

Міністерство освіти і науки України
УКРАЇНСЬКА ІНЖЕНЕРНО-ПЕДАГОГІЧНА АКАДЕМІЯ

І. О. Бардус, М. І. Лазарєв, А. О. Ніценко

БАЗИ ДАНИХ
У СХЕМАХ
(НА ОСНОВІ ФУНДАМЕНТАЛІЗОВАНОГО ПІДХОДУ)

Навчальний посібник
для студентів спеціальностей:

- 015.10 Професійна освіта (Комп'ютерні технології)
- 015.02 Професійна освіта (Видавничо-поліграфічна справа)
- 121 Інженерія програмного забезпечення
- 122 Комп'ютерні науки та інформаційні технології
- 123 Комп'ютерна інженерія
- 124 Системний аналіз
- 125 Кібербезпека

Харків
Видавництво «Діса плюс»
2017

УДК 004.6(075.8)
Б24

Схвалено Вченою радою
Української інженерно-педагогічної академії
Протокол № 2 від 26.09.2017 р.

Рецензенти:

В. Г. Хоменко, доктор педагогічних наук, професор, завідувач кафедри комп'ютерних технологій в управлінні та навчанні й інформатики Бердянського державного педагогічного університету;

О. П. Нечуйвітер, доктор фізико-математичних наук, професор, завідувач кафедри інформаційних комп'ютерних та поліграфічних технологій Української інженерно-педагогічної академії.

Бардус І. О.

Б24 Бази даних у схемах (на основі фундаменталізованого підходу) : навч. посіб. /
І. О. Бардус, М. І. Лазарев, А. О. Ніценко. – Харків : Вид-во «Діса плюс», 2017. – 133 с.

ISBN 978-617-7384-77-8

В посібнику в компактній та наочній формі схем представлено головні концепції проектування і побудови баз даних, основні моделі баз даних, архітектури систем керування базами даних, сучасні напрямки розвитку технологій баз даних, питання, які пов'язані з експлуатацією баз даних.

Навчальний посібник орієнтований на використання в навчальному процесі в якості допоміжного засобу до лекційного, практичного, лабораторного забезпечення, а також для самостійної роботи, підготовки до контрольних робіт, заліків, іспитів, курсового та дипломного проектування.

Видання призначене студентам спеціальностей: 015.10 Професійна освіта (Комп'ютерні технології), 015.02 Професійна освіта (Видавничо-поліграфічна справа), 121 Інженерія програмного забезпечення, 122 Комп'ютерні науки та інформаційні технології, 123 Комп'ютерна інженерія, 124 Системний аналіз, 125 Кібербезпека, а також може бути корисним викладачам, розробникам баз даних, прикладним програмістам, всім бажаючим підвищити свою кваліфікацію в галузі інформаційних технологій.

© Бардус І. О., Лазарев М. І., Ніценко А. О., 2017

© УІПА, 2017

ЗМІСТ

УМОВНІ ПОЗНАЧЕННЯ.....	9
ВСТУП.....	10
ТЕМА 1. ВСТУП ДО БАЗ ДАНИХ	13
1.1. Поняття інформації, предметної області та інформаційної системи.....	14
1.2. Поняття бази даних	15
1.2.1. Визначення поняття бази даних на основі філософських категорій «загальне», «особливе», «конкретне»	15
1.2.2. Семантична модель поняття база даних.....	16
1.3. Поняття системи керування базою даних.....	17
1.3.1. Визначення поняття системи керування базою даних на основі філософських категорій «зміст» і «форма»	17
1.3.2. Трирівнева архітектура систем керування базами даних ANSI-SPARC.....	18
1.3.3. Зв'язки «загальне», «конкретне», «особливе» в структурі системи керування базами даних	19
1.3.4. Причинно-наслідкові зв'язки між предметною областю, базою даних та системою керування базою даних.....	20
1.3.5. Семантична модель поняття системи керування базою даних	21
Контрольні питання.....	22
Завдання для самостійної роботи	22
ТЕМА 2. ОСНОВНІ МОДЕЛІ ПРЕДСТАВЛЕННЯ ДАНИХ У БАЗАХ ДАНИХ.....	23
2.1. Класифікація моделей баз даних	24
2.2. Вибір моделі бази даних	25
2.3. Ієрархічна модель бази даних	26
2.3.1. Основні поняття та приклад побудови ієрархічної моделі бази даних.....	26
2.3.2. Переваги та недоліки представлення даних у вигляді ієрархічної моделі бази даних	27
2.3.3. Семантична модель поняття ієрархічної бази даних.....	28
2.4. Мережева модель бази даних	29

2.4.1. Основні поняття та приклад побудови мережевої моделі бази даних.....	29
2.4.2. Переваги та недоліки представлення даних у вигляді мережевої моделі бази даних.....	30
2.4.3. Семантична модель поняття мережевої бази даних.....	31
2.5. Модель даних «Сутність-Зв'язок».....	32
2.5.1. Основні поняття та приклад побудови моделі даних «Сутність-Зв'язок».....	32
2.5.2. Переваги та недоліки представлення даних у вигляді моделі даних «Сутність-Зв'язок».....	33
2.5.3. Семантична модель поняття моделі даних «Сутність-Зв'язок».....	34
2.6. Реляційна модель бази даних.....	35
2.6.1. Основні поняття та приклад побудови реляційної моделі бази даних.....	35
2.6.2. Переваги та недоліки представлення даних у вигляді реляційної моделі бази даних.....	36
2.6.3. Семантична модель поняття реляційної бази даних.....	37
2.7. Об'єктно-орієнтована модель бази даних.....	38
2.7.1. Основні поняття та приклад побудови об'єктно-орієнтованої моделі бази даних.....	38
2.7.2. Переваги та недоліки представлення даних у вигляді об'єктно-орієнтованої моделі бази даних.....	39
2.7.3. Семантична модель поняття об'єктно-орієнтованої бази даних.....	40
2.8. Документно-орієнтована модель бази даних.....	41
2.8.1. Основні поняття та приклад побудови документно-орієнтованої моделі бази даних.....	41
2.8.2. Переваги та недоліки представлення даних у вигляді документно-орієнтованої моделі бази даних.....	42
2.8.3. Семантична модель поняття документно-орієнтованої бази даних.....	43
Контрольні питання.....	44
Завдання для самостійної роботи.....	45

ТЕМА 3. ПРОЕКТУВАННЯ ТА РОЗРОБКА БАЗ ДАНИХ НА КОНЦЕПТУАЛЬНОМУ РІВНІ	46
3.1. Етапи життєвого циклу інформаційних систем.....	47
3.2. Етапи життєвого циклу бази даних.....	48
3.3. Семантична модель поняття технології розробки та використання баз даних	49
3.4. Етапи планування баз даних	50
3.5. Концептуальне (інфологічне) проектування баз даних	51
3.6. Моделювання предметної області на основі моделі «Сутність- Зв'язок» (ER-моделювання).....	52
3.6.1. Типи сутностей.....	52
3.6.2. Атрибути.....	53
3.6.3. Ключі.....	54
3.6.4. Зв'язки	55
3.6.5. Структурні обмеження	56
3.6.6. Розширена ER-модель (EER)	57
Контрольні питання.....	58
Завдання для самостійної роботи	59
ТЕМА 4. ПРОЕКТУВАННЯ ТА РОЗРОБКА БАЗ ДАНИХ НА ЛОГІЧНОМУ РІВНІ	60
4.1. Логічне (дatalogічне) моделювання баз даних	61
4.2. Реляційна алгебра та числення.....	62
4.2.1. Реляційні алгебра. Операції реляційної алгебри	62
4.2.2. Перейменування, вибірка, проєкція.....	63
4.2.3. Об'єднання відношень	64
4.2.4. Різниця відношень	65
4.2.5. Декартів добуток.....	66
4.2.6. Перетин відношень.....	67
4.2.7. З'єднання відношень	68
4.2.8. Природне з'єднання.....	69
4.2.9. Тета-з'єднання	70
4.2.10. Напівз'єднання.....	71
4.2.11. Антисз'єднання	72

4.2.12. Зовнішнє з'єднання	73
4.2.13. Ділення відношень.....	74
4.3. Реляційна модель бази даних.....	75
4.3.1. Основні поняття реляційної моделі даних.....	75
4.3.2. Відношення, властивості відношень	76
4.3.3. Представлення	77
4.3.4. Реляційні ключі	78
4.3.5. Реляційна цілісність.....	79
4.4. Перетворення концептуальної моделі бази даних у логічну	80
4.4.1. Сильна сутність, слабка сутність, зв'язок 1:М.....	80
4.4.2. Зв'язок 1:1	81
4.4.3. Зв'язок N:М та багатозначний зв'язок	82
4.4.4. Складний зв'язок	83
4.4.5. Зв'язок клас - підклас (Mandatory/And)	84
4.4.6. Зв'язок клас - підклас (Optional/Or)	85
4.4.7. Зв'язок клас - підклас (Mandatory/Or).....	86
4.4.8. Зв'язок клас - підклас (Optional/And).....	87
4.5. Нормалізація	88
4.5.1. Функціональні залежності між відношеннями	88
4.5.2. Поняття нормалізації відношень у реляційній моделі даних.....	89
4.5.3. Ненормалізована та перша нормальна форма (1НФ).....	90
4.5.4. Повна функціональна залежність та друга нормальна форма (2НФ)	91
4.5.5. Транзитивна залежність та третя нормальна форма (3НФ).....	92
4.5.6. Нормальна форма Бойса-Кодда, четверта та п'ята нормальні форми (НФБК, 4НФ, 5НФ).....	93
Контрольні питання.....	94
Завдання для самостійної роботи	95
ТЕМА 5. ПРОЕКТУВАННЯ ТА РОЗРОБКА БАЗ ДАНИХ НА ФІЗИЧНОМУ РІВНІ	96
5.1. Фізичне проектування баз даних.....	97
5.2. Мова запитів до баз даних SQL.....	98

5.2.1. Загальна структура мови	98
5.2.2. Нотація посібника для лексичних діаграм SQL	99
5.2.3. Основні типи даних SQL	100
5.2.4. Команда створення таблиць.....	101
5.2.5. Команда оновлення структури таблиці	102
5.2.6. Команди перейменування, видалення таблиці та видалення її змісту	103
5.2.7. Команда вибірки даних з таблиці	104
5.2.8. Команди вставки, оновлення та видалення даних з таблиці.....	106
Контрольні питання.....	107
Завдання для самостійної роботи	107
ТЕМА 6. СИСТЕМИ КЕРУВАННЯ БАЗАМИ ДАНИХ.....	109
6.1. Функції системи керування базами даних.....	110
6.2. Структура системи керування базами даних	111
6.3. Паралельна обробка даних. Транзакції	112
6.3.1. Поняття транзакції.....	112
6.3.2. Керування паралельним доступом	113
6.3.3. Рівні ізолюваності транзакцій	114
6.3.4. Відновлення бази даних.....	115
6.3.5. Механізми відновлення баз даних після збоїв	116
6.4. Розподілені бази даних та системи керування базами даних .	117
6.4.1. Основні поняття.....	117
6.4.2. Структурна схема розподіленої бази даних	118
6.4.3. Приклад архітектури розподіленої СКБД.....	119
6.5. Технологія обробки інформації OLAP	120
6.6. Технологія обробки транзакцій OLTP	121
6.7. Сховища даних	122
6.8. Ретроспективний аналіз розвитку баз даних та систем керування базами даних	123
6.9. Перспективи розвитку баз даних та систем керування базами даних.....	125
Контрольні питання.....	126

Завдання для самостійної роботи	127
ПРЕДМЕТНИЙ ПОКАЖЧИК	128
БІБЛІОГРАФІЧНИЙ СПИСОК	132
Основна використана література	132
Рекомендована література для поглибленого вивчення курсу	133

УМОВНІ ПОЗНАЧЕННЯ



Мета



Зміст



Контрольні питання



Завдання для самостійної роботи



Приклад



Умовна нотація



Модель даних



Технологія розробки



Система керування базами даних



Філософські засади



Математичні засади



Семантична модель

ВСТУП

На сучасному етапі інформаційного розвитку суспільства бази даних стали невід'ємною частиною нашого повсякденного життя. Вони введені й успішно використовуються практично в усіх галузях людської діяльності. Однією з найбільш розповсюджених задач, що виникає при користуванні комп'ютером, є використання інформаційних систем, заснованих на базах даних. Тому при підготовці майбутніх ІТ-фахівців навчальна дисципліна «Бази даних» є фундаментальною та обов'язковою для вивчення.

Метою курсу «Бази даних» є засвоєння основних знань з проектування й організації баз даних (БД), вивчення мови структурованих запитів SQL, ознайомлення з сучасними системами управління БД (СКБД), набуття практичних умінь і навичок проектування, розроблення й експлуатації баз даних та інформаційних систем, створених на їхній основі.

Даний навчальний посібник розроблено відповідно до останніх тенденцій в професійній підготовці майбутніх ІТ-фахівців на основі фундаменталізованого підходу.

Для спрощення сприйняття великого обсягу різнопланових понять з баз даних та систем керування базами даних навчальний матеріал посібника структуровано за тріадою ключових понять курсу: об'єкт – технологія його розробки – інструмент (рис. 1).

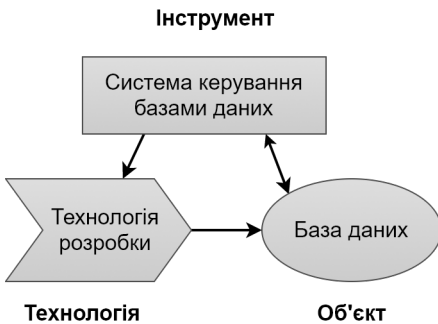


Рис. 1. Тріада ключових понять навчальної дисципліни «Бази даних»

Зміст даного навчального посібника повністю відповідає державним стандартам підготовки бакалаврів зі спеціальностей: 015.10 «Професійна освіта (Комп'ютерні технології)», 015.02 «Професійна освіта. Видавничо-поліграфічна справа», 121 «Інженерія програмного забезпечення», 122 «Комп'ютерні науки та інформаційні технології», 123 «Комп'ютерна інженерія», 124 «Системний аналіз», 125 «Кібербезпека».

Структура змісту посібника побудована на основі принципу поетапної деталізації. Для формування початкового цілісного уявлення

про бази даних, системи керування БД та технології їх розробки в першій та другій темах посібника наведено у стислому вигляді із зазначенням зв'язків між ними за допомогою філософських категорій «матеріальне» та «ідеальне», «одиничне» і «загальне», «елемент» і «система», «явище» і «сутність», «зміст» і «форма», «причина» і «наслідок», «можливість» і «дійсність». У наступних темах ці поняття і зв'язки між ними поступово деталізуються.

Фундаменталізований підхід до навчання баз даних покликаний для забезпечення ефективної професійної підготовки майбутніх ІТ-фахівців до продуктивної професійної діяльності. Важливу роль при пошуку нових технічних рішень удосконалення або створення нових об'єктів професійної діяльності ІТ-фахівця відіграють знання закономірностей розвитку цих об'єктів в ретроспективі, вміння їх аналізувати і використовувати для виявлення резервів їх розвитку, визначення доцільності вдосконалення або створення принципово нових технічних або програмних систем. Тому значну увагу в посібнику приділено ретроспективному аналізу розвитку баз даних та систем керування ними. Системний ретроспективний аналіз БД та СКДБ дозволяє визначити основні причини виникнення (нові характеристики) кожного покоління із зазначенням нового призначення, структури, принципу дії (математичні закони, на основі яких будуються моделі даних), та згодом визначити перспективи їх розвитку.

Тому ключові поняття «база даних», «модель даних», «система керування базою даних», «технологія розробки баз даних» представлені в посібнику у вигляді моделі на основі їх семантичних ознак призначення (R), складу (S), принципів дії (D) та характеристик (H) (рис. 2).

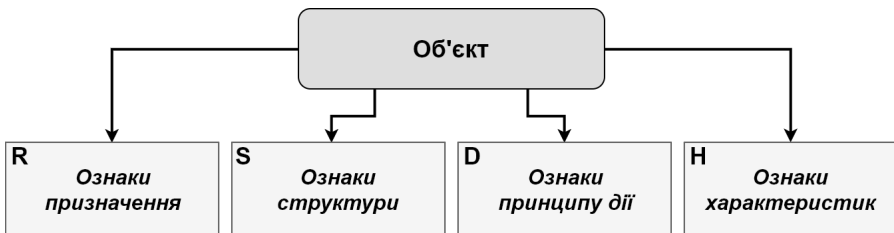


Рис. 2. Семантична модель поняття

Головною особливістю навчального посібника є представлення навчального матеріалу у вигляді структурно-логічних схем, що покращує усвідомлення, запам'ятовування та формування продуктивного професійного мислення ІТ-фахівців. Для спрощення користування навчальними матеріалами посібник містить умовні позначення їх

структурних елементів: мети, змісту теми, контрольних питань, завдань для самостійної роботи, прикладів, умовних нотацій, філософських та математичних засад та ін.

Навчальний посібник орієнтований на використання в навчальному процесі в якості допоміжного засобу до лекційного, лабораторного забезпечення, а також для самостійної роботи, підготовки до контрольних робіт, заліків, іспитів, курсового та дипломного проектування.

В першу чергу посібник адресований студентам вищих навчальних закладів, які вивчають дисципліну «Бази даних» та викладачам, а також може бути корисним розробникам баз даних, прикладним програмістам, фахівцям у галузі інформаційних технологій, всім бажаючим підвищити свою кваліфікацію в галузі інформаційних технологій.

Тема 1

ВСТУП ДО БАЗ ДАНИХ



МЕТА

Оволодіння знаннями з основних понять та принципів побудови баз даних і систем керування базами даних

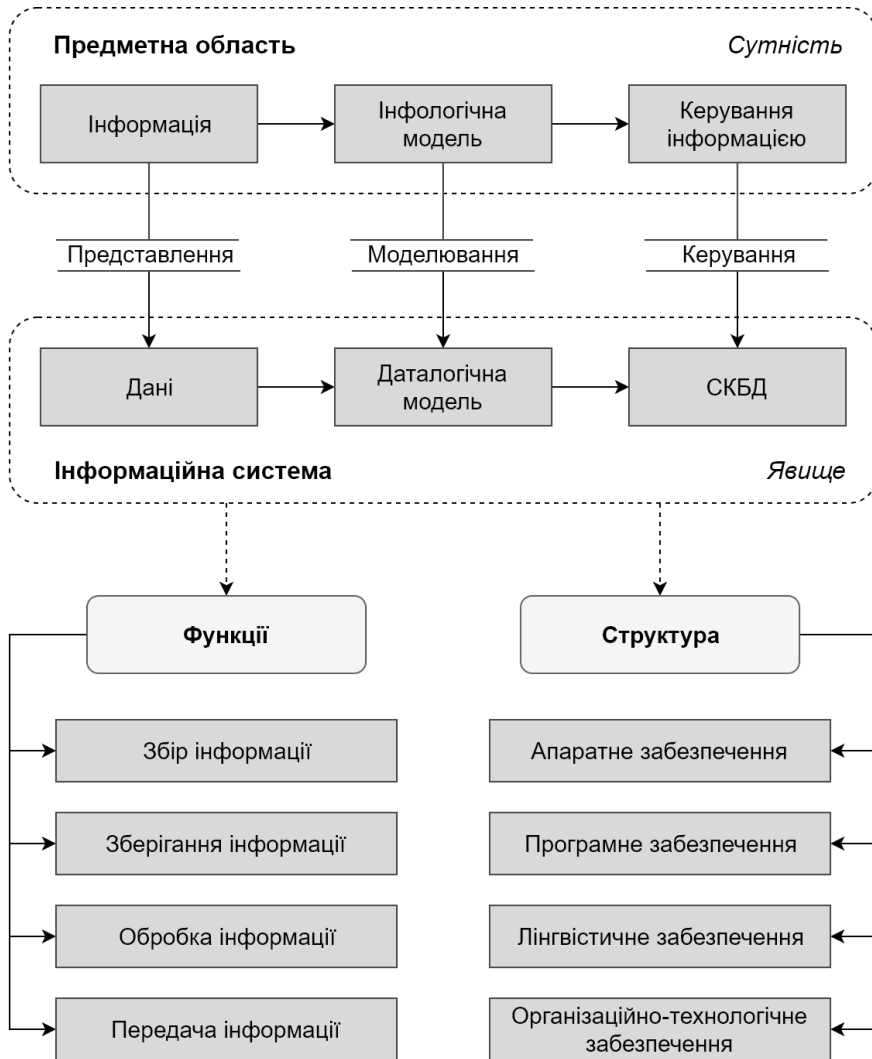


ЗМІСТ

- 1.1. Поняття інформації, предметної області та інформаційної системи**
- 1.2. Поняття бази даних**
 - 1.2.1. Визначення поняття бази даних на основі філософських категорій «загальне», «особливе», «конкретне»
 - 1.2.2. Семантична модель поняття бази даних
- 1.3. Поняття системи керування базою даних**
 - 1.3.1. Визначення поняття системи керування базою даних на основі філософських категорій «зміст» і «форма»
 - 1.3.2. Трирівнева архітектура системи керування базами даних ANSI-SPARC
 - 1.3.3. Зв'язки «загальне», «конкретне», «особливе» в структурі системи керування базами даних
 - 1.3.4. Причинно-наслідкові зв'язки між предметною областю, базою даних та системою керування базою даних
 - 1.3.5. Семантична модель поняття системи керування базою даних



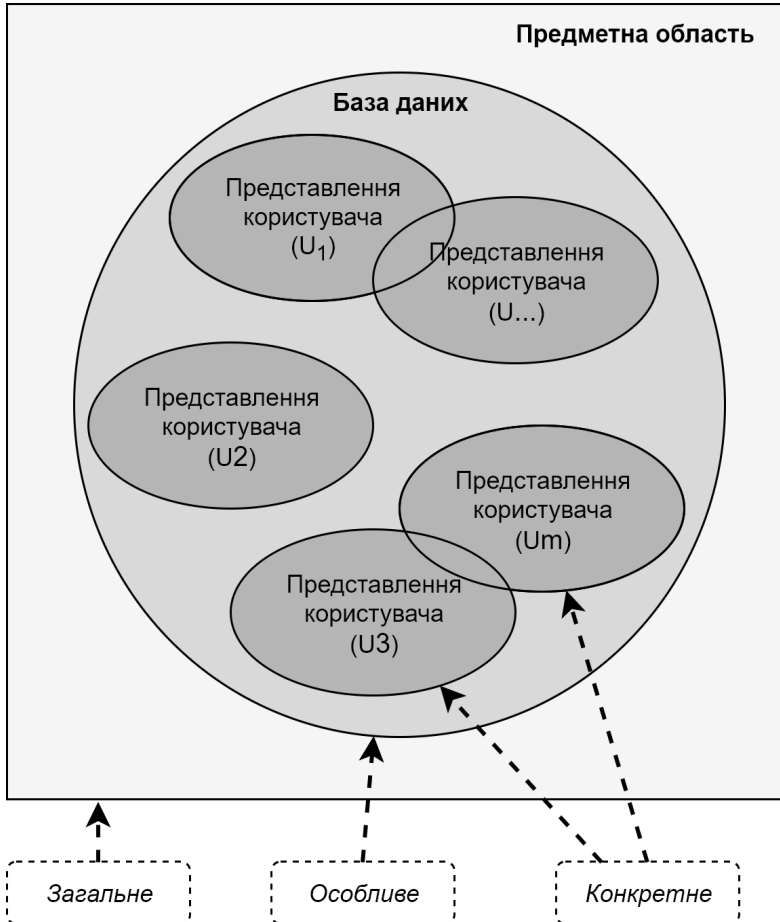
1.1. Поняття інформації, предметної області та інформаційної системи





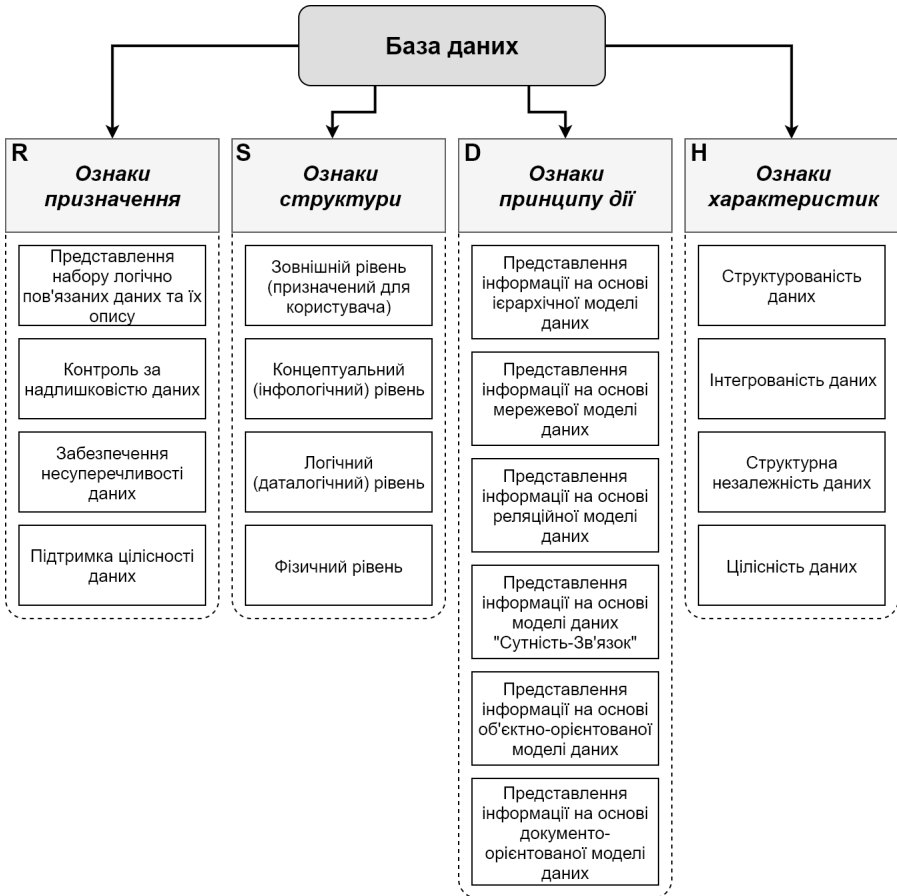
1.2. Поняття бази даних

1.2.1. Визначення поняття бази даних на основі філософських категорій «загальне», «особливе», «конкретне»





1.2.2. Семантична модель поняття бази даних

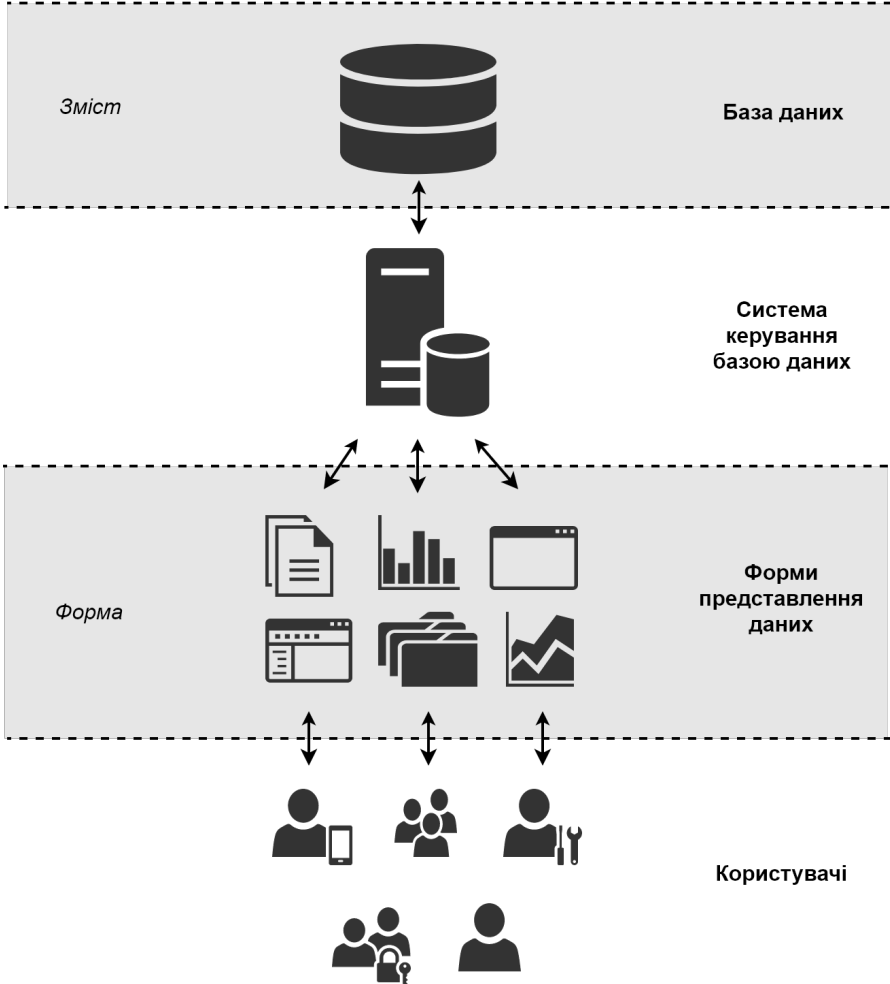




1.3. Поняття системи керування базою даних

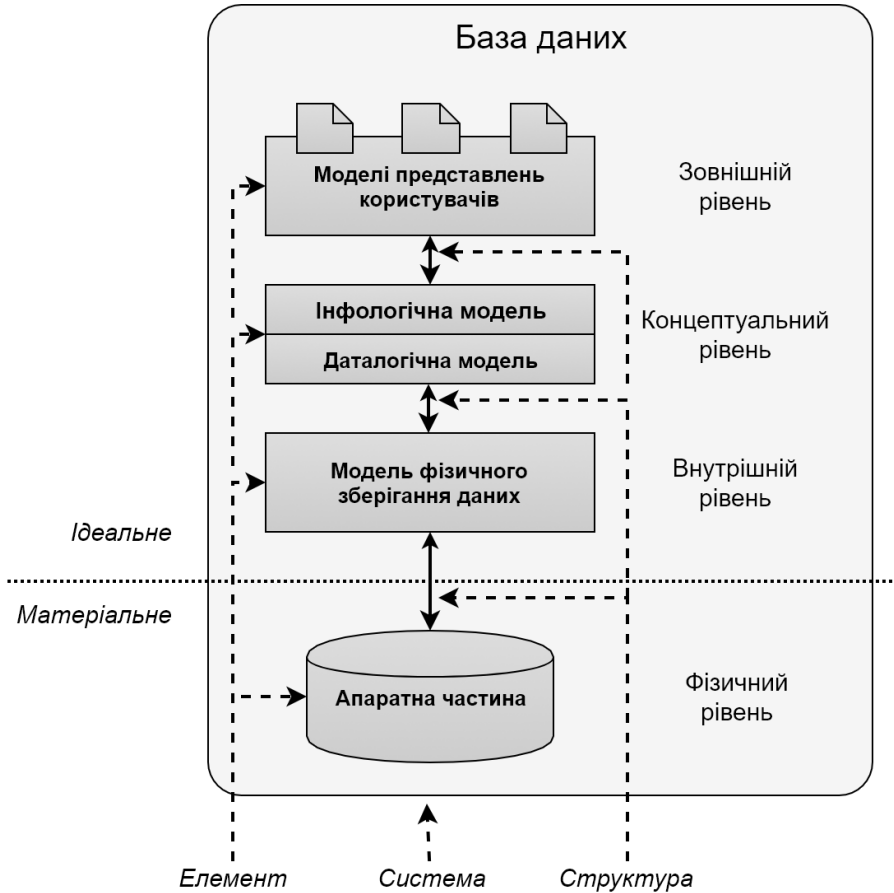


1.3.1. Визначення поняття системи керування базою даних на основі філософських категорій «зміст» і «форма»





1.3.2. Трирівнева архітектура систем керування базами даних ANSI-SPARC



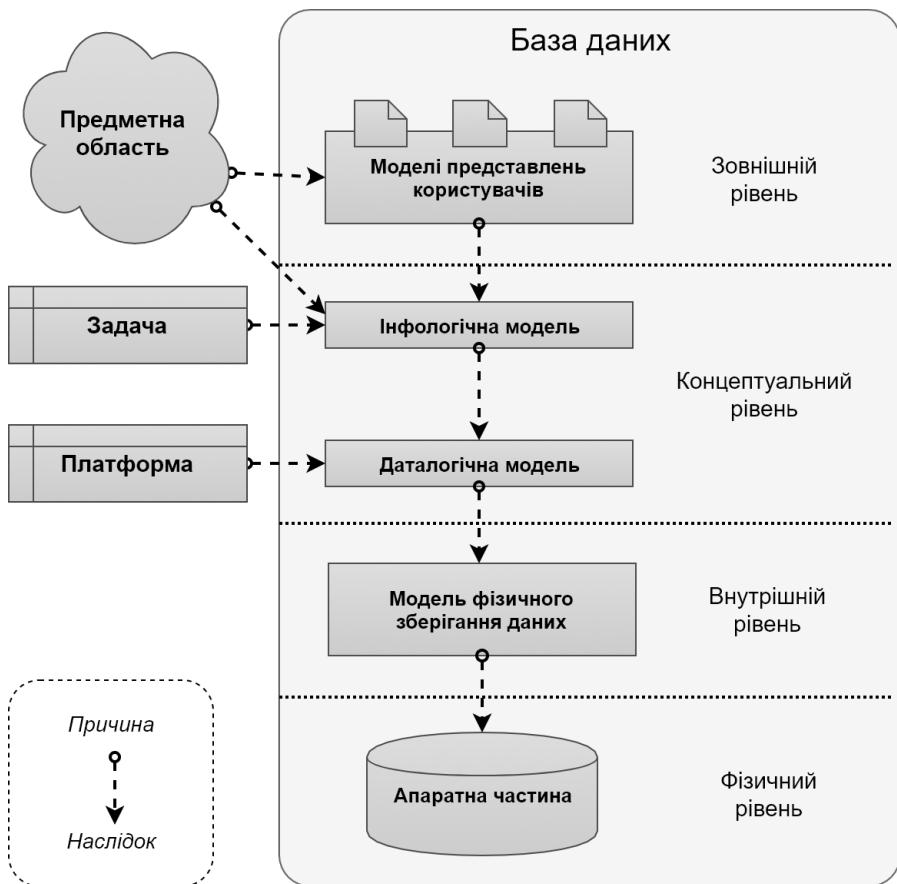


1.3.3. Зв'язки «загальне», «конкретне», «особливе» в структурі системи керування базами даних



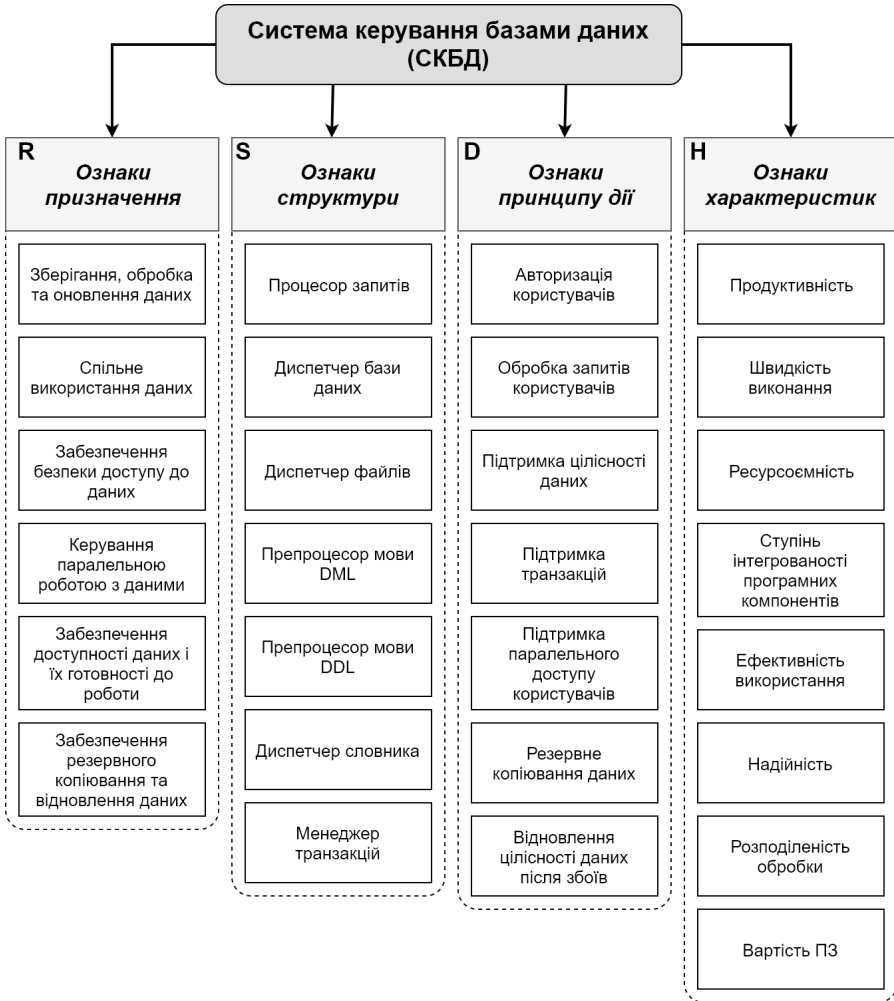


1.3.4. Причинно-наслідкові зв'язки між предметною областю, базою даних та системою керування базою даних





1.3.5. Семантична модель поняття системи керування базою даних





Контрольні питання

1. Дайте визначення поняттям: «інформація», «предметна область» та «інформаційна система».
2. Дайте визначення поняття «база даних».
3. Поясніть зв'язки між структурними компонентами бази даних на основі філософських категорій.
4. Назвіть призначення баз даних.
5. Назвіть структуру баз даних.
6. Назвіть принципи дії, на яких будуються бази даних.
7. Назвіть основні характеристики баз даних.
8. Дайте визначення поняття системи керування базою даних.
9. Поясніть зв'язки між структурними компонентами систем керування базами даних на основі філософських категорій.
10. Назвіть призначення систем керування базами даних.
11. Назвіть структуру систем керування базами даних.
12. Назвіть принципи дії, на яких будуються системи керування базами даних.
13. Назвіть основні характеристики систем керування базами даних.
14. Дайте визначення тривірневої архітектури системи керування базами даних ANSI-SPARC.



Завдання для самостійної роботи

1. Наведіть приклади предметних областей і прикладних задач, пов'язаних з цими областями.
2. Наведіть приклади різних представлень користувачів у межах однієї й тієї ж предметної області.
3. На основі аналізу інтернет-ресурсів назвіть сучасні системи керування базами даних. Які з них є найпопулярнішими?

Тема 2

ОСНОВНІ МОДЕЛІ ПРЕДСТАВЛЕННЯ ДАНИХ У БАЗАХ ДАНИХ



МЕТА

Оволодіння знаннями про класифікацію моделей представлення даних у базах даних, вміннями обирати модель бази даних залежно від предметної області та поставлених цілей

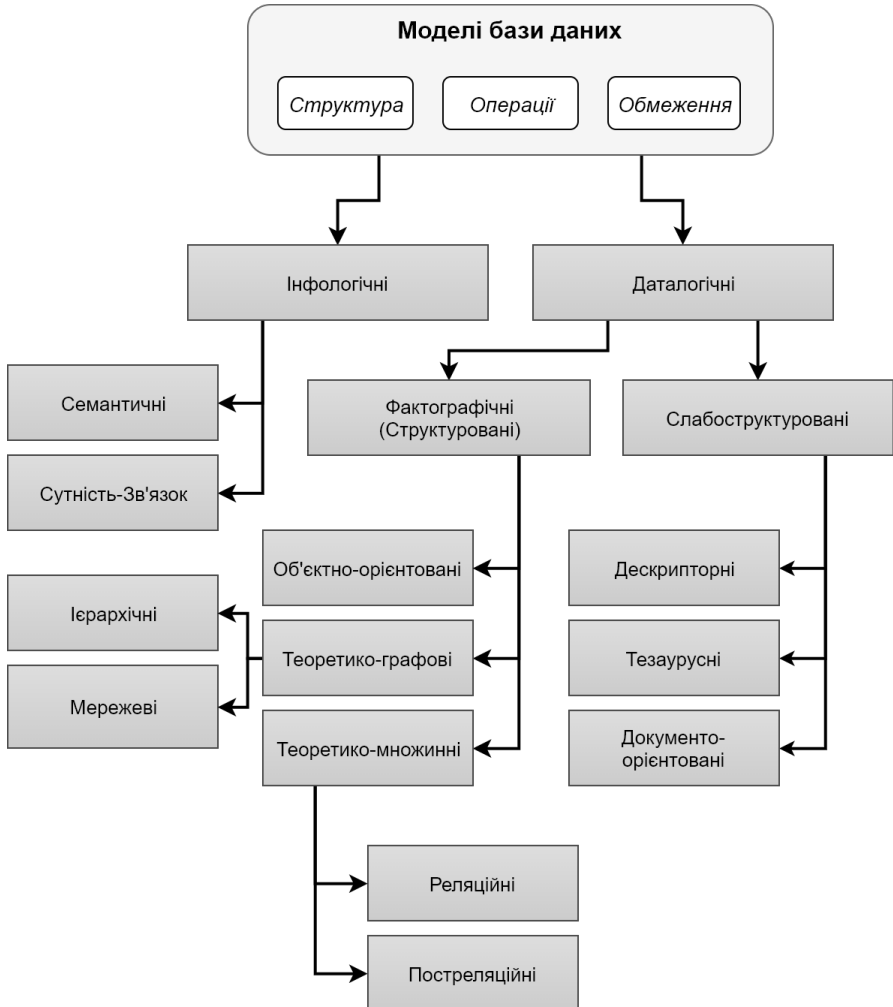


ЗМІСТ

- 2.1. Класифікація моделей баз даних
- 2.2. Вибір моделі бази даних
- 2.3. Ієрархічна модель бази даних
- 2.4. Мережева модель бази даних
- 2.5. Модель даних «Сутність-Зв'язок»
- 2.6. Реляційна модель бази даних
- 2.7. Об'єктно-орієнтована модель бази даних
- 2.8. Документно-орієнтована модель бази даних

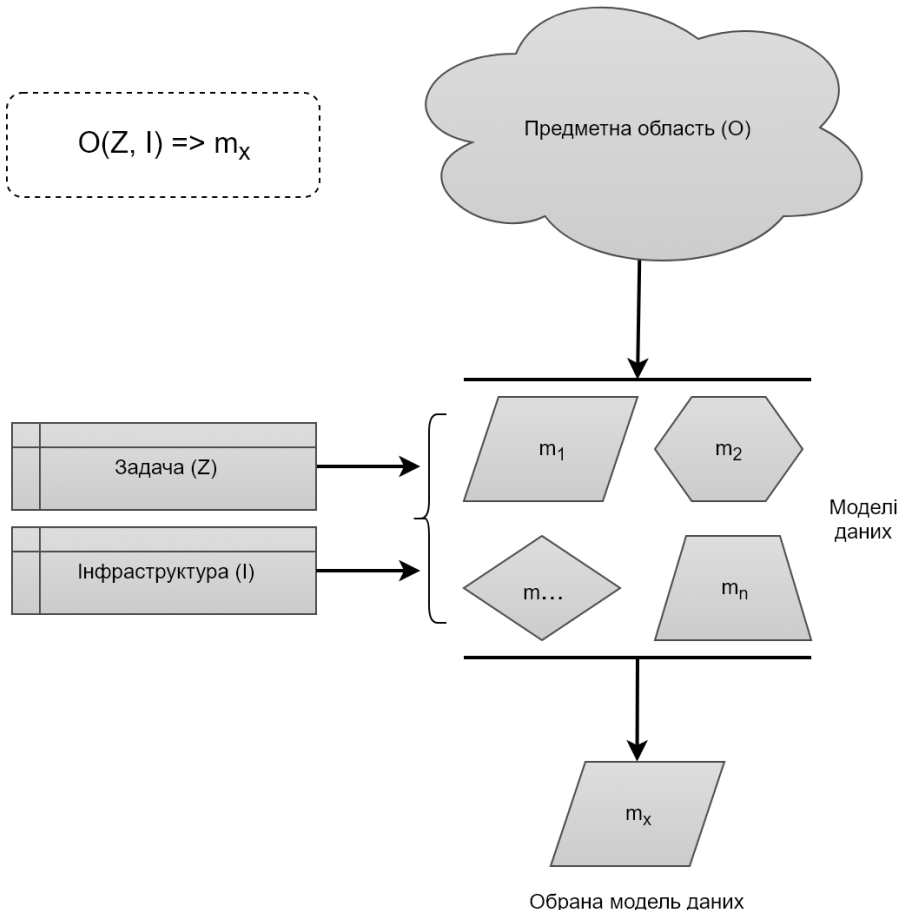


2.1. Класифікація моделей баз даних





2.2. Вибір моделі бази даних





2.3. Ієрархічна модель бази даних



2.3.1. Основні поняття та приклад побудови ієрархічної моделі бази даних

Нульовий рівень

Кореневий сегмент

є предком сегментів першого рівня

Сегменти першого рівня

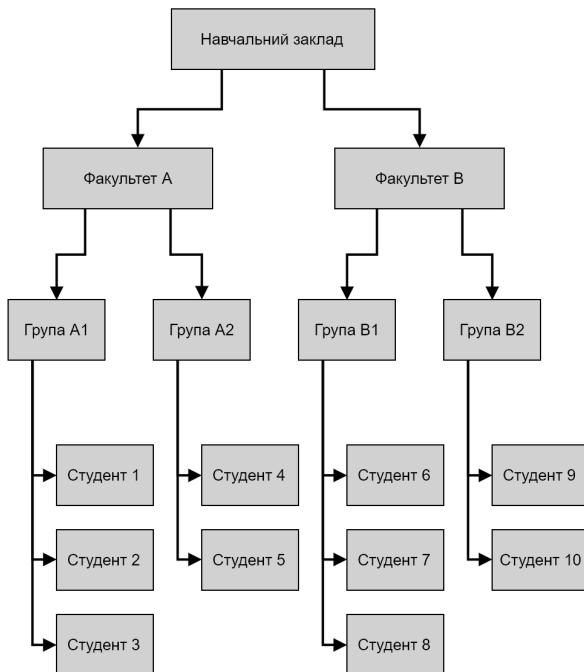
є нащадками кореневого сегменту та предками сегментів другого рівня

Сегменти другого рівня

є нащадками сегментів першого рівня та предками сегментів третього рівня

Сегменти третього рівня

є нащадками сегментів другого рівня



Ключові поняття

Сегмент

Предок

Нащадок

Основні правила

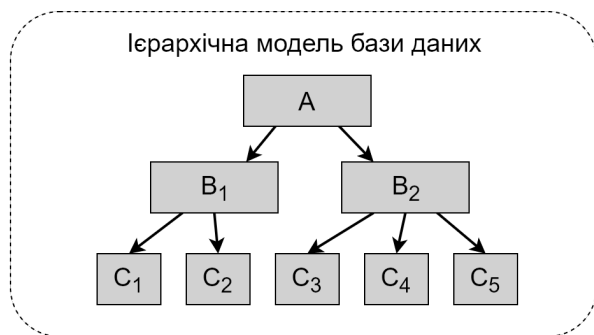
Кожен сегмент може бути предком декількох сегментів, які, в свою чергу, будуть вважатися для нього нащадками (зв'язок 1:М)

Кожен сегмент може мати тільки одного предка

Тільки один сегмент може бути кореневим



2.3.2. Переваги та недоліки представлення даних у вигляді ієрархічної моделі бази даних



Переваги

Концептуальна простота

Структура ієрархічної моделі, відносини між різними шарами якої інтуїтивно зрозумілі

Безпека бази даних

Безпеку бази даних забезпечує СКБД

Незалежність даних

Зміна типу даних викликає його автоматичну заміну за допомогою СКБД у всій базі даних

Цілісність даних

Оскільки дочірній сегмент завжди автоматично пов'язаний зі своїм предком, ієрархічна модель таким чином завжди забезпечує цілісність БД.

Ефективність

Ієрархічна модель бази даних дуже ефективна, коли в БД міститься великий обсяг даних зі зв'язком 1 : M

Недоліки

Складність керування

Будь-які зміни в структурі БД викличуть необхідність змін у всіх прикладних програмах, які отримують доступ до бази даних.

Нестача структурної незалежності

Зміни в структурі БД можуть привести до проблем з прикладними програмами, які до цього працювали правильно

Складність програмування

Програмісти та кінцеві користувачі повинні точно знати, яким чином фізично дані розміщені в БД для того, щоб отримати до них доступ.

Обмеження в реалізації

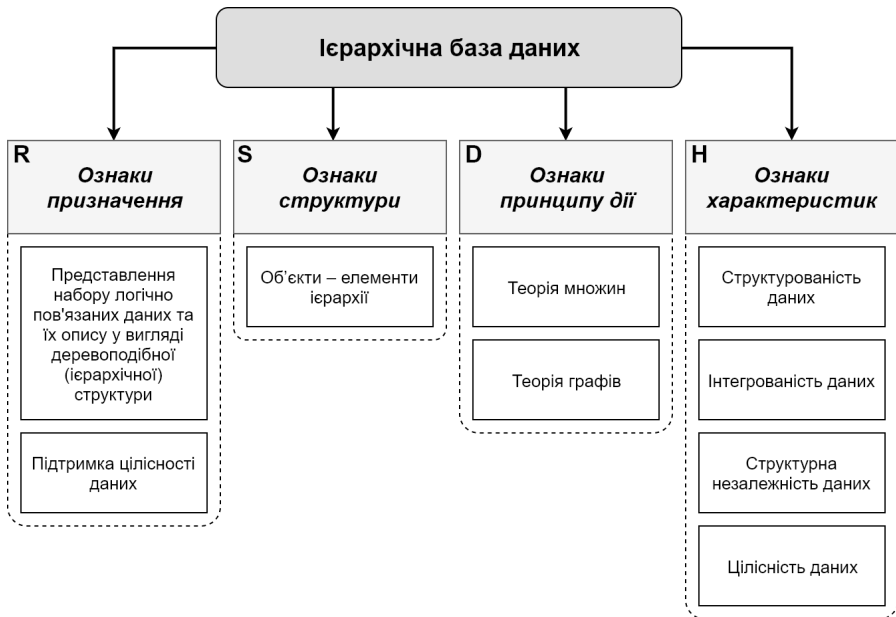
Більшість зв'язків не можуть бути зображені схемою 1 : M

Нестача стандартизації

Не існує стандартизованого набору ключових понять і компонентів, на базі яких можна було би створити модель, що містила би опис всіх необхідних стандартів



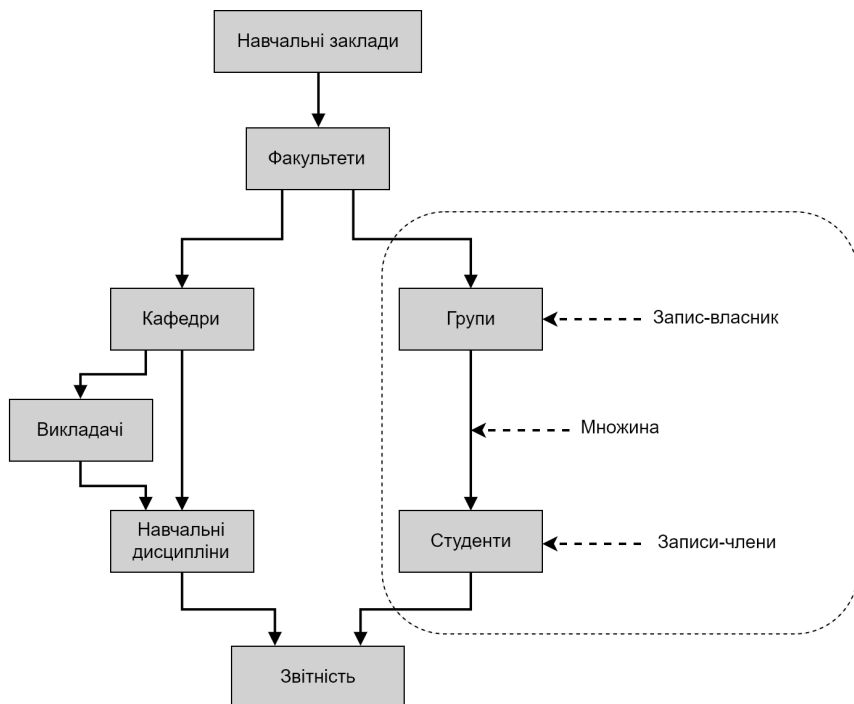
2.3.3. Семантична модель поняття ієрархічної бази даних





2.4. Мережева модель бази даних

2.4.1. Основні поняття та приклад побудови мережевої моделі бази даних



Ключові поняття

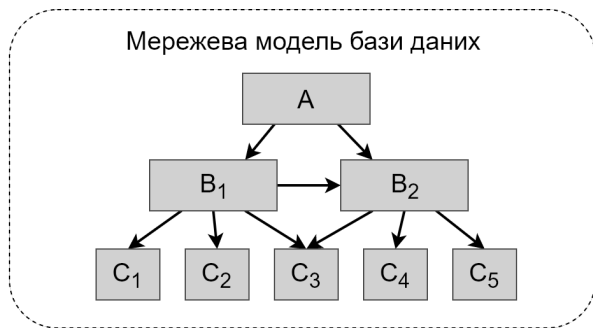
- Запис
- Множина
- Запис-власник
- Запис-член

Основні правила

- Множина є зв'язком 1:М між записом-власником та записами-членами
- Запис-член може з'являтися більше ніж в одній множині



2.4.2. Переваги та недоліки представлення даних у вигляді мережевої моделі бази даних



Переваги

Концептуальна простота

Абстрактне представлення бази даних є досить простим, що спрощує проектування

Підтримка інших типів зв'язків

Зв'язок M:N простіше реалізується в мережевій моделі, ніж в ієрархічній

Гнучкий доступ до даних

Гнучкість доступу до даних в мережевій моделі значно вище, ніж в ієрархічній моделі і в системах файлів

Цілісність даних

Відносини власник / член забезпечують цілісність бази даних

Незалежність

Зміни у властивостях даних не потребують переробки тих ділянок прикладних програм, де здійснюється доступ до даних

Відповідність стандартам

Виникнення мережевої моделі, в деякій мірі, пов'язане зі стандартами, розробленими і прийнятими в 70-х рр. XX ст.

Недоліки

Складність системи у цілому

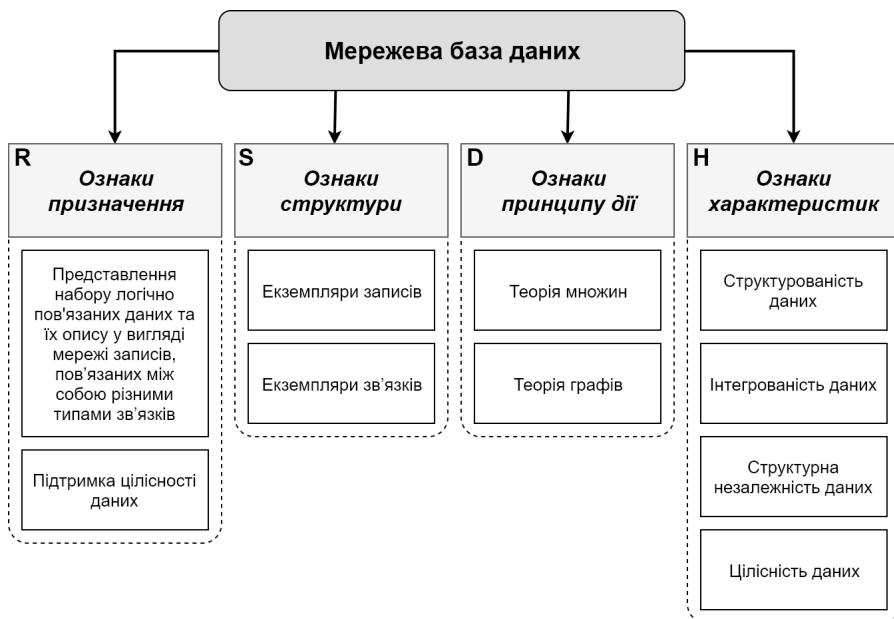
Складність системи знижує її ефективність (навігаційний метод доступу до даних)

Нестача структурної незалежності

Навігаційна система доступу до даних ускладнює проектування, реалізацію, розробку додатків, використання і управління



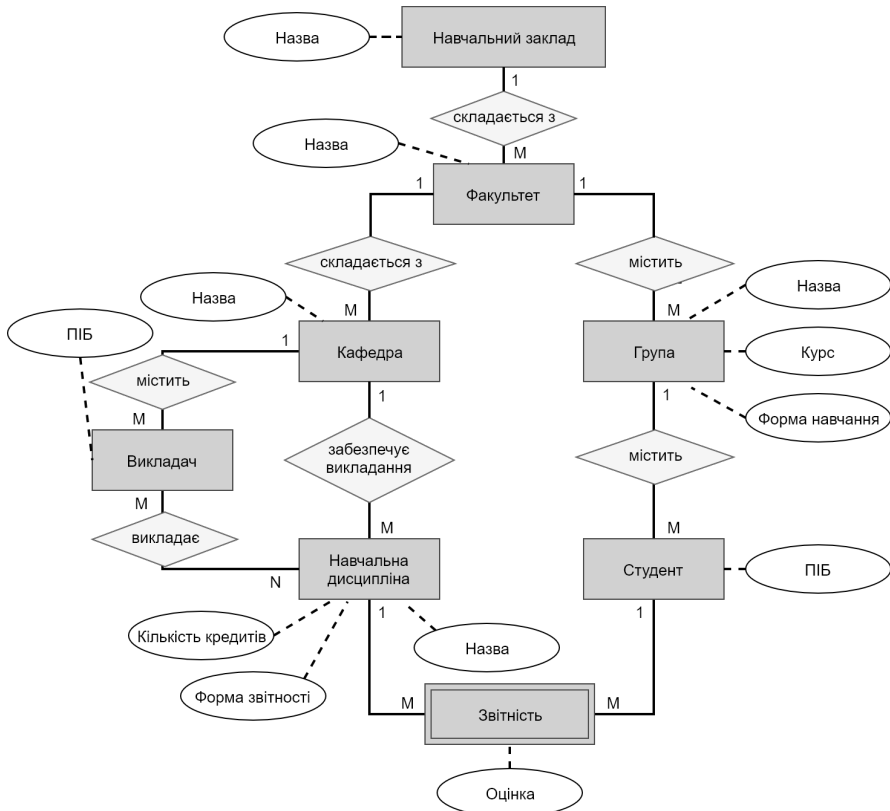
2.4.3. Семантична модель поняття мережевої бази даних





2.5. Модель даних «Сутність-Зв'язок»

2.5.1. Основні поняття та приклад побудови моделі даних «Сутність-Зв'язок»



Ключові поняття



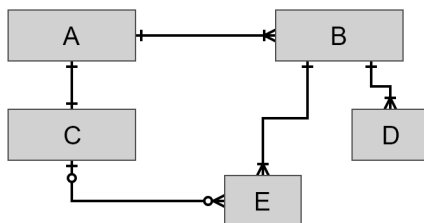
Основні правила

- Сутність описується набором атрибутів. Кожен атрибут описує окрему властивість сутності
- Зв'язок описує з'єднання між даними. Більшість зв'язків описує з'єднання між двома сутностями. Існує три типи зв'язків між даними: "один-до-багатьох" (1 : M), "багато-до-багатьох" (M : N) і "один-до-одного" (1 : 1).
- Зв'язок зображується на ER-діаграмі ромбом, сполученим з відповідною сутністю. Назва зв'язку (в дієслівній формі) записується всередині ромбу



2.5.2. Переваги та недоліки представлення даних у вигляді моделі даних «Сутність-Зв'язок»

Модель даних "Сутність - Зв'язок" (ER)



Переваги

Концептуальна простота

ER-модель дає дуже просте і наочне уявлення про основні логічні об'єкти БД та існуючі між ними зв'язки

Наочне представлення

ER-модель дає проектувальникам баз даних, програмістам і кінцевим користувачам просте наочне уявлення про дані і зв'язки між ними. Тому ER-модель є надзвичайно ефективним засобом, що інтегрує різні уявлення про дані в єдине робоче середовище

Інтеграція з реляційною моделлю БД

ER-модель прекрасно ув'язується з реляційною моделлю БД. Така інтеграція допомагає добре структурувати процес проектування реляційних БД

Недоліки

Недостатні можливості представлення обмежень

Існує багато важливих обмежень, які неможливо змоделювати методами ER-моделі

Обмеження можливостей представлення зв'язків

Зв'язки представляються як дещо, що відбувається між сутностями. Тому зв'язки між атрибутами всередині сутностей не можуть бути представлені засобами ER-моделі

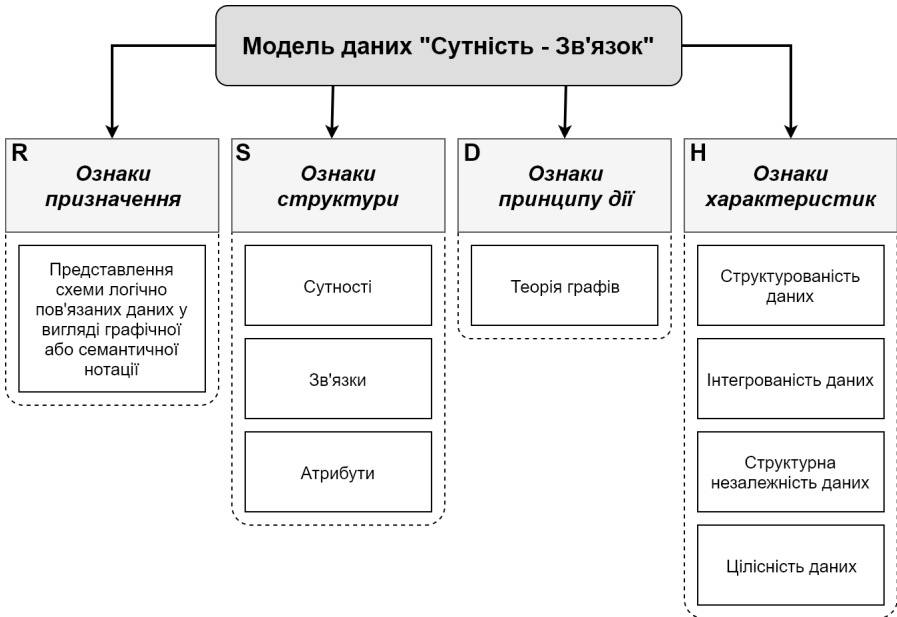
Відсутність мови маніпулювання даними

Втрата інформаційного наповнення

Ця модель сильно "переповнюється", якщо в ній відобразити всі її атрибути. Тому проектувальники баз даних зазвичай уникають повного відображення атрибутів, таким чином зменшуючи інформаційне наповнення ER-моделі



2.5.3. Семантична модель поняття моделі даних «Сутність-Зв'язок»

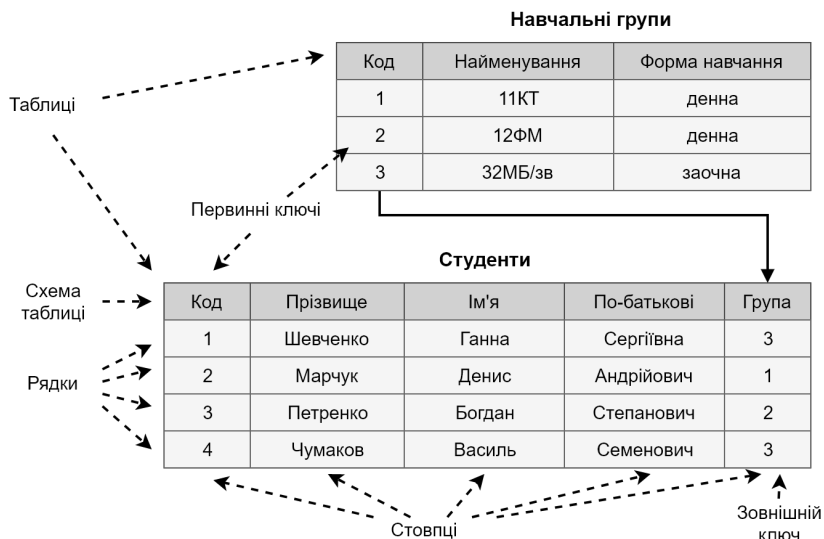




2.6. Реляційна модель бази даних



2.6.1. Основні поняття та приклад побудови реляційної моделі бази даних



Ключові поняття

Таблиця

Первинний ключ

Зовнішній ключ

Схема таблиці

Основні правила

Кожна таблиця (relation - відношення) являє собою матрицю, яка складається з набору рядків та стовпців, що перетинаються

Кожна комірка таблиці повинна мати атомарне (семантично неподільне) значення

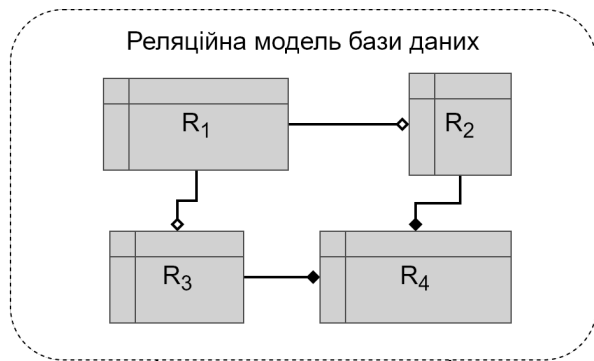
Кожний стовпчик містить дані тільки одного типу. Кожний рядок є окремим записом даних

Таблиці мають містити стовпчик з унікальними для цієї таблиці даними - первинний ключ

Таблиці можуть містити стовпці зі значеннями первинних ключів інших таблиць. Такі стовпці мають назву "зовнішні ключі". За їх допомогою таблиці можуть посилатися одна на одну



2.6.2. Переваги та недоліки представлення даних у вигляді реляційної моделі бази даних



Переваги

Концептуальна простота

Реляційна модель є логічною моделлю, що не залежить від специфіки фізичного представлення даних

Структурна незалежність даних

Зміни в структурі БД не впливають на можливість доступу до даних з боку СКБД

Простота проектування, реалізації, керування та використання

Довільні запити до БД

Реляційні БД мають потужні та гнучкі можливості зі створення довільних запитів за допомогою стандартної мови запитів SQL

Потужна система керування БД

Реляційна СУБД виконує велику кількість складних задач та приховує фізичний рівень складності системи як від проектувальників, так і від користувачів

Недоліки

Суттєві вимоги до обладнання та системного програмного забезпечення

Потужна РСКБД, що приховує від очей користувача всю складність системи, вимагає значних системних і апаратних ресурсів

Можливість "скоростиглих" проектів і реалізацій

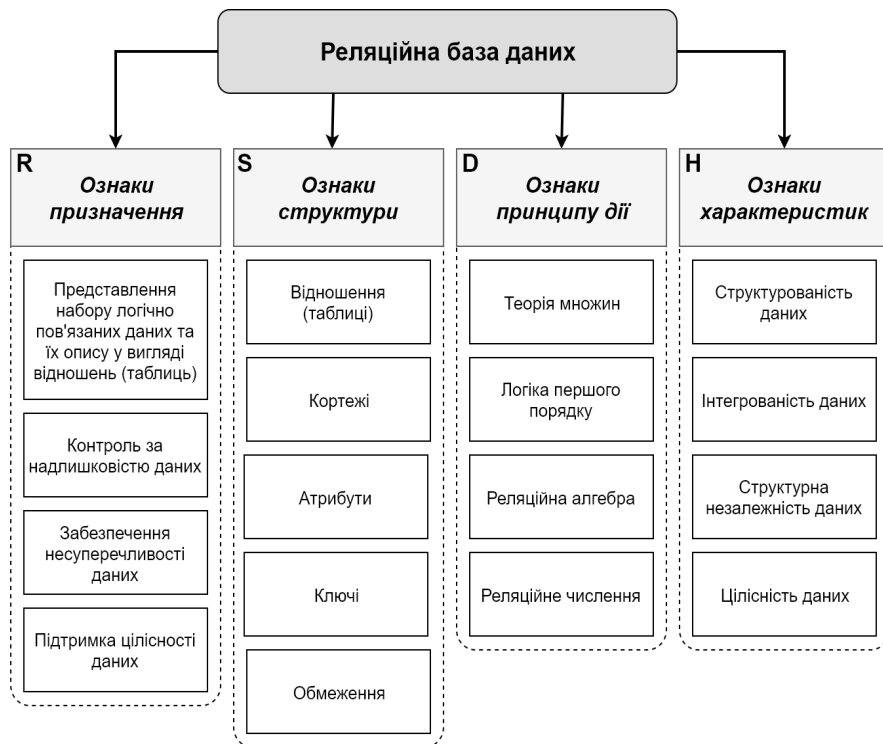
Концептуальна простота системи допускає можливість поганого використання гарної системи слабо підготовленим користувачем

Проблема "інформаційних острівців"

Може виникнути проблема "інформаційних острівців", оскільки окремі користувачі і підрозділи вважають можливим розробляти свої власні додатки



2.6.3. Семантична модель поняття реляційної бази даних

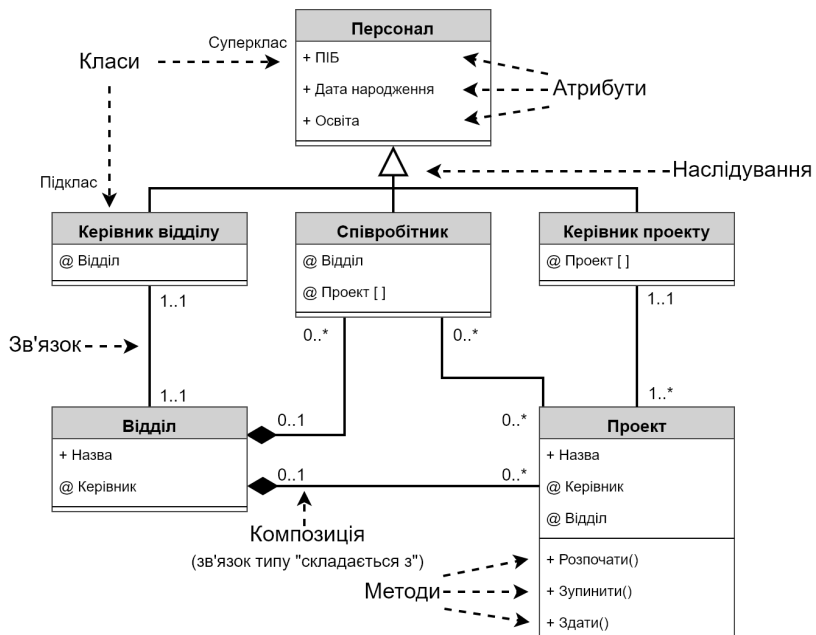




2.7. Об'єктно-орієнтована модель бази даних



2.7.1. Основні поняття та приклад побудови об'єктно-орієнтованої моделі бази даних



Ключові поняття

- Клас
- Наслідування
- Суперклас
- Підклас
- Об'єкт
- Атрибут
- Метод
- Зв'язок

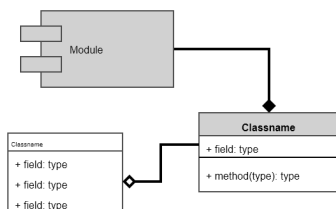
Основні правила

- Об'єкти, які спільно використовують одні й ті самі характеристики, групуються в класи. Клас являє собою сукупність подібних об'єктів зі структурою спільного доступу (атрибути) і поведінкою (методи).
- Метод класу являє собою реальну дію, що визначає поведінку об'єкта
- Класи організовані в ієрархію класів. Ієрархія класів схожа на перевернуте дерево, в якому кожен клас має тільки одного предка
- Наслідування - це можливість об'єкта всередині ієрархії класів наслідувати атрибути і методи класів, що структурно розташовані вище за нього.



2.7.2. Переваги та недоліки представлення даних у вигляді об'єктно-орієнтованої моделі бази даних

Об'єктно-орієнтована модель бази даних



Переваги

Додавання семантичного наповнення

Додавання семантичного наповнення робить модель даних більш значущою

Семантичне наповнення зовнішнього представлення

Як і ER-моделі, об'єктно-орієнтована модель представляє зв'язки в наочній формі. Однак наочне представлення об'єктно-орієнтованої моделі містить семантичне наповнення, що спрощує візуалізацію складних відносін всередині і між об'єктами

Цілісність даних

Так само як і ієрархічна, об'єктно-орієнтована модель використовує наслідування для захисту цілісності бази даних. Однак об'єкти ООБД містять більшу кількість типів зв'язків і більш складні зв'язки

Структурна незалежність та незалежність даних

Автономія об'єктів об'єктно-орієнтованої моделі гарантує структурну незалежність і незалежність даних

Недоліки

Складна навігація доступу до даних

Метод доступу до даних схожий на стиль ієрархічної і мережевої моделі

Складність вивчення та програмування

Об'єкти дуже складні, і той факт, що вони можуть мати значне семантичне наповнення, робить їх складними для проектування і реалізації

Повільне виконання транзакцій

Складність конфігурації і підвищені системні вимоги призводять до уповільнення виконання транзакцій

Нестача стандартизації

Поки не існує стандартів для об'єктно-орієнтованої моделі, а також немає стандартного методу доступу до даних. Цей недолік створює проблеми при доступі до даних від різних джерел



2.7.3. Семантична модель поняття об'єктно-орієнтованої бази даних

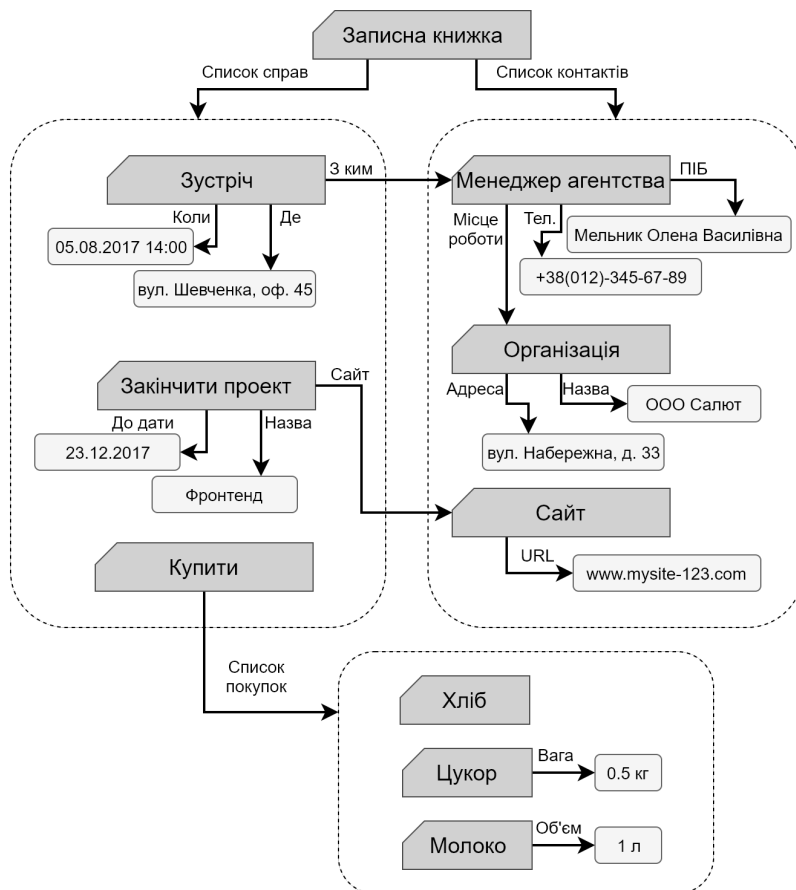




2.8. Документо-орієнтована модель бази даних



2.8.1. Основні поняття та приклад побудови документо-орієнтованої моделі бази даних



Ключові поняття

Документ

Колекція

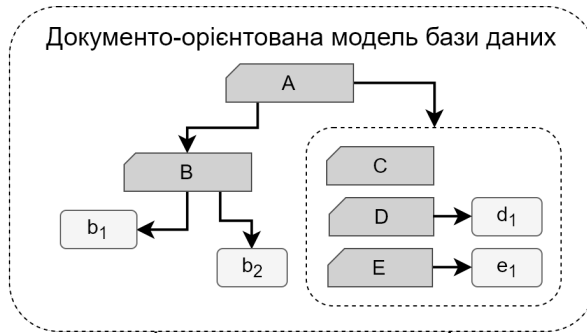
Атрибут

Основні правила

Характерною особливістю слабоструктурованих даних є те, що описова інформація, яка зазвичай виділяється в окрему схему, присутня в самих даних



2.8.2. Переваги та недоліки представлення даних у вигляді документо-орієнтованої моделі бази даних



Переваги

Семантичне наповнення

Дані у документо-орієнтованій БД не можуть бути описані за допомогою єдиної схеми. Замість цього, дані описують самі себе, тобто кожен елемент даних має особисту семантичну схему

Структурна незалежність та незалежність даних

Характерною особливістю слабоструктурованих даних є те, що описова інформація, яка зазвичай виділяється в окрему схему, присутня в самих даних. Тому дані в документо-орієнтованій БД не залежать від будь-яких зовнішніх структур та схем

Відповідність стандартам опису даних

Наявність загальних стандартів для представлення даних, таких як XML, JSON, YAML

Простота проектування, реалізації, керування та використання

Висока масштабованість

Ключові переваги NoSQL баз в розподілених системах полягають в процедурах шаринга і реплікації.

Недоліки

Відсутність транзакцій

Відсутність єдиної схеми та специфіка області застосування документо-орієнтованих БД поки що не дають змоги реалізувати повноцінний механізм транзакцій

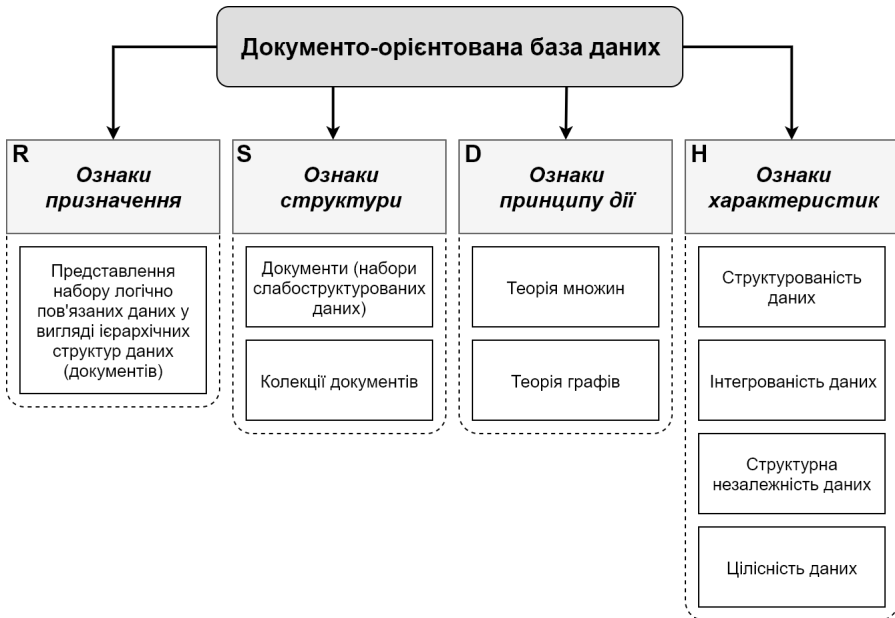
Відсутність стандартизованої мови маніпулювання даними

Недостатня цілісність даних

Задача контролю цілісності даних ніяк не регулюється СКБД та повинна бути реалізована у кодї додатків, що працюють з цією БД



2.8.3. Семантична модель поняття документо-орієнтованої бази даних





Контрольні питання

1. Назвіть моделі даних, які використовуються при інфологічному моделюванні баз даних.
2. Назвіть моделі даних, які використовуються при даталогічному моделюванні баз даних.
3. Від яких умов залежить вибір моделі бази даних.
4. Дайте визначення ієрархічної моделі бази даних. Наведіть приклад.
5. Назвіть основні переваги та недоліки ієрархічної моделі бази даних.
6. На основі яких математичних законів та теорій будуються ієрархічні моделі даних?
7. Дайте визначення мережевої моделі бази даних. Наведіть приклад.
8. Назвіть основні переваги та недоліки мережевої моделі бази даних.
9. На основі яких математичних законів та теорій будуються мережеві моделі даних?
10. Дайте визначення мережевої моделі бази даних. Наведіть приклад.
11. Назвіть основні переваги та недоліки мережевої моделі бази даних.
12. На основі яких математичних законів та теорій будуються мережеві моделі даних?
13. Дайте визначення моделі даних «Сутність-Зв'язок». Наведіть приклад.
14. Назвіть основні переваги та недоліки моделі даних «Сутність-Зв'язок».
15. На основі яких математичних законів та теорій будуються моделі даних «Сутність-Зв'язок»?
16. Дайте визначення реляційної моделі бази даних. Наведіть приклад.
17. Назвіть основні переваги та недоліки реляційної моделі бази даних.



18. На основі яких математичних законів та теорій будуються реляційні моделі даних?
19. Дайте визначення об'єктно-орієнтованої моделі бази даних. Наведіть приклад.
20. Назвіть основні переваги та недоліки об'єктно-орієнтованої моделі бази даних.
21. На основі яких математичних законів та теорій будуються об'єктно-орієнтовані моделі даних?
22. Дайте визначення документо-орієнтованої моделі бази даних. Наведіть приклад.
23. Назвіть основні переваги та недоліки документо-орієнтованої моделі бази даних.
24. На основі яких математичних законів та теорій будуються документо-орієнтовані моделі даних?



Завдання для самостійної роботи

1. Наведіть приклади предметних областей, що доцільно реалізовувати за допомогою ієрархічної моделі даних.
2. Наведіть приклади предметних областей, що доцільно реалізовувати за допомогою мережевої моделі даних.
3. Наведіть приклади предметних областей, що доцільно реалізовувати за допомогою реляційної моделі даних.
4. Наведіть приклади предметних областей, що доцільно реалізовувати за допомогою об'єктно-орієнтованої моделі даних.
5. Наведіть приклади предметних областей, що доцільно реалізовувати за допомогою документо-орієнтованої моделі даних.

Тема 3

ПРОЕКТУВАННЯ ТА РОЗРОБКА БАЗ ДАНИХ НА КОНЦЕПТУАЛЬНОМУ РІВНІ



МЕТА

Оволодіння знаннями про життєвий цикл інформаційних систем та баз даних, етапи проектування та розробки баз даних, вміннями створювати концептуальні схеми предметної області на основі моделі «Сутність-Зв'язок»



ЗМІСТ

- 3.1. Етапи життєвого циклу інформаційних систем
- 3.2. Етапи життєвого циклу баз даних
- 3.3. Семантична модель поняття технології розробки та використання баз даних
- 3.4. Етапи планування баз даних
- 3.5. Концептуальне (інфологічне) проектування баз даних
- 3.6. Моделювання предметної області на основі моделі «Сутність-Зв'язок» (ER-моделювання)
 - 3.6.1. Типи сутностей
 - 3.6.2. Атрибути
 - 3.6.3. Ключі
 - 3.6.4. Зв'язки
 - 3.6.5. Структурні обмеження
 - 3.6.6. Розширена ER-модель (EER)

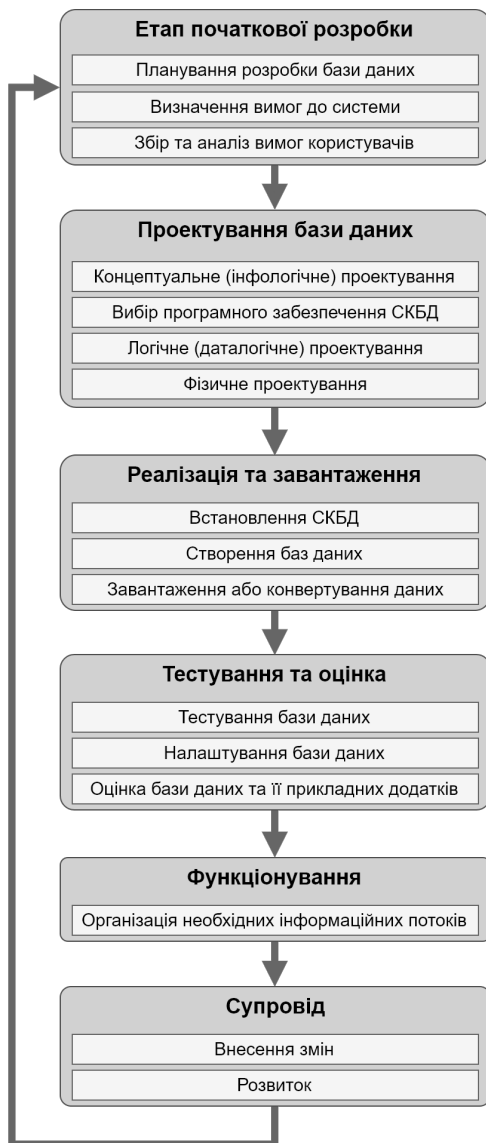


3.1. Етапи життєвого циклу інформаційних систем



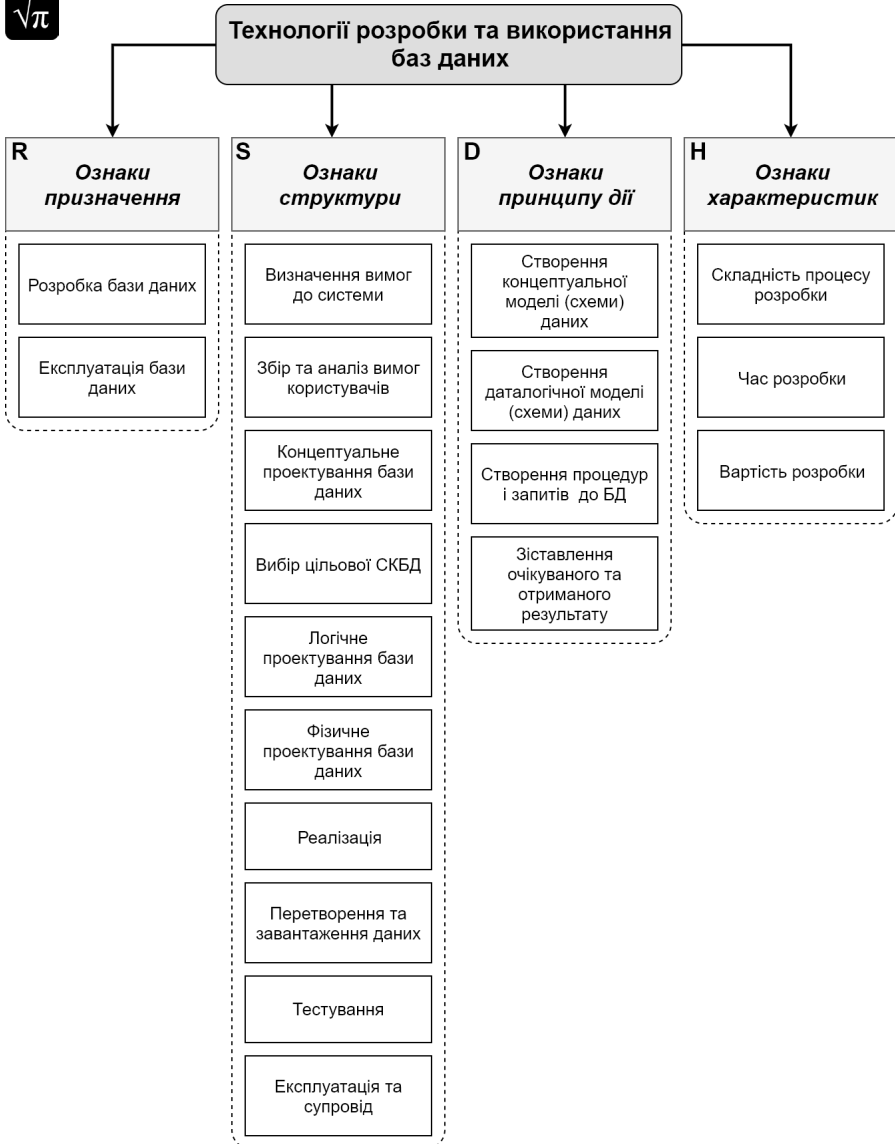


3.2. Етапи життєвого циклу бази даних





3.3. Семантична модель поняття технології розробки та використання баз даних





3.4. Етапи планування баз даних

Планування розробки бази даних

Підготовчі дії, що дозволяють з максимально можливою ефективністю реалізувати етапи життєвого циклу програми бази даних

Визначення бізнес-планів і цілей організації з подальшим виділенням її потреб в інформаційних технологіях

Оцінка показників вже існуючих інформаційних систем з метою виявлення їх сильних і слабких сторін

Оцінка можливостей використання інформаційних технологій для досягнення переваг перед конкурентами



Визначення вимог до системи

Визначення діапазону дій і завдань бази даних та її додатків, складу їх користувачів і областей застосування



Збір та аналіз вимог користувачів

Процес збору та аналізу інформації про ту частину організації, робота якої буде підтримуватися за допомогою створюваного додатка бази даних, а також використання цієї інформації для визначення вимог користувачів до створюваної системи

Опис даних, що використовуються та створюються в системі

Збір детальних відомостей про способи використання та створення даних

Визначення усіх допоміжних вимог до системи

Ця інформація потім аналізується для визначення вимог (або засобів), які повинні бути реалізовані у системі, що розробляється. Ці вимоги описуються в документах - **специфікація вимог** до нової системи бази даних



3.5. Концептуальне (інфологічне) проектування баз даних

Концептуальне (інфологічне) проектування

Створення концептуального представлення бази даних, що включає в себе визначення типів найважливіших сутностей і існуючих між ними зв'язків та атрибутів.

Щоб досягти незалежності від даних, не можна включати в визначення концептуальної мови будь-які вказівки про структури зберігання або методи доступу. Визначення концептуальної мови повинні відноситися тільки до змісту інформації

Створення локальної концептуальної моделі даних виходячи з представлень про предметну область для кожної групи користувачів

Визначення типів сутностей

Визначення типів зв'язків

Визначення атрибутів і зв'язування їх з типами сутностей і зв'язків

Визначення доменів атрибутів

Визначення атрибутів, які є потенційними і первинними ключами

Перевірка моделі на відсутність надлишковості

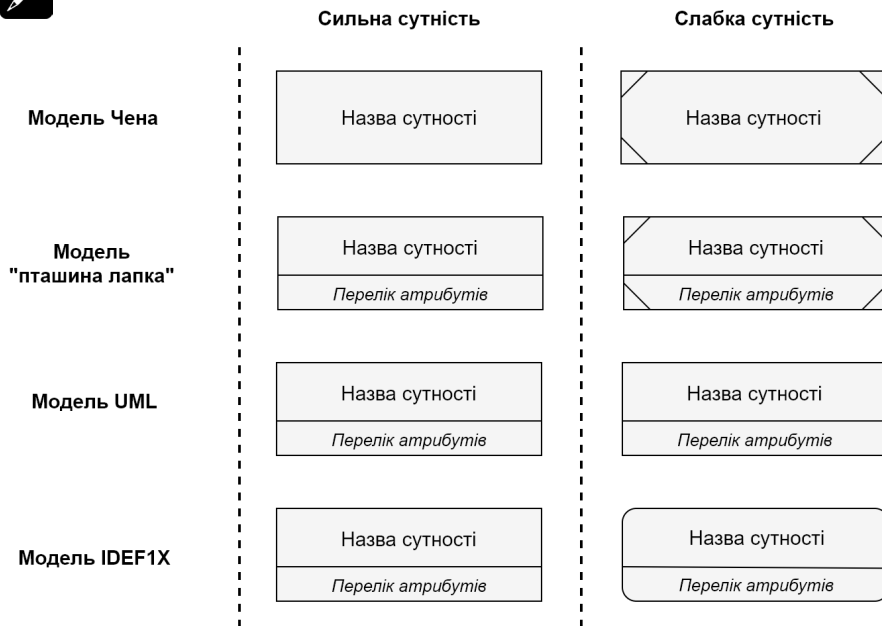
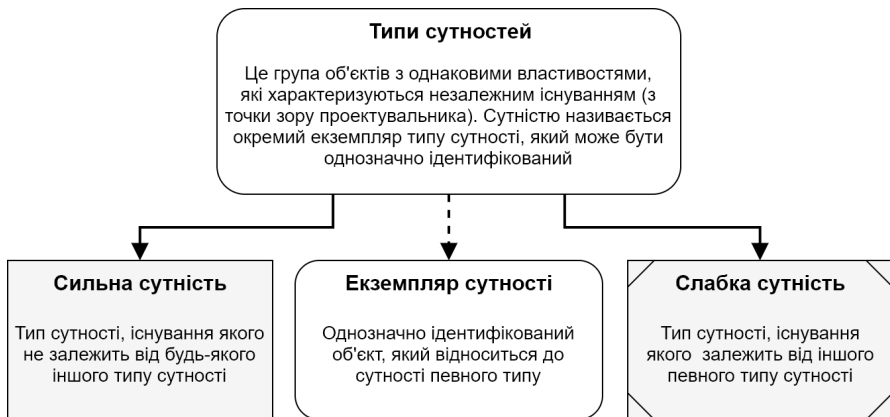
Перевірка відповідності локальної концептуальної моделі до конкретних транзакцій користувачів

Обговорення локальних концептуальних моделей даних з кінцевими користувачами.



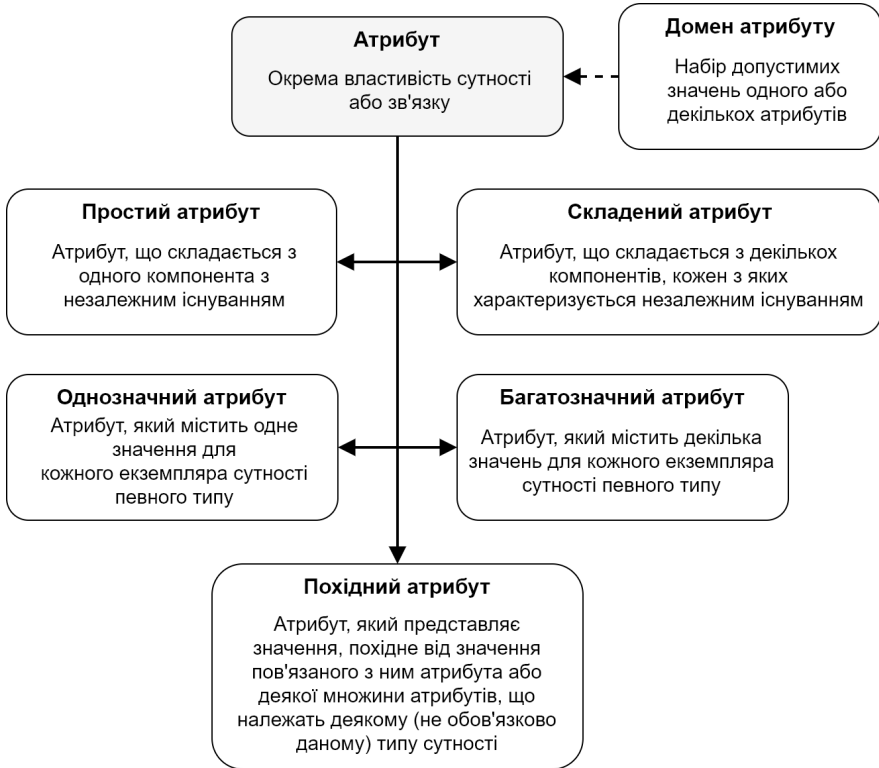
3.6. Моделювання предметної області на основі моделі «Сутність-Зв'язок» (ER-моделювання)

3.6.1. Типи сутностей





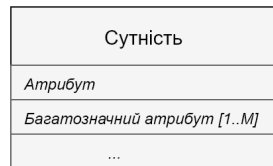
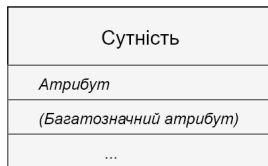
3.6.2. Атрибути



Модель Чена



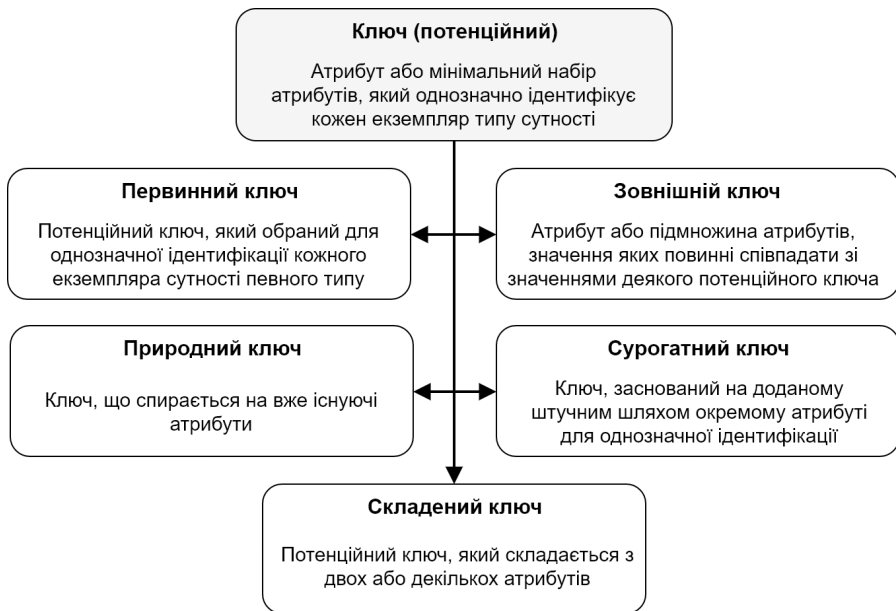
Модель "пташина лапка"
Модель IDEF1X



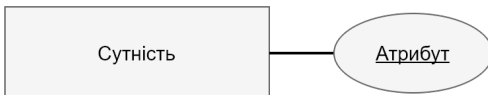
Модель UML



3.6.3. Ключі



Модель Чена



Модель "пташина лапка"
Модель IDEF1X

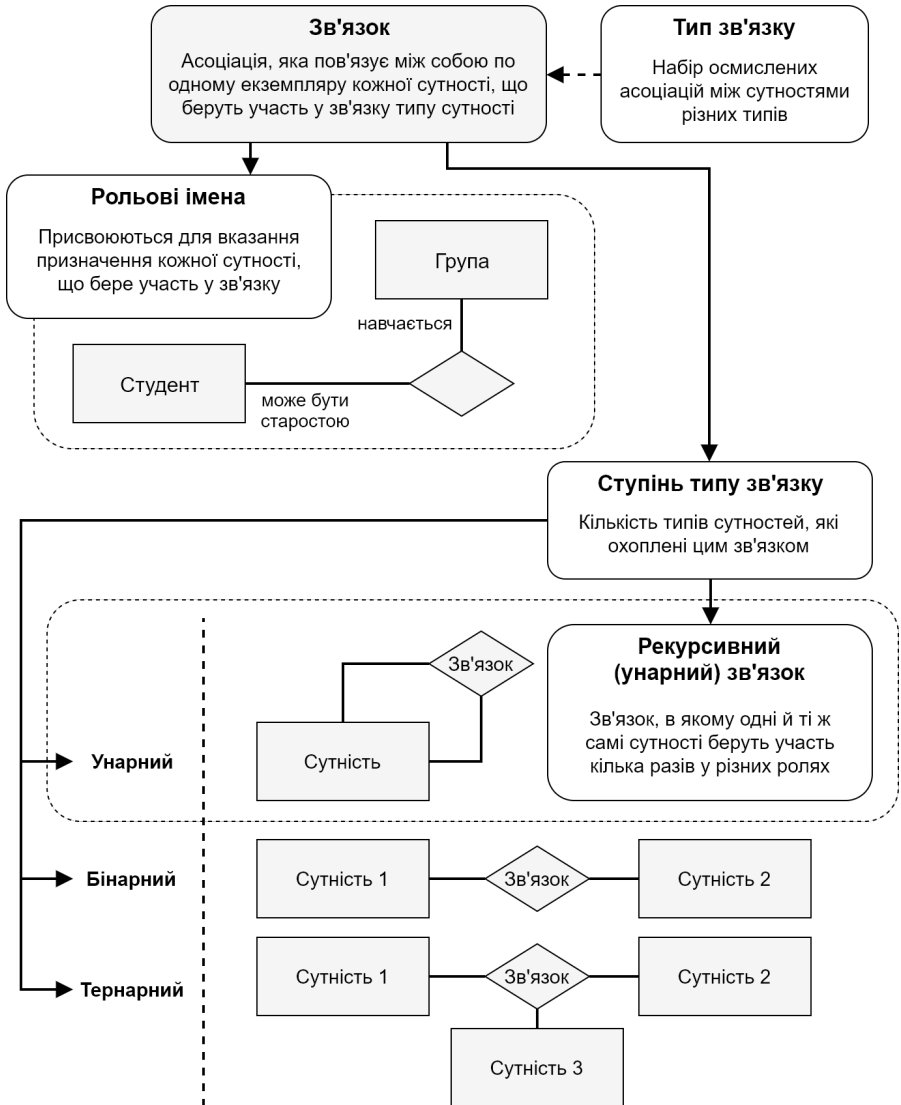


Модель UML



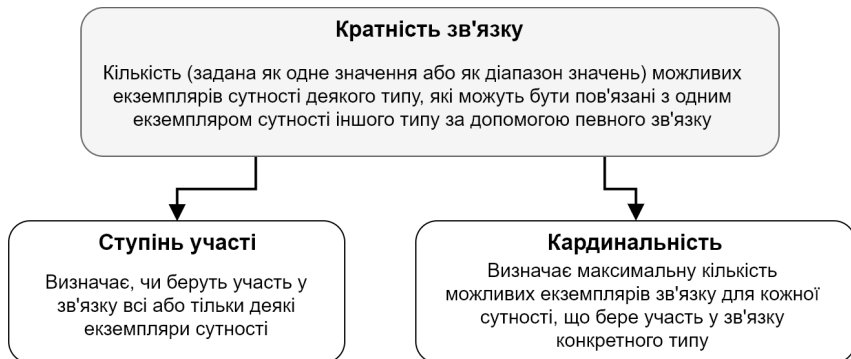


3.6.4. Зв'язки





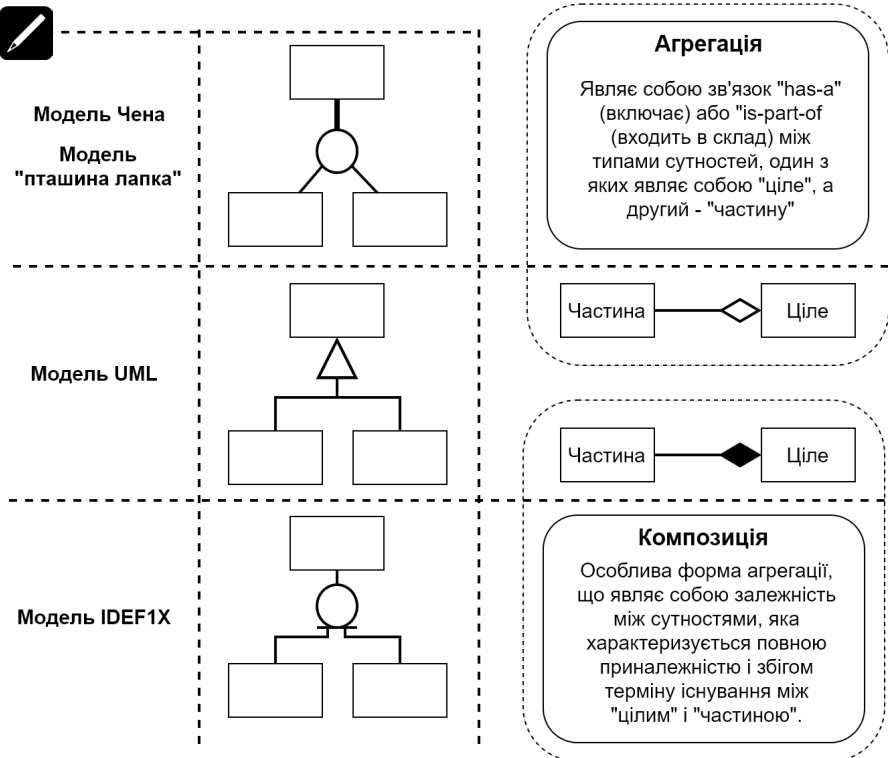
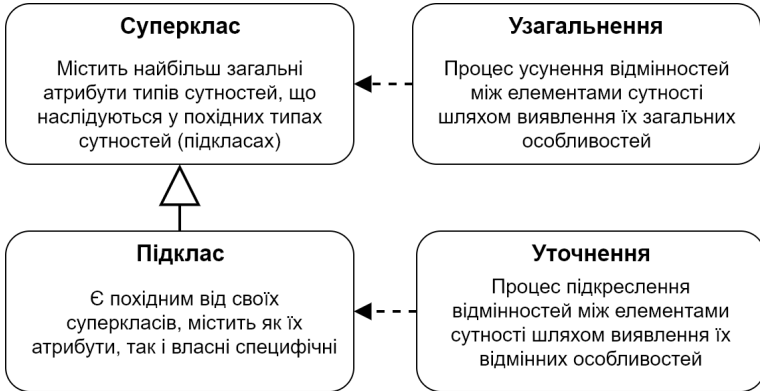
3.6.5. Структурні обмеження



	Обов'язковий одиничний	Обов'язковий множинний	Необов'язковий одиничний	Необов'язковий множинний
Модель Чена	1	M	1	M
Модель "пташина лапка"		>	○	○>
Модель UML	1..1	1..M	0..1	0..M
Модель IDEF1X	1 ●	P ●	Z ●	●



3.6.6. Розширена ER-модель (EER)





Контрольні питання

1. Назвіть етапи життєвого циклу інформаційних систем.
2. Назвіть етапи життєвого циклу баз даних.
3. Опишіть семантичні ознаки поняття технології розробки та використання баз даних.
4. Назвіть етапи планування баз даних.
5. Назвіть етапи концептуального (інфологічного) проектування баз даних.
6. Опишіть типи сутностей ER-моделі.
7. Вкажіть, в чому різниця між сутностями сильного та слабого типів, наведіть приклад кожного з них.
8. Опишіть різницю між атрибутом та його доменом.
9. Опишіть та наведіть приклади простих і складених атрибутів.
10. Опишіть та наведіть приклади однозначних, багатозначних і похідних атрибутів.
11. Опишіть, які типи зв'язків представлені у ER-моделі та наведіть приклади одно-, дво-, три- і чотирибічних зв'язків.
12. Які обмеження кратності поширюються на типи зв'язків ER-моделі?
13. Дайте визначення суперкласу та підкласу.
14. Опишіть зв'язок між суперкласом та його підкласом.
15. Опишіть та наведіть приклад процесу наслідування атрибутів.
16. В чому полягають основні причини введення понять суперкласів і підкласів у EER-моделі?
17. Опишіть та порівняйте процеси уточнення та узагальнення.
18. Опишіть та порівняйте поняття агрегування та композиції, а також наведіть приклад застосування кожного з них.



Завдання для самостійної роботи

1. Створіть інфологічну модель предметної області «Бібліотека» за допомогою графічної нотації «Сутність-Зв'язок».
2. Створіть інфологічну модель предметної області «Школа» за допомогою графічної нотації «Сутність-Зв'язок».
3. Створіть інфологічну модель предметної області «Супермаркет» за допомогою графічної нотації «Сутність-Зв'язок».
4. Створіть інфологічну модель предметної області «Інтернет-магазин» за допомогою графічної нотації «Сутність-Зв'язок».
5. Створіть інфологічну модель предметної області «Кінотеатр» за допомогою графічної нотації «Сутність-Зв'язок».
6. Створіть інфологічну модель предметної області «Лікарня» за допомогою графічної нотації «Сутність-Зв'язок».
7. Створіть інфологічну модель предметної області «Туристична агенція» за допомогою графічної нотації «Сутність-Зв'язок».

Тема 4

ПРОЕКТУВАННЯ ТА РОЗРОБКА БАЗ ДАНИХ НА ЛОГІЧНОМУ РІВНІ



МЕТА

Оволодіння знаннями про принципи логічного моделювання баз даних, теорію реляційної алгебри та числення, принципи побудови реляційної моделі бази даних, нормалізацію та нормальні форми, вміннями перетворювати концептуальні моделі баз даних у логічні, нормалізувати відношення реляційної моделі бази даних.



ЗМІСТ

- 4.1. Логічне (дatalogічне) моделювання баз даних**
- 4.2. Реляційна алгебра та числення**
- 4.3. Реляційна модель бази даних**
 - 4.3.1. Основні поняття реляційної моделі даних
 - 4.3.2. Відношення, властивості відношень
 - 4.3.3. Представлення
 - 4.3.4. Реляційні ключі
 - 4.3.5. Реляційна цілісність
- 4.4. Перетворення концептуальної моделі бази даних у логічну**
- 4.5. Нормалізація**
 - 4.5.1. Функціональні залежності між відношеннями
 - 4.5.2. Поняття нормалізації відношень у реляційній моделі даних
 - 4.5.3. Ненормалізована та перша нормальна формам
 - 4.5.4. Повна функціональна залежність та друга нормальна форма
 - 4.5.5. Третя нормальна форма
 - 4.5.6. Нормальна форма Бойса-Кодда, четверта та п'ята нормальні форми



4.1. Логічне (даталогічне) моделювання баз даних

Логічне (даталогічне) проектування

Перетворення концептуального представлення в логічну структуру бази даних, включаючи проектування відносин з урахуванням обраного типу СКБД.

Логічна модель даних є джерелом інформації для етапу фізичного проектування. Вона надає розробнику фізичної моделі даних засоби проведення всебічного аналізу різних аспектів роботи з даними, що має виключно важливе значення для вибору дійсно ефективного проектного рішення

Перетворення розробленої локальної концептуальної моделі даних у локальну логічну модель даних та її перевірка

Усунення особливостей локальної логічної моделі, несумісних з обраною логічною моделлю

Перевірка логічної моделі за допомогою правил нормалізації

Перевірка відповідності локальної логічної моделі до конкретних транзакцій користувачів

Визначення вимог підтримки цілісності даних

Злиття локальних логічних моделей даних в єдину глобальну модель даних

Перевірка глобальної логічної моделі даних

Перевірка можливостей розширення моделі в майбутньому



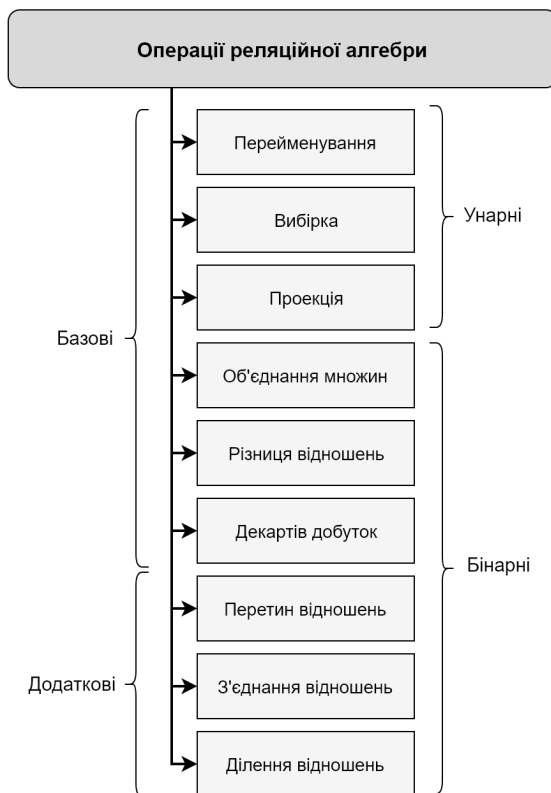
4.2. Реляційна алгебра та числення

4.2.1. Реляційні алгебра. Операції реляційної алгебри

Реляційна алгебра

Це теоретична мова операцій, які дозволяють створювати на основі одного або декількох відношень інше відношення без змін самих вихідних відношень.

Обидва операнда та результат є відношеннями, тому результати однієї операції можуть бути застосовані до іншої операції. Це дозволяє створювати вкладені вирази реляційної алгебри, але при будь-якій глибині вкладеності результатом є відношення.





4.2.2. Перейменування, вибірка, проєкція

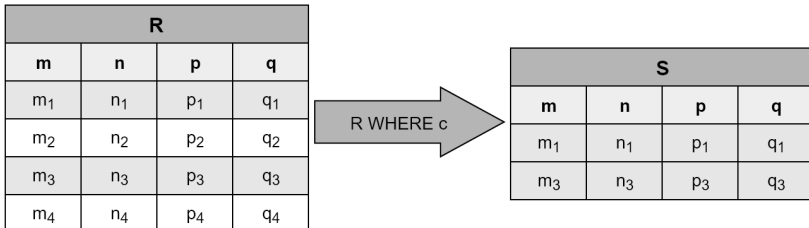
Перейменування

Результат застосування оператора над відношенням R ідентичний R за винятком того, що поле b в усіх кортежах перейменовується на поле a. Цей оператор застосовується для простого перейменування атрибута відношення, або самого відношення.



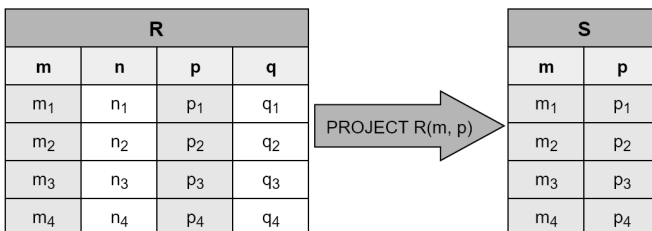
Вибірка

Операція вибірки застосовується до одного відношення R і визначає результуюче відношення S, яке містить тільки ті кортежі (рядки) з відношення R, які задовільняють заданій умові (предикату)



Проекція

Операція вибірки застосовується до одного відношення R і визначає нове відношення S, яке містить вертикальну підмножину відношення R, що створюється шляхом вилучення значень вказаних атрибутів та виключення з результату рядків-дублікатів





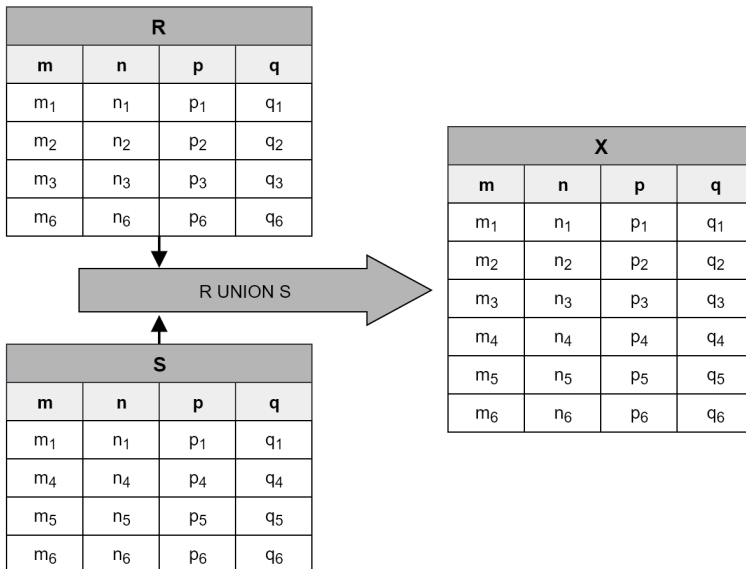
4.2.3. Об'єднання відношень

Об'єднання відношень

Об'єднання двох відношень R та S визначає нове відношення, яке включає всі кортежі, що містяться тільки в R, тільки в S, одночасно в R та S, причому всі дублікати кортежів виключені. При цьому відношення R та S мають бути сумісними за об'єднанням

Сумісність за об'єднанням

Відношення є сумісними за об'єднанням, якщо схеми двох відношень збігаються, тобто складаються з однакової кількості атрибутів, причому кожна пара відповідних атрибутів має однаковий домен

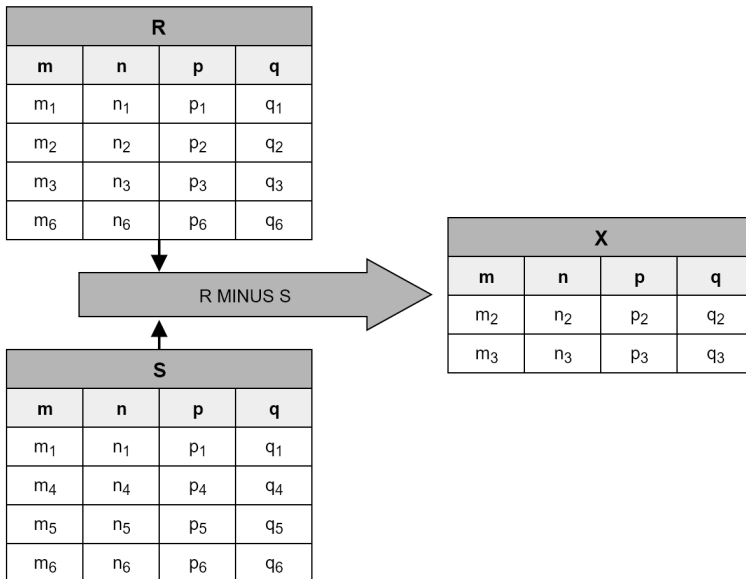




4.2.4. Різниця відношень

Різниця відношень

Різниця двох відношень R та S складається з кортежів, які є у відношенні R, але відсутні у відношенні S. Причому відношення R та S мають бути сумісними за об'єднанням

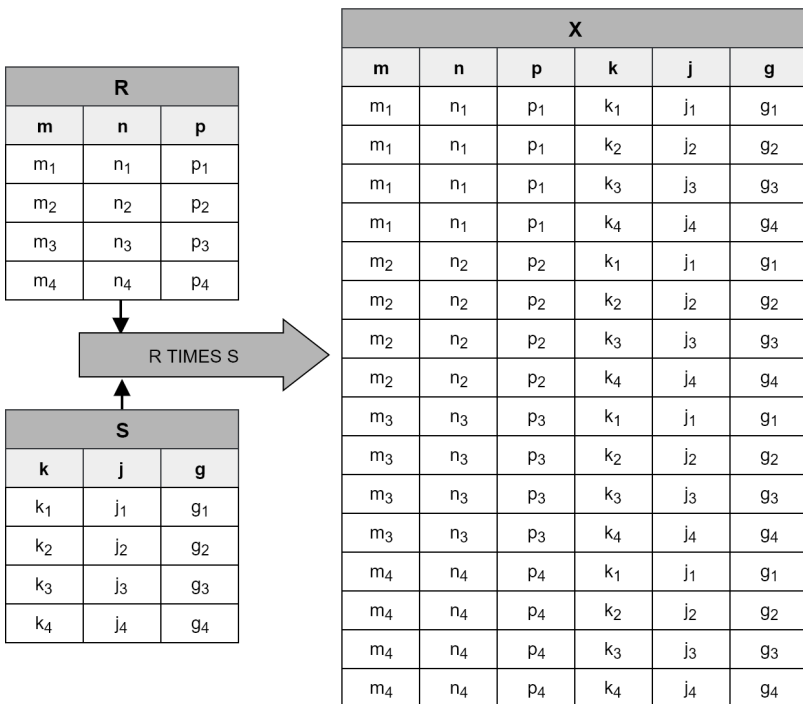




4.2.5. Декартів добуток

Декартів добуток

Операція декартового добутку визначає нове відношення, яке є результатом конкатенації (тобто зчеплення) кожного кортежу з відношення R з кожним кортежем з відношення S

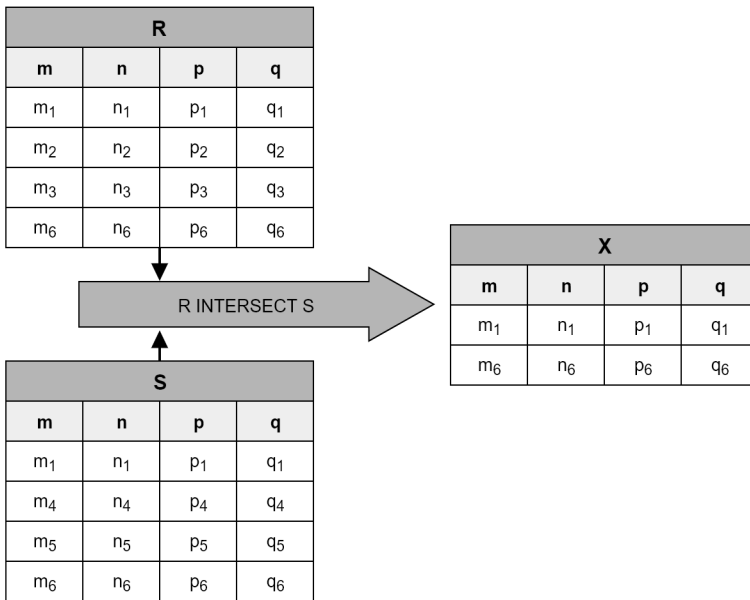




4.2.6. Перетин відношень

Перетин відношень

Операція перетину визначає відношення, яке містить кортежі, присутні як у відношенні R, так і у відношенні S. Відношення R та S повинні бути сумісними за об'єднанням

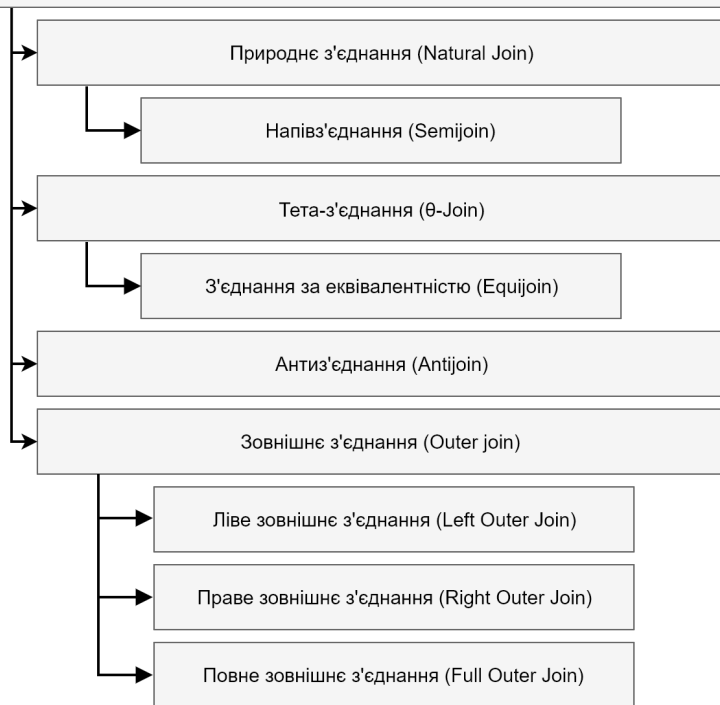




4.2.7. З'єднання відношень

З'єднання відношень

Операція з'єднання є похідною від операції декартова добутку, оскільки вона еквівалентна операції вибірки з декартового добутку двох операндів-відношень тих кортежів, які задовільняють умові, що вказана у предикаті з'єднання в якості формули вибірки

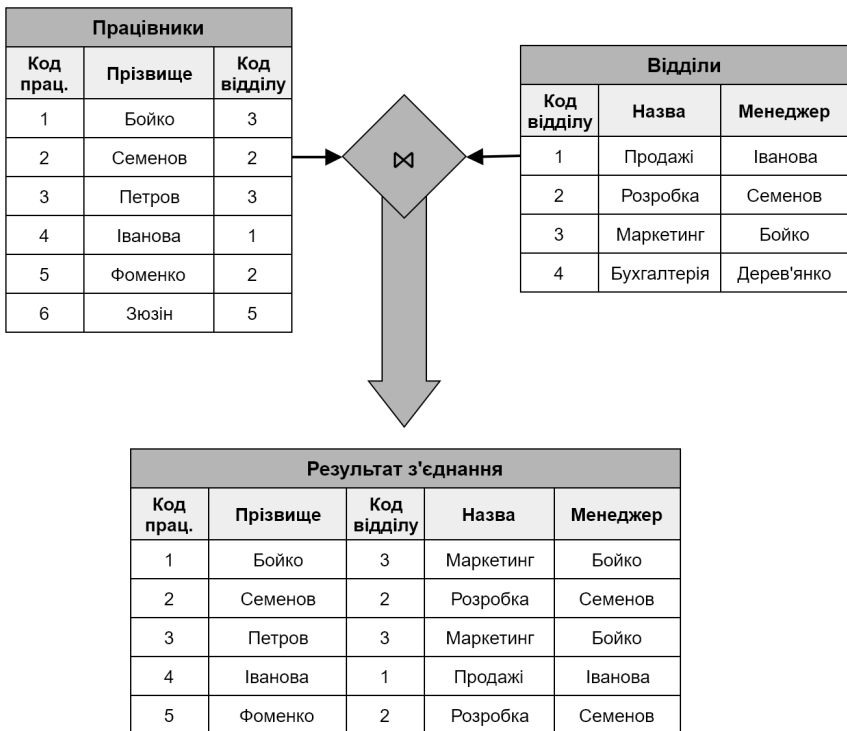




4.2.8. Природне з'єднання

Природне з'єднання

Природне з'єднання (\bowtie) - це бінарний оператор, який записується як $(R \bowtie S)$, де R та S є відношеннями. Результатом природнього з'єднання є набір усіх комбінацій кортежів в R та S , які рівні за їхніми загальними іменами атрибутів





4.2.9. Тета-з'єднання

Тета-з'єднання

Тета-з'єднання (θ) - це бінарний оператор, який записується як $(R \bowtie S)$, або $(R \bowtie_{\theta} S)$
 $a \theta b$ $a \theta v$

де R та S є відношеннями, a та b є іменами атрибутів, θ - бінарний реляційний оператор із множини $\{<, \leq, =, >, \geq\}$, v є постійним значенням. Результатом природного з'єднання є набір усіх комбінацій кортежів в R та S , які задовільняють θ .

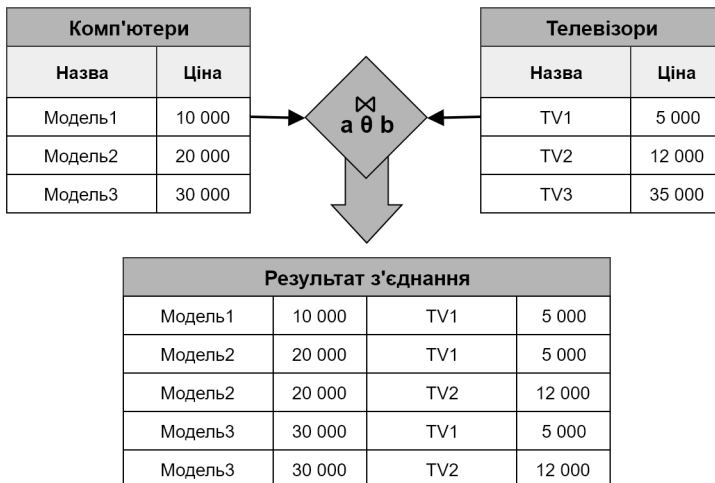
Результат тета-з'єднання є визначеним за умови того, що схеми R та S є роз'єданими, тобто не містять спільних атрибутів

З'єднання за еквівалентністю

З'єднання за еквівалентністю є окремим випадком тета-з'єднання, коли оператор θ є оператором еквівалентності ($=$)



Таблиці "Комп'ютери" та "Телевізори" містять списки відповідних моделей та їх цін. Покупець хоче придбати комп'ютер та телевізор, але не хоче витратити на телевізор коштів більше, ніж коштує комп'ютер. Надати усі можливі варіанти, що задовільняють такій вимозі

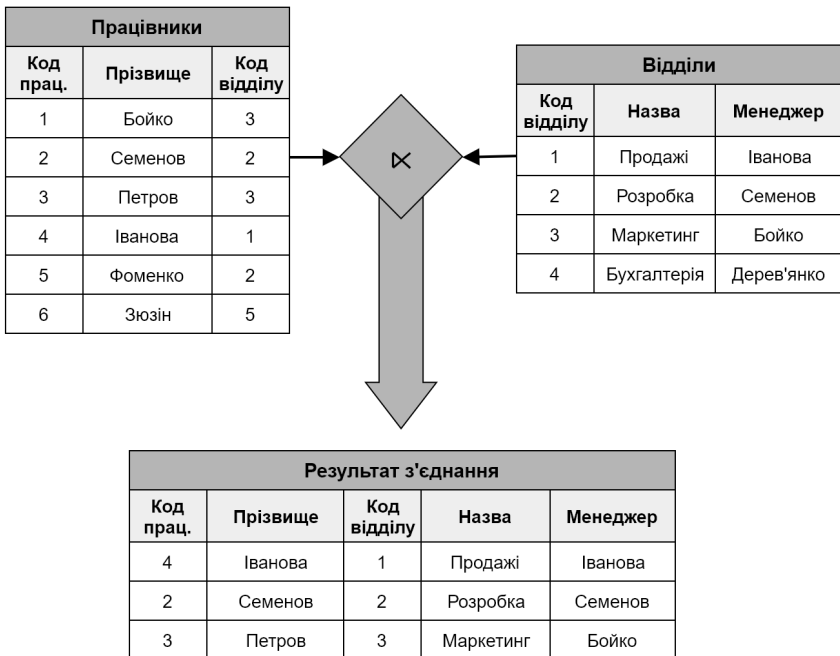




4.2.10. Напівз'єднання

Напівз'єднання

Напівз'єднання (ліве: \bowtie , та праве: \bowtie) - це бінарний оператор, подібний до природнього з'єднання, який записується як $(R \bowtie S)$ або $(R \bowtie S)$, де R та S є відношеннями. Результатом напівз'єднання є набір усіх комбінацій кортежів в R , для яких є відповідний кортеж у S (та навпаки - для правого з'єднання), що містить однакові значення у атрибутах з однаковими загальними іменами

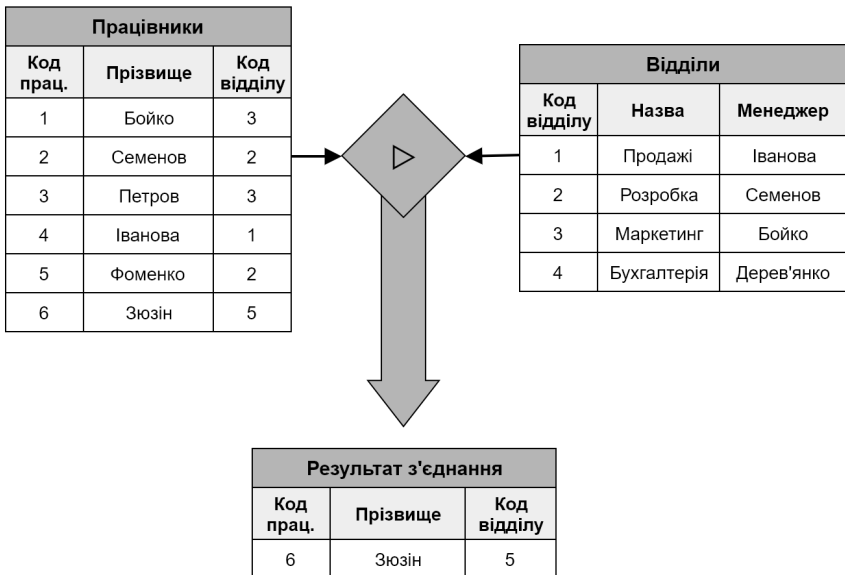




4.2.11. Антиз'єднання

Антиз'єднання

Антиз'єднання (\triangleright) - це бінарний оператор, подібний до напівз'єднання, який записується як $(R \triangleright S)$ або $(R \times S)$, де R та S є відношеннями. Результатом антиз'єднання є набір усіх комбінацій кортежів в R , для яких немає відповідних кортежів у S , що містять однакові значення у атрибутах з однаковими загальними іменами





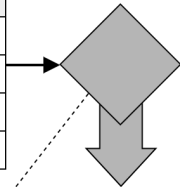
4.2.12. Зовнішнє з'єднання

Зовнішнє з'єднання

Зовнішнє з'єднання - це бінарний оператор, R та S є відношеннями. Результатом зовнішнього з'єднання є набір усіх комбінацій кортежів в R та S, які рівні за їхніми загальними іменами атрибутів, а також кортежів, що не мають відповідних кортежів у іншому відношенні.
 Для позначення відсутніх значень у другому відношенні використовується значення NULL.
 Розрізняють три види зовнішнього з'єднання - **ліве**, **праве** та **повне**, залежно від напрямку з'єднання відношень з неповною відповідністю кортежів.

Працівники		
Код прац.	Прізвище	Код відділу
2	Семенов	2
4	Іванова	1
5	Фоменко	2
6	Зюзін	5

Відділи		
Код відділу	Назва	Менеджер
1	Продажі	Іванова
2	Розробка	Семенов
4	Бухгалтерія	Дерев'янку



Результат з'єднання				
Код прац.	Прізвище	Код відділу	Назва	Менеджер
2	Семенов	2	Розробка	Семенов
4	Іванова	1	Продажі	Іванова
5	Фоменко	2	Розробка	Семенов
6	Зюзін	5	NULL	NULL



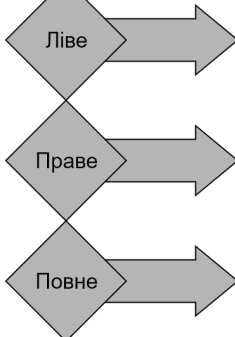
6	Зюзін	5	NULL	NULL
---	-------	---	------	------

або

NULL	NULL	4	Бухгалтерія	Дерев'янку
------	------	---	-------------	------------

або

6	Зюзін	5	NULL	NULL
NULL	NULL	4	Бухгалтерія	Дерев'янку



4.2.13. Ділення відношень



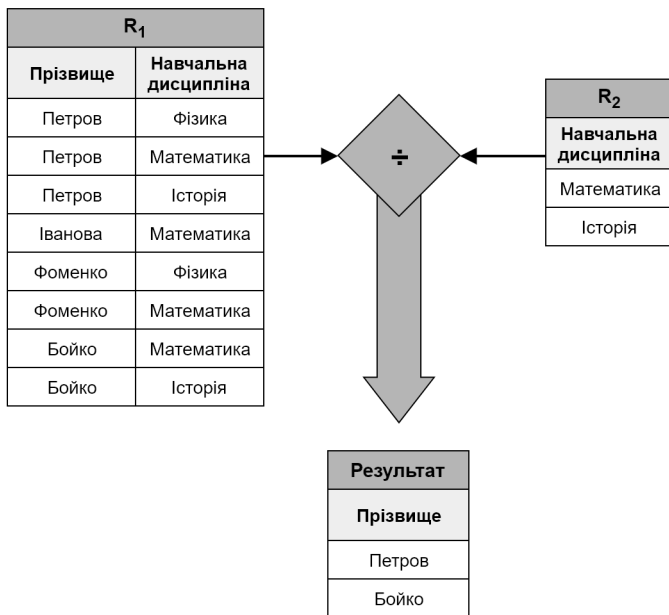
Ділення відношень

Ділення відношень - це бінарна операція, яка записується як $(R \div S)$, де R та S є відношеннями. Результатом операції ділення є набір кортежів відношення R , визначених на множині атрибутів S , які відповідають комбінації всіх кортежів відношення S



Перше відношення R_1 містить список усіх студентів та навчальні дисципліни, що вони вивчають. Треба отримати список студентів, які вивчають всі дисципліни, наведені у другому відношенні R_2 .

Ця операція може бути виконана шляхом ділення першого відношення на друге.



4.3. Реляційна модель бази даних

4.3.1. Основні поняття реляційної моделі даних



Відношення	Плоска таблиця, що містить стовпці та рядки
Атрибут	Іменований стовпець відношення
Домен атрибуту	Набір допустимих значень одного або декількох атрибутів
Кортеж	Рядок відношення
Ступінь	Кількість атрибутів, які містить відношення
Кардинальність	Кількість кортежів, які містить відношення
Реляційна база даних	Набір нормалізованих відношень, що різняться за назвою
Реляційна схема	Іменоване відношення, що визначено на основі множини пар атрибутів та імен доменів
Схема реляційної бази даних	Множина реляційних схем, що різняться за назвою

Альтернативна термінологія

Офіційний варіант	Альтернативний варіант 1	Альтернативний варіант 2
Відношення	Файл	Файл
Кортеж	Рядок	Запис
Атрибут	Стовпець	Поле



4.3.2. Відношення, властивості відношень





4.3.3. Представлення

Базове відношення

Іменоване відношення, що відповідає сутності у концептуальній схемі, кортежі якого фізично зберігаються у базі даних

Представлення

Динамічний результат однієї або декількох реляційних операцій над базовими відношеннями з метою створення деякого іншого відношення. Представлення є віртуальним відношенням, яке реально в базі даних не існує, але створюється за вимогою окремого користувача в момент надходження цієї вимоги

Призначення представлень

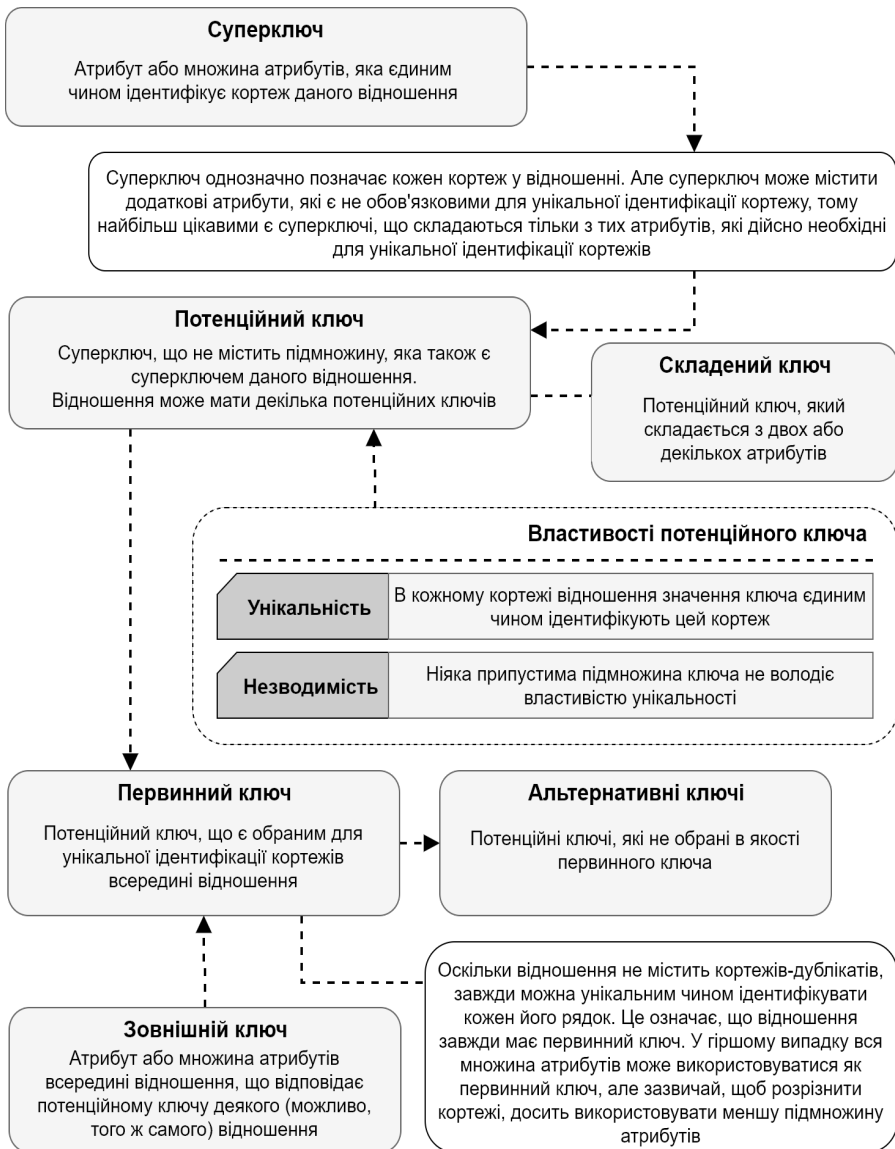
Надає потужний і гнучкий механізм захисту, що дозволяє приховати деякі частини бази даних від певних користувачів. Користувач не матиме відомостей про існування будь-яких атрибутів або кортежів, відсутніх в доступних йому представленнях

Дозволяє організувати доступ користувачів до даних найбільш зручним для них чином, тому одні й ті ж дані в один і той же час можуть розглядатися різними користувачами абсолютно різними способами

Дозволяє спрощувати складні операції з базовими відношеннями. Наприклад, якщо представлення буде визначено на основі поєднання двох відношень, то користувач зможе виконувати над ним прості унарні операції вибірки і проєкції, які будуть автоматично конвертовані засобами СКБД в еквівалентні операції з виконанням з'єднання базових відношень



4.3.4. Реляційні ключі





4.3.5. Реляційна цілісність

Порожнє значення (NULL)

Вказує, що значення атрибуту на даний час невідоме або неприйнятне для цього кортежу

Порожнє значення (яке умовно позначається як NULL) слід розглядати як логічну величину "невідомо". Іншими словами, або це значення не входить в область визначення деякого кортежу, або ніяке значення ще не задано

Реляційні обмеження цілісності

Цілісність сутностей

У базовому відношенні жоден атрибут первинного ключа не може містити відсутні значення, що позначаються як NULL

За визначенням, первинний ключ - це мінімальний ідентифікатор, який використовується для унікальної ідентифікації кортежів. Це означає, що ніяка підмножина первинного ключа не може бути достатньою для унікальної ідентифікації кортежів. Якщо допустити присутність NULL в будь-якій частині первинного ключа, це рівнозначно твердженню, що не всі його атрибути необхідні для унікальної ідентифікації кортежів, що суперечить визначенню первинного ключа

Цілісність посилань

Якщо у відношенні існує зовнішній ключ, то значення зовнішнього ключа повинно або відповідати значенням потенційного ключа деякого кортежу в його базовому відношенні або зовнішній ключ повинен повністю складатися із значень NULL

Корпоративні обмеження цілісності

Додаткові правила підтримки цілісності даних, що визначаються користувачами або адміністраторами баз даних



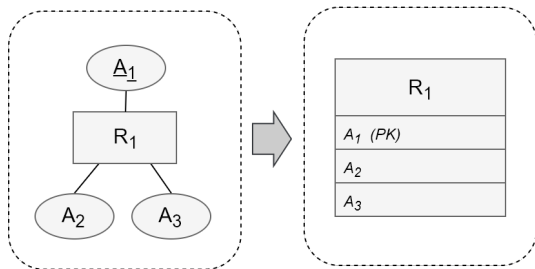
4.4. Перетворення концептуальної моделі бази даних у логічну



4.4.1. Сильна сутність, слабка сутність, зв'язок 1:М

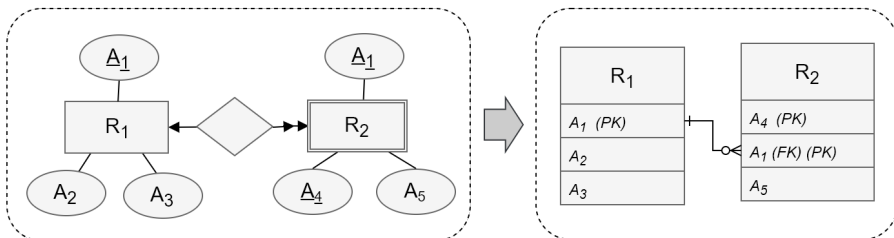
Сильна сутність

Створення відношення, яке включає всі прості атрибути



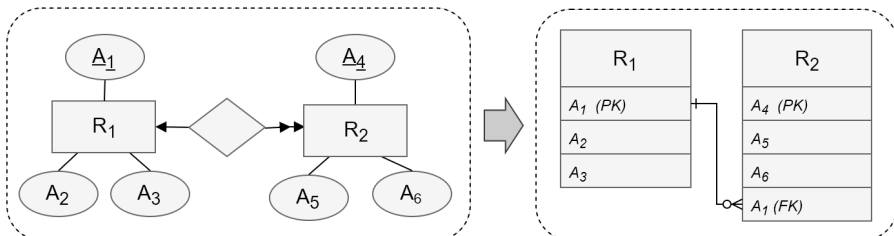
Слабка сутність

Створення відношення, яке включає всі прості атрибути. Після перетворення зв'язку з сутністю-володарем необхідно визначити первинний ключ



Зв'язок 1:М

Передача первинного ключа сутності "один" на відношення "багато" в якості зовнішнього ключа. На бік "багато" передаються також всі атрибути зв'язку





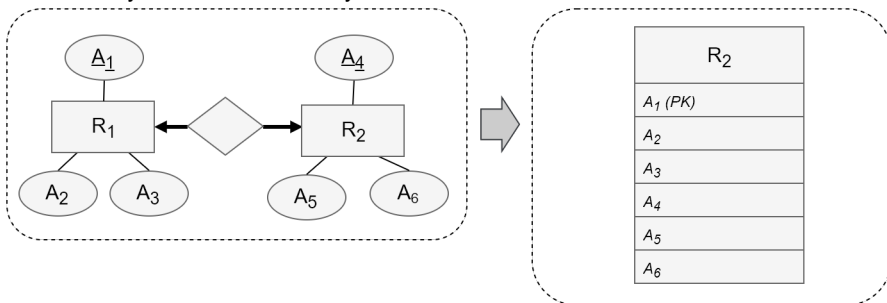
4.4.2. Зв'язок 1:1



Зв'язок 1:1

Сутності об'єднуються в одне відношення

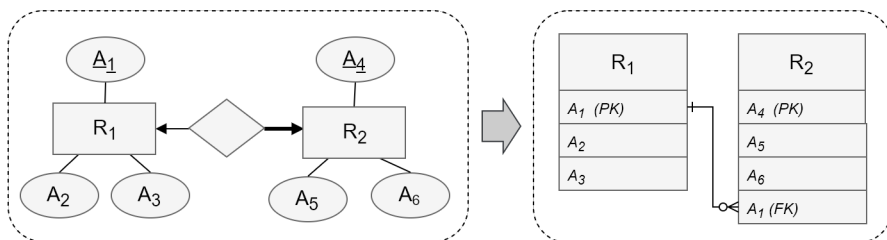
Обов'язкова участь обох боків зв'язку



Зв'язок 1:1

Передача первинного ключа на "обов'язковий" бік для використання в якості зовнішнього ключа у відношенні, що представляє "необов'язковий" бік

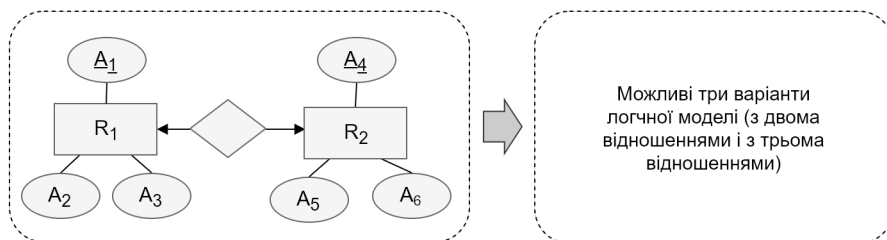
Обов'язкова участь одного боку зв'язку



Зв'язок 1:1

У разі відсутності додаткової інформації вибір буде довільним

Необов'язкова участь обох боків зв'язку



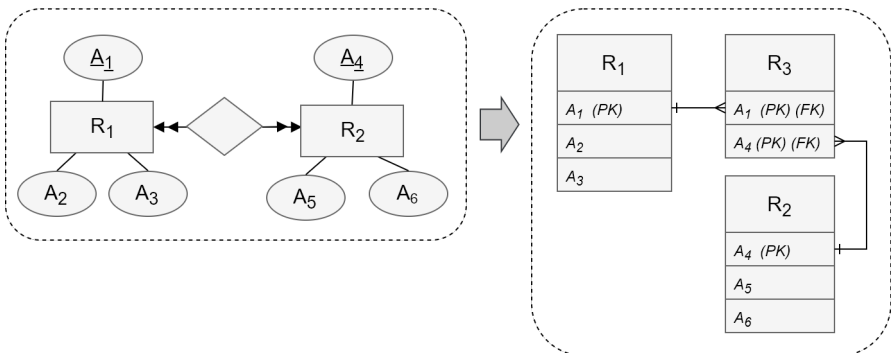


4.4.3. Зв'язок N:M та багатозначний зв'язок



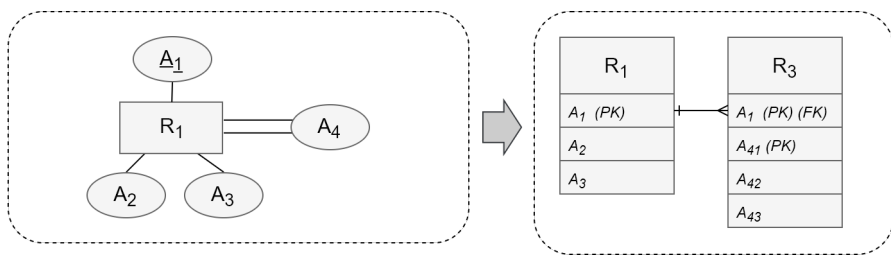
Зв'язок N:M

Створення відношення, яке являє собою зв'язок, і включення всіх атрибутів зв'язку. Передача в нове відношення копії первинного ключа з кожної сутності-володаря для використання в якості зовнішнього ключа



Багатозначний зв'язок

Створення відношення, яке являє собою багатозначний атрибут, і передає в нове відношення копії первинного ключа із сутності-володаря для використання в якості зовнішнього ключа

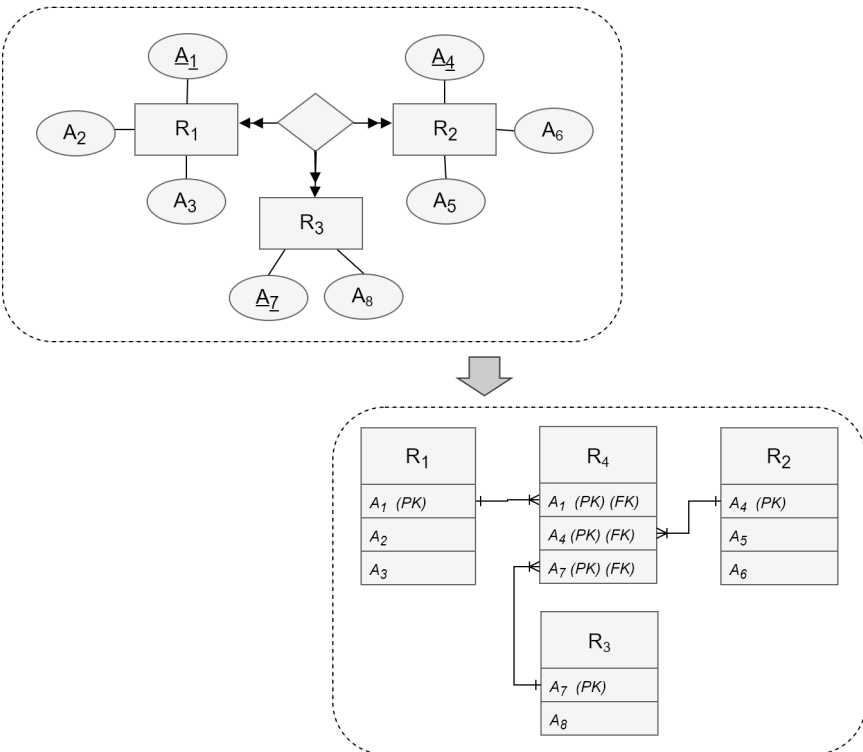




4.4.4. Складний зв'язок

Складний зв'язок

Створення відношення, яке являє собою зв'язок, і включення всіх атрибутів зв'язку. Передача в нове відношення копії первинного ключа з кожної сутності-володаря для використання в якості зовнішнього ключа



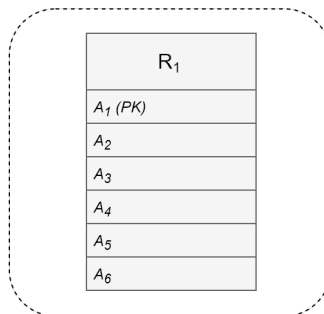
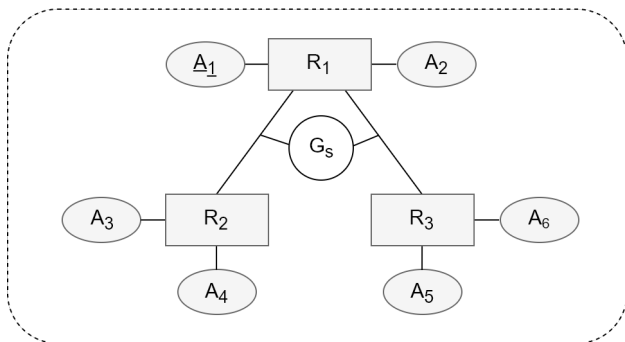


4.4.5. Зв'язок клас - підклас (Mandatory/And)



Зв'язок клас - підклас

Обов'язкова участь зв'язку суперклас/ підклас (Mandatory), дозволяється перетин підкласів (And)



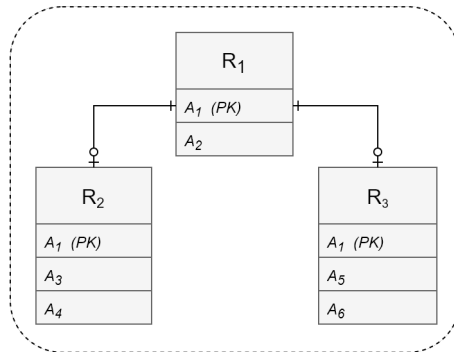
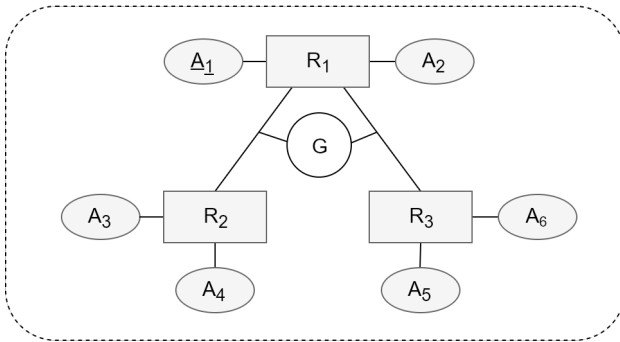


4.4.6. Зв'язок клас - підклас (Optional/Or)



Зв'язок клас - підклас

Необов'язкова участь зв'язку суперклас/підклас (Optional), не дозволяється перетин підкласів (Or)



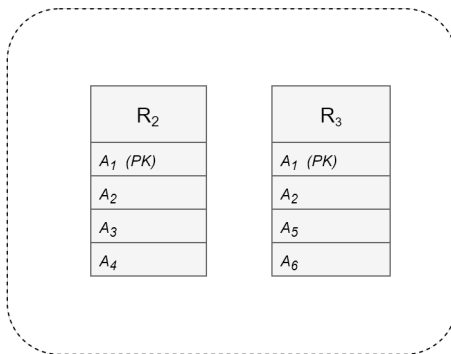
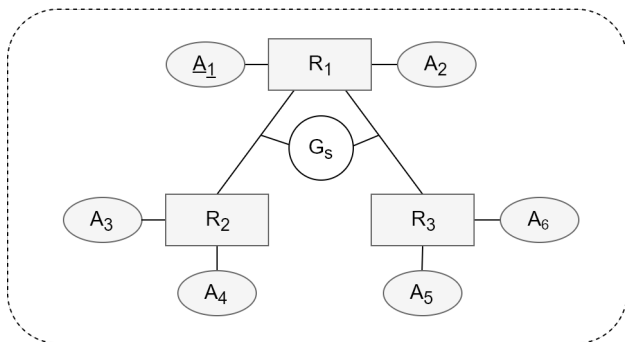


4.4.7. Зв'язок клас - підклас (Mandatory/Or)



Зв'язок клас - підклас

Обов'язкова участь зв'язку суперклас/підклас (Mandatory), не дозволяється перетин підкласів (Or)

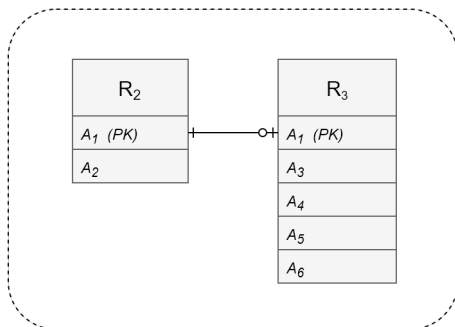
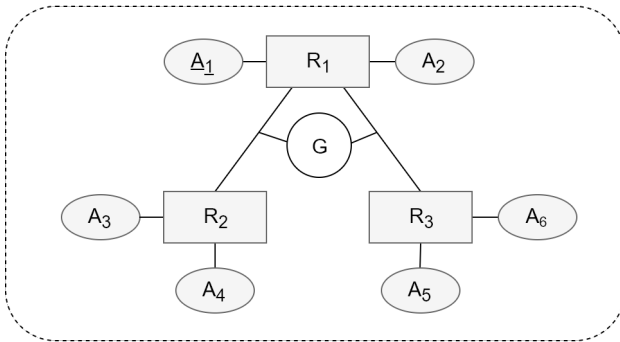




4.4.8. Зв'язок клас - підклас (Optional/And)

Зв'язок клас - підклас

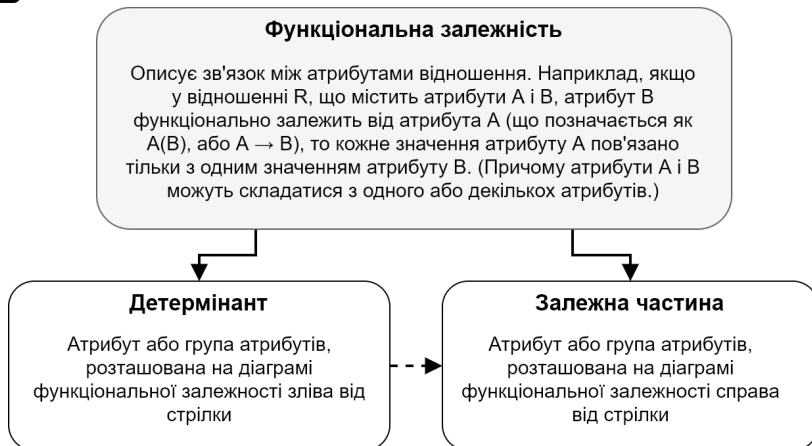
Необов'язкова участь зв'язку суперклас/ підклас (Optional), дозволяється перетин підкласів (And)





4.5. Нормалізація

4.5.1. Функціональні залежності між відношеннями



Правила виводу функціональних залежностей

Визначають способи виведення нових функціональних залежностей із заданих

Рефлексивність	якщо B є підмножиною A	\Rightarrow	то $A \rightarrow B$
Доповнення	якщо $A \rightarrow B$	\Rightarrow	то $(A, C) \rightarrow (B, C)$
Транзитивність	якщо $A \rightarrow B$ і $B \rightarrow C$	\Rightarrow	то $A \rightarrow C$
Самовизначення	$A \rightarrow A$		
Декомпозиція	якщо $A \rightarrow (B, C)$	\Rightarrow	то $A \rightarrow B$ і $A \rightarrow C$
Об'єднання	якщо $A \rightarrow B$ і $A \rightarrow C$	\Rightarrow	то $A \rightarrow (B, C)$
Композиція	якщо $A \rightarrow B$ і $C \rightarrow D$	\Rightarrow	то $(A, C) \rightarrow (B, D)$



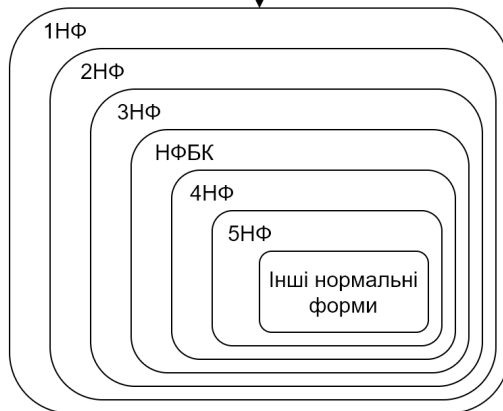
4.5.2. Поняття нормалізації відношень у реляційній моделі даних

Нормалізація

Формальний метод аналізу відношень на основі їх первинного ключа (або потенційних ключів) і існуючих функціональних залежностей. Він включає ряд правил, що можуть використовуватися для перевірки окремих відношень таким чином, щоб вся база даних могла бути нормалізована до бажаного ступеня. Якщо деяка вимога не задовольняється, то відношення, що суперечить даній вимозі має бути розділене на декілька відношень, кожне з яких (окремо) задовольняє всім вимогам нормалізації

Найчастіше нормалізація здійснюється у вигляді декількох послідовно виконуваних етапів, кожен з яких відповідає певній нормальній формі, що володіє відомими властивостями. В ході нормалізації формат відношень стає все більш обмеженим (строгим) і менш сприйнятливим до аномалій оновлення

При роботі з реляційною моделлю даних важливо розуміти, що для створення відношень прийнятної якості обов'язково тільки виконання вимог першої нормальної форми (1НФ). Всі інші форми можуть використовуватися за бажанням проектувальників. Але для того щоб уникнути аномалій оновлення, нормалізацію рекомендується виконувати як мінімум до третьої нормальної форми (3НФ)





4.5.3. Ненормалізована та перша нормальна форма (1НФ)



Ненормалізована форма (ННФ)

Відношення, що містить одну або декілька груп даних, що повторюються

Реєстр здачі в оренду								
Клієнт ID	Клієнт	Об'єкт ID	Адреса об'єкту	Оренда з	Оренда до	Оплата	Власник ID	Власник
H12	Хоменко Василь	S23	вул. Сумська, 12	1.10.2015	1.10.2016	6500	K02	Коваль Сергій
		S58	вул. Шевченка, 41	1.05.2016	1.11.2017	5800	T09	Тимчук Олег
C29	Чуб Олена	S58	вул. Правди, 80	1.07.2016	1.08.2017	7200	T09	
		S58	вул. Шевченка, 41	1.12.2017	1.12.2018	5800	T09	
		S23	вул. Сумська, 12	1.11.2016	1.10.2018	6500	K02	Коваль Сергій

Перша нормальна форма (1НФ)

Відношення, у якого на перетині кожного рядка та кожного стовпчика міститься одне і тільки одне значення

Реєстр здачі в оренду								
Клієнт ID	Клієнт	Об'єкт ID	Адреса об'єкту	Оренда з	Оренда до	Оплата	Власник ID	Власник
H12	Хоменко Василь	S23	вул. Сумська, 12	1.10.2015	1.10.2016	6500	K02	Коваль Сергій
H12	Хоменко Василь	S58	вул. Шевченка, 41	1.05.2016	1.11.2017	5800	T09	Тимчук Олег
C29	Чуб Олена	P26	вул. Правди, 80	1.07.2016	1.08.2017	7200	T09	Тимчук Олег
C29	Чуб Олена	S58	вул. Шевченка, 41	1.12.2017	1.12.2018	5800	T09	Тимчук Олег
C29	Чуб Олена	S23	вул. Сумська, 12	1.11.2016	1.10.2018	6500	K02	Коваль Сергій



4.5.4. Повна функціональна залежність та друга нормальна форма (2НФ)

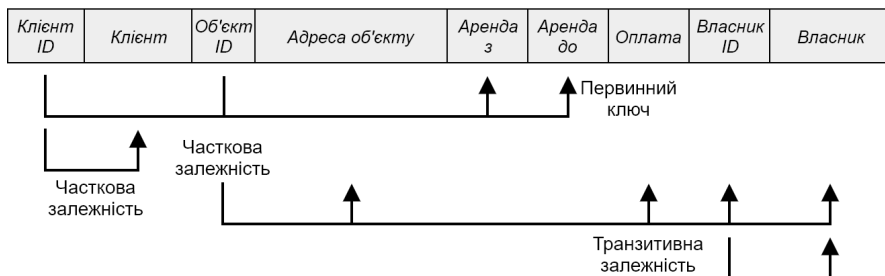
Повна функціональна залежність

Якщо А і В - атрибути відношення, то атрибут В знаходиться в повній функціональній залежності від атрибута А, якщо атрибут В є функціонально залежним від А, але не залежить від жодної власної підмножини атрибута А.

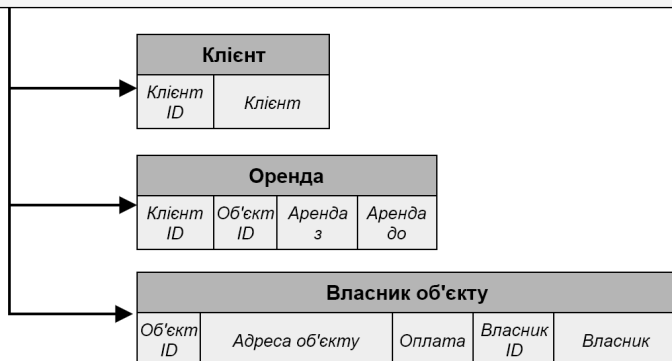


Друга нормальна форма (2НФ)

Відношення, яке знаходиться в першій нормальній формі і кожен атрибут якого, що не входить до складу первинного ключа, характеризується повною функціональною залежністю від цього первинного ключа



Для перетворення таблиці в форму 2НФ необхідно створити нові таблиці таким чином, щоб атрибути, що не входять до первинного ключа, були переміщені в них разом з копією тієї частини первинного ключа, з якою ці атрибути пов'язані повною функціональною залежністю





4.5.5. Транзитивна залежність та третя нормальна форма (ЗНФ)



Транзитивна залежність

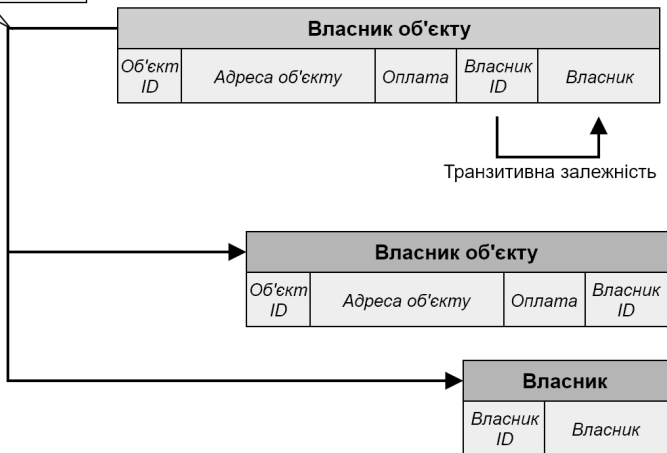
Якщо для атрибутів А, В і С деякого відношення існують залежності виду $A \rightarrow B$ і $B \rightarrow C$, це означає, що атрибут С транзитивно залежить від атрибута А через атрибут В (за умови, що атрибут А функціонально не залежить ні від атрибута В, ні від атрибута С)



Третя нормальна форма (ЗНФ)

Відношення, яке знаходиться в першій та в другій нормальних формах і не має атрибутів, що не входять до первинного ключа атрибутів, які перебували б у транзитивній функціональній залежності від цього первинного ключа

Нормалізація





4.5.6. Нормальна форма Бойса-Кодда, четверта та п'ята нормальні форми (НФБК, 4НФ, 5НФ)

Нормальна форма Бойса-Кодда (НФБК)

Відношення знаходиться в НФБК тоді і тільки тоді, коли кожен його детермінант є потенційним ключем



Різниця між ЗНФ і НФБК полягає в тому, що функціональна залежність $A \rightarrow B$ допускається щодо ЗНФ, якщо атрибут B є первинним ключем, а атрибут A не обов'язково є потенційним ключем. Тоді як щодо НФБК ця залежність допускається тільки тоді, коли атрибут A є потенційним ключем. Отже, нормальна форма Бойса-Кодда є більш строгою версією форми ЗНФ, оскільки кожне відношення НФБК є також відношенням ЗНФ, але не будь-яке відношення ЗНФ є відношенням НФБК

Багатозначна залежність

Являє собою таку залежність між атрибутами відношення (наприклад, A, B і C), що кожне значення A являє собою множину значень для A і множину значень для C . Однак множини значень для B і C не залежать один від одного



Четверта нормальна форма (4НФ)

Відношення в нормальній формі Бойса-Кодда, яке не містить нетривіальних багатозначних залежностей

Залежність з'єднань без втрат

Властивість декомпозиції, яке гарантує відсутність фіктивних рядків при відновленні первісного відношення за допомогою операції природного з'єднання



П'ята нормальна форма (5НФ)

Відношення без залежностей з'єднань без втрат



Контрольні питання

1. Опишіть етапи логічного (дatalogічного) проектування.
2. Що таке реляційна алгебра та як вона пов'язана з реляційною моделлю даних?
3. Дайте визначення операціям перейменування, вибірки та проекції, наведіть приклади.
4. Дайте визначення операціям об'єднання, різниці, декартового добутку та перетину відношень, наведіть приклади.
5. Опишіть призначення операцій з'єднання відношень.
6. Опишіть різницю між такими операціями з'єднання, як природнє з'єднання, напівз'єднання, тета-з'єднання, з'єднання за еквівалентністю та антиз'єднання.
7. Опишіть різницю між операціями зовнішнього з'єднання.
8. Назвіть основні поняття реляційної моделі даних.
9. Дайте визначення поняттю відношення реляційної моделі даних. Які особливості йому притаманні?
10. Що таке представлення та чим воно відрізняється від відношення?
11. Опишіть поняття реляційного ключа та його різновиди.
12. Що таке реляційна цілісність? Які різновиди обмежень реляційної цілісності існують?
13. Надайте загальний опис методів перетворення концептуальної моделі у логічну.
14. Надайте визначення поняття функціональної залежності.
15. Опишіть основні типи правил функціональної залежності.
16. Назвіть призначення методів нормалізації даних.
17. Назвіть характеристики таблиці в ненормалізованій формі та опишіть спосіб перетворення такої таблиці до відношення у першій нормальній формі.
18. Назвіть нормальну форму, якій, як мінімум, повинно задовільняти кожне відношення. Надайте визначення такої форми.
19. Сформулюйте поняття повної функціональної залежності. Як воно пов'язане з другою нормальною формою? Наведіть



Контрольні питання

приклад.

20. Сформулюйте поняття транзитивної залежності. Як воно пов'язане з третьою нормальною формою? Наведіть приклад.
21. Опишіть призначення нормальної форми Бойса-Кодда. В чому полягає різниця між нею та третьою нормальною формою?



Завдання для самостійної роботи

1. Перетворіть інфологічну модель «Бібліотека» із завдання №1 до теми 3 у даталогічну реляційну модель та здійсніть її нормалізацію мінімум до 3НФ.
2. Перетворіть інфологічну модель «Школа» із завдання №2 до теми 3 у даталогічну реляційну модель та здійсніть її нормалізацію мінімум до 3НФ.
3. Перетворіть інфологічну модель «Супермаркет» із завдання №3 до теми 3 у даталогічну реляційну модель та здійсніть її нормалізацію мінімум до 3НФ.
4. Перетворіть інфологічну модель «Інтернет-магазин» із завдання №4 до теми 3 у даталогічну реляційну модель та здійсніть її нормалізацію мінімум до 3НФ.
5. Перетворіть інфологічну модель «Кінотеатр» із завдання №5 до теми 3 у даталогічну реляційну модель та здійсніть її нормалізацію мінімум до 3НФ.
6. Перетворіть інфологічну модель «Лікарня» із завдання №6 до теми 3 у даталогічну реляційну модель та здійсніть її нормалізацію мінімум до 3НФ.
7. Перетворіть інфологічну модель «Туристична агенція» із завдання №7 до теми 3 у даталогічну реляційну модель та здійсніть її нормалізацію мінімум до 3НФ.

Тема 5

ПРОЕКТУВАННЯ ТА РОЗРОБКА БАЗ ДАНИХ НА ФІЗИЧНОМУ РІВНІ



МЕТА

Оволодіння знаннями про принципи фізичного проектування баз даних, мову запитів до баз даних SQL, вміннями реалізувати даталогічну модель бази даних на фізичному рівні за допомогою мови SQL



ЗМІСТ

- 5.1. Фізичне проектування баз даних**
- 5.2. Мова запитів до баз даних SQL**
 - 5.2.1. Загальна структура мови
 - 5.2.2. Нотація посібника для лексичних діаграм SQL
 - 5.2.3. Основні типи даних SQL
 - 5.2.4. Команда створення таблиць
 - 5.2.5. Команда оновлення структури таблиці
 - 5.2.6. Команда перейменування, видалення таблиці та видалення її змісту
 - 5.2.7. Команда вибірки даних з таблиці
 - 5.2.8. Команди вставки, оновлення та видалення даних з таблиці



5.1. Фізичне проектування баз даних

Фізичне проектування

Прийняття рішень про те, як логічна модель буде фізично реалізована в базі даних, що створюється з використанням обраної СКБД

Фізична модель залежить як від програмного забезпечення (СКБД, операційна система), так і від обладнання (комп'ютери, носії інформації, мережева інфраструктура). Вона визначає не тільки різні типи записів даних, але також існуючі індекси, способи подання полів, фізичну впорядкованість записів та ін.

Перенесення глобальної логічної моделі даних в середовище цільової СКБД

Реалізація обмежень предметної області

Аналіз транзакцій

Вибір файлової структури

Визначення індексів

Визначення вимог до засобів довготривалого зберігання інформації

Визначення вимог до кешування даних

Розробка механізмів захисту даних

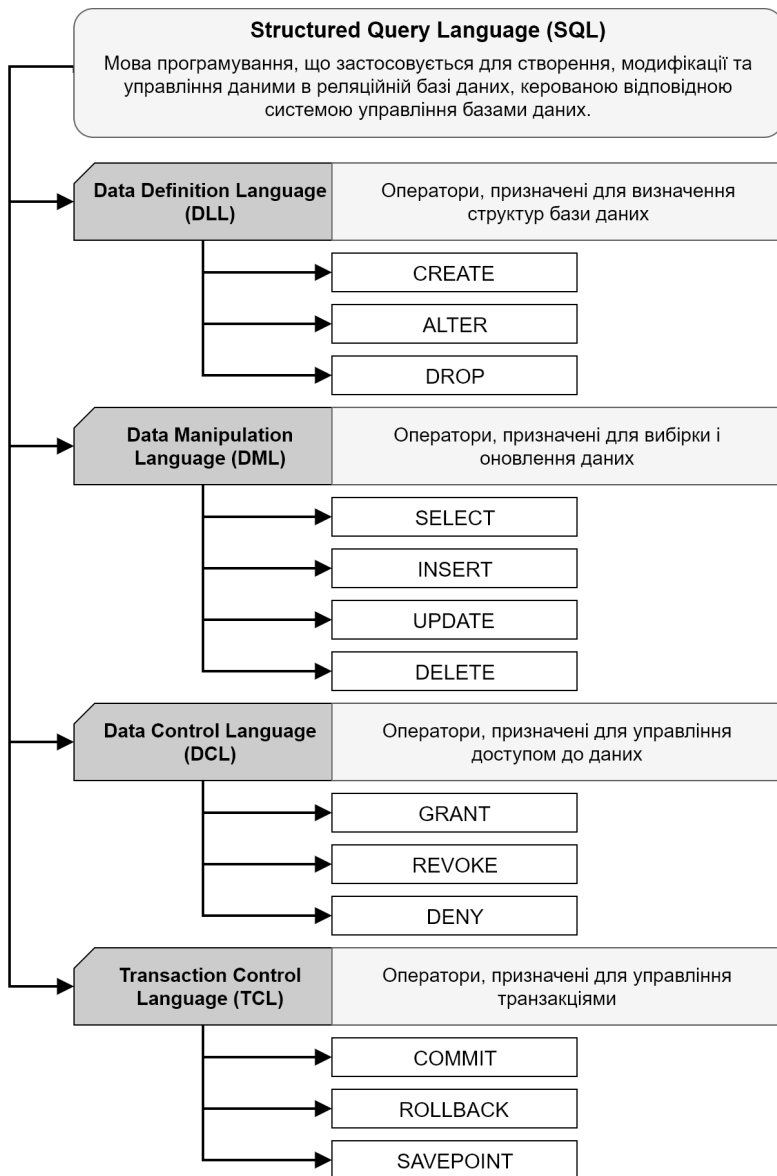
Аналіз необхідності впровадження контрольованої надлишковості

Організація моніторингу та налаштування функціонування операційної системи




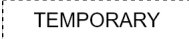
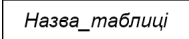
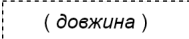
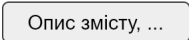
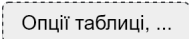
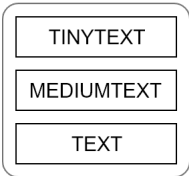


5.2. Мова запитів до баз даних SQL

5.2.1. Загальна структура мови



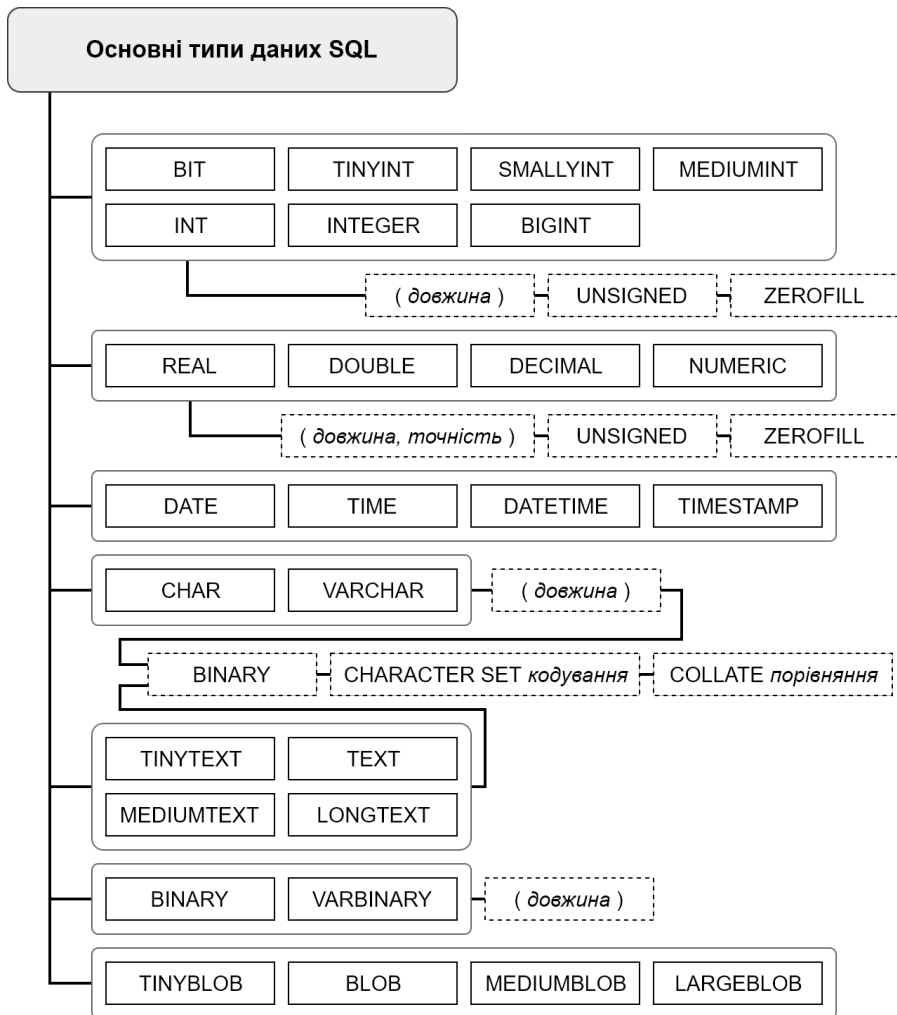


5.2.2. Нотація посібника для лексичних діаграм SQL

 CREATE	Обов'язковий оператор або ключове слово
 TEMPORARY	Необов'язковий оператор або ключове слово
 <i>Назва_таблиці</i>	Обов'язкове значення
 (<i>довжина</i>)	Необов'язкове значення
 Опис змісту, ...	Обов'язкова іменована лексична група. Може повторюватись, якщо стоять три крапки
 Опції таблиці, ...	Необов'язкова іменована лексична група. Може повторюватись, якщо стоять три крапки
	Вибір одного варіанта з декількох
	Послідовність лексем
	Зв'язок між назвою лексичної групи та її змістом

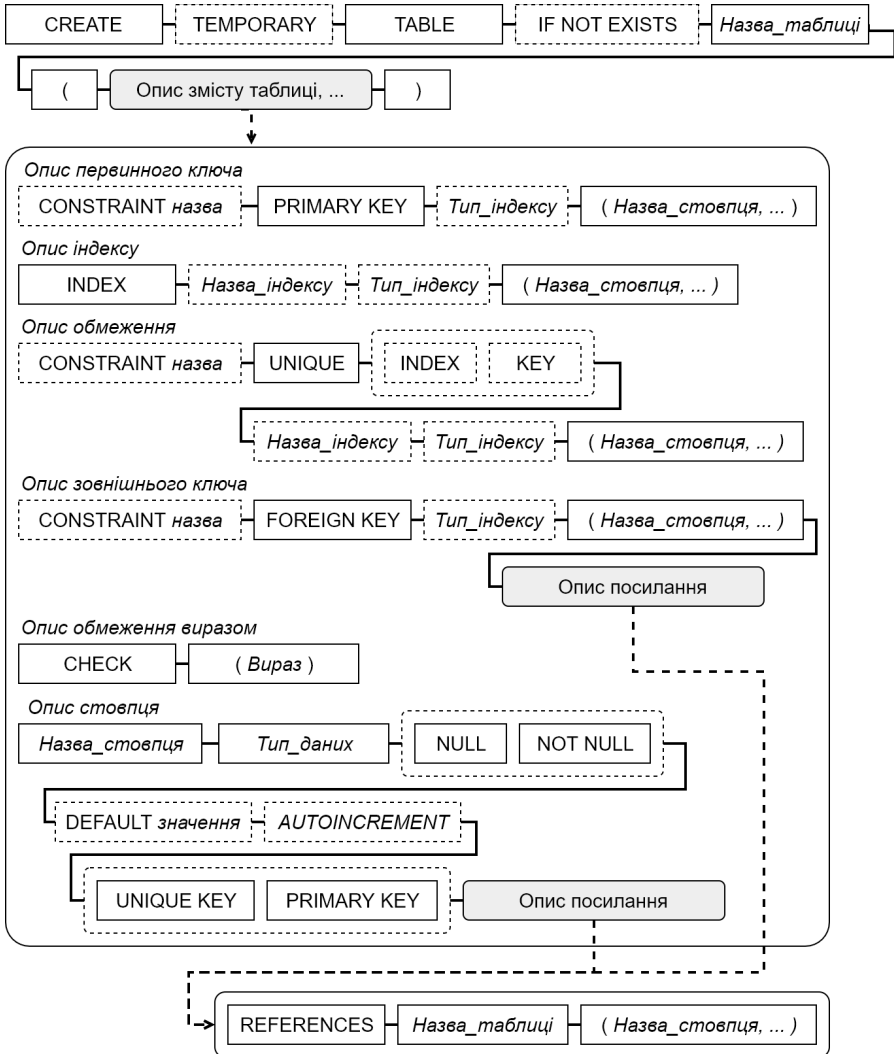


5.2.3. Основні типи даних SQL



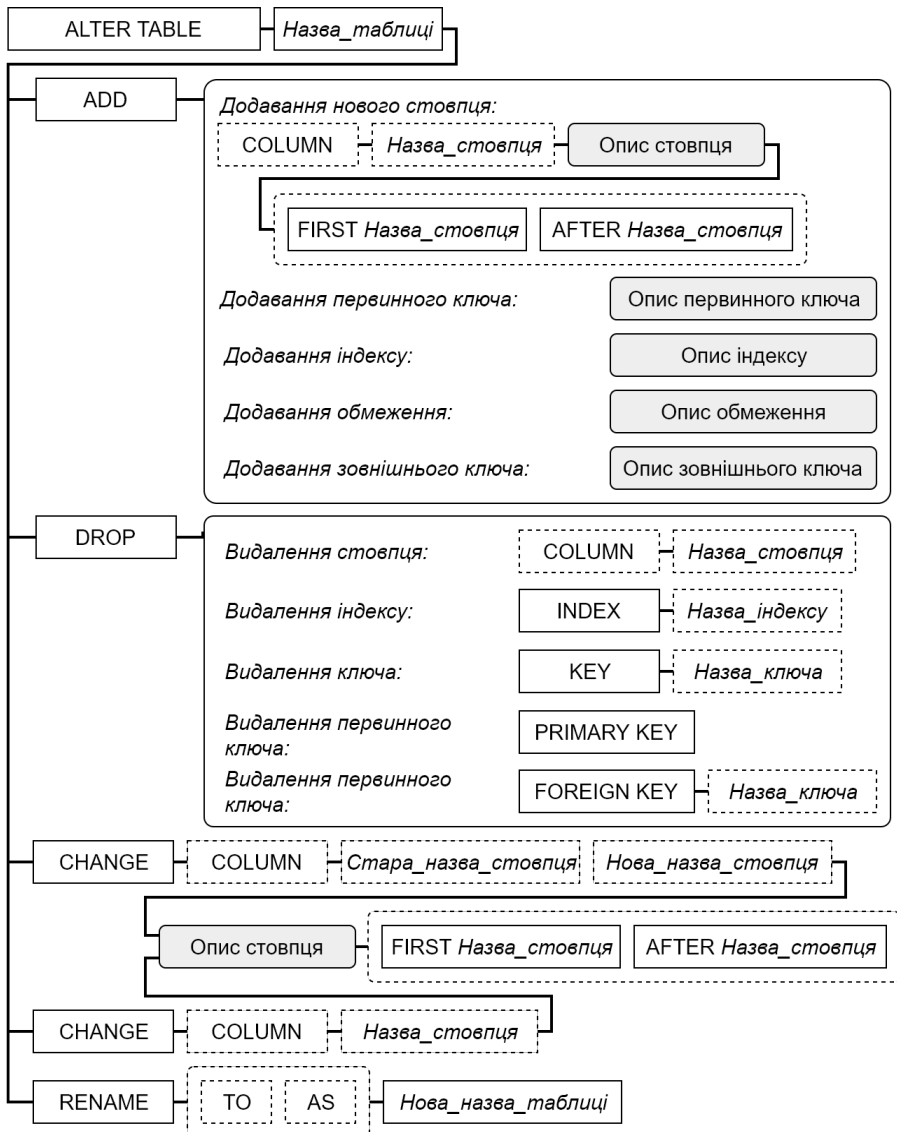


5.2.4. Команда створення таблиць



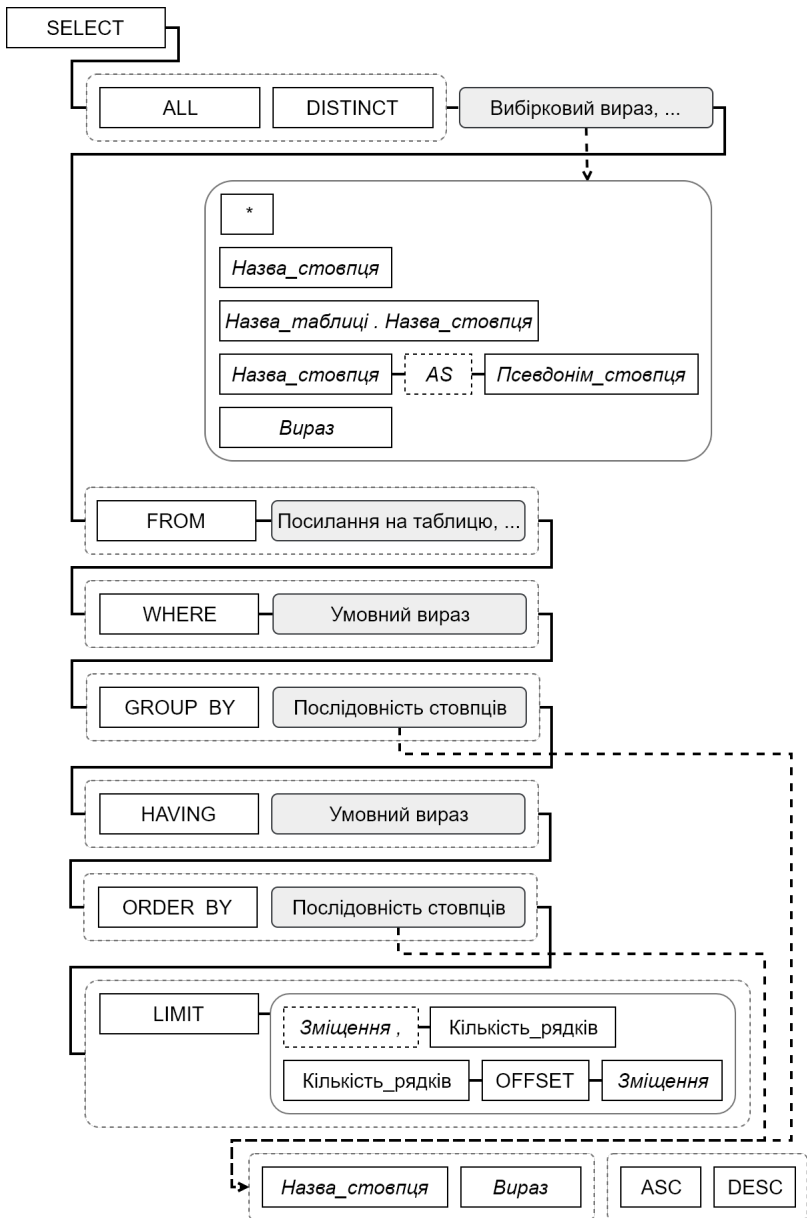


5.2.5. Команда оновлення структури таблиці



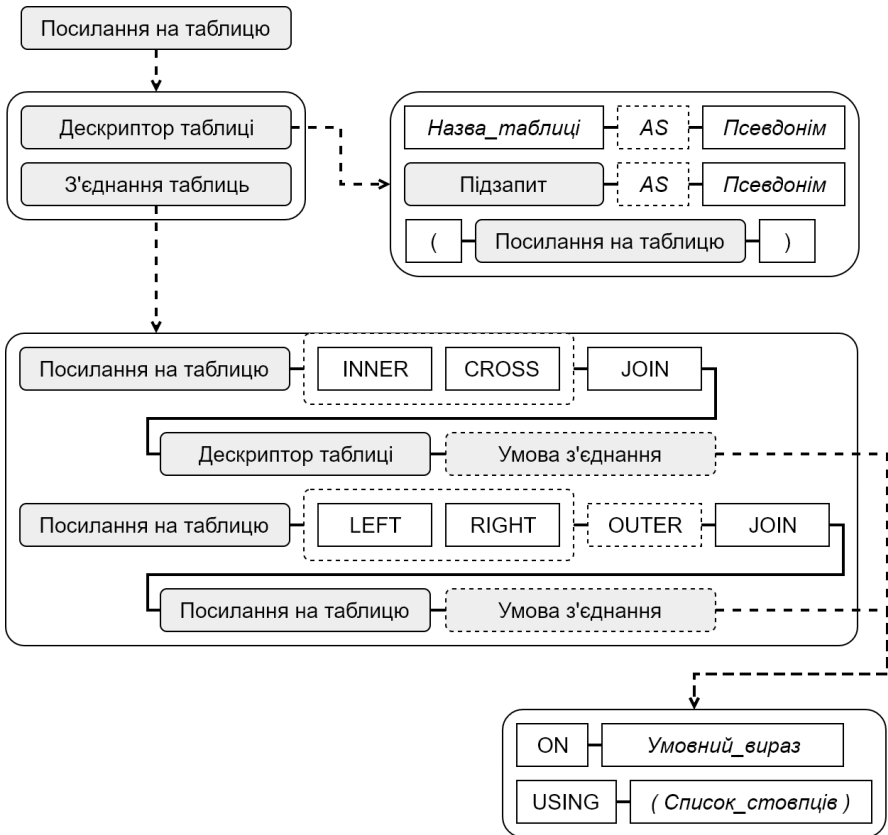


5.2.7 . Команда вибірки даних з таблиці



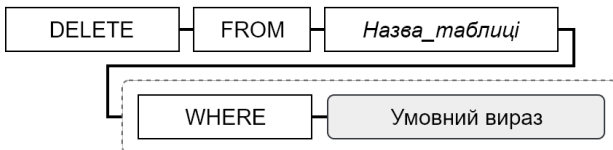
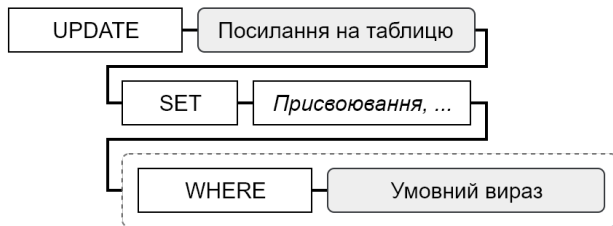
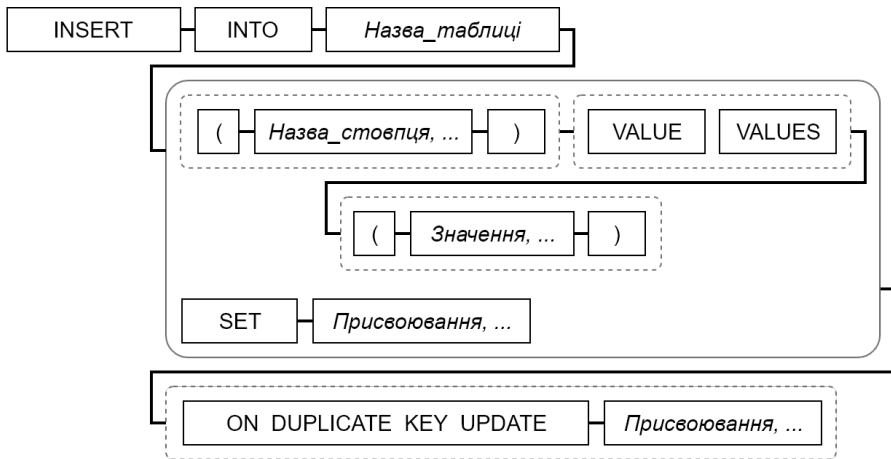


(Продовження опису команди вибірки даних з таблиці)





5.2.8. Команди вставки, оновлення та видалення даних з таблиці





Контрольні питання

1. Назвіть етапи фізичного проектування бази даних
2. Дайте визначення поняттю SQL.
3. Яку структуру має мова SQL?
4. Які основні типи даних підтримує мова SQL?
5. Опишіть синтаксис та можливості команди створення таблиць.
6. Опишіть синтаксис та можливості команди оновлення таблиць.
7. Опишіть синтаксис та можливості команд перейменування, видалення таблиць і видалення їх змісту.
8. Опишіть синтаксис та можливості команди вибірки даних з таблиці.
9. Опишіть синтаксис та можливості команди вибірки даних з декількох таблиць.
10. Опишіть синтаксис та можливості команд вставки, оновлення та видалення даних з таблиці.



Завдання для самостійної роботи

1. Перетворіть реляційну модель «Бібліотека» із завдання №1 до теми 4 у послідовність SQL-команд. Створіть команди для заповнення бази тестовими даними.
2. Перетворіть реляційну модель «Школа» із завдання №2 до теми 4 у послідовність SQL-команд. Створіть команди для заповнення бази тестовими даними.
3. Перетворіть реляційну модель «Супермаркет» із завдання №3 до теми 4 у послідовність SQL-команд. Створіть команди для заповнення бази тестовими даними.
4. Перетворіть реляційну модель «Інтернет-магазин» із завдання №4 до теми 4 у послідовність SQL-команд. Створіть команди для заповнення бази тестовими даними.



5. Перетворіть реляційну модель «Кінотеатр» із завдання №5 до теми 4 у послідовність SQL-команд. Створіть команди для заповнення бази тестовими даними.
6. Перетворіть реляційну модель «Лікарня» із завдання №6 до теми 4 у послідовність SQL-команд. Створіть команди для заповнення бази тестовими даними.
7. Перетворіть реляційну модель «Туристична агенція» із завдання №7 до теми 4 у послідовність SQL-команд. Створіть команди для заповнення бази тестовими даними.

Тема 6

СИСТЕМИ КЕРУВАННЯ БАЗАМИ ДАНИХ



МЕТА

Оволодіння знаннями про функції та структуру систем керування базами даних, паралельну обробку даних, транзакції, розподілені бази даних, технології обробки даних OLAP та OLTP, сховища даних, вміннями здійснювати паралельну обробку даних у базах даних



ЗМІСТ

- 6.1. Функції системи керування базами даних**
- 6.2. Структура системи керування базами даних**
- 6.3. Паралельна обробка даних. Транзакції**
 - 6.3.1. Поняття транзакції
 - 6.3.2. Керування паралельним доступом
 - 6.3.3. Рівні ізолюваності транзакцій
 - 6.3.4. Відновлення бази даних
 - 6.3.5. Механізми відновлення баз даних після збоїв
- 6.4. Розподілені бази даних та системи керування базами даних**
 - 6.4.1. Основні поняття
 - 6.4.2. Структурна схема розподіленої бази даних
 - 6.4.3. Приклад архітектури розподіленої СКБД
- 6.5. Технологія обробки інформації OLAP**
- 6.6. Технологія обробки транзакцій OLTP**
- 6.7. Сховища даних**
- 6.8. Ретроспективний аналіз розвитку баз даних та систем керування базами даних**
- 6.9. Перспективи розвитку баз даних та систем керування базами даних**

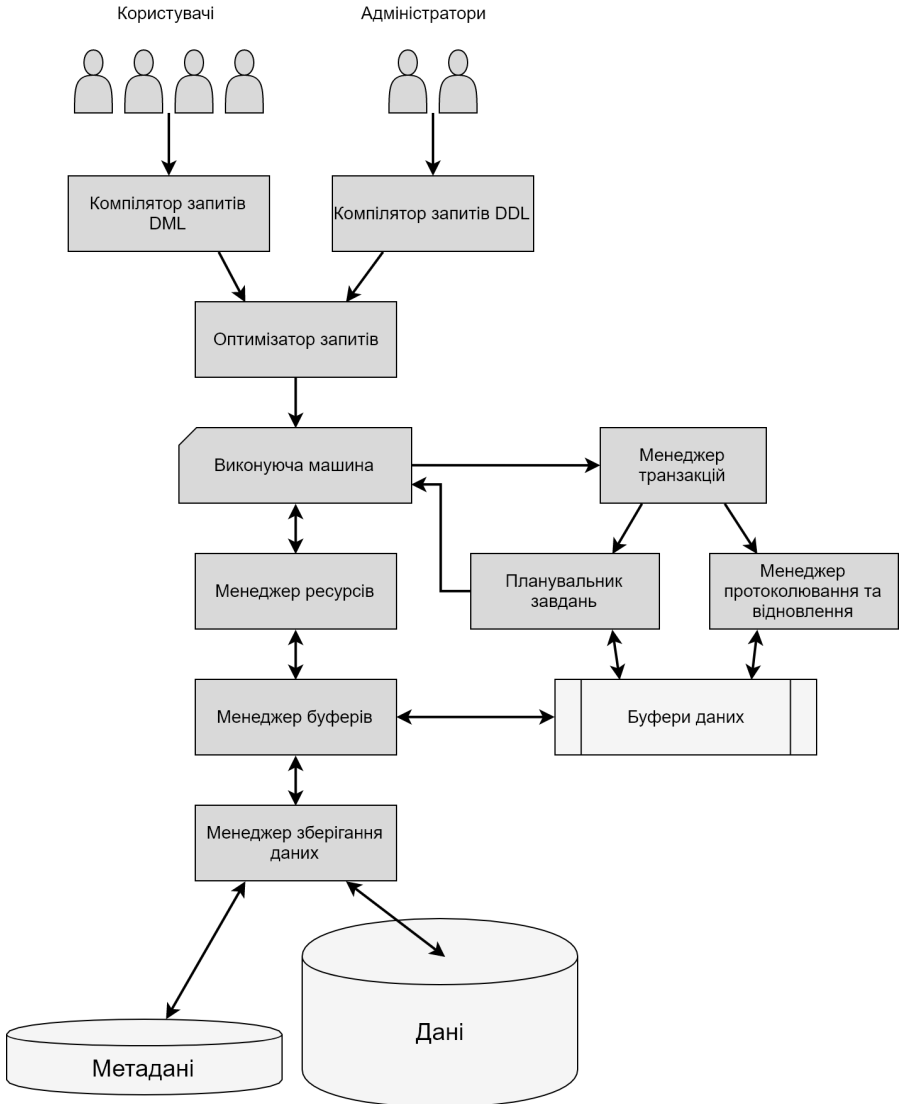


6.1. Функції системи керування базами даних





6.2. Структура системи керування базами даних





6.3. Паралельна обробка даних. Транзакції

6.3.1. Поняття транзакції

Транзакція

Група послідовних операцій з базою даних, яка являє собою логічну одиницю роботи з даними. Транзакція може бути виконана або цілком і успішно, дотримуючись цілісності даних і незалежно від інших транзакцій, що виконуються паралельно, або не виконана взагалі, і тоді вона не повинна призвести ніякого ефекту

Властивості транзакцій (ACID)

Атомарність <i>Atomicity</i>	Будь-яка транзакція є неподільною одиницею роботи, яка може бути або виконана вся цілком, або не виконана взагалі. За забезпечення атомарності відповідає підсистема відновлення СУБД
Узгодженість <i>Consistency</i>	Система повинна перебувати в узгодженому, несуперечливому стані до початку дії транзакції і по її завершенню. При цьому вона може перебувати в неузгодженому стані протягом виконання транзакції, проте ця неузгодженість завдяки іншим властивостям — атомарності та ізолюваності — не буде видимою за межами транзакції
Ізолюваність <i>Isolation</i>	Жодні проміжні зміни не повинні бути видимі за межами транзакції аж до її завершення. Питання ізоляції стає актуальним при одночасній роботі багатьох транзакцій з одними й тими самими даними. Згідно з цією вимогою, якщо дві транзакції намагатимуться змінити одні й ті самі дані, то одну з них буде відхилено або призупинено до завершення другої
Довговічність <i>Durability</i>	Гарантує, що незалежно від інших проблем після відновлення працездатності системи результати завершених транзакцій будуть збережені. Іншими словами, якщо користувач отримав повідомлення про успішне завершення транзакції, то він може бути впевнений, що дані будуть збережені та відновлені у випадку збоїв



6.3.2. Керування паралельним доступом

Керування паралельним доступом

Процес організації одночасного виконання в базі даних різних операцій доступу, що гарантує запобігання їх впливу одна на одну



Виконання операцій транзакцій чергується і забезпечується їх паралельне виконання, що призводить до підвищення загальної продуктивності системи та надає можливості багатьом користувачам виконувати свої операції з базою даних одночасно.

Незважаючи на те, що кожна з транзакцій може сама по собі виконуватися цілком коректно, подібне чергування операцій здатне приводити до невірних результатів, через що цілісність і узгодженість бази даних будуть порушені

Можливі проблеми паралельного виконання транзакцій

Втрачене оновлення	При одночасній зміні одного блоку даних різними транзакціями, одна із змін втрачається
«Брудне» читання	Читання даних, які додані чи змінені транзакцією, що потім не підтвердиться (відкотиться)
Неповторюване читання	При повторному читанні в рамках однієї транзакції, раніше прочитані дані з'являються зміненими
Фантомне читання	Одна транзакція в ході свого виконання декілька разів вибирає множину рядків за одними і тими ж критеріями. Інша транзакція в інтервалах між цими вибірками додає чи видаляє рядки або змінює стовпці деяких рядків, що використовується в критеріях вибірки першої транзакції, і успішно закінчується. В результаті отримуємо, що одні і ті ж вибірки в першій транзакції дають різні множини рядків



6.3.3. Рівні ізолюваності транзакцій

Рівень ізолюваності транзакцій

Значення, що задає рівень, при якому в транзакції дозволяються неузгоджені дані, тобто ступінь ізолюваності однієї транзакції від іншої. Більш високий рівень ізолюваності підвищує точність даних, але при цьому може знижуватись кількість транзакцій, що виконуються паралельно. З іншого боку, більш низький рівень ізолюваності дозволяє виконувати більше паралельних транзакцій, але знижує точність даних

Рівні ізоляції транзакцій згідно зі стандартом SQL-92

Впорядкованість (рівень 3)	Найбільш високий рівень ізолюваності; транзакції повністю ізолюються одна від одної. На цьому рівні результати паралельного виконання транзакцій для бази даних у більшості випадків можна вважати такими, що збігаються з послідовним виконанням тих же транзакцій (по черзі в будь-якому порядку)
Повторюваність читання (рівень 2)	Рівень, при якому читання одного і того ж рядку чи рядків в транзакції дає однаковий результат. (Поки транзакція не закінчена, ніякі інші транзакції не можуть змінити ці дані)
Читання фіксованих даних (рівень 1)	Закінчене читання, при якому відсутнє «брудне» читання (тобто, читання одним користувачем даних, що не були зафіксовані в БД командою COMMIT). Проте, в процесі роботи однієї транзакції інша може бути успішно закінчена, і зроблені нею зміни зафіксовані. В підсумку, перша транзакція буде працювати з іншим набором даних. Це проблема неповторюваного читання
Читання незафіксованих даних (рівень 0)	Найнижчий рівень ізоляції. Він гарантує тільки відсутність втрачених оновлень. Якщо декілька транзакцій одночасно намагались змінювати один і той же рядок, то в кінцевому варіанті рядок буде мати значення, визначений останньою успішно виконаною транзакцією



6.3.4. Відновлення бази даних

Відновлення бази даних

Процес повернення бази даних до узгодженого стану, що був втрачений внаслідок збою або відмовлення

Деякі причини, що здатні визвати відмови





6.3.5. Механізми відновлення баз даних після збоїв

Механізм резервного копіювання

Призначений для періодичного створення резервних копій бази даних

Резервна копія бази даних використовується в разі пошкодження або руйнування файлів бази даних у вторинній пам'яті. Резервне копіювання може виконуватися для всієї бази даних в цілому або для частини, що змінилася (тобто інкрементно). В останньому випадку копія буде містити відомості тільки про зміни, що накопилися з моменту створення попередньої повної або інкрементної резервної копії системи

Журнал транзакцій

Фіксує поточний стан транзакцій і зміни, що вносяться в базу даних

Логічна відмова

Відкат (Rollback)

Журнал сканується в зворотному напрямку, і всі записи транзакції, що скасовуються, витягуються з журналу аж до позначки початку транзакції. Згідно витягнутої інформації виконуються дії, що скасовують дії транзакції, а в журнал записуються компенсуючі записи

Фізична відмова

Прогін (Rollforward)

Журнал сканується в прямому напрямку, починаючи від попередньої контрольної точки. Всі записи витягуються з журналу аж до кінця журналу та послідовно виконуються

Контрольні точки

Момент синхронізації між базою даних і журналом реєстрації транзакцій. У цей момент всі буфери системи примусово записуються у вторинну пам'ять системи

Диспетчер відновлення

Забезпечує відновлення узгодженого стану бази даних, порушеного в результаті відмови



6.4. Розподілені бази даних та системи керування базами даних

6.4.1. Основні поняття

Розподілена база даних

Набір логічно пов'язаних між собою сукупностей даних (і їх описів), що розділяються, які фізично розподілені в деякій комп'ютерній мережі.

Розподілена СКБД

Програмний комплекс, призначений для управління розподіленими базами даних, що забезпечує прозорий доступ користувачів до розподіленої інформації

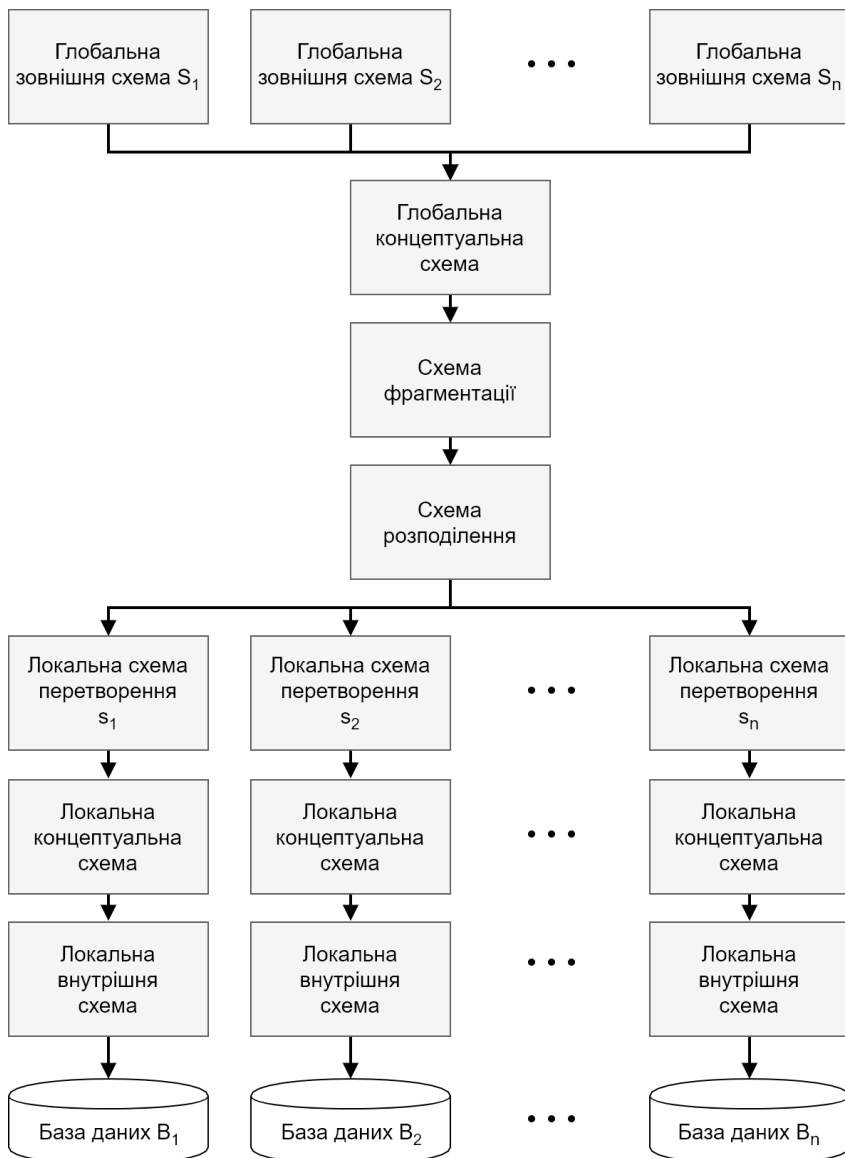
Для кінцевого користувача розподілена база даних повинна мати вигляд та функціонувати так, як нерозподілена, тобто створювати ілюзію звичайної нерозподіленої БД

Переваги розподілених СКБД



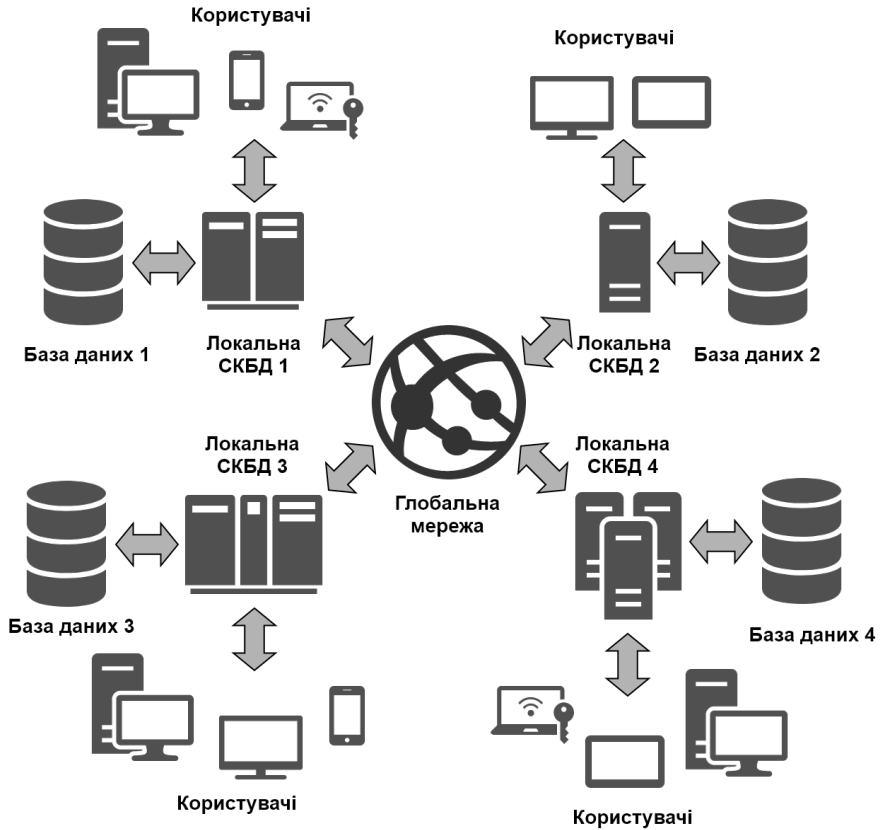


6.4.2. Структурна схема розподіленої бази даних





6.4.3. Приклад архітектури розподіленої СКБД





6.5. Технологія обробки інформації OLAP

OLAP

Online Analytical Processing — аналітична обробка в реальному часі. Технологія обробки інформації, що дозволяє швидко отримувати відповіді на багатовимірні аналітичні запити. Бази даних, сконфігуровані для OLAP, використовують багатовимірні моделі даних, що дозволяє виконувати складні аналітичні та спеціалізовані запити за короткий проміжок часу. Вони запозичують окремі аспекти навігаційних та ієрархічних баз даних, які є швидшими за реляційні БД.

Головна причина використання OLAP для обробки запитів — це швидкість. Реляційні БД зберігають сутності в окремих таблицях, які зазвичай добре нормалізовані. Ця структура зручна для операційних БД (системи OLTP), але складні багатовимірні запити в ній виконуються відносно повільно. Зручнішою моделлю для виконання запитів (але не для внесення змін) є просторова БД. OLAP робить миттєвий знімок реляційної БД і структурує її в просторову модель для запитів. Заявлений час обробки запитів в OLAP становить близько 0,1% від аналогічних запитів до реляційної БД.



OLAP-куб

Ядром будь-якої OLAP-системи є ідея OLAP-куба (багатовимірний куб, або гіперкуб). OLAP-структура, створена з робочих даних, називається OLAP-кубом. Він складається з чисельних фактів (розмірів), розподілених за вимірами. Зазвичай куб створюється за допомогою з'єднання таблиць із застосуванням схеми «зірка», або схеми «сніжинка». В центрі «зірки» знаходиться таблиця, яка містить ключові факти, за якими робляться запити. Множинні таблиці з вимірами приєднані до таблиці фактів. Ці таблиці показують, як можуть аналізуватися агреговані реляційні дані.

Типи

Багатовимірна OLAP (Multidimensional OLAP — MOLAP)

Реляційна OLAP (Relational OLAP — ROLAP)

Гібридна OLAP (Hybrid OLAP — HOLAP)



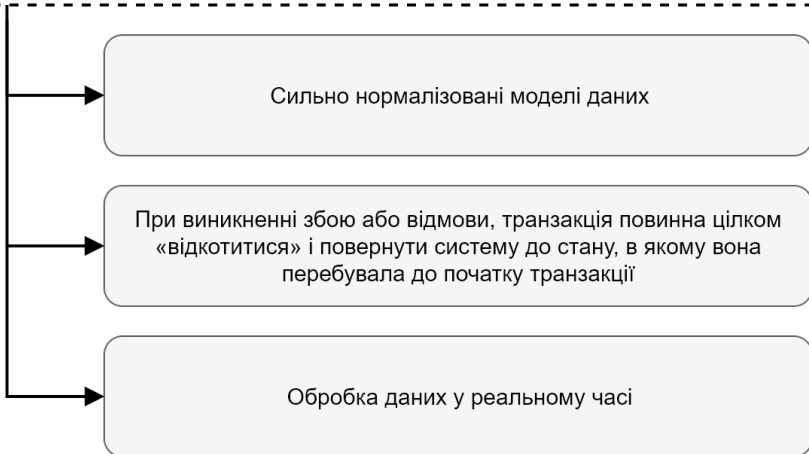
6.6. Технологія обробки транзакцій OLTP

OLTP

Online Transaction Processing — онлайнна обробка транзакцій. Спосіб організації БД, при якому система працює з невеликими за розмірами транзакціями, що йдуть великим потоком, і при цьому клієнту потрібний від системи максимально швидкий час відповіді

OLTP-системи проектуються, налаштовуються і оптимізуються для виконання максимальної кількості транзакцій за короткі проміжки часу. Показником ефективності є кількість транзакцій, що виконуються за секунду. Зазвичай аналітичні можливості OLTP-систем сильно обмежені (або взагалі відсутні)

Вимоги





6.7. Сховища даних

Сховище даних

Предметно орієнтований, інтегрований, незмінний набір даних, що підтримує хронологію, і здатний бути комплексним джерелом достовірної інформації для оперативного аналізу та прийняття рішень

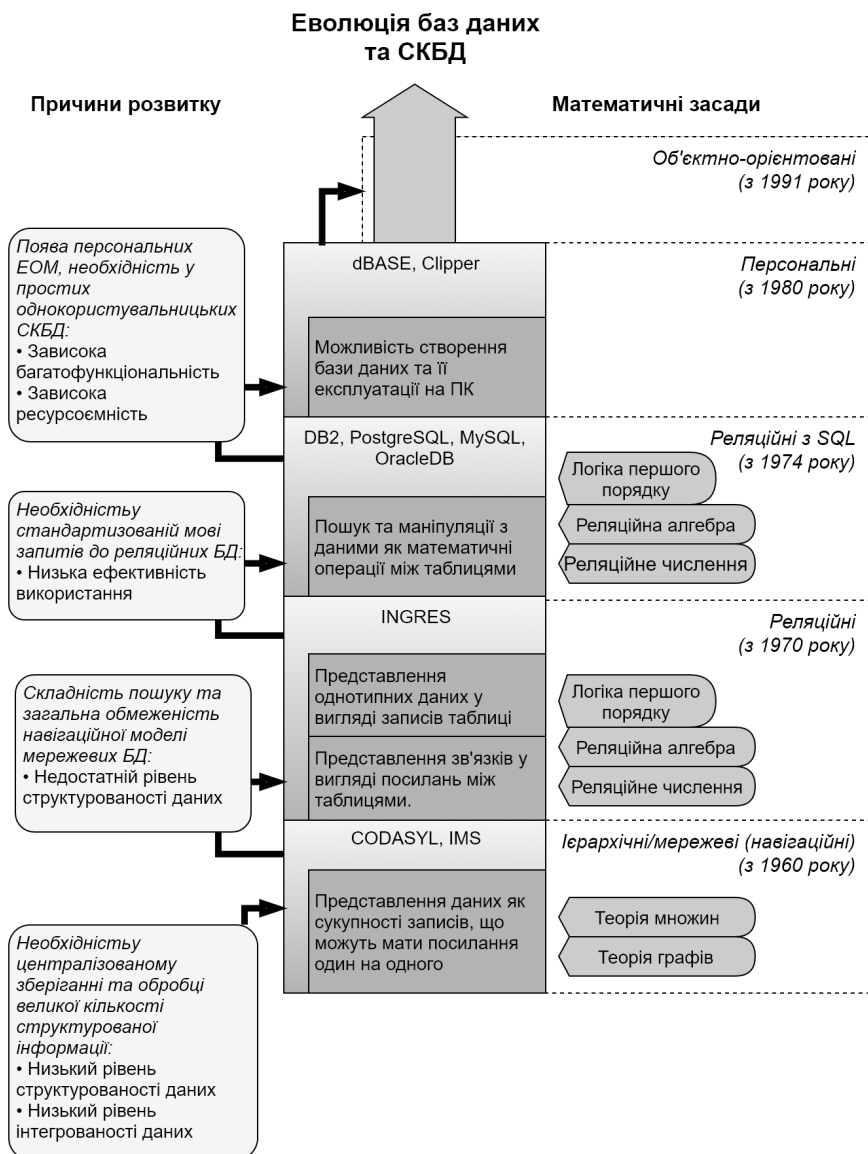
В основі концепції сховища даних (СД) лежить розподіл інформації, що використовують в системах оперативної обробки даних (OLTP) і в системах підтримки прийняття рішень (СППР). Такий розподіл дозволяє оптимізувати як структури даних оперативного зберігання для виконання операцій введення, модифікації, знищення та пошуку, так і структури даних, що використовуються для аналізу. В СППР ці два типи даних називаються відповідно оперативними джерелами даних (ОДД) та сховищем даних

Принципи організації сховищ

<p>Проблемно-предметна орієнтованість</p>	<p>Дані об'єднуються в категорії і зберігаються відповідно до областей, які вони описують, а не до додатків, які вони використовують</p>
<p>Інтегрованість</p>	<p>Дані об'єднані так, щоб вони задовольняли всім вимогам підприємства в цілому, а не єдиній функції бізнесу</p>
<p>Незмінність</p>	<p>Дані в сховищі даних не створюються, тобто надходять із зовнішніх джерел, не коригуються і не видаляються</p>
<p>Залежність від часу</p>	<p>Дані в сховищі є точними і коректними тільки в тому випадку, коли вони прив'язані до певного проміжку або моменту часу</p>



6.8. Ретроспективний аналіз розвитку баз даних та систем керування базами даних





6.9. Перспективи розвитку баз даних та систем керування базами даних





Контрольні питання

1. Назвіть основні функції системи керування базами даних.
2. Назвіть основні компоненти системи бази даних і поясніть функцію кожної з них.
3. Дати визначення транзакції та її основним властивостям.
4. Що таке паралельний доступ до бази даних та яким чином він здійснюється?
5. Перелічте та опишіть можливі проблеми паралельного виконання транзакцій.
6. Що таке рівні ізолюваності транзакцій? Опишіть ці рівні.
7. Які причини здатні визвати відмову або збій у базі даних? Що таке відновлення бази даних?
8. Опишіть механізми відновлення бази даних після збоїв.
9. Дайте визначення розподіленій базі даних та розподіленій системі керування базами даних. У чому полягає різниця цих двох понять?
10. Назвіть основні переваги розподілених систем керування базами даних.
11. Опишіть приблизну схему розподіленої бази даних.
12. Надайте характеристику технології обробки інформації OLAP.
13. Що таке OLAP-куб?
14. Надайте характеристику технології обробки інформації OLTP.
15. Порівняйте задачі, що виконують системи OLAP та OLTP.
16. Що таке сховище даних та які функції воно виконує?
17. Наведіть приклади задач, що потребують використання сховища даних.
18. Назвіть та опишіть найбільш перспективні напрямки розвитку баз даних та систем керування базами даних.



Завдання для самостійної роботи

1. Проаналізуйте схему прикладу архітектури розподіленої СКБД на ст. 119. На її основі створіть можливу схему розподіленої системи керування базою даних «Бібліотека» із завдання №1 до теми 5.
2. Проаналізуйте схему прикладу архітектури розподіленої СКБД на ст. 119. На її основі створіть можливу схему розподіленої системи керування базою даних «Школа» із завдання №2 до теми 5.
3. Проаналізуйте схему прикладу архітектури розподіленої СКБД на ст. 119. На її основі створіть можливу схему розподіленої системи керування базою даних «Супермаркет» із завдання №3 до теми 5.
4. Проаналізуйте схему прикладу архітектури розподіленої СКБД на ст. 119. На її основі створіть можливу схему розподіленої системи керування базою даних «Інтернет-магазин» із завдання №4 до теми 5.
5. Проаналізуйте схему прикладу архітектури розподіленої СКБД на ст. 119. На її основі створіть можливу схему розподіленої системи керування базою даних «Кінотеатр» із завдання №5 до теми 5.
6. Проаналізуйте схему прикладу архітектури розподіленої СКБД на ст. 119. На її основі створіть можливу схему розподіленої системи керування базою даних «Лікарня» із завдання №6 до теми 5.
7. Проаналізуйте схему прикладу архітектури розподіленої СКБД на ст. 119. На її основі створіть можливу схему розподіленої системи керування базою даних «Туристична агенція» із завдання №7 до теми 5.

ПРЕДМЕТНИЙ ПОКАЖЧИК

- агрегація 57
- альтернативний ключ 78
- антиз'єднання 68, 72
- архітектура ANSI-SPARC 18
- атомарність 113
- атрибут 53, 75

- багатозначний атрибут 53
- багатозначна залежність 94
- база даних 16, 18, 19
- базове відношення 77
- «брудне» читання 114

- відкат 117
- відновлення бази даних 116
- відношення 75, 76
- вибірка 63
- внутрішній рівень 18, 19, 20
- втрачене оновлення 114

- дatalogчна модель даних 18, 19, 20
- декартів добуток 66
- детермінант 89
- диспетчер відновлення 117
- ділення відношень 74
- довговічність 113
- документо-орієнтована модель бази даних 41, 42, 43
- домен атрибуту 53, 75
- друга нормальна форма (2НФ) 92

- екземпляр сутності 52

- життєвий цикл інформаційної системи 47
- життєвий цикл бази даних 48
- журнал транзакцій 117

- залежність з'єднань без втрат 94
- зв'язок 55, 80, 81, 82, 83, 84, 85, 86, 87

- з'єднання відношень 68
зовнішнє з'єднання 68, 73
зовнішній ключ 54, 78
зовнішній рівень 18, 19, 20
- єрархічна модель бази даних 26, 27, 28
ізолюваність (транзакції) 113
інфологічна модель даних 18, 19, 20
інформаційна система 14
інформація 14
- кардинальність 56, 75
керування паралельним доступом 114
ключ 54
композиція 57
контрольна точка 117
концептуальний рівень 18, 19, 20
корпоративні обмеження цілосності 79
кортеж 75
кратність зв'язку 56
куб OLAP 121
- мережева модель бази даних 29, 30, 31
механізм резервного копіювання 117
мова запитів SQL 99, 100, 101, 102, 103, 104, 105, 106, 107
модель бази даних 24, 25
модель даних «Сутність-Зв'язок» 32, 33, 34, 52
- напівз'єднання 68, 71
ненормалізована форма (ННФ) 91
неповторюване читання 114
нормалізація 89, 90
нормальна форма Бойса-Кодда (НФБК) 94
- об'єднання відношень 64
об'єктно-орієнтована модель бази даних 38, 39, 40
однозначний атрибут 53
- первинний ключ 54, 78
перейменування (операція) 63

- перетин відношень 67
 перша нормальна форма (1НФ) 91
 підклас 57
 повна функціональна залежність 92
 порожнє значення (NULL) 79
 потенційний ключ 54, 78
 похідний атрибут 53
 предметна область 14, 19, 25
 представлення 77
 представлення користувачів 18, 19, 20
 природнє з'єднання 68, 69
 природний ключ 54
 прогін 117
 проекція 63
 простий атрибут 53
 п'ята нормальна форма (5НФ) 94
- рекурсивний зв'язок 55
 реляційна алгебра 62
 реляційна модель бази даних 35, 36, 37, 75
 реляційна схема 75
 реляційна цілісність 79
 реляційний ключ 78
 рівень ізолюваності транзакцій 115
 різниця відношень 65
 розподілена база даних 118
 розподілена система керування базами даних 118
 розширена ER-модель 57
 рольове ім'я 55
- сильна сутність 52, 80
 система керування базами даних 17, 111, 112
 слабка сутність 52, 80
 складений атрибут 53
 складений ключ 54, 78
 структурне обмеження 56
 ступінь типу зв'язку 55
 ступінь участі 56
 суперклас 57
 суперключ 78

- сурогатний ключ 54
сутність 52
сховище даних 123
- тета-з'єднання 68, 70
технологія OLAP 121
технологія OLTP 122
тип зв'язку 55
тип сутності 52
типи даних SQL 101
транзакція 113, 114
транзитивна залежність 93
третя нормальна форма 93
- узагальнення 57
узгодженість 113
уточнення 57
- фантомне читання 114
фізична модель даних 18, 19, 20
фізичне проектування 98
фізичний рівень 18, 19, 20
функціональна залежність 89
- цілісність посилань 79
цілісність сутностей 79
- четверта нормальна форма (4НФ) 94

БІБЛІОГРАФІЧНИЙ СПИСОК

Основна використана література

1. Гайна Г. А. Основы проектирования баз данных : навч. посіб. / Г. А. Гайна. – К. : КНУБА, 2005. – 204 с.
2. Гайдамакин Н. А. Автоматизированные информационные системы, базы и банки данных. Вводный курс / Н. А. Гайдамакин. – М. : Гелиос АРВ, 2002. – 368 с.
3. Гарсиа-Молина Г. Системы баз данных / Г. Гарсиа-Молина, Д. Ульман, Д. Уидом. – М. : Изд. дом «Вильямс», 2003. – 1088 с.
4. Григорьев Ю. А. Банки данных / Ю. А. Григорьев, Г. И. Ревунков. – М. : Изд. МГТУ им. Н.Э.Баумана, 2002. – 320 с.
5. Грофф Дж. Энциклопедия SQL / Дж. Грофф, П. Вайнберг. – СПб. : Питер, 2003. – 896 с.
6. Грошев А. С. Основы работы с базами данных / А. С. Грошев. – М. : Национальный открытый университет «ИНТУИТ», 2016. – 255 с.
7. Дейт К. Дж. Введение в системы баз данных / К. Дж. Дейт. – К. : Диалектика, 1998. – 784 с.
8. Диго С. М. Проектирование и использование баз данных / С. М. Диго. – М. : Финансы и статистика, 1995. – 208 с.
9. Карпова Т. С. Базы данных: модели, разработка, реализация / Т. С. Карпова. – СПб. : Питер, 2001. – 304 с.
10. Когаловский М. Р. Энциклопедия технологий баз данных / М. Р. Когаловский. – М. : Финансы и статистика, 2002. – 800 с.
11. Конноли Т. Базы данных. Проектирование, реализация и сопровождение. Теория и практика / Т. Конноли, К. Бегг. – М. : Изд. дом «Вильямс», 2003. – 1440 с.
12. Кренке Д. Теория и практика построения баз данных / Д. Кренке. – СПб. : Питер, 2003. – 800 с.
13. Малыхина М. П. Базы данных: основы, проектирование, использование / М. П. Малыхина. – СПб. : БХВ-Петербург, 2004. – 512 с.
14. Роб П. Системы баз данных: проектирование, реализация и управление / П. Роб, К. Коронел. – СПб. : БХВ-Петербург, 2004. – 1040 с.

15. Советов Б. Я. Базы данных : учеб. для прикладного бакалавриата / Б. Я. Советов, В. В. Цехановский, В. Д. Чертовской. – 2-е изд. – М. : Изд. «Юрайт», 2015. – 463 с.
16. Хомоненко А. Д. Базы данных / А. Д. Хомоненко, В. М. Цыганков, М. Г. Мальцев. – СПб. : Корона принт, 2004. – 736 с.

Рекомендована література для поглибленого вивчення курсу

1. Дейт К. Дж. Введение в системы баз данных / К. Дейт; пер. с англ. : В. С. Минаева, И. А. Маслаковой. – М. : Наука, 1980. – 464 с.
2. Джексон Г. Проектирование реляционных баз данных для использования с микроЭВМ / Г. Джексон; пер. с англ. А. Н. Елькова; под ред. канд. физ.-мат. наук С. А. Платонова. – М. : Мир, 1991. – 252 с.
3. Карпова Т. С. Базы данных: модели, разработка, реализация : учеб. / Т. С. Карпова. – СПб. ; М. ; Х. : Питер, 2002. – 304 с.
4. Карпуша В. Д. Моделювання та проектування реляційних баз даних : навч. посіб. для вищ. навч. закладів / В. Д. Карпуша, Б. Є. Панченко; Сумський держ. університет. – Суми : [б. в.], 2010. – 386 с.
5. Мартэн Д. Базы данных. Практические методы / Д. Мартэн; пер. с фр., под ред. А. В. Шилейко, пер. с фр., под ред. Т. И. Шилейко. – М. : Радио и связь, 1983. – 168 с.
6. Реляційні бази даних: табличні алгебри та SQL-подібні мови : моногр. / В. Н. Редько [та ін.]. – Київ : Академперіодика, 2001. – 198 с.

Навчальне видання

БАРДУС Ірина Олександрівна,
ЛАЗАРЄВ Микола Іванович,
НІЦЕНКО Андрій Олександрович

**БАЗИ ДАНИХ
У СХЕМАХ
(НА ОСНОВІ ФУНДАМЕНТАЛІЗОВАНОГО ПІДХОДУ)**

Навчальний посібник

В авторській редакції

Підп. до друку 02.10.2017. Формат 60x84 1/16. Папір офсетний. Друк цифровий. Гарнітура Arial. Ум.- друк. арк. 9.81. Наклад 300 прим. Зам № 277. Ціна договірна.

Видавництво «Діса плюс»
Тел. (057) 768-03-15

Свідоцтво про внесення суб'єкта видавничої справи до Державного реєстру видавців, виготівників та розповсюджувачів видавничої продукції: серія ДК № 4047 від 15.04.2011 р.

Надруковано в друкарні ТОВ «ПромАрт» 61023, м. Харків,
вул. Весніна, 12.
тел. (057) 717-28-80 e-mail: promart_order@ukr.net