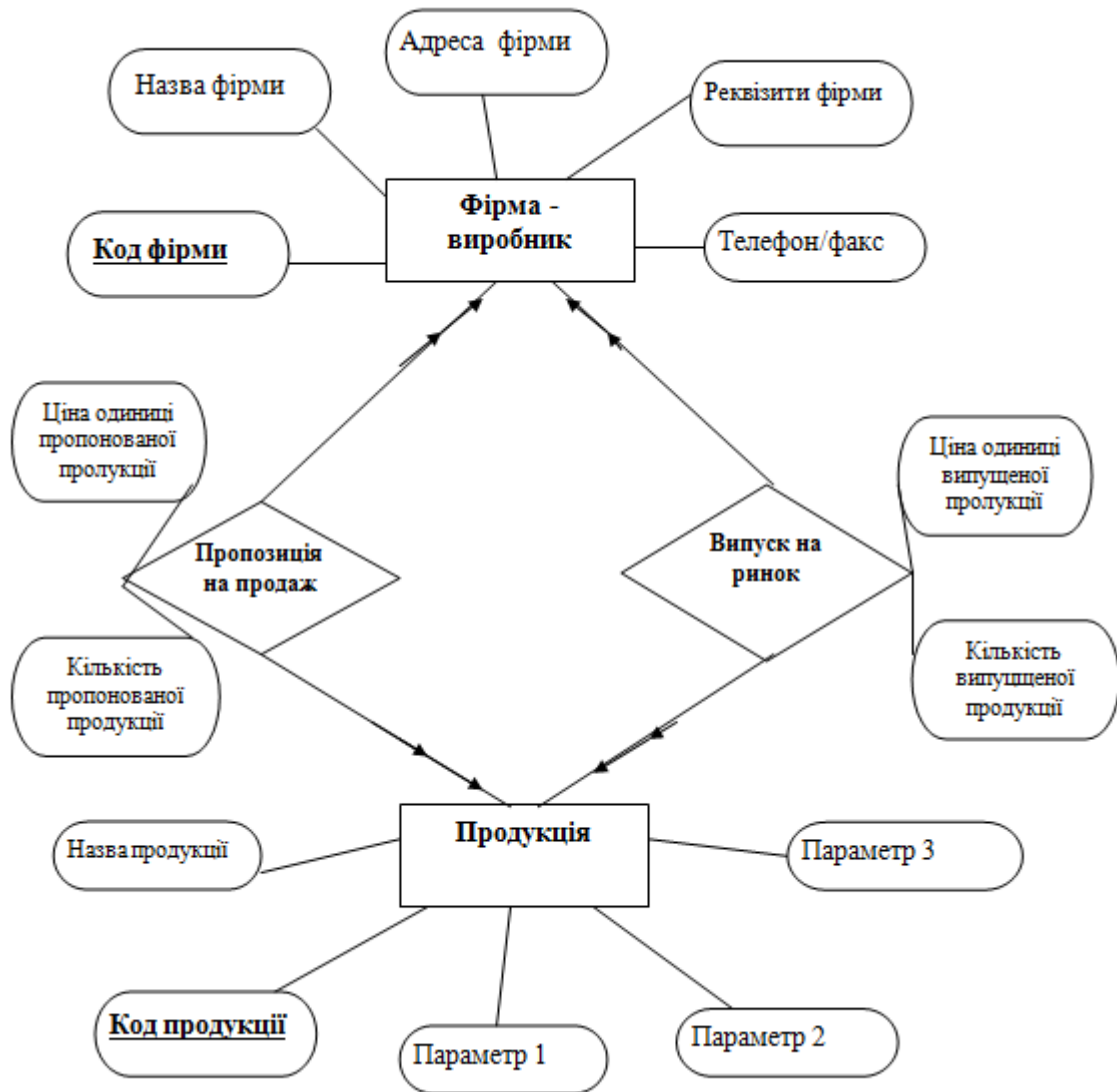


Рисунок 1.14 - Перший варіант ER-діаграми фрагмента БД ІС обліку випуску продукції на ринок

Агрегація — абстракція даних, що дає змогу трактувати сукупність різноманітних за природою об'єктів як новий об'єкт. Наприклад, сукупність об'єктів "Студент", "Дисципліна", "Викладач", "Оцінка" в межах певної предметної області може бути подана у вигляді агрегованого об'єкта "Екзамен" з атрибутами "Студент", "Дисципліна", "Викладач", "Оцінка".

Узагальнення — абстракція даних, що дає змогу трактувати клас різних подібних типів об'єктів як один поименований узагальнений тип об'єкта.

Наприклад, при організації БД предметної області міської товарно-сировинної біржі дані про брокерів міських брокерських контор, які укладають угоди купівлі-продажу певного товару, недоцільно зберігати в різних масивах і відповідно зображати у вигляді окремих типів сутностей, оскільки кожний брокер у день торгів може бути і продавцем, і покупцем. Цей варіант моделі, показаний на Рис.16, характеризується необґрунтованим дублюванням даних. Доцільно, використавши поняття узагальнення, зберігати єдиний масив даних про брокерів, а угоду зобразити у вигляді петлі зв'язку між брокерами, як це показано на рис.1.17. Наступний приклад — використання поняття **узагальнення** при об'єднанні різних типів сутностей — стосується організації БД предметна область обліку лізингових контрактів, коли підприємства-рентери укладають угоди на довгострокову оренду устаткування, що належить підприємству-ліссору. Тип сутності "Ліссор" подібний до типу сутності "Рентер" (вони описуються однаковими атрибутами). Тому в даному разі використовують принцип узагальнення для об'єднання цих типів сутностей в одну — "Підприємство". Зв'язок "Лізинговий контракт" відображає укладення лізингової угоди. Тип сутності "Устаткування" підпорядкований типу сутності "Підприємство".

При цьому під устаткуванням залежно від постановки задачі обліку можна розуміти як множину видів устаткування, що пропонується ліссорами для укладання угоди, так і устаткування, яке фактично становить предмет угоди. У такому випадку ER-діаграма включає цикл зв'язку між типами сутностей "Підприємство" й "Устаткування", як це показано на рис.1.15.

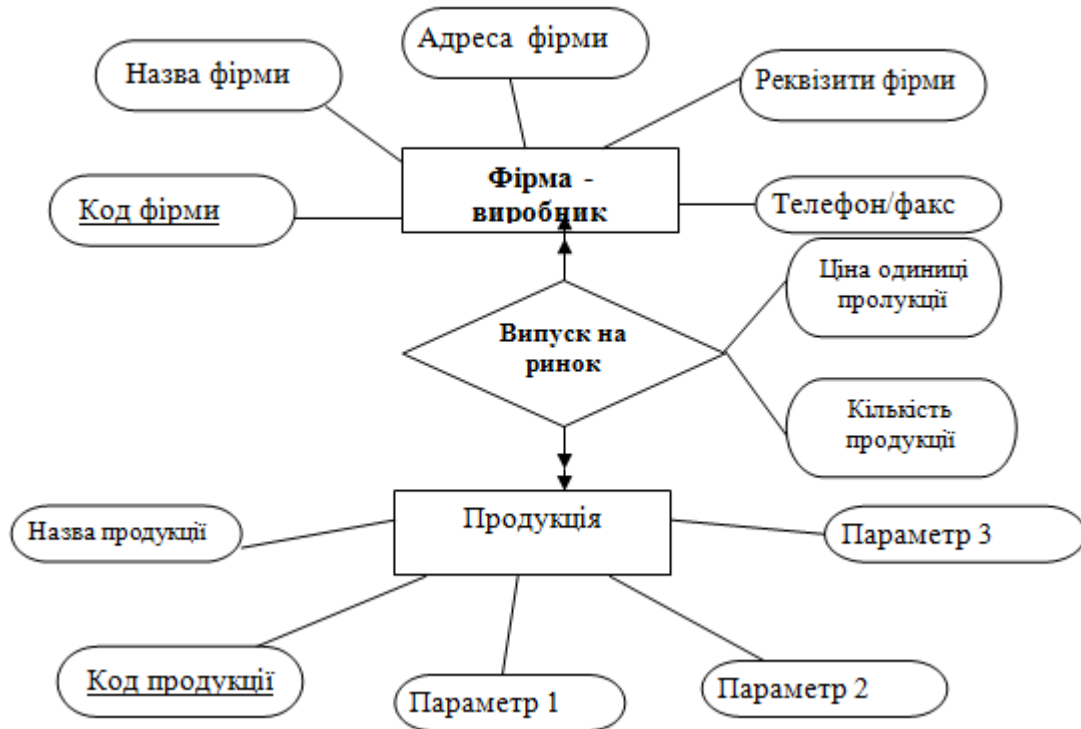


Рисунок 1.15 - Другий варіант ER-діаграми фрагмента БД ІС обліку випуску продукції

1.6. Даталогічне проектування бази даних

Точно розмежувати інфологічний і фізичний етапи проектування БД досить важко через відсутність установленної термінології. Тому прийнято вважати, що на етапі інфологічного проектування дані розглядають без урахування специфіки СУБД, що використовується, а особливості фізичного зберігання БД у пам'яті ЕОМ включають в опис її структури на етапі фізичного проектування. Етап між інфологічним і фізичним проектуванням, на якому одержують СУБД-орієнтовану схему БД, прийнято називати проектуванням реалізації або даталогічним проектуванням.

Проектування реалізації БД — перехід до даталогічної концептуальної моделі даних (СУБД-орієнтованої моделі).

Вирішальними при виборі СУБД є такі чинники:

— **абстрактна структура або модель даних, яку підтримує СУБД** (ієрархічна, мережна, реляційна, об'єктно-орієнтована, об'єктно-реляційна, багатовимірна, гібридна).

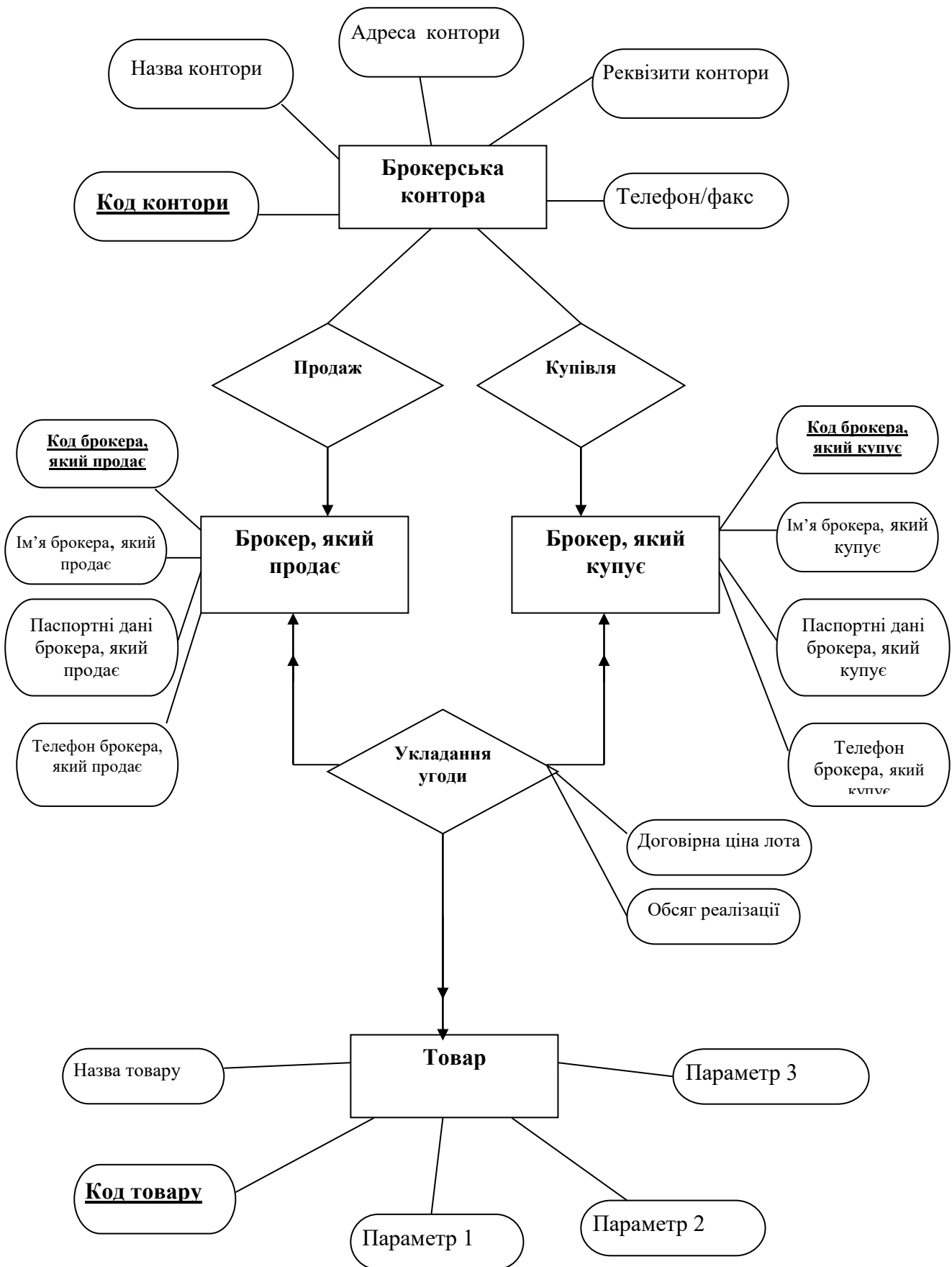


Рисунок 1.16 - Перший варіант ER-діаграми фрагмента БД ІС товарно-сировинної біржі

— Поняття даталогічної або СУБД-орієнтованої моделі передбачає передусім вибір моделі для БД і, відповідно, вибір СУБД, що підтримує таку модель. Загалом стан реальної предметної області відображається складною мережною моделлю, яка включає фрагменти простих мережних та ієрархічних структур. СУБД реляційного типу на сьогодні є настільки популярними, що здебільшого не залишають розробникам БД права вибору типу моделі даних. Тому реальна даталогічна модель зводиться до реляційної;

— **швидкість оброблення інформації** (перевага віддається більш високошвидкісним пакетам).

При виборі СУБД важливими є також чинники, що визначають вибір технічного та програмного видів забезпечення. Всі сучасні СУБД забезпечують можливість мережного оброблення інформації (багатокористувальний режим), тому цей чинник не є критичним при виборі СУБД.

Кожна нова СУБД сумісна з попередніми версіями своїх, а іноді не тільки своїх пакетів, але несумісна з попередніми версіями технічного забезпечення. Тому надмірне захоплення новинками протипоказане за наявності відсталої від потреб СУБД техніки.

При виборі лінгвістичного забезпечення вирішальною є наявність національних драйверів, які забезпечують адекватні введення-виведення, зберігання та маніпулювання даними. Особливу увагу при цьому слід приділити можливості адекватного сортування даних засобами СУБД.

Інфологічна концептуальна модель є більш загальним, порівняно з даталогічною концептуальною моделлю, відображенням стану предметної області. Даталогічна модель потребує більш детального опису стану об'єктів предметної області й зв'язків між ними через необхідність відображення моделі даних у пам'яті обчислювальної системи.

Тут можуть з'явитися додаткові об'єкти зберігання в БД. Наприклад, ER-діаграму фрагмента БД предметної області обліку переміщення матеріалів на складі, що відображає поставку матеріалів сторонніми організаціями і реалізацію надлишків матеріалів стороннім організаціям, можна зобразити двома варіантами.

Перший з них показано на рис.1.19. Дії "Одержання" та "Реалізація" подано різними зв'язками.

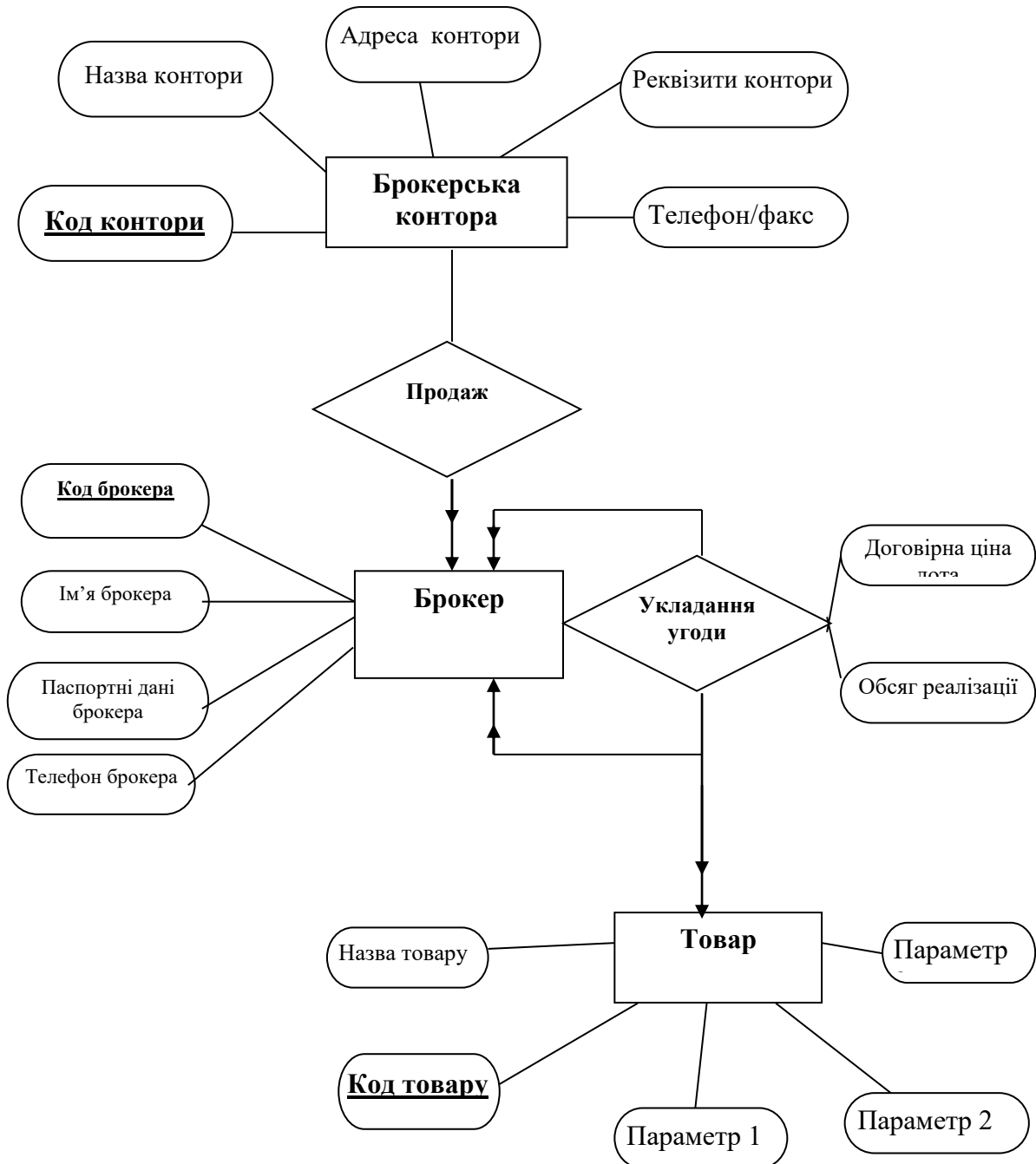


Рисунок 1.17 - Другий варіант ER-діаграми фрагмента БД ІС товарно-сировинної біржі

Другий варіант цього фрагмента зображено на рис.1.20. Контакти зі сторонніми організаціями тут подано у вигляді одного зв'язку "Зовнішня операція" з додатковим атрибутом "Код операції", не зафіксованим у словнику даних. Атрибут може набувати двох значень: 1 — "Одержання" і 0 — "Реалізація". Другий варіант фрагмента має переваги над першим. Вони зумовлені

тим, що замість двох масивів — зв'язок між типами сутностей "Матеріал" і "Зовнішня організація" використано лише один — "Зовнішня операція". Новий атрибут "Код операції" не спричиняє істотної надмірності в набір даних. На концептуальному інфологічному рівні можливим є використання як першого, так і другого варіантів ER-діаграми. Перехід до більш раціональної моделі можна виконати на даталогічному рівні, на якому відбувається уточнення та оптимізація структури БД.

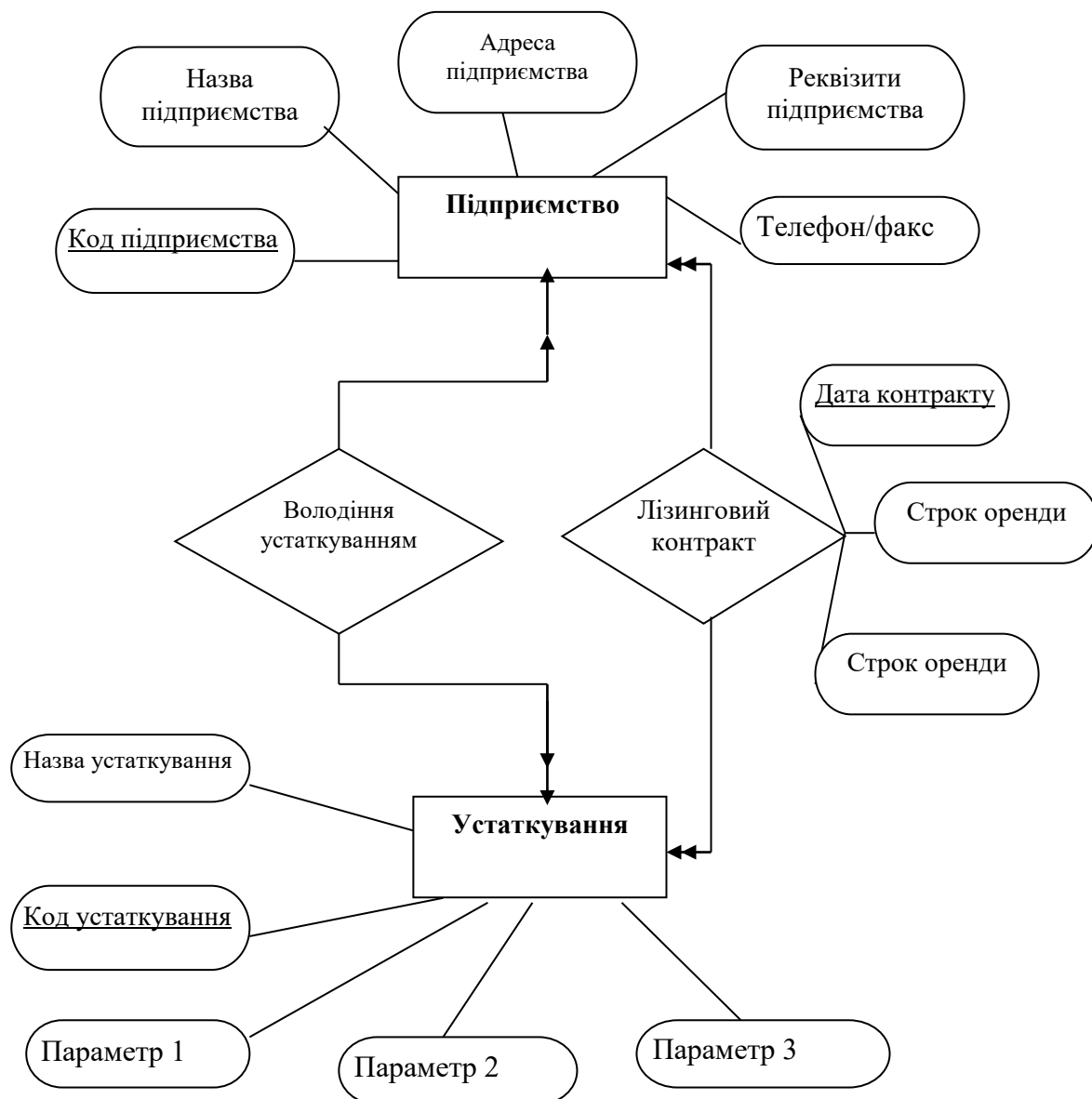


Рис.1.18. ER-діаграма фрагмента БД ІС обліку лізингових контрактів

На етапі проектування реалізації БД також можливим є доповнення одержаної інформаційної моделі предметної області інформаційними структурами (у вигляді ER-діаграм), що зумовлено додатково виявленими запитами до БД.

Надалі цей етап проектування є переходом від одержаної концептуальної інфологічної моделі до даталогічної або СУБД-орієнтованої моделі даних предметної області.

Графічно даталогічна модель є відображенням інфологічної. У ній тип сутності зображують у вигляді поіменованого прямокутника, кожна клітина якого відповідає атрибуту сутності. Ключ підкреслюється. Зв'язки між записами відображають у вигляді спрямованих ліній з зазначенням типів зв'язку 1:1, 1:Б, Б:Б. Дані перетину при зв'язку типу Б:Б указуються поруч із лінією зв'язку. Переходячи до даталогічної моделі даних, варто керуватися такими правилами:

Правило 1. Складна мережна модель є найпоширенішою моделлю даних, оскільки вона відображає складність зв'язків між об'єктами предметної області. Тому, як правило, даталогічна концептуальна модель даних відображається у вигляді складної мережної моделі. **Даталогічну модель** стосовно задачі обліку переміщення матеріалів, що відповідає другому варіанту фрагмента ER-діаграми (див. рис.20), показано на рис.21. Це неоднорідна складна мережна модель.

Правило 2. Від складної мережної моделі здійснюється перехід до простої мережної моделі. При цьому позбавляються зв'язків типу Б:Б, вводячи додатковий запис, який обов'язково повинен містити ключі записів, що з'єднуються. На рис.22 зображено неоднорідну просту мережну модель, що відповідає складній моделі на рис.21.



Рисунок 1.19 - Перший варіант ER-діаграми фрагмента БД ІС обліку переміщення матеріалів

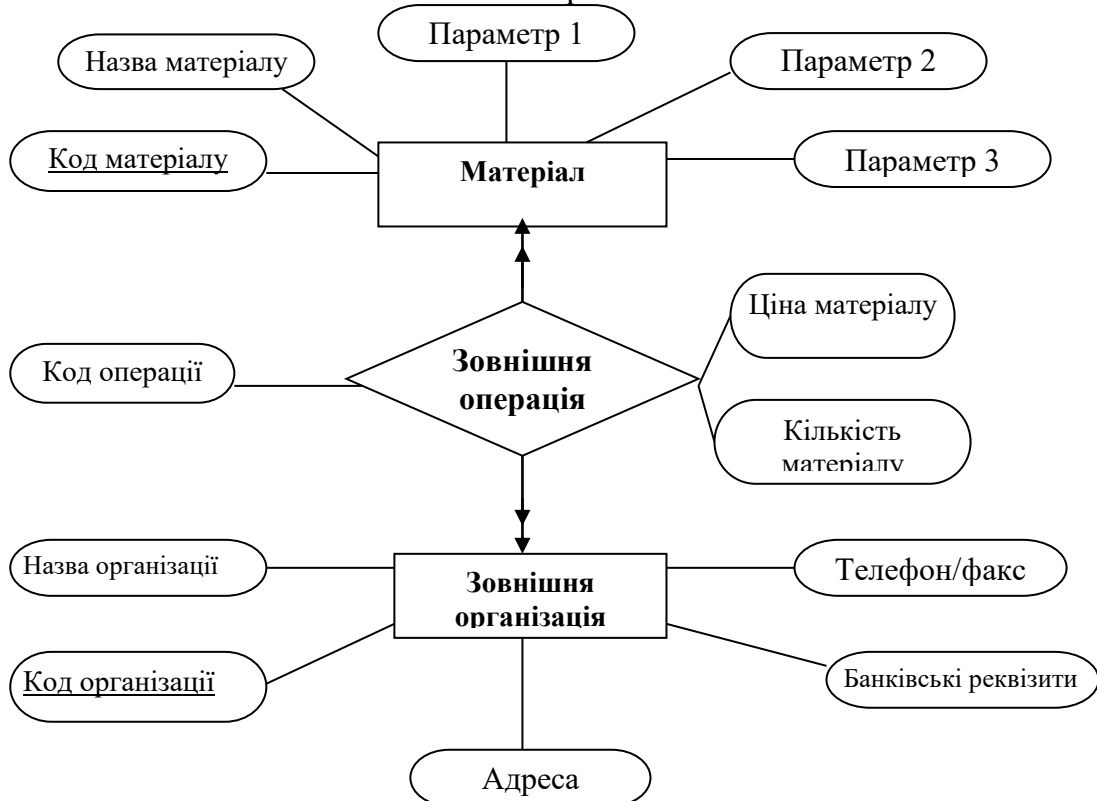


Рисунок 1.20 - Другий варіант ER-діаграми фрагмента БД ІС обліку переміщення матеріалів

МАТЕРІАЛ

<u>Код матеріалу</u>	Назва матеріалу	Параметр 1	Параметр 2	Параметр 3

Ціна матеріалу
Кількість матеріалу
Код операції

ЗОВНІШНЯ ОРГАНІЗАЦІЯ

<u>Код організації</u>	Назва організації	Адреса	Банківські реквізити	Телефон/ факс

Рисунок 1.21- Даталогічна модель фрагмента БД ІС обліку переміщення матеріалів у вигляді складної мережі

МАТЕРІАЛ

<u>Код матеріалу</u>	Назва матеріалу	Параметр 1	Параметр 2	Параметр 3

ЗОВНІШНЯ ОПЕРАЦІЯ

<u>Код матеріалу</u>	<u>Код організації</u>	Ціна матеріалу	Кількість матеріалу	Код операції

ЗОВНІШНЯ ОРГАНІЗАЦІЯ

<u>Код організації</u>	Назва організації	Адреса	Банківські реквізити	Телефон/ факс

Рисунок 1.22 - Даталогічна модель фрагмента БД ІС обліку переміщення матеріалів у вигляді простої мережі

Правило 3. При усуненні "залежності від шляху" в ієрархічних фрагментах моделі даних у породжені записи включають ключі породжувальних записів. Якщо екземпляри породжених записів ураховують у межах породжувальних записів, то первинним ключем породженого запису стає складний ключ. Він складається з первинного ключа породжувального запису та первинного атрибута породженого запису.

СКЛАД

<u>Код складу</u>	Назва складу	Адреса складу	Телефон складу

ВИРІБ

<u>Код виробу</u>	Назва виробу	Параметр 1	Параметр 2	Параметр 3	Параметр 4	Ціна	Кількість

Рисунок 1.23- Ієрархічна даталогічна модель фрагмента БД ІС обліку виробів на складі з залежним записом "Виріб"

На рис.1.23 показано ієрархічну даталогічну модель фрагмента БД ІС обліку виробів на складі, що відповідає ER-діаграмі (рис.1.9). У даному випадку атрибут породженого запису "Код виробу" означає нумерацію виробів у межах кожного складу.

Після усунення "залежності від шляху", як це показано на рис.1.24, ключ породженого запису "Виріб" стає складним. Він складається з ключа породжувального запису "Склад" і коду виробу на складі.

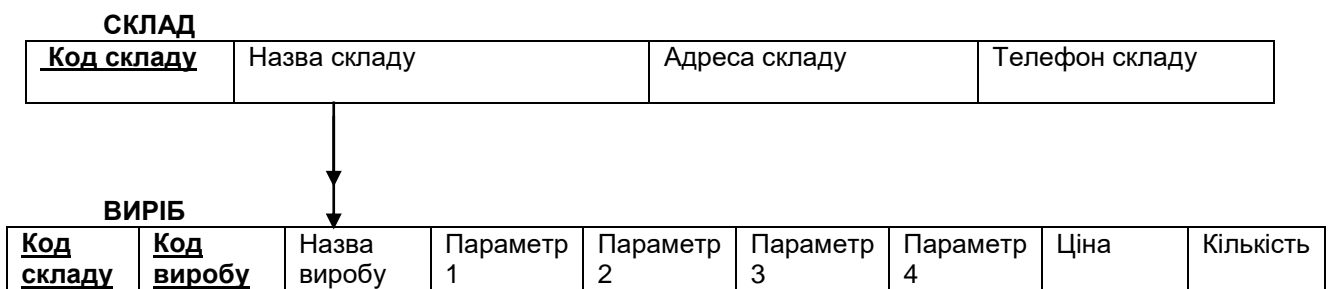


Рисунок 1.24 - Ієрархічна даталогічна модель фрагмента БД ІС обліку виробів на складі з незалежним записом "Виріб"

Якщо об'єкти породжених записів ураховують у межах усієї БД, то первинним ключем породженого запису є його первинний атрибут, а первинний ключ породжувального запису надходить до породженого запису і стає зовнішнім ключем, необхідним для з'єднання записів за запитами до БД. Результатом усунення "залежності від шляху" є те, що всі записи в моделі даних стають автономними, незалежними від породжувальних записів.

Правило 4. У мережній моделі поняття "породжувальний" і "породжений" записи умовні, тому що точкою входу в ній може бути будь-який запис. Вважати записи породжувальними та породженими можна тільки після вибору точки входу. Тоді "залежність від шляху" усувають так само, як і в ієрархічній структурі даних. У прикладі з фрагментом БД, зображеному на рис.1.22, запис "Зовнішня операція" містить ключі породжувальних записів, тому в цій моделі залежності від шляху немає. Приклад, який показано на рис.1.25, ілюструє просту мережну модель із залежним записом. Це фрагмент БД ІС обліку матеріалів на складі свідчить, що кожен матеріал залежить від складу, на якому він зберігається, і постачається певним постачальником.

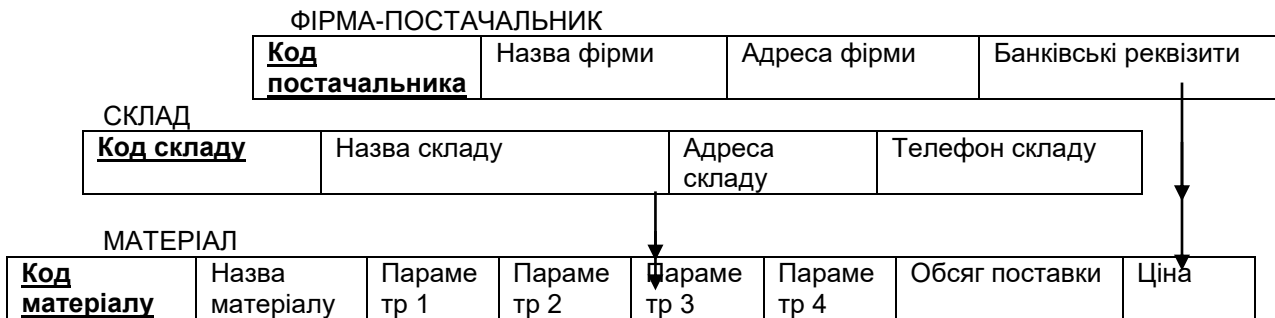


Рисунок 1.25 - Даталогічна модель БД ІС обліку матеріалів на складі з залежним записом "Матеріал"

Запис "Матеріал" стане автономним, якщо міститиме ключі записів "Склад" та "Фірма-постачальник" або як зовнішні ключі, або як компоненти складного первинного ключа "Код постачальника" + "Код складу" + "Код матеріалу". Рис.1.26 ілюструє останній варіант ключа запису "Матеріал".

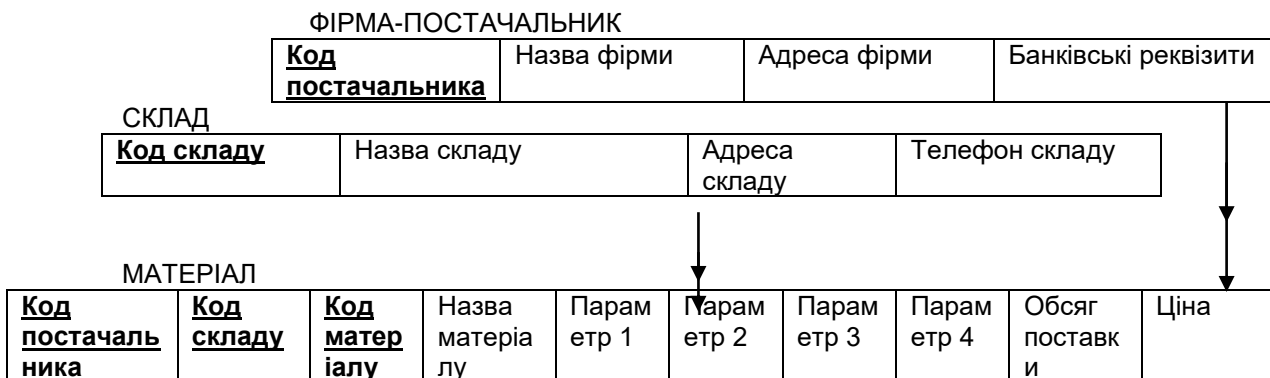


Рис.1.26 Даталогічна модель фрагмента БД ІС обліку матеріалів на складі з незалежним записом "Матеріал"

Правило 5. Фрагменти структури, що містять цикли та петлі, спрощують по-різному залежно від виду зв'язку між типами сутностей. За наявності зв'язків типу Б:Б у циклі або в петлі вводять додатковий запис із ключами записів, які з'єднуються.

Стосовно фрагмента БД ІС товарно-сировинної біржі (див. рис.1.17) даталогічна модель даних є складною мережею (рис.1.27).

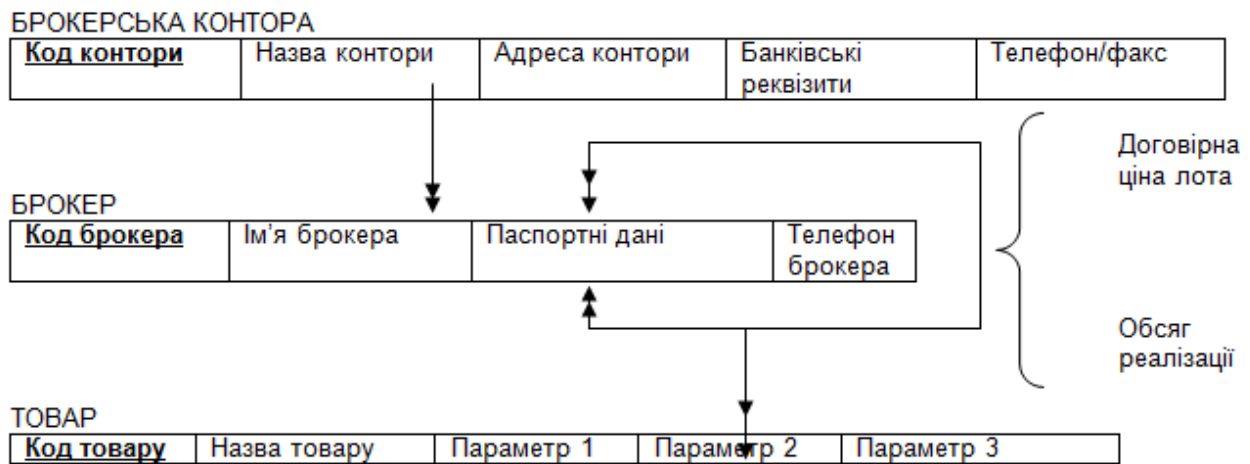


Рисунок 1.27 - Даталогічна складна мережна модель даних фрагмента БД ІС товарно-сировинної біржі

На рис.1.28 показано даталогічну просту мережну модель даних, в якій усунений зв'язок типу Б:Б і залежність від шляху. Ця модель відповідає складній мережній моделі даних, яку зображено на рис.1.27



Рисунок 1.28 - Даталогічна проста мережна модель даних фрагмента БД ІС товарно-сировинної біржі

Атрибути "Код контори 1" та "Код контори 2" належать домену "Код контори" і визначають коди контор брокера-продавця та брокера-покупця товару. Атрибути "Код брокера 1" та "Код брокера 2" належать домену "Код брокера" і визначають коди брокера-продавця та брокера-покупця у своїх конторах. Запис "Укладання угоди" має складний ключ. Перші два атрибути запису визначають брокера-продавця, два наступних — брокера-покупця, п'ятий — код товару. Стосовно фрагмента БД ІС обліку лізингових контрактів (див. рис.1.18)

дatalogічна модель є неоднорідною складною мережною (рис.1.28). На рис.1.29 зображено просту мережну модель даних, в якій усунений зв'язок типу Б:Б і залежність від шляху. Ця модель відповідає складній мережній моделі даних, яку показано на Рис.1.28.

ПІДПРИЄМСТВО

<u>Код підприємства</u>	Назва підприємства	Адреса підприємства	Банківські реквізити	Телефон/факс
-------------------------	--------------------	---------------------	----------------------	--------------

УСТАТКУВАННЯ

<u>Код устаткування</u>	Назва устаткування	Параметр 1	Параметр 2
-------------------------	--------------------	------------	------------

Дата контракту

Строк оренди
Сума до сплати

Рисунок 1.29 - Дatalogічна модель фрагмента БД ІС обліку лізингових контрактів у вигляді складної мережі

Первинний ключ запису "Лізинговий контракт" є складним: "Код підприємства 1" + "Код підприємства 2" + "Код устаткування" + "Дата контракту". Атрибути "Код підприємства 1" та "Код підприємства 2" належать домену "Код підприємства" і визначають код ліссора та код рентера відповідно.



Рис.1.30. Дatalogічна модель фрагмента БД ІС обліку лізингових контрактів у вигляді простої мережі

Правило 6. Після того як усі записи дatalogічної моделі даних стали автономними, тобто незалежними від породжувальних записів, вони можуть бути подані у вигляді схем окремих відношень. Дatalogічна реляційна модель даних містить набір схем відношень або записів, які графічно не пов'язані між собою. Схему відношення зображують у вигляді вектора. Назва вектора відповідає назві

запису. В дужках через кому іде перелік імен атрибутів. Первинний ключ підкреслюють.

Наприклад, стосовно рис.1.30 даталогічна реляційна модель даних має такий вигляд:

ПІДПРИЄМСТВО(Код підприємства, Назва підприємства, Адреса підприємства, Банківські реквізити, Телефон_факс);

ЛІЗИНГОВИЙ КОНТРАКТ(Код підприємства 1, Код підприємства 2, Код устаткування, Дата контракту, Строк оренди, Сума до сплати оренди);

УСТАТКУВАННЯ(Код підприємства, Код устаткування, Назва устаткування, Параметр 1, Параметр 2)

Правило 7. Даталогічна реляційна модель даних має складатися з набору схем відношень, які неможливо звести. Відношення цієї моделі мають бути зведені як мінімум до третьої нормальної форми.

Правило 8. Первинні ключі відношень використовуються при проектуванні унікальних первинних ключових або індексних файлів, за допомогою яких здійснюється пошук даних у пам'яті обчислювальної системи.

Правило 9. У відношенні порядок атрибутів у складному ключі не має значення, а при проектуванні індексного або ключового файлу порядок полів, що відповідають цим атрибутам, має принципове значення. Сортування даних ключа виконується спочатку за першим атрибутом, а в межах однакових екземплярів першого атрибута ключа — за наступним атрибутом і т.д.

Правило 10. Одержану даталогічну модель аналізують з метою оптимізації (зокрема, оцінюється ефективність схеми БД за обсягом оброблення).

Кожна прикладна задача характеризується частотою виконання та кількістю звернень **LRA** (*Logical records access*) до логічних записів БД. Кількість **LRA** може бути оцінена за даними таблиці "Задача—дані" та даними про частоту виконання кожної прикладної задачі.

Загальна кількість **LRA** в усіх прикладних задачах є одним з основних критеріїв ефективності схеми БД. **LRA** характеризує загальну кількість звернень до фізичних записів БД, що містяться в зовнішній пам'яті. Чим вищий цей показник у системі, тим нижча швидкість оброблення. Тому прикладна задача з

найбільшим значенням **LRA** має бути досліджена з метою вдосконалення структури даних, зменшення частоти виконання, вирішення питання про доцільність оптимізації зберігання записів у процесі експлуатації БД. Останню обставину особливо важливо враховувати тоді, коли файл даних через великий розмір повністю не вміщується в оперативній пам'яті. Вибірка записів такого файлу, відсортованого за ключем, може забирати багато часу. Це зумовлено тим, що близькі за ключем записи можуть фізично знаходитись на диску на значній відстані один від одного. Такі файли треба час від часу перезаписувати, заздалегідь проіндексувавши їх за первинним ключем. Це мінімізує час оброблення записів.

За результатами аналізу розроблюють пропозиції щодо вдосконалення схеми БД і режимів оброблення даних.

1.7 Фізичне проектування бази даних

Фізична організація даних істотно впливає на експлуатаційні характеристики БД, що проектується, оскільки вона зумовлена з прив'язкою БД до фізичної пам'яті.

Фізичне проектування БД — відображення даталогічної концептуальної моделі даних у пам'ять обчислювальної системи.

Наявні інструментальні засоби розроблення прикладних програм (ОС та СУБД) не потребують проектування фізичних записів, оскільки використовують фізичні записи, прийняті в цих інструментальних засобах.

На цьому етапі проектують структуру файлів даних і ключових (індексних) файлів, а також методи доступу до записів даних, які полягають у виборі первинного та вторинного шляхів доступу до даних.

Первинні шляхи доступу визначають способи пошуку записів БД, а вторинні включають міжфайлові зв'язки й альтернативні точки входу для доступу до записів, що зберігаються, за допомогою вторинних ключів.

Програмний продукт налагоджують на контрольному прикладі, який охоплює усі можливі запити до БД.

1.8 Вибір системи управління базами даних

Система управління базами даних — програмна система, призначена для створення на ЕОМ спільної для багатьох додатків БД, підтримання її в актуальному стані і забезпечення ефективного доступу користувачів до даних.

Залежно від особливостей функціонування розрізняють два класи СУБД — загального призначення і спеціалізовані. **СУБД загального призначення** не орієнтовано на конкретну ПС або на інформаційні потреби конкретної групи користувачів. Кожна така система реалізується як програмний засіб, здатний функціонувати на певній моделі ПЕОМ в певній операційній обстановці, і постачається багатьом користувачам як комерційний виріб. **СУБД спеціалізованого призначення** мають засоби налаштування на роботу з конкретною БД за конкретного застосування.

Основними засобами СУБД є: - засоби задання (опису) структури БД; - засоби конструювання екранних форм, призначених для введення, перегляду й оброблення даних у діалоговому режимі; - засоби створення запитів для вибірки даних за заданими умовами, а також їх оброблення; - засоби створення звітів з БД для виведення на друк результатів оброблення; - мовні засоби — макроси, вбудована алгоритмічна мова, мова запитів, які використовуються для реалізації нестандартних алгоритмів оброблення даних; - засоби створення додатків користувача (генератори додатків, засоби створення меню і панелі керування додатками), що дають змогу об'єднати різні операції з БД в єдиний технологічний процес.

Відповідно до моделі та структур даних, які підтримують СУБД, організують БД.

З появою ПК настав новий період в розвитку технології БД. З'явилися СУБД, які дозволяли зберігати значні об'єми інформації, вони мали зручний інтерфейс для заповнення даних, засоби для генерації різних звітів, запитів, мову програмування для розробки модулів.

Особливості цього етапу такі:

- всі СУБД були розраховані на створення БД в основному з монопольним доступом;
- більшість СУБД мали розвинутий і зручний інтерфейс користувача;
- в СУБД були відсутні засоби підтримки цілості посилання, структурної цілості бази даних (декілька файлів);
- наявність монопольного режиму роботи фактично привела до відміни функції адміністрування БД і в зв'язку з цим – до відсутності інструментальних засобів адміністрування БД;
- наступні СУБД мали невеликі вимоги до апаратного забезпечення ПК (наприклад, програми, працювали на Clipper; PC 286)

Приклади: СУДБ (DBASE III+, VI), FoxPro, Clipper, Paradox.

Розподілені бази даних. Добре відомо, що історія розвивається по спіралі, тому після процесу “персоналізації” почався зворотній процес – інтеграція. А саме, збільшується кількість локальних мереж, все більше інформації передається між комп'ютерами, гостро стає задача узгодженості даних, що зберігаються і обробляються в різних місцях, але логічно пов'язаних між собою, виникають задачі, пов'язані з паралельною обробкою транзакцій – послідовностей операцій над БД. Успішне вирішення цих задач приводить до появи розподілених баз даних, що зберігають всі переваги локальних СУБД, і в той же час дозволяють організувати паралельну обробку інформації і підтримку цілості БД.

Особливості цього етапу:

- практично всі сучасні СУБД забезпечують підтримку нової реляційної моделі (а саме: структурної цілості, тобто представлення всіх даних у вигляді таблиць; мовної цілості, тобто мова маніпулювання даними високого рівня (SQL); цілісність посилання)
- більшість сучасних СУБД розраховані на багатоплатформні архітектури комп'ютерів і на різні ОС;
- розвинуті засоби адміністрування БД з реалізацією загальної концепції засобів захисту даних;

- майже всі сучасні СУБД мають засоби підключення клієнтських додатків, розроблених з використанням локальних (настільних) СУБД;
- розробка стандартів в межах мов опису та маніпулювання даними, починаючи з SQL89, SQL92, SQL99 та технологій по обміну даними між різними СУБД, наприклад, ODBC (Open DataBase Connectivity)
- роботи пов'язані з концепцією об'єктно-орієнтованих БД – (MS Access 97,2000,2003, 2010, 2016, Oracle 7.3, Oracle 8.4, MS SQL 7.0, SQL Base).

Наступний етап характеризується появою нової технології доступу до даних – **Інтернет**. Основна відмінність цього підходу від технології клієнт – сервер полягає в тому, що зникає необхідність використання спеціалізованого клієнтського програмного забезпечення. Для роботи з віддаленою базою даних використовується стандартний браузер Інтернет, наприклад Explorer, або Netscape Navigator (аналогічно WWW). При цьому код написаний на мові Java, Java-script, Perl і “вбудований” в HTML-сторінки, що завантажуються користувачем, відслідковує всі дії користувача і транслює їх в SQL-запити до бази даних, виконуючи тим самим ту роботу, якою в технології клієнт-сервер займається клієнтська програма. Зручність такого підходу стала причиною того, що він став використовуватись не тільки для віддаленого доступу до баз даних, а й для користування локальною мережею підприємства. В такому випадку для підключення нового користувача до можливості використовувати дану задачу (алгоритм обробки – простий) не вимагає установки додаткового клієнтського програмного забезпечення. Але алгоритмічно складні задачі рекомендуються реалізувати в архітектурі клієнт-сервер з розробкою спеціального програмного забезпечення.

Однією з найважливіших тенденцій розвитку СУБД є розроблення універсальної СУБД, здатної інтегрувати в базі традиційні та нетрадиційні дані — тексти, рисунки, звук і відео, сторінки HTML та ін. Це є особливо актуальним для Web. При побудові такої СУБД використовують два підходи: об'єктно-реляційний — вдосконалення наявних реляційних СУБД й об'єктний. Шляхом створення об'єктно-реляційних СУБД пішли такі фірми, як IBM, Informix, Oracle.

Фірмою IBM розроблено об'єктно-реляційну СУБД DB2 для ОС AIX та OS/2. Корпорація Microsoft зробила ставку на об'єктно-орієнтований інтерфейс OLEDB, що забезпечує доступ до даних Microsoft SQL Server (реляційна СУБД). Фірма Sybase орієнтується на використання спеціалізованих серверів, а інтеграцію даних має намір провести іншими засобами, тобто йде шляхом створення об'єктно-реляційної СУБД Adaptive-Server.

Інформаційні сховища на основі СУБД з паралельним обробленням розраховано на багатопроцесорні системи. Такі СУБД за типом архітектури можуть бути без поділу ресурсів і зі спільним використанням дискового простору. У першому випадку за кожним із процесорів закріплюються виділені області пам'яті та диски, що забезпечує високу швидкість оброблення даних. У другому випадку оперативна пам'ять і місце на диску розподіляються між усіма процесорами.

Прикладами СУБД без поділу ресурсів є системи: DB2 (IBM), Informix Online Dynamic (Informix), Navigation Server (Sybase). СУБД зі спільним використанням пам'яті є система Adabas D 6.1 (Software AG). У СУБД Oracle 7.2 забезпечується краще перенесення інформації на різні платформи.

Вибір СУБД доцільно здійснювати не тільки за типом архітектури та якістю зовнішнього інтерфейсу, а, передусім, на основі функціональних можливостей. Суттєвим критерієм вибору є здатність оброблення складних запитів (і швидкість оброблення), можливість перенесення даних між платформами.

2. Курсова робота з дисципліни "Бази даних та СУБД"

Метою написання курсової роботи є: - систематизація, закріплення та розширення теоретичних і практичних знань з проектування баз даних інформаційної системи з використанням сучасних інформаційних технологій на підставі аналізу економіко-інформаційного середовища предметної області; - розвиток навиків ведення самостійної роботи та розробки проектних рішень з інформаційного та програмного забезпечення інформаційної системи.

2.1 Завдання курсових робіт

Завдання: розробити проект бази даних відповідно до інформаційних потреб вказаної інформаційної системи.

Інформаційна система повинна забезпечувати виконання вказаних нижче операцій над базами:

1. База складається з декількох зв'язаних таблиць з використанням (по необхідності) зв'язків «один до одного», «один до багатьох», «багато до одного»;
2. Для полегшення перегляду і пошуку використовується упорядкованість за допомогою індексації або (у крайньому випадку) сортування;
3. Вся інформація піддається переглядові і редагуванню. Записи, що містять багато полів, можна переглядати як у табличному, так і в посторінковому вигляді (у вигляді карток);
4. Інформація, що піддається стандартизації, зберігається в окремих таблицях-довідниках, що також піддаються корегуванню (з дотриманням цілісності);
5. Програма дозволяє проводити пошук або вибірку інформації з довільного запиту (фільтрові). При невдалому завданні умови запиту, можливе його корегування (з метою звуження або розширення зони пошуку).

2.2 Варіанти курсових робіт

Варіант 1

Інформаційна система: Облік обслуговування абонентів міської телефонної станції.

Інформаційні потреби. Керівництву міської телефонної станції необхідна інформація з обліку обслуговування абонентів. З цією метою необхідно зберігати відомості про абонентів міста, про тарифи міських, міжміських та міжнародних переговорів, про тривалість розмов абонентів, про оплату абонентами послуг міської телефонної мережі та про їх заборгованість міській телефонній мережі

Варіант 2

Інформаційна система: Облік виконання робіт на підприємстві.

Інформаційні потреби. Керівництву підприємства необхідна оперативна інформація про облік виконання робіт працівниками підприємства. З цією метою необхідно зберігати дані про працівників, про види робіт, які працівники виконують, про проекти, в розробці яких працівники приймають участь, а також про клієнтів – замовників цих проектів. При цьому кожний працівник може виконувати декілька видів робіт в кожному проекті і може бути зайнятий в декількох проектах. Один і той же вид робіт може виконуватись різними працівниками. Погодинний тариф установлений для кожного працівника окремо, а нарахування залежить від часу, затраченого на виконання робіт.

Варіант 3

Інформаційна система: автоматизація збереження періодичних видань.

Інформаційні потреби. З метою автоматизації пошуку місця збереження періодичних видань, потрібна інформація про сховища періодичних видань, про тематику періодичних видань та самих виданнях за багато років. Між тематикою видання (науковий напрям) та сховищем немає зв'язку. В кожному сховищі може знаходитись багато різноманітних видань, але кожне видання відноситься до одної тематики (до одного наукового напрямку).

Варіант 4

Інформаційна система: облік обслуговування клієнтів рекламного агентства .

Інформаційні потреби. Керівництву рекламного агентства необхідна оперативна інформація про облік обслуговування клієнтів. Для цього необхідно зберігати дані про рекламних агентів, рекламодавців, видах рекламних послуг, обслуговування рекламодавців агентами. Кожний рекламодавець може замовляти різноманітні види рекламних послуг та обслуговуватись різними агентами. Кожний агент може обслуговувати багато рекламодавців відносно прийому замовлень на різноманітні послуги. Кожна послуга може бути запропонована багатьом рекламодавцям багатьма агентами. Вартість послуги довгий час залишається постійною (реєстрація змін вартості послуги в задачі не

передбачена). В один і той же день один і той же рекламодавець не звертається з повторним проханням надати одну й ту саму рекламну послугу.

Варіант 5

Інформаційна система: Облік використання обладнання на підприємстві.

Інформаційні потреби. Керівництву підприємства необхідна інформація з обліку використання обладнання на підприємстві. Для цього потрібно зберігати дані про цехи, про продукцію, що виготовляється на цьому підприємстві та дані про використання обладнання. При цьому обладнання закріплене за цехом, кожний вид обладнання використовується при виготовленні багатьох видів продукції і кожний вид продукції, як правило проходить технологічний ряд виготовлення в багатьох цехах на різноманітному обладнанні.

Варіант 6

Інформаційна система: облік обслуговування клієнтів банку.

Інформаційні потреби. Керівництву банку необхідна оперативна інформація з обліку обслуговування клієнтів банку. Банк має декілька філіалів, які розміщені в різних містах України. В кожному місті може бути декілька філіалів, і клієнт може обслуговуватись в кожному філіалі банку. Кожний клієнт має один розрахунковий рахунок в своєму банку.

Варіант 7

Інформаційна система: облік видачі та погашення кредитів.

Інформаційні потреби. Керівництву банків необхідна оперативна інформація з обліку видачі та погашення кредитів. Для цього потрібно зберігати інформацію про банки України, організації, які є клієнтами певних банків, про ведення рахунків клієнтів, про видачу організаціям кредитів сторонніми для них банками, про погашення кредитів. Організації можуть мати декілька розрахункових рахунків в одному або декількох банках і можуть брати кредити в банках, в яких не мають розрахункових рахунків. Операція видачі та погашення кредиту здійснюється не безпосередньо між банком і організацією, а між банком-кредитором та одним із банків, які ведуть рахунки організації замовника.

Варіант 8

Інформаційна система: облік обслуговуванні клієнтів банку

Інформаційні потреби. Керівництву банків необхідна оперативна інформація з обліку обслуговування клієнтів. Центральні відділення банків знаходяться в різних містах України, філіали банків можуть знаходитись в містах, відмінних від міст, де знаходяться центральні відділення. Клієнти банку знаходяться в містах, можливо відмінних від міст, де знаходяться банки і філіали. Клієнти можуть обслуговуватись у всіх філіалах тих банків, де у них є розрахункові рахунки.

Варіант 9

Інформаційна система: облік основних засобів.

Інформаційні потреби. Керівництву підприємства необхідна інформація з обліку основних засобів. Для цього потрібно зберігати інформацію про надходження основних засобів, переміщення та вибуття, нарахування амортизаційних відрахувань. Комплекс задач з обліку основних засобів повинен забезпечувати автоматизацію ведення бухгалтерського обліку ОЗ, занесення первинних документів, виконання необхідних розрахунків, формування вихідних документів, реалізацію запитів і довідок до БД, підготовку бухгалтерських проводок облікових операцій для відображення в бухгалтерському балансі.

Варіант 10

Інформаційна система: облік заявок на торги на товарно – сировинній біржі.

Інформаційні потреби. Керівництву товарно – сировинної біржі необхідна оперативна інформація з обліку заявок на торги. Для цього потрібно зберігати дані про брокерські контори, з якими співробітничает біржа, про брокерів, які в них працюють, про заявлені ними товари на різні торги. при цьому кожний брокер працює тільки в одній конторі, на торг він має можливість заявити багато товарів, і кожний товар може бути замовлений різними брокерами.

Варіант 11

Інформаційна система: облік укладання договорів на товарно – сировинній біржі.

Інформаційні потреби. Керівництву товарно – сировинної біржі необхідна оперативна інформація з обліку укладання договорів купівлі-продажу. Для цього необхідно зберігати дані про брокерські контори, з якими співробітничав біржа, про брокерів, які в них працюють, про укладання договорів купівлі-продажу виставлених на торги товарів. Кожний брокер працює тільки в одній конторі, в день торгів він може виступати як в ролі продавця, так і в ролі покупця декількох товарів. Кожний товар може бути проданий (куплений) декількома брокерами - учасниками торгів. Інформація про наступну дату торгів замінює інформацію про попередню дату.

Варіант 12

Інформаційна система: облік реалізації товарів через кредитні картки.

Інформаційні потреби. Керівництву супермаркету необхідна оперативна інформація з обліку реалізації товарів покупцям супермаркету через кредитні картки. Для цього потрібно зберігати дані про відділи, касові апарати кожного відділу, товари, які зберігаються на центральному складі, покупців, які обслуговуються через кредитні картки та їх покупки. Інформація про покупки клієнта накопичується протягом декількох годин, а потім передається у філіал банку згідно банківських реквізитів, що вказані в кредитній картці клієнта. Покупець за цей час може придбати декілька товарів, скориставшись різними касовими апаратами. Кожний товар може бути придбаний різними покупцями через декілька касових апаратів.

Варіант 13

Інформаційна система: облік розподілу товарів між відділами супермаркету.

Інформаційні потреби. Керівництву супермаркету необхідна оперативна інформація з обліку розподілу товарів між відділами супермаркету. Для цього необхідно зберігати дані про відділи, склади відділів, товари, що розподіляються кожного ранку між складами відділів, про збереження товарів на цих складах. Причому на кожному складі зберігається множина товарів, і в різних відділах

може продаватись однаковий товар. Облік відділів і товарів виконується в межах розроблюваної підсистеми, облік складів в межах кожного відділу.

Варіант 14

Інформаційна система: автоматизована підтримка маркетингових досліджень.

Інформаційні потреби. З метою підтримки маркетингових досліджень керівництву підприємства-виробника потрібно володіти інформацією відносно конкурентів, які випускають аналогічну продукцію, про споживачів цієї продукції та реалізацію продукції на ринку. При цьому кожне підприємство випускає і реалізує декілька видів продукції багатьом споживачам. Кожний вид продукції випускається багатьма виробниками і реалізується багатьом споживачам. Ціна тієї ж самої продукції різних виробників може бути різною.

Варіант 15

Інформаційна система: облік пасажирських авіаперевезень.

Інформаційні потреби. Керівництву аеропорту необхідна інформація з обліку пасажирських авіаперевезень між містами на внутрішніх і зовнішніх авіалініях. З цією метою потрібно зберігати дані про держави, міста, аеропорти, літаки, авіаперевезення пасажирів. Причому кожний аеропорт любого міста і держави може бути зв'язаний авіалінією з любим іншим аеропортом. Кожний літак може здійснювати рейс із різних аеропортів відправлення в різні аеропорти призначення. Приналежність літаків певним авіакомпаніям в даній задачі не враховується. Облік держав, міст, літаків виконується в межах розроблюваної підсистеми, облік аеропортів - в межах кожного міста.

Варіант 16

Інформаційна система: автоматизації формування розкладу занять факультету.

Інформаційні потреби. Деканату факультету з метою автоматизації формування розкладу занять факультету потрібно зберігати дані про викладачів, дисципліни, навчальні аудиторії та про проведення занять. Кожний викладач може проводити заняття з різноманітних дисциплін в різних навчальних групах і в

різних аудиторіях. Одну й ту саму дисципліну можуть вести різні викладачі різним навчальним групам та в різних аудиторіях. В кожній групі заняття проводяться різними викладачами, з різних дисциплін та в різних аудиторіях. В одній і тій самій аудиторії заняття проводяться різними викладачами, з різних дисциплін, для різних навчальних груп.

Варіант 17

Інформаційна система: облік успішності студентів факультету.

Інформаційні потреби. Деканату факультету необхідна оперативна інформація з обліку успішності студентів факультету в період сесії. З цією метою потрібно зберігати дані про викладачів, дисципліни, навчальні групи, студентів, кураторів, результатах складання заліків та екзаменів. Кожна група складається із багатьох студентів і має одного куратора. Кожен викладач може бути куратором одної групи і приймати заліки та екзамени у багатьох студентів і з багатьох дисциплін. З кожної дисципліни велика кількість студентів складають заліки та екзамени різним викладачам. Кожний студент складає декілька заліків і екзаменів з багатьох дисциплін різним викладачам. Передбачається можливість реєстрації ліквідації студентами заборгованості із заліків та екзаменів.

Варіант 18

Інформаційна система: автоматизація діяльності центру зайнятості.

Інформаційні потреби. Керівництву обласного центру зайнятості необхідна оперативна інформація про вакансії і претендентів на них. Для цього потрібно зберігати дані про підприємства (організації), вакантні місця, які є в наявності на цих підприємствах (в організаціях), про клієнтів центру зайнятості – претендентів на ці вакантні місця та про результати співбесіди з претендентами на вакансії. Причому на одну вакансію може бути декілька претендентів і кожному клієнту центру зайнятості надається можливість вибору місця роботи через декілька вакансій. Обговорення з кожним клієнтом однієї й тієї ж вакансії може здійснюватись декілька разів (в різні дні).

Варіант 19

Інформаційна система: обліку обслуговування клієнтів страхової компанії.

Інформаційні потреби. Керівництву страхової компанії з метою автоматизації обліку обслуговування клієнтів необхідно зберігати інформацію про види страхових полісів, які випускаються компанією, службовців компанії – страхових агентів, фізичних осіб, які беруть страхові поліси та про реалізацію страхових полісів службовцями компанії фізичним особам. Кожна фізична особа може придбати декілька полісів і може обслуговуватись декількома агентами. Кожний агент може обслуговувати декількох фізичних осіб з приводу придбання різних полісів. Кожний вид полісу може бути реалізований багатьом фізичним особам через посередництво різних агентів. Вартість полісу залишається незмінною (реєстрація змін вартості в даній задачі не передбачається). В один і той же день один і той же клієнт не звертається повторно з приводу одного і того ж самого виду страхування.

Варіант 20

Інформаційна система: обліку обслуговування клієнтів фондового магазину.

Інформаційні потреби. Керівництву фондового магазину потрібно мати оперативну інформацію з обліку обслуговування клієнтів. З цією метою необхідно зберігати дані про організації – емітенти, цінні папери, які вони випускають, про фізичних осіб, які купують цінні папери, про реалізацію цінних паперів фізичним особам. Кожен цінний папір має помітку приналежності певному емітенту. Кожна фізична особа може придбати декілька видів цінних паперів різних емітентів. Кожен вид цінних паперів може бути придбаний багатьма фізичними особами. Інформація про цінні папери поновлюється в день емісії. Ціна реалізації залежить від дати реалізації. Облік організацій – емітентів та фізичних осіб виконується в межах підсистеми, що розробляється, облік цінних паперів – в межах організацій емітентів.

Варіант 21

Інформаційна система: Облік запасів та витрат деталей на підприємстві.

Інформаційні потреби. Керівництву підприємства потрібно мати оперативну інформацію про запаси та витрати деталей на підприємстві. Деталі

зберігаються в коморах цехів (кожна комора закріплена за певною ділянкою цеху). В коморі цехів деталі поступають з центрального складу підприємства. Одні й ті самі деталі використовуються для виробництва на різних ділянках одного цеху та в різних цехах. Облік цехів і деталей виконується в межах розробленої підсистеми, облік комір – в межах кожного цеху.

Варіант 22

Інформаційна система: автоматизація пошуку літератури в бібліотечних фондах.

Інформаційні потреби. З метою автоматизації пошуку та запиту необхідного літературного джерела по міжбібліотечному абоненту необхідно зберігати інформацію про міста, де розташовані великі наукові бібліотеки, про наукові бібліотеки і книги, які вони мають в своїх фондах. Одна й та сама книга може бути представлена в різних бібліотеках в різноманітній кількості екземплярів.

Варіант 23

Інформаційна система: Облік сировини та готової продукції на підприємстві.

Інформаційні потреби. Керівництву підприємства потрібно мати оперативну інформацію про надходження сировини і її параметрів, а також про підприємства – постачальники сировини, про підприємства – споживачі готової продукції, про види готової продукції та про її реалізацію. Підприємство закупляє сировину у декількох постачальників. В один і той же день у одного й того ж постачальника здійснюється не більше одної закупки. Підприємство випускає декілька видів продукції. Кожний вид продукції може реалізовуватись багатьом споживачам. В один і той же день одному й тому ж споживачу один і той же вид готової продукції реалізується не більше одного разу. Тарифи на сировину залежать від її параметрів, тарифи на готову продукцію – від виду цієї продукції.

Варіант 24

Інформаційна система: облік руху матеріалів між складами підприємства

Інформаційні потреби. Керівництву підприємства необхідно мати оперативну інформацію з обліку руху матеріалів між складами підприємства та сторонніми організаціями. Для цього потрібно зберігати дані про склади, матеріали, які там зберігаються, про сторонні для підприємства організації, про рух матеріалів, які постачають сторонні організації та реалізацію залишків матеріалів стороннім організаціям. На кожному складі зберігаються різні матеріали, але кожний матеріал зберігається на певному складі. Кожна організація може постачати або отримувати різні матеріали, а кожний матеріал може надходити від різних організацій та реалізуватись різним організаціям.

Варіант 25

Інформаційна система: облік використання будівельних матеріалів та облік робіт будівельної компанії.

Інформаційні потреби. Керівництву будівельної компанії необхідно мати оперативну інформацію з обліку використання будівельних матеріалів та обліку робіт. Для цього потрібно зберігати дані про будівлі, які будуються будівельною компанією, про витрати матеріалів на будівництво, про зайнятість робочих на будівництві. Для всіх будівель потрібні різноманітні матеріали в різних кількостях. На різних етапах проекту працюють різні бригади: арматурників, мулярів, штукатурів. Склад бригад та чисельність є непостійними. Робочий може працювати в одній бригаді простим робочим і керувати іншою бригадою (в різний період). Бригади складаються так, як вимагається для конкретного об'єкту.

Варіант 26

Інформаційна система: облік виробництва тваринницької продукції в сільськогосподарських формуваннях регіону.

Інформаційні потреби. Для здійснення адміністративного управління на рівні регіону галуззю тваринництва відділу розвитку тваринництва обласного управління необхідна оперативна інформація з виробництва тваринницької продукції та вихідного поголів'я в сільськогосподарських формуваннях районів області. Для цього необхідно зберігати дані про чисельність, склад та структуру виробленої і реалізованої продукції тваринництва в розрізі господарств, також

про парування маточного поголів'я, про рух рогатої та іншої худоби в сільських господарствах району. Ця інформація надходить на початку кожного місяця з районів області. Районним відділам інформація надається головними зоотехніками сільськогосподарських підприємств.

Варіант 27

Інформаційна система: Автоматизований аналіз зовнішньоекономічної діяльності.

Інформаційні потреби: Для здійснення регулювання зовнішньоекономічною діяльністю необхідне накопичення, обробки та збереження даних про цю діяльність суб'єктами господарської діяльності України (а саме в Сумській області) й іноземними суб'єктами. Такими даними є дані про експортно-імпорتنі операції з товарами, послугами, операції з давальницькою сировиною та давальницькою продукцією, дані про іноземні інвестиції. База даних повинна містити відомості, достатні для вирішення таких задач: - загальний аналіз зовнішньоекономічною діяльністю; - аналіз операцій з експорту та імпорту товарів; - формування обороті на бартерній основі; - формування оборотів з давальницькою сировиною та давальницькою продукцією; - географія зовнішньоторговельних партнерів; - аналіз операцій з експорту та імпорту послуг; - територіальна структура зовнішньоторговельного обороту Сумської області.

2.2 Методичні вказівки щодо написання курсової роботи.

Курсова робота складається з вступу, двох розділів, висновків, списку використаної літератури та додатків.

У вступі вказується актуальність використання бази даних як засобу автоматизованого вирішення задач заданої, відповідно варіанту, предметної області.

Перший розділ називається "Теоретичні основи проектування баз даних". В цьому розділі студент вказує мету проектування бази даних (БД) та наводить основні теоретичні положення для визначення складу даних, призначених

збереженню в базі даних та забезпеченню виконання запитів користувачів, для здійснення їх аналізу та структуризації.

Процес проектування БД описується як послідовність переходів від неформального словесного опису інформаційної структури предметної області до формалізованого опису об'єктів предметної області в термінах деякої моделі та фізичної реалізації бази даних в середовищі вибраної СУБД. З цією метою дається коротка характеристика таких основних етапів проектування як:

5. Системний аналіз і словесний опис інформаційних об'єктів предметної сфери.
6. Проектування інфологічної моделі предметної області.
7. Даталогічне або логічне проектування БД, тобто опис БД в термінах прийнятої даталогічної моделі даних (реляційна, ієрархічна, сіткова).
8. Фізичне проектування БД, тобто вибір ефективного розміщення БД на зовнішніх носіях для забезпечення найбільш ефективної роботи додатку.

Теоретичні викладки можна доповнювати узагальненими схемами (наприклад, "Етапи життєвого циклу бази даних", "ER-діаграми інфологічної моделі бази даних").

Другий розділ називається "Проект бази даних інформаційної системи..." з указівкою назви інформаційної системи відповідно варіанту. Цей розділ містить такі пункти:

1. Постановка задачі
2. Передпроектний аналіз предметної області
3. Концептуальна інфологічна модель бази даних
4. Проектування реалізації бази даних
5. Фізичне проектування бази даних

В пункті "Постановка задачі" наводяться інформаційні потреби користувачів інформаційної системи, характеризуються інформаційні потоки при вирішенні задач предметної області, формуються зведені інформаційні таблиці та таблиці, що містять значення реквізитів документів, які циркулюють в предметній області.

В пункті "Передпроектний аналіз предметної області" аналізується інформаційна залежність всіх служб чи окремих користувачів. Цю залежність можна відобразити функціональною схемою. Далі складається список інформаційних документів, що обробляються вручну, результати оформляються таблицею "Список інформаційних документів". Розробляється схема існуючого технологічного процесу оброблення інформації при вирішенні задач предметної області. Наступним кроком буде формування у вигляді таблиці родо-видового списку елементів даних вхідних та вихідних документів. Таблиця містить такі колонки:

Номер п/п	Назва елемента	Фактичний/ обчислюваний	Призначення
-----------	----------------	----------------------------	-------------

На основі отриманого списку елементів даних заповнюється таблиця "Словник даних", де вказуються всі виявлені інформаційні одиниці:

Номер п/п	Назва елемента	Ідентифікатор	Тип даних, довжина	Призначення
-----------	----------------	---------------	-----------------------	-------------

На основі сформованого інформаційного списку, словника даних та аналізу існуючого технологічного процесу оброблення інформації на фірмі (установі, службі і т.п.) виділяються локальні задачі виконання окремих функцій у підсистемі, що проектується. На основі виділених задач формується схема існуючого технологічного процесу оброблення інформації з розбиттям його на окремі задачі. Результати аналізу предметної області зводяться в таблицю зв'язку "Задача—дані" :

Таблиця зв'язку "Задача—дані"

Номер п/п	Назва задачі	Тип	Частота виконання	Обсяг	Відділ	Елементи даних
-----------	--------------	-----	-------------------	-------	--------	----------------

В пункт "Передпроектний аналіз предметної області" формулюються також вимоги до БД. Забезпечення БД необхідними властивостями відбувається на різних етапах її проектування. Зведені воедино вимоги до БД є стратегічним

планом для системного аналітика. Такими вимогами є функціональна повнота, узгодженість, відновлюваність, безпечність, ефективність та ін. Вимоги до оброблення даних ґрунтуються інформаційними запитамі потенційних користувачів ІС. Для кожного користувача розробляються ескізи екранних форм введення та відображення інформації.

В пункті "Концептуальна інфологічна модель бази даних" проектується локальні інформаційні структури та здійснюється їх об'єднання. Локальні подання потенційних користувачів розроблюваної ІС практично збігаються з задачами, виділеними в пункті "Передпроектний аналіз предметної області". Інформаційні структури подаються ER-моделями (схеми, на яких вказуються інформаційні сутності, атрибути цих сутностей та зв'язки між ними). При об'єднанні локальних інформаційних структур із ER-діаграм вилучають елементи даних, що дублюються. В кожному зв'язку залишають лише йому притаманні атрибути. В результаті об'єднання локальних ER-діаграм отримують структуру глобальної ER-діаграми.

Далі в зазначеному пункті обґрунтовують вибір системи управління базами даних. Система управління базами даних — програмна система, призначена для створення на ЕОМ спільної для багатьох додатків БД, підтримання її в актуальному стані і забезпечення ефективного доступу користувачів до даних. Основними засобами СУБД є: - засоби задання (опису) структури БД; - засоби конструювання екранних форм, призначених для введення, перегляду й оброблення даних у діалоговому режимі; - засоби створення запитів для вибірки даних за заданими умовами, а також їх оброблення; - засоби створення звітів з БД для виведення на друк результатів оброблення; - мовні засоби — макроси, вбудована алгоритмічна мова, мова запитів, які використовуються для реалізації нестандартних алгоритмів оброблення даних; - засоби створення додатків користувача (генератори додатків, засоби створення меню і панелі керування додатками), що дають змогу об'єднати різні операції з БД в єдиний технологічний процес.

Відповідно до моделі та структур даних, які підтримують СУБД, організують БД.

Вибір СУБД доцільно здійснювати не тільки за типом архітектури та якістю зовнішнього інтерфейсу, а й передусім на основі функціональних можливостей. Суттєвим критерієм вибору є здатність оброблення складних запитів (і швидкість оброблення), можливість перенесення даних між платформами.

Дається коротка характеристика вибраної СУБД (наприклад, СУБД MS Access).

Після вибору СУБД наявна модель доповнюється додатковими локальними інформаційними структурами, пов'язаними з запитом потенційних користувачів системи. Вони зображуються у вигляді ER-діаграм, які доповнюють ER-діаграму, створену на етапі концептуального проектування БД. Одержана в даному разі інфологічна модель предметної області не відображає тільки можливий запит адміністратора БД, пов'язаний з її власною структурою.

У найпростішому випадку адміністратора БД цікавить відповідь на запит: "Список імен таблиць (з зазначенням місця їх розташування, кількості записів, розмірів у байтах, дат створення, імен розробників), в яких є поле з іменем "XXXXXXXXXX". Такий запит актуальний, коли є потреба змінити структуру існуючої таблиці БД. Для реалізації такого запиту може бути використана екранна форма, зображена на рис.31.

З цією метою створюється словник БД, який можна назвати БД над БД. ER-діаграму словника БД показано на рис.32. Тут виділяють типи сутностей "Таблиця БД" та "Поле", які мають таку специфікацію.

ТАБЛИЦЯ БД (Ім'я таблиці, Кількість логічних записів, Кількість байтів, Дата створення, Ім'я розробника).

Поле: XXXXXXXXXXXX включено в такі таблиці				
Ім'я таблиці	Кількість записів	Кількість байт	Дата розробки	Ім'я розроблювача
xxxxxxxxxx	9999	99999999	мм/дд/рр	xxxxxxxxxxxxxx
xxxxxxxxxx	9999	99999999	мм/дд/рр	xxxxxxxxxxxxxx
xxxxxxxxxx	9999	99999999	мм/дд/рр	xxxxxxxxxxxxxx
xxxxxxxxxx	9999	99999999	мм/дд/рр	xxxxxxxxxxxxxx

Рисунок 31 - Ескіз екранної форми запиту до словника БД

ПОЛЕ ТАБЛИЦІ (Ім'я поля, Тип, Довжина, Точність). Зв'язок "Включення" між типами сутностей не має власних атрибутів.

Словник даних поповнюється новими елементами — даними про дані.

В пункті "Проектування реалізації бази даних" здійснюється формулювання СУБД-орієнтованої логічної структури, або даталогічної концептуальної моделі БД. На цьому етапі виконується перетворення інфологічної моделі на даталогічну модель. Наявність зв'язків між сутностями типу Б:Б дає змогу подати ER-діаграму у вигляді складної мережної моделі даних. Для усунення зв'язку типу Б:Б у модель даних вводять додаткові записи, створюються додаткові відношення. Дані перетину, що належать обом типам сутностей, будуть атрибутами додаткових записів. Завдяки цьому складна мережа зводиться до простої мережної моделі. Даталогічна реляційна модель, або схема БД, складається з набору відношень, що не зводяться.

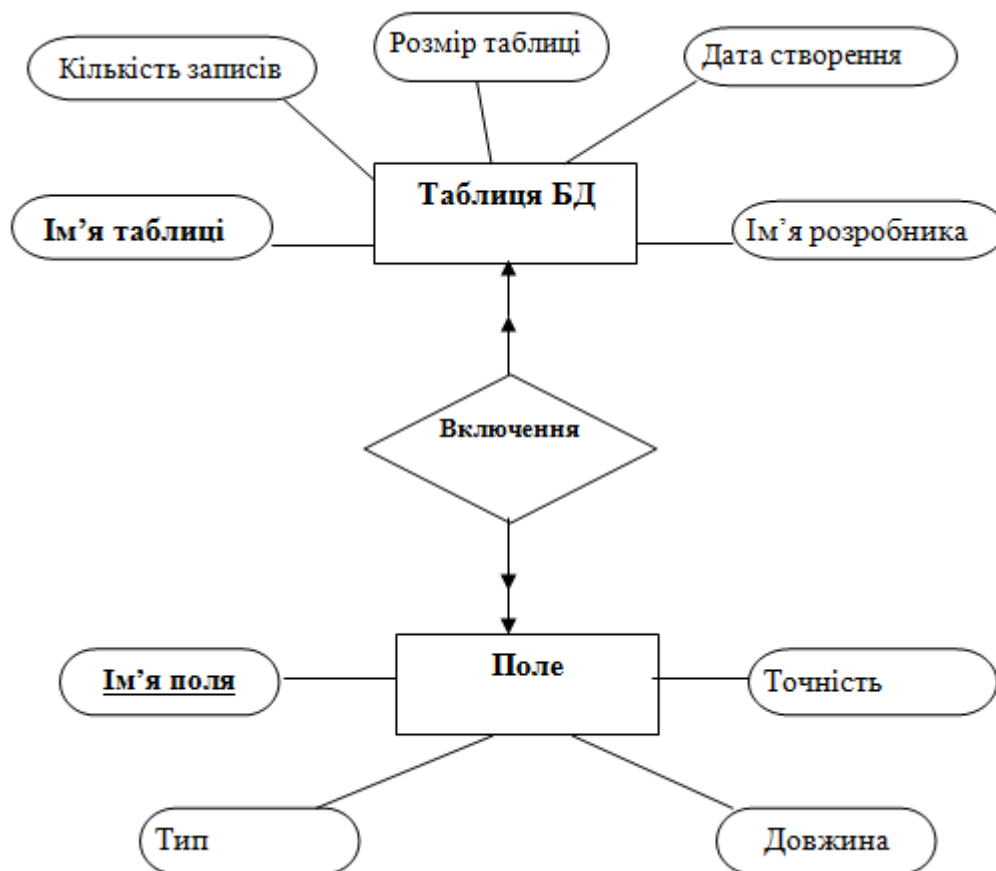


Рисунок 32 - ER-діаграму словника БД

В пункті "Фізичне проектування бази даних" вказується, що структура таблиць даних відповідає реляційній схемі та словнику даних, наведених в попередніх пунктах.

Вибравши середовище реалізації СУБД Access, створюється файл бази даних та відповідні об'єкти. Наводиться схема даних, копії екранних форм, приклади звітів, запитів та текст модуля головної екранної форми.

У висновку вказується які властивості характеризують спроектовану базу даних та що забезпечують впровадження інформаційної системи.

Додаток може містити приклади звітів, запитів, копії первинних документів, значення реквізитів яких заносяться в базу даних.

Список використаних джерел

1. Шпортько О.В. Розробка баз даних в СУБД Microsoft Access 2010/2013. Практикум: Навчальний посібник для студентів вищих навчальних закладів / О.В. Шпортько, Л.В. Шпортько. -Рівне: О. Зень, 2016. – 184 с.
2. Тарасов О.В. Проектування баз даних : навч. посіб. / О.В. Тарасов, В.В. Федько, М.Ю. Лосєв. – Х. : Вид. ХНЕУ, 2011. – 200 с.
3. Лобок О.П. Організація баз даних та знань. Теоретичні основи проектування, реалізації та використання баз даних : навч. посіб. / О.П. Лобок. – К.: НУХТ, 2013. – 262 с.
4. В.М. Юрчишин, Б.В. Клим, В.Б. Кропивницька. Організація баз даних. Навчальний посібник – Івано-Франківськ: "Факел", 2010. – 224 с.
5. 3. Гайдаржи В.І., Ізварін І.В. Бази даних в інформаційних системах. – К.; Університет «Україна», 2018. – 418 с.
6. Карпова И. Базы данных. Учебное пособие. СПб.: «Питер», 2013. – 240 с.
7. Конноли Т., Бегг К. Базы данных. Проектирование, реализация и сопровождение. Теория и практика. – К.: Вильямс, 2016. – 1440 с.
8. Пасічник В. В., Резніченко В. А. Організація баз даних та знань: підручник для ВНЗ. – К.: Видавнича група ВНУ, 2006. – 384с.
9. Гайна Г. А. Основи проектування баз даних: Навчальний посібник. – К: КНУБА, 2005. – 204 с.
10. Яргер Р.Дж., Риз Дж., Кинг Т. MySQL и mSQL: Базы данных для небольших предприятий и Интернета; СПб: Символ- Плюс, 2013. – 560 с.
11. Тімонін В.О. Конспект лекцій з дисципліни «Основи систем баз даних» для студентів за напрямом підготовки 6.050201 Системна інженерія. – Харків:

- ХНАДУ, 2016. – 88 с. (в електронній формі).
12. Тімонін В.О. Методичні вказівки до проведення лабораторних робіт з дисципліни «Основи систем баз даних» для студентів за напрямом підготовки 6.050201 Системна інженерія. – Харків: ХНАДУ, 2016. – 118 с. (в електронній формі).
 13. Буй Д.Б. Теорія нормалізації в реляційних базах даних: сучасний стан / Д.Б. Буй, А.В. Пузікова [Електронний ресурс]. – Режим доступу: http://ekmair.ukma.edu.ua/bitstream/handle/123456789/8935/Bui_Teoriia_normalizatsii_v_reliatsiinykh.pdf
 14. <https://tiacher.org.ua/>
 15. <http://www.znannya.org/>
 16. Бази даних в проектуванні і реалізації інформаційних систем https://stud.com.ua/77194/informatika/bazi_danih_proektuvanni_realizatsiyi_informatsiynih_sistem
 17. Бази даних та інформаційні системи: <https://wiki.cuspu.edu.ua/index.php>
 18. http://msoffice.if.ua/Zvitu_Access/Index07.htm

ДОДАТОК Приклад виконання курсової роботи

БАЗИ ДАНИХ ТА СУБД

**Методичні вказівки щодо
виконання курсової роботи**

Суми, РВВ, Сумський національний аграрний університет,
вул. Г. Кондрат'єва, 160

Підписано до друку: _____ 2022 р. Формат А5: Гарнітура TimesNewRoman

Тираж 100 примірників

Замовлення

Ум. друк.арк 1.3
