

3. ОФОРМЛЕННЯ КУРСОВОЇ РОБОТИ

Обсяг пояснювальної записки до курсової роботи, незалежно від змісту, складає 25-35 сторінок. Тексту роботи має передувати титульна сторінка, підписане керівником завдання, та зміст. Титульна сторінка оформлюється за зразком, наведеним у додатку або в [1]. Зміст містить назви усіх розділів із зазначенням сторінки початку матеріалу розділу. У кінці текстової частини наводиться список використаних джерел інформації та додатки. Список використаних джерел інформації повинен бути оформлений відповідно до державних стандартів, узагальнені вимоги яких наведено в [1].

3.1. Структура пояснювальної записки до курсової роботи

Пояснювальна записка повинна мати такі розділи:

Вступ

1. Опис предметної сфери та постановка задачі.
2. Концептуальна модель бази даних. Діаграми потоків даних.
3. Інфологічна модель бази даних. Діаграми «сутність – зв'язок».
4. Даталогічна модель бази даних. Схеми функціональних залежностей. Нормалізація відношень.
5. Фізична модель бази даних. Засоби забезпечення цілісності.
6. Опис реалізації проекту бази даних.

Висновки.

Джерела інформації.

Нижче наведено концептуальні вимоги до змісту кожного з перелічених вище пунктів та приклади виконання відповідних робіт для демонстративної інформаційної системи «Предмети у приміщеннях» (ІСПП). Текст прикладів буде наведено іншим шрифтом.

3.2. Титульний аркуш, зміст і вступ

Як вже наводилося вище, титульна сторінка оформлюється за зразком, наведеним у додатку або в [1]. Обов'язкові розділи і пункти змісту наведено вище. У вступі необхідно вказати відомості про призначення інформаційної системи (підсистеми, задачі, автоматизованого робочого місця тощо) та сутність інформаційних задач, які

можна розв'язувати при використанні бази даних, що має бути створена. Необхідно також навести відомості про подібні системи, що отримані в результаті аналізу як літературних джерел, так і джерел із мережі Інтернет.

3.3. Опис предметного середовища і постановка задачі

При проектуванні інформаційних систем (ИС) постає задача визначення структури та організації даних, тобто виявлення елементів, структур, в які групуються елементи даних, та зв'язків між елементами та структурами даних. Вирішенню цієї задачі покликана сприяти інформаційні моделі (ІМ), задачі (комплексу задач, підсистеми, системи). Тому спочатку зупинимося на різновидах ІМ у залежності від етапу або рівня проектування інформаційної системи та її бази даних.

Процес створення ІМ починається з визначення предметної сфери та аналізу концептуальних вимог ряду користувачів. При цьому треба передбачати вимоги майбутніх задач, які виникають у процесі розвитку ІС та її БД відповідно. Вимоги окремих користувачів інтегруються у якесь єдине «узагальнене» представлення, яке називають *концептуальною моделлю* (КМ) [3-5, 7].

Після створення концептуальної моделі будують *логічну модель* (ЛМ), яка визначає структуру даних: структурні одиниці даних та їх елементи і зв'язки між елементами даних незалежно від їх змісту та середовища зберігання. Вона може бути реляційною, ієрархічною або мережною (сітковою). Інколи її розглядають сумісно з обраною СКБД. Для користувачів виділяються підмножини ЛМ, вони називаються зовнішніми моделями або підсхемами. Обсяг бази даних у межах, що наведені у п. 2, як правило, вписується в рамки підсхеми. Логічна модель відображається у *фізичну модель*. Якщо структура даних, що визначена у ЛМ, не може бути реалізована у середовищі обраної СКБД, то коректують ЛМ, що може привести, у свою чергу, до коректування КМ.

Із перелічених інформаційних моделей найбільш формалізованою є логічна модель, тобто модель даних. Фізична модель є похідною від структур і організації даних, що використовуються в конкретній СКБД. Найменш формалізованою є концептуальна модель. Вона може бути чисто вербальною, а частіше – це опис предметної сфери, опис вхідної інформації, перелік вхідних форм та вихідних звітів і т. д. При цьому використовуються різного роду таблиці, схеми та інші форми подання. В останні роки намагання фахівців з баз даних якимось

чином формалізувати процес створення та представлення КМ привів до певних результатів, наслідком яких стало виділення ще одного рівня проектування бази даних, який хронологічно і технологічно знаходиться між КМ і ЛМ. Відповідну інформаційну модель називають *інфологічною моделлю* (ІЛМ). При уведенні ІЛМ наступну інформаційну модель прийнято називати *даталогічною моделлю*.

На рис. 1 показано етапність побудови вказаних моделей та основні проектні рішення, що мають бути отримані на цих етапах (рівнях проектування).

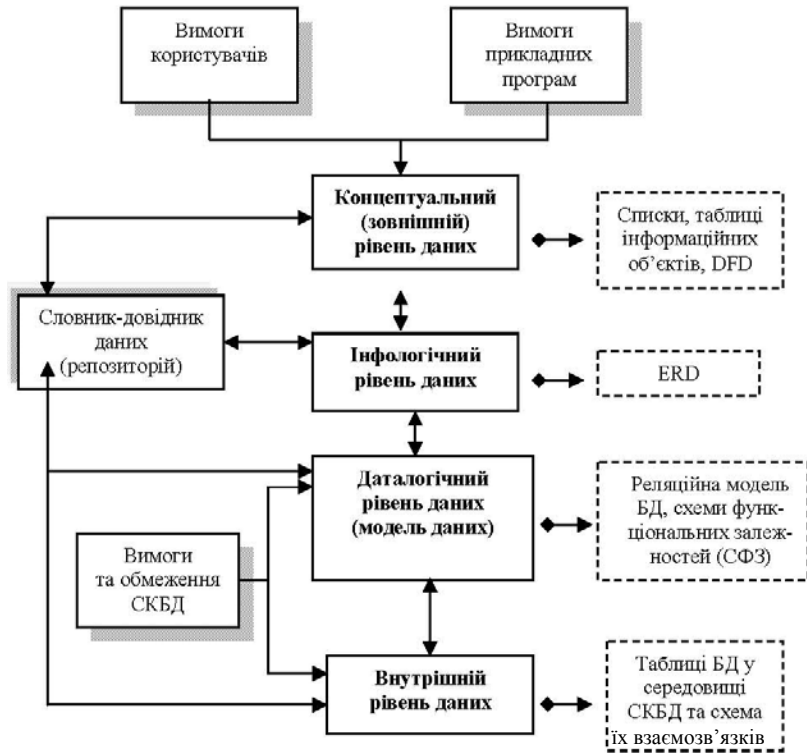


Рис. 1. Схема взаємозв'язку інформаційних моделей по рівнях:
 DFD – Data Flow Diagram (діаграма потоків даних – ДПД);
 ERD – Entity-Relationship Diagram
 (діаграма «сутність – зв'язок» – ДСЗ).

Тепер перейдемо до змісту робіт, що мають бути виконані в даному пункті. Аналіз предметного середовища повинен включати описи або формулювання:

- об'єкта дослідження та характеристику його організаційно-функціональної структури;
- функціональної сутності системи (підсистеми, комплексу задач, задачі);
- опис вхідних даних, у т. ч. уніфікованих форм документації;
- нормативно-довідкової інформації, що використовується (класифікатори, довідники, норми, тощо);
- основних процесів перетворення даних та нормативних документів, що їх визначають (інструкції, правила, методичні матеріали, регламенти, тощо).
- вихідні повідомлення (форми, звіти, сторінки) усіх видів, що формуються при розв'язанні задачі з визначенням їх структури, періодичності та способу формування.

На основі проведеного аналізу здійснюється постановка задачі, формулюється мета роботи та визначаються основні вимоги до:

- функціональної сутності інформаційної системи;
- до структури й складу бази даних, що проектується;
- програмного забезпечення;
- технічного забезпечення.

Наскрізний приклад проектування бази даних ІСПП. Призначення, цілі, задачі та основні вимоги до ІСПП.

Об'єктом дослідження курсової роботи є інформаційне забезпечення коменданта приміщення, технічного відділу, бухгалтерії і диспетчерського підрозділу навчального закладу щодо приміщень, предметів, що там знаходяться, та їх характеристик у залежності від функціональних задач цих підрозділів.

Предметом дослідження даної курсової роботи є база даних про приміщення і предмети, що там знаходяться, та інформаційна система, що може бути створена на основі цієї бази даних.

Інформаційна система (ІС) має надавати вказаним підрозділам інформацію:

- *Коменданту будинку:* про площі кімнат (аудиторій) – для розподілення зон між прибиральницями; меблеві предмети (столи, стільці, шафи, тощо) – для оперативного обліку і своєчасної заміни або ремонту.

- *Навчальному відділу*: про площі і типи кімнат, кількість посадочних місць у кімнатах – для складання розкладу занять.
- *Технічному відділу*: про лабораторне устаткування й прилади, в т. ч. комп'ютери – для оперативного обліку і своєчасної заміни або ремонту.
- *Бухгалтерії*: про усі предмети, що знаходяться в кімнатах, та їх залишкову вартість – для бухгалтерського обліку й нарахувань амортизації предметів.

Вхідні дані уводяться в ІС:

- із актів інвентаризації – на початковому етапі;
- на основі актів уведення або виведення основних фондів (предметів й обладнання) із експлуатації. Форми цих актів є уніфікованими.

Довідкова інформація, що має використовувати:

- типи приміщень (лекційна аудиторія, лабораторія, тощо);
- типи предметів. Як мінімум, має бути два типи предметів: підзвітні коменданта і підзвітні технічного відділу.

Нормативна інформація, що має використовуватися: норми амортизаційних нарахувань на предмети.

Основну частину *процесів перетворення даних* складають операції уведення, корегування та видалення інформації, а також групування та розрахунку сум, середніх, максимальних та мінімальних значень по обраних угрупованнях, підрахунку кількості об'єктів групування. Розрахунковим результатом, що визначається нормативним документом (інструкція Мінфіну України), є нарахування на амортизацію предметів. Воно здійснюється раз на рік за формулою:

$$A = k_i * Spoc,$$

де A – сума нарахування на амортизацію предмета;

k_i – нормативний коефіцієнт амортизації, встановлений для i -ї групи основних засобів (предметів);

$Spoc$ – початкова вартість предмета, яка, при уведенні предмета в експлуатацію, дорівнює вартості придбання.

Після нарахування амортизації нова залишкова вартість обраховується як

$$S_{зал_нова} = S_{зал_стара} - A.$$

Для першого року експлуатації $S_{зал_стара} = Spoc$. Якщо $S_{зал_нова} < 0$, то нова залишкова вартість визначається рівною нулю і на цей предмет у подальшому амортизація не нараховується.

Вихідними повідомленнями, що формуються при розв'язанні задачі, є запити у формі таблиць, екранні форми і звіти на папері. Нижче наводиться попереднє визначення їх номенклатури, сутності, періодичності та способу формування (табл. 1).

Таблиця 1

Уточнений попередній перелік вихідних повідомлень

Позначення	Сутність вихідного повідомлення	Підрозділ-замовник	Періодичність	Спосіб формування
П10	Площі кімнат і поверхів	Комендант	За необхідністю	Для кімнат – вибірка, для поверхів – сума
П20	Предмети підзвітності по кімнатах, поверхах	Комендант	За необхідністю	Вибірка і групування
П30	Кімнати, де знаходяться конкретні предмети підзвітності	Комендант	За необхідністю	Вибірка і групування
П40	Предмети підзвітності, що вилучені з експлуатації	Комендант	Щотижня	Вибірка, групування і сума
П50	Нові предмети підзвітності, що встановлені	Комендант	Щомісяця	Вибірка, групування і сума
П60	Предмети підзвітності по кімнатах, поверхах	Технічний відділ	За необхідністю	Вибірка і групування
П70	Кімнати, де знаходяться конкретні предмети підзвітності	Технічний відділ	За необхідністю	Пошук за ознаками
П80	Предмети підзвітності, що вилучені з експлуатації	Технічний відділ	Щомісяця	Вибірка, групування і сума

Закінчення табл. 1

П90	Нові предмети підзвітності, що встановлені	Технічний відділ	Один раз на півроку	Вибірка, групування і сума
П100	Типи аудиторій, їх площа та кількість	Диспетчерська	За необхідністю	Вибірка, групування, сума
П110	До якого типу належить конкретна аудиторія та її площа	Диспетчерська	За необхідністю	Пошук за ознаками
П120	Кількість комп'ютерів у комп'ютерних класах	Диспетчерська	Щомісяця	Вибірка, групування, сума
П130	Вартість предметів, що встановлені в аудиторії	Бухгалтерія	За необхідністю	Пошук за ознаками
П140	Предмети, що не мають залишкової вартості	Бухгалтерія	За необхідністю	Пошук за ознаками
П150	Зведена балансова відомість амортизаційних нарахувань та залишкової вартості предметів	Бухгалтерія	Щорічно	Групова операція за методикою (див. вище)
П160	Зведена відомість руху предметів	Бухгалтерія	Щомісяця	Вибірка, групування, сума

На основі наведених даних сформулюємо мету роботи та основні вимоги до функціональної сутності ІС, до структури й складу її БД та технічного й програмного забезпечення.

Метою даної курсової роботи є позбавлення працівників вищеперелічених підрозділів від рутинної роботи щодо ведення інформації про приміщення й предмети, що там знаходяться, швидкого отримання різноманітних довідок та формування регламентованих звітів, у тому числі й нарахування амортизації.

Вимоги до функціональної сутності ІС полягають у забезпеченні формування інформації, що наведена в табл.1, і потреб, які ще можуть виникнути у процесі розвитку ІС.

Структури й склад бази даних мають моделювати такі інформаційні об'єкти (ІО), як «предмети», «кімнати» та їх властивості, а також зв'язки між названими ІО та властивості цих зв'язків, і забезпечити отримання інформації, потрібної користувачам (табл. 1).

Вимоги до програмного забезпечення ПК: операційна система – Windows 200X/NT/XP; система керування базою даних – Access 200X.

Вимоги до технічного забезпечення: персональні комп'ютери, об'єднані в локальну комп'ютерну мережу, в перелічених вище підрозділах. Мінімальна конфігурація ПК: оперативна й зовнішня пам'ять, що будуть достатніми для нормальної роботи базових програмних засобів (див. вище), а також наявність 100 Мб зовнішньої пам'яті безпосередньо для бази даних.

Кінець фрагмента наскрізного прикладу (призначення, цілі, задачі та основні вимоги до ІСПП).

3.4. Концептуальна модель інформаційної системи. Діаграми потоків даних

Як уже наводилося, *концептуальна модель* визначає вимоги користувачів бази даних, які оформлені у вигляді таблиць, схем інформаційних потоків, схем взаємозв'язків між показниками техніко-економічної інформації, порядку їх розрахунку і т. п. та коментарів до них. Таким чином, проектування КМ засновано на аналізі задач обробки даних, що мають розв'язуватися в інформаційній системі, і включає такі роботи:

- аналіз інформаційних потреб користувачів/замовника та попереднє визначення звітів та/або форм, які їм потрібні;
- аналіз елементів даних, що використовуються в кожному із звітів та/або формі;
- вилучення синонімії, омонімії та узагальнення атрибутів;
- визначення основних процедур перетворення інформації (бізнес-правил).

Наскрізний приклад проектування бази даних. Створення концептуальної моделі ІСПП.

Після попереднього визначення інформаційних потреб користувачів (табл. 1) була проведена додаткова робота з користувачами, зокрема виявлялися додаткові інформаційні потреби та визначалися

вид (тип) представлення вихідної інформації: запит (літера Q в позначенні), форма (F) чи звіт (R). Результати роботи наведені у табл. 2.

Таблиця 2

Попередній перелік вихідних повідомлень

Позначення	Сутність вихідного повідомлення	Підрозділ-замовник	Періодичність	Спосіб формування	Подання
Q10	Площі кімнат і поверхів	Комендант	За необхідністю	Для кімнат – вибірка, для поверхів – сума	Запит
F11	Поверхи та кімнати на поверхах разом з їх характеристиками	Комендант	За необхідністю	Для кімнат – вибірка, для поверхів – сума	Форма
R12	Площі кімнат і поверхів станом на перше число нового року	Комендант	Раз на рік	Для кімнат – вибірка, для поверхів – сума	Звіт
R20	Предмети підзвітності по кімнатах, поверхах	Комендант	За необхідністю	Вибірка і групування	Звіт
F21	Предмети підзвітності по кімнатах, поверхах	Комендант	За необхідністю	Вибірка і групування	Ієрархічна форма
Q30	Кімнати, де знаходяться конкретні предмети підзвітності	Комендант	За необхідністю	Вибірка і групування	Запит
R40	Предмети підзвітності, що вилучені з експлуатації	Комендант	Щотижня	Вибірка, групування і сума	Звіт

Продовження табл. 2

R50	Нові предмети підзвітності, що встановлені в кімнатах	Комендант	Що-місяця	Вибірка, групування і сума	Звіт
F51	Нові предмети, що встановлені на поверхах	Комендант	За необхідністю	Вибірка і групування	Ієрархічна форма
Q52	Нові предмети, що встановлені в певній кімнаті	Комендант	За необхідністю	Вибірка	Запит
R60	Предмети підзвітності по кімнатах, поверхах	Технічний відділ	Що-місяця	Вибірка і групування	Звіт
F61	Предмети підзвітності по кімнатах, поверхах	Технічний відділ	За необхідністю	Вибірка і групування	Ієрархічна форма
F70	Кімнати, де знаходяться конкретні предмети підзвітності	Технічний відділ	За необхідністю	Пошук за ознаками	Ієрархічна форма
R71	Розміщення предметів підзвітності по кімнатах	Технічний відділ	Щоквартально	Вибірка і групування	Звіт
Q72	Предмети, термін амортизації яких закінчується у поточному році	Технічний відділ	2 рази на рік	Вибірка і групування	Запит

Продовження табл. 2

R80	Предмети підзвітності, що вилучені з експлуатації	Технічний відділ	Що-місяця	Вибірка, групування і сума	Звіт
Q81	Комп'ютери, що вилучені з експлуатації	Технічний відділ	За необхідністю	Вибірка за ознаками	Запит
R90	Нові предмети підзвітності, що встановлені в аудиторіях	Технічний відділ	Один раз на півроку	Вибірка, групування і сума	Звіт
Q91	Кімнати, де у поточному році встановлені комп'ютери	Технічний відділ	За необхідністю	Вибірка за ознаками	Запит
Q100	Типи аудиторій, їх площа та кількість	Диспетчерська	За необхідністю	Вибірка, групування, сума	Запит
Q110	До якого типу належить конкретна аудиторія та її площа	Диспетчерська	За необхідністю	Пошук за ознаками	Запит
R111	Кімнати, де у поточному році встановлені комп'ютери	Диспетчерська	Раз на рік	Вибірка, групування, сума	Звіт
F120	Кількість комп'ютерів у комп'ютерних класах	Диспетчерська	Що-місяця	Вибірка, групування, сума	Ієрархічна Форма
R121	Оснащеність комп'ютерних класів	Диспетчерська	За необхідністю	Вибірка, групування, сума	Звіт

Закінчення табл. 2

Q130	Вартість предметів, що встановлені у певній аудиторії	Бухгалтерія	За необхідністю	Вибірка, групування, сума	Запит
R131	Відомість нарахування амортизації	Бухгалтерія	Щорічно	За методикою нарахування	Звіт
F132	Вартість предметів по аудиторіях	Бухгалтерія	Щорічно	Вибірка, групування, сума	Форма
Q140	Предмети, що не мають залишкової вартості	Бухгалтерія	За необхідністю	Пошук за ознаками	Запит
R150	Зведена балансова відомість амортизаційних нарахувань та залишкової вартості предметів	Бухгалтерія	Щорічно	Групова операція за методикою (див. вище)	Звіт
R160	Зведена відомість руху предметів	Бухгалтерія	Що-місяця	Вибірка, групування, сума	Звіт

Тепер виконаємо аналіз елементів даних, що використовуються в кожному із запитів, звітів та/або форм.

На основі аналізу інформаційних потреб користувачів можна зробити попередній висновок, що у наведених у табл. 2 повідомленнях використовуються такі атрибути (табл. 3).

Таблиця 3

**Попередній перелік атрибутів,
що вживаються у вихідних повідомленнях**

№	Атрибут	№	Атрибут
1	Поверх	7	Предмет
2	Загальна площа кімнат на поверсі	8	Тип предмета
3	Кімната	9	Початкова вартість предмета
4	Тип кімнати	10	Термін амортизації предмета
5	Площа кімнати	11	Кількість предметів у кімнаті
6	Дата останнього ремонту кімнати	12	Дата встановлення предметів у кімнаті

У результаті уточнення сутності запитів із майбутніми користувачами встановлено використання атрибутів у вихідних повідомленнях (табл. 4).

Таблиця 4

Використання атрибутів у вихідних повідомленнях

Позначення повідомлення	Атрибути Вихідні повідомлення	Поверх											
		Загальна площа кімнат на поверсі	Кімната	Тип кімнати	Площа кімнати	Дата останнього ремонту кімнати	Предмет	Тип предмета	Початкова вартість предмета	Термін амортизації предмета	Кількість предметів у кімнаті	Дата встановлення предметів у кімнаті	
Номера атрибутів		1	2	3	4	5	6	7	8	9	10	11	12
Q10	Площі кімнат і поверхів	+	+	+		+							
F11	Поверхи та кімнати на поверхах разом з їх характеристиками	+	+	+	+	+	+						
	Площі кімнат і поверхів станом на перше число нового року	+	+	+		+							
R20	Предмети підзвітності по кімнатах, поверхах	+		+				+		+		+	+

Продовження табл. 4

F21	Предмети підзвітності по кімнатах, поверхах	+		+					+	+				+	+
Q30	Кімнати, де знаходяться конкретні предмети підзвітності			+					+					+	
R40	Предмети підзвітності, що вилучені з експлуатації			+					+					+	+
R50	Нові предмети підзвітності, що встановлені в кімнатах			+					+	+	+	+	+	+	+
F51	Нові предмети, що встановлені на поверхах								+	+	+	+	+	+	+
Q52	Нові предмети, що встановлені у певній кімнаті			+					+					+	+
R60	Предмети підзвітності по кімнатах, поверхах	+		+					+	+				+	+
F61	Предмети підзвітності по кімнатах, поверхах	+		+					+					+	
F70	Кімнати, де знаходяться конкретні предмети підзвітності			+					+	+					
R71	Розміщення предметів підзвітності по кімнатах			+					+		+	+	+	+	+
Q72	Предмети, термін амортизації яких закінчується у поточному році			+					+		+	+	+	+	+
R80	Предмети підзвітності, що вилучені з експлуатації			+					+					+	
Q81	Комп'ютери, що вилучені з експлуатації			+					+	+				+	
R90	Нові предмети підзвітності, що встановлені в аудиторіях			+					+	+	+	+	+	+	+
Q91	Кімнати, де у поточному році встановлені комп'ютери			+					+	+					+
Q100	Типи аудиторій, їх площа та кількість			+	+	+									
Q110	До якого типу належить конкретна аудиторія та її площа			+	+										
R111	Кімнати, де у поточному році встановлені комп'ютери			+					+	+				+	+
F120	Кількість комп'ютерів у комп'ютерних класах			+	+				+	+					
R121	Оснащеність комп'ютерних класів			+	+				+	+					
Q130	Вартість предметів, що встановлені в певній аудиторії			+					+		+	+		+	
R131	Відомість нарахування амортизації								+	+	+	+			+
F132	Вартість предметів по аудиторіях			+					+	+	+	+			+
Q140	Предмети, що не мають залишкової вартості								+	+	+	+			+

Закінчення табл. 4

R150	Зведена балансова відомість амортизаційних нарахувань та залишкової вартості предметів			+				+	+	+	+	+	+
R160	Зведена відомість руху предметів			+				+		+	+	+	+

Аналіз назв і семантики атрибутів та повідомлень показує, що ефект омонімії (одна й та ж назва має різну семантику) відсутній. Що стосується синонімії (одне й теж поняття має різні назви), то вона має місце (кімната, приміщення, аудиторія), але немає особливого сенсу переробляти цю частину концептуальної моделі, тому вона залишається, при цьому просто треба мати на увазі, що це в даному прикладі одне й теж.

Що стосується визначення основних процедур перетворення інформації (бізнес-правил), то вони були наведені у табл. 3.

Кінець фрагмента наскрізного прикладу (створення концептуальної моделі ІСПП)

Однією із важливих компонент регламентованого представлення КМ, крім таблиць, в яких представлені форми, звіти, атрибути, організаційні структури, що беруть участь у формування первинної інформації та процесах її перетворення, є:

- функціональна модель – стандарт IDEF0;
- діаграми потоків даних (ДПД) або Data Flow Diagram (DFD);
- діаграми потоків робіт (ДПР) або Work Flow Diagram (WFD) – стандарт IDEF3.

Примітка. Наведені вище стандарти були прийняті у США в авіабудівній промисловості при впровадженні інформаційних технологій в управління. Перші чотири літери розшифровуються таким чином: ICAM DEFinition, де ICAM (Information Computer Aided Manufacturing) – програма впровадження інформаційних технологій в управління.

Усі три типи схем являють собою ієрархію діаграм: на самому верхньому рівні створюються так звані контекстні діаграми, центральною ланкою якої є сама інформаційна система (підсистема, комплекс задач, задача, автоматизоване робоче місце тощо). До позначення центральної ланки додаються інші елементи в залежності від типу діаграми. Потім центральна ланка підлягає декомпозиції, в результаті чого отримують діаграми першого рівня, потім деталізують ланки першого рівня і отримують діаграми другого рівня і так далі. Така декомпозиція продовжується, створюючи багаторівневу ієрархію

діаграм, до тих пір, поки не буде досягнуто такий рівень декомпозиції, на котрому компоненти стають елементарними і деталізувати їх далі неможливо.

Всі три типи діаграм методологічно пов'язані між собою в такій послідовності: IDEF0 → DFD → WFD, але їх всі разом розробляють для великих проектів. Враховуючі обсяг бази даних курсової роботи, обов'язковим компонентом проекту є DFD. Для тих, хто цікавиться методологією названих вище діаграм, можна порекомендувати звернутися до [2, 6].

А зараз розглянемо методологію створення діаграм потоків даних. В основі даної методології (методології Гейна-Сарсона (Gane/Sarson) [6]) лежить створення моделі ІС, що аналізується: реально існуючої або яка проектується. Згідно з методологією модель системи визначається як ієрархія діаграм потоків даних, що описують асинхронний процес перетворення інформації від її уведення в систему до видачі користувачу. Діаграми верхніх рівнів ієрархії (контекстні діаграми) визначають підсистеми (АРМ, задачі, тощо) ІС або основні процеси та їх зв'язки із зовнішніми входами та виходами. Вони деталізуються за допомогою діаграм нижнього рівня. Джерела інформації (зовнішні сутності) породжують інформаційні потоки (потоки даних), які переносять інформацію до підсистем або процесів. Ті, у свою чергу, перетворюють інформацію і породжують нові потоки, які переносять інформацію до інших процесів або підсистем, накопичувачів даних або зовнішніх сутностей – споживачів інформації. Таким чином, основними компонентами діаграм потоків даних є:

- зовнішні сутності;
- системи/підсистеми;
- процеси;
- накопичувачі даних;
- потоки даних.

Зупинимось на нотації, прийнятої для представлення DFD. Системи/підсистеми і процеси на схемах прийнято позначати або прямокутником із закругленими кутами – верхні фігури на рис. 2, або кругом (еліпсом, овалом) – нижні фігури на рис. 1. Системи (підсистеми, комплекси задач, задачі, автоматизовані робочі місця) відрізняються від процесів побудовою назв. *Назва системи* у своїй основі має *іменник*, наприклад: «Інформаційна система про приміщення і предмети», «АРМ робітника бухгалтерії», «Ведення інформації про предмети» тощо. *Назва процесу* у своїй основі має *дієслово*, наприклад: «Увести інформацію про новий предмет», «Оновити дату ремонту кімнати», «Розрахувати амортизацію

предметів» тощо. Як правило, системи вживаються на верхніх рівнях DFD, процеси – на нижніх. Контекстна діаграма не має номера (нульовий рівень ієрархії), системи або процеси першого рівня декомпозиції мають номери 1, 2, ..., N, підсистеми або процеси другого рівня декомпозиції мають номери 1.1, 1.2, ... 1.m, 2.1, 2.2 і т. д.

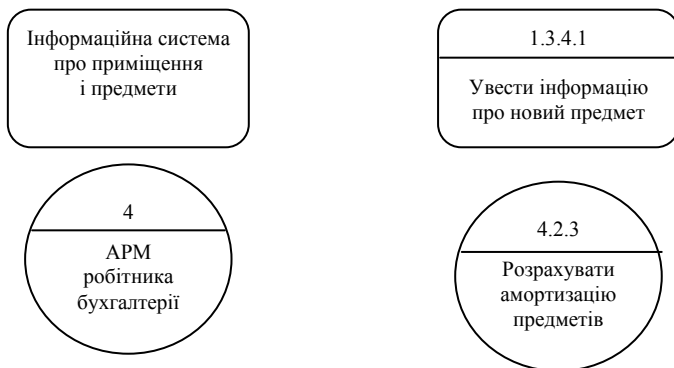


Рис. 2. Приклади представлення систем і процесів різними способами в DFD

У даних матеріалах буде застосовуватися нотація представлення систем і процесів у вигляді прямокутників із закругленими кутами.

Зовнішні сутності на DFD прийнято позначати прямокутниками (ліва фігура на рис. 3), при цьому рекомендується прямокутники показувати з тінню при освітленні зверху і зліва (права фігура на рис. 3). В даних матеріалах буде використовуватися друга нотація.



Рис. 3. Приклади представлення зовнішніх сутностей

Накопичувачі даних позначаються смугою або без визначення лівої і правої граней (ліва фігура на рис. 4), або без визначення тільки правої грані (права фігура на рис. 4). Накопичувачі, як системи і процеси, нумеруються, але нумерація їх незалежна: попереду номера ставиться літера D (або Д), після якої ставиться порядковий номер. Якщо в діаграмах IDEF0 і WFD накопичувачами можуть бути будь-яке сховище, де зберігаються як матеріальні, так і нематеріальні об'єкти,

то в DFD під накопичувачами мають на увазі файли та/або бази даних. У даних матеріалах буде використовуватися друга нотація.



Рис. 4. Приклади представлення накопичувачів

Потоки даних на DFD позначаються направленими лініями (лініями із стрілками) між переліченими вище елементами діаграм. Кожна лінія має назву потоку інформації. На верхніх рівнях вони мають узагальнені назви («Інформація про предмети для бухгалтерії», «Дані про норми амортизації» тощо), а на нижніх рівнях – більш конкретні («Звіт R31», «Дані про норми амортизації Мінфіну від №» тощо). Тут теж використовують різні нотації: прямі лінії (рис. 5) або випуклі лінії (рис. 6).

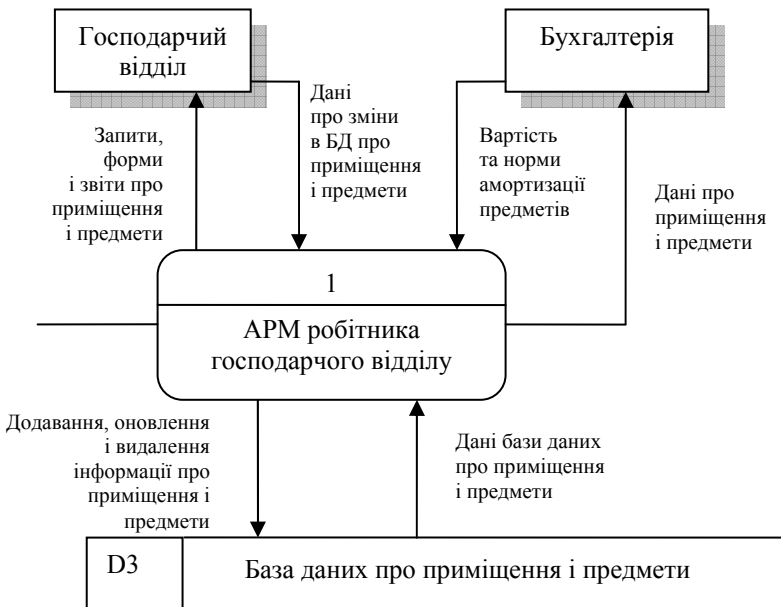


Рис. 5. Приклади представлення потоків даних прямими лініями



Рис. 6. Приклади представлення потоків даних випуклими лініями

Використаємо викладену вище методологію для створення ієрархії діаграм для інформаційної системи про аудиторії та предмети, що там встановлені.

Наскрізний приклад проектування бази даних. Діаграми потоків даних

Контекстна DFD ІСПП показана на рис. 7.

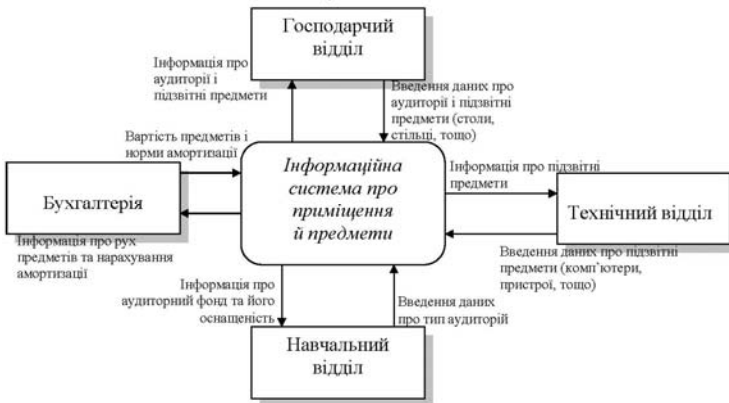


Рис. 7. Контекстна діаграма потоків даних

Нижче в даному прикладі наведено набір DFD тільки для АРМ працівника господарчого відділу (рис. 8-11).

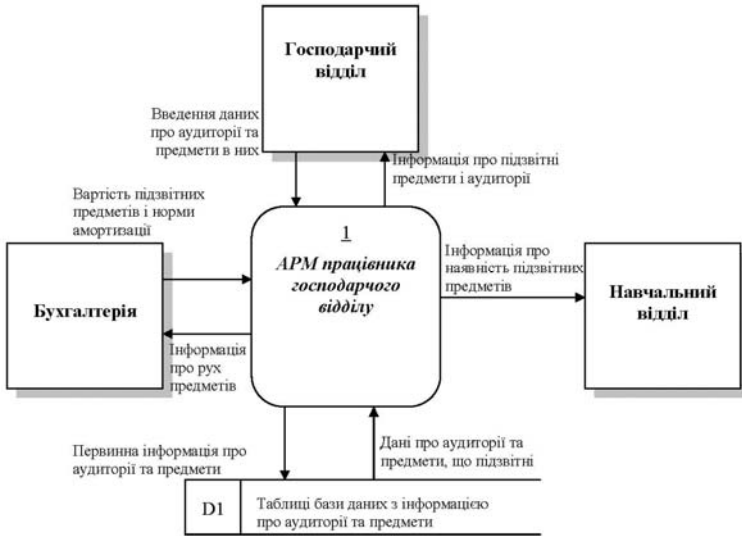


Рис. 8. Фрагмент DFD першого рівня

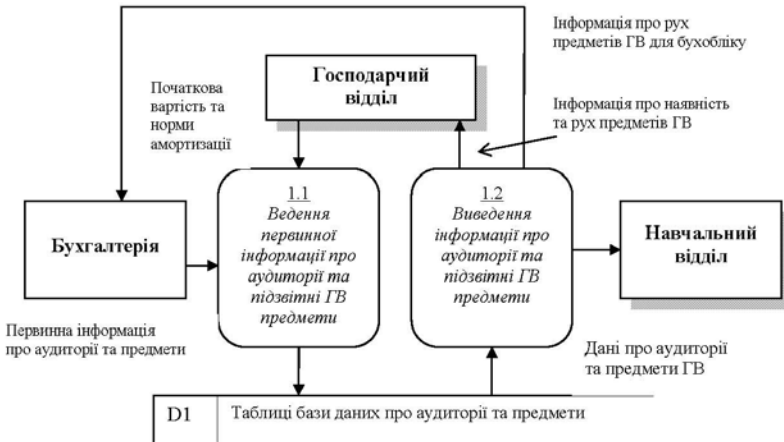


Рис. 9. Фрагмент DFD другого рівня

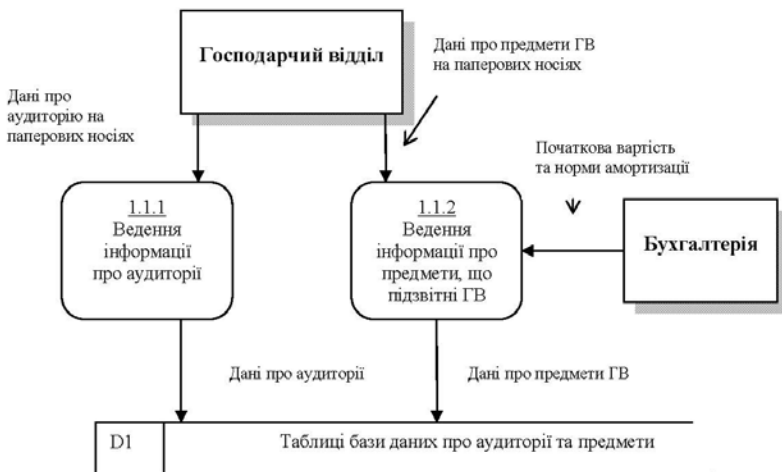


Рис. 10. Фрагмент DFD третього рівня



Рис. 11. Фрагмент DFD четвертого рівня

Інші DFD тут не наводяться. У таблиці 5 наведена ієрархія діаграм потоків даних інформаційної системи. Маючи ієрархію діаграм потоків даних, складання інших DFD для даної ІС не представляє труднощів.

Кінець фрагмента наскрізного прикладу (діаграми потоків даних)

Ієрархія діаграм потоків даних ІСПП

Вершина діаграми	Підсистеми та процеси
1	АРМ працівника господарчого відділу (ГВ)
1.1	Ведення первинної інформації про аудиторії та підзвітні ГВ предмети
1.1.1	Ведення інформації про аудиторії
1.1.1.1	Додати інформацію про аудиторії
1.1.1.2	Оновити інформацію про аудиторії
1.1.1.3	Видалити інформації про аудиторії
1.1.2	Ведення інформації про підзвітні ГВ предмети
1.1.2.1	Додати інформації про підзвітні ГВ предмети
1.1.2.2	Оновити інформації про підзвітні ГВ предмети
1.1.2.3	Видалити інформації про підзвітні ГВ предмети
1.2	Виведення інформації про аудиторії та підзвітні ГВ предмети
1.2.1	Виведення інформації про аудиторії
1.2.1.1	Виведення інформації про аудиторії у вигляді запитів
1.2.1.2	Виведення інформації про аудиторії у вигляді форм
1.2.1.3	Виведення інформації про аудиторії у вигляді звітів
1.2.2	Виведення інформації про підзвітні ГВ предмети
1.2.2.1	Виведення інформації про підзвітні ГВ предмети у вигляді запитів
1.2.2.	Виведення інформації про підзвітні ГВ предмети у вигляді форм
1.2.2.	Виведення інформації про підзвітні ГВ предмети у вигляді звітів
2	АРМ працівника технічного відділу (ТВ)
2.1	Ведення інформації про підзвітні ТВ предмети
2.1.1	Додати інформації про підзвітні ТВ предмети
2.1.2	Оновити інформації про підзвітні ТВ предмети
2.1.3	Видалити інформації про підзвітні ТВ предмети
2.2	Виведення інформації про підзвітні ТВ предмети
2.2.1	Виведення інформації про підзвітні ТВ предмети у вигляді запитів
2.2.2	Виведення інформації про підзвітні ТВ предмети у вигляді форм

2.2.3	Виведення інформації про підзвітні ТВ предмети у вигляді звітів
3	АРМ працівника бухгалтерії
3.1	Ведення інформації про норми амортизації предметів
3.2	Формування документів для проведення інвентаризації предметів в аудиторіях
3.2.1	Уведення предметів в експлуатацію
3.2.2	Переміщення предметів
3.2.3	Нарахування амортизації і встановлення нової вартості предметів
3.2.4	Вилучення предметів із експлуатації
4	АРМ працівника навчального відділу
4.1	Ведення інформації про типи аудиторій
4.2	Отримання інформації про характеристики аудиторій
4.3	Отримання інформації про предмети в аудиторіях
4.3.1	Отримання інформації про оснащеність аудиторій комп'ютерами
4.3.2	Отримання інформації про оснащеність аудиторій пристроями та лабораторним обладнанням

Таким чином, у результаті аналізу інформаційної системи об'єкта на основі створення DFD-діаграм (при розробці середніх і великих проектів ще й на основі IDEF0-діаграм, що передують DFD-діаграмам):

- визначають головну функцію системи, що проектується, наприклад, «Надавати інформаційне супроводження процесам встановлення, переміщення та списання предметів в аудиторіях»;
- описують в об'єкті, що моделюється, бізнес-процеси, наприклад: «Увести дані про встановлення нових предметів в аудиторії», «Отримати звіт про ремонти приміщень у минулому році»;
- визначити потоки даних, що обробляються визначеними бізнес-процесами, наприклад, «Акт про списання предмета», «Звіт про ремонти приміщень у минулому році»;
- завершують створення концептуальної моделі системи, яка являє собою опис об'єкта, властивих йому бізнес-правил, потоків і сховищ даних.

Відмітимо, що потоки даних – це прообрази процедур прикладного програмного забезпечення, а сховища даних – таблиці бази даних. Але більш детальне визначення таблиць, складу їх атрибутів та взаємозв'язків здійснюється на наступних етапах проектування бази даних.

3.5. Інфологічна модель бази даних. Діаграми «сутність – зв'язок»

Як наводилося вище, діаграми потоків даних на наступних етапах проектування БД використовуються як певні специфікації (вимоги) процедур уведення та обробки даних. Після цього приступають безпосередньо до проектування бази даних. Першим процесом цього етапу проектування є моделювання даних із врахуванням концептуальної моделі інформаційної системи. Ціль моделювання даних (створення інфологічної моделі) складається в забезпеченні розроблювача ІС концептуальною схемою бази даних у формі однієї моделі або декількох локальних моделей, що відносно легко можуть бути перетворені в будь-яку модель даних: реляційну, ієрархічну, мережну. Найбільш розповсюдженим засобом абстрактного представлення структур баз даних є ER-модель або модель «сутність – зв'язок». Вона уперше була введена Питером Ченом (Peter Chen) у 1976 р. [3, 5, 7], а графічне представлення (нотація) цієї моделі відомо як ERD (Entity-Relationship Diagram) або ER-схема (ER-діаграма). На даний час існує декілька нотацій ERD. Нижче для її представлення використовується нотація Чена, в якій схема складається з елементів трьох основних типів:

- а) множин сутностей;
- б) атрибутів;
- в) зв'язків.

Перший крок інфологічного моделювання – аналіз концептуальної моделі, включаючи DFD, з метою виділення *сутностей*. Якщо етап створення концептуальної моделі не виконувався, то витяг інформації для виділення *сутностей* здійснюється шляхом проведення інтерв'ю із замовником(ами) системи або анкетування, тобто виконується певна частина робіт концептуального рівня.

Сутність (Entity) – реальний або уявлюваний об'єкт, що має істотне значення для розглянутої предметної сфери, інформація про яке підлягає збереженню. На ERD сутність представляється або прямокутником (рис. 11, ліва фігура), або прямокутником із закругле-

ними кутами (рис. 11, права фігура). В даних матеріалах буде використовуватися друга форма представлення сутностей. У середині фігури наводиться назва сутності.

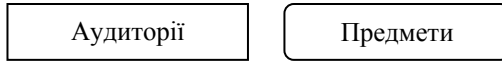


Рис. 11. Варіанти графічного зображення сутності

Кожна сутність повинна:

- мати унікальний ідентифікатор (ім'я);
- однозначно ідентифікуватися і відрізнитися від усіх інших екземплярів даного типу сутності;
- мати один або декілька атрибутів, що або належать сутності, або успадковуються через зв'язок з іншою сутністю;
- мати один або декілька атрибутів, що однозначно ідентифікують кожен екземпляр сутності.

Сутність може являти собою *супертип*, коли одна сутність є узагальнюючим поняттям для групи подібних сутностей, які називаються підтипами.

Кожній сутності відповідає якась множина атрибутів (attributes), що являють собою властивості сутності.

Наскрізний приклад проектування бази даних ІСПП. Інфологічне моделювання. Виділення сутностей

Звертаючись до аналізу, що наведений у концептуальній моделі, можна дійти висновку, що сутності можуть бути ідентифіковані через їх важливість до тих чи інших користувачів бази даних. Так, є важливими для:

– ГВ – *поверхи, кімнати (аудиторії) і предмети*, за які він відповідає, та дані стосовно площі аудиторій і поверхів, а також дані стосовно номенклатури і кількості меблевих предметів та термінів їх експлуатації;

– ТВ важливі *аудиторії*, але на всі, а тільки ті, де знаходяться *предмети*, за які він відповідає, а також дані стосовно номенклатури і кількості предметів технічного напрямку та термінів їх експлуатації;

– НВ важливі *поверхи, аудиторії* і усі *предмети* в аудиторіях. На підставі площ аудиторій та їх функціонального призначення цей відділ класифікує аудиторії на *типи кімнат*: для лекційних занять, для групових занять, лабораторії і т. д., та виділяє предмети, за які він відповідає.

Для бухгалтерії важлива інформація, пов'язана з кількістю, вартістю, датою встановлення та термінами амортизації предметів. Таким чином, з бухгалтерією не пов'язуються додаткові сутності – з її боку висуваються додаткові вимоги до атрибутів наведених вище сутностей.

Виходячи з вищенаведеного, виділяються 4 сутності (поверхи, приміщення, типи приміщень, предмети), що зображуються на діаграмі в такий спосіб (рис. 12).



Рис. 12. Сутності предметної сфери

Оскільки для ГВ і ТВ важливі одна й та ж сутність «Предмети», але різні її підмножини, то названа сутність являє собою супертип, яка містить два підтипи: «Предмети, що підзвітні ГВ» та «Предмети, що підзвітні ТВ» (рис. 13).

Кінець фрагмента наскрізного прикладу (виділення сутностей в ERD)



Рис. 13. Супертип «Предмети» та його підтипи

Наступним кроком моделювання є ідентифікація зв'язків.

Зв'язок (relationship) – поійменована асоціація між двома сутностями, що має значення для даної предметної сфери. Це асоціація між сутностями, при якій, як правило, кожен екземпляр однієї сутності, яка називається батьківською сутністю, асоційований з довільною (у тому числі нульовою) кількістю екземплярів іншої сутності, яка іменується сутністю-нащадком, а кожен екземпляр сутності-нащадка асоційований у точності з одним екземпляром сутності-батька. Таким чином, екземпляр сутності-нащадка може існувати тільки при існуванні сутності батька. Зв'язок може давати ім'я, що виражається граматичним оборотом дієслова і що поміщається у ромбі. Ім'я кожного зв'язку між двома даними сутностями повинне бути унікальним, але імена зв'язків у моделі не зобов'язані бути унікальними. Ім'я зв'язку

завжди формується з погляду батька. Кожна сутність може мати будь-яку кількість зв'язків з іншими сутностями моделі. На лінії зв'язку (ребрі графа) ставиться також степінь зв'язку між сутностями.

Між двома сутностями, наприклад, **A** і **B** можливі чотири види зв'язків.

Перший тип – зв'язок *один-до-одного* (1:1): у кожен момент часу кожному представнику (екземпляру) сутності **A** відповідає 1 чи 0 представників сутності **B** (рис. 14а). На рис. 14б показано зв'язок типу 1:1 для сутностей **Студент** і **Стипендія**. Студент може не «заробити» стипендію, одержати звичайну чи одну з підвищених стипендій.

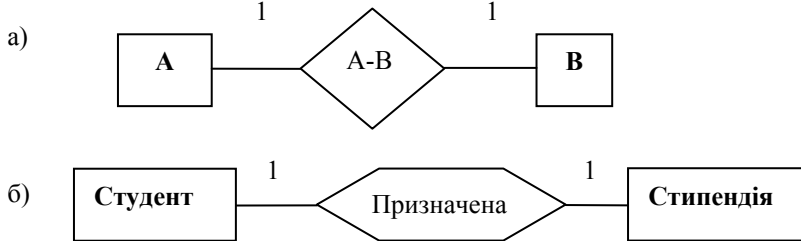


Рис. 14. Приклади зв'язків 1:1

Примітка. На рис. 14 і наступних ER-схемах, виходячи з міркувань зручності, для позначення зв'язку буде використовуватися не тільки ромб, а ще й ромбовидний шестикутник.

Другий тип – зв'язок *один-до-багатьох* (1:N): одному представнику сутності **A** відповідають 0, 1 чи декілька представників сутності **B** (рис. 15а). На рис. 15б показано зв'язок типу 1:N для сутностей **Квартири** й **Мешканці**: квартира може пустувати, в ній може жити один або кілька мешканців.

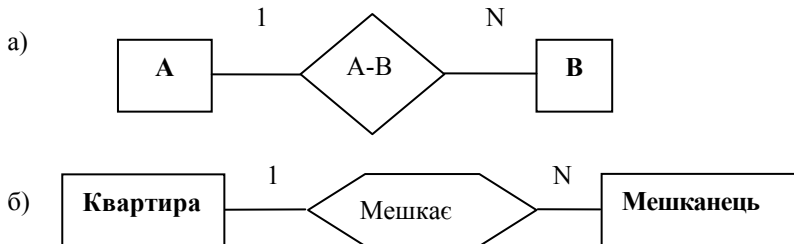


Рис. 15. Приклади зв'язків 1:N

Оскільки між двома сутностями можливі зв'язки в обох напрямках, то існує ще два типи зв'язку *багато-до-одного* (N:1) і *багато-до-багатьох* (M:N).

Приклад 1. Якщо зв'язок між сутностями **Замовники** і **Виробники** називається *Замовлення*, то існує чотири можливих представлення зв'язків:

1:1 – замовник робить замовлення тільки одному виробнику або виготовляє продукцію самотужки;

1:N – замовник робить замовлення одному виробнику або кільком виробникам, або виготовляє продукцію самотужки;

N:1 – виробник або не виготовляє продукцію, або виготовляє для одного замовника, або виготовляє для кількох замовників.

M:N – один виробник може не виробляти продукції для замовників, може виготовляти для одного й більше замовників, а замовник може не замовляти продукцію, або замовляти в одного й більше виробників.

Як уже відмічалось, властивості можуть мати не тільки сутності, але й зв'язки. Особливо це характерно для зв'язків типу M:N. На рис. 16 показано зв'язок між сутностями **Триместри** і **Дисципліни**, де він має такі властивості, як кількість **аудиторних годин** дисципліни у даному триместрі та вид триместрового **контролю** (іспит, залік, рейтинг).

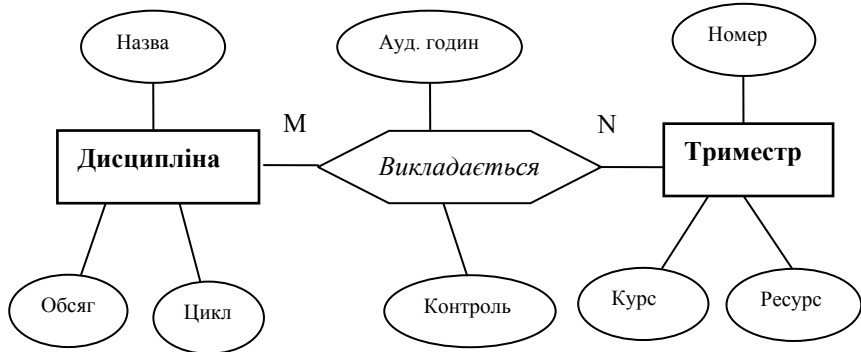


Рис. 16. Приклад наявності властивостей для зв'язку

Якщо ця взаємозалежність (зв'язок) між сутностями має атрибути, то вона називається *асоціацією*.

Характер зв'язків між сутностями не обмежується перерахованими. Існують і більш складні зв'язки:

- різні зв'язки між одними й тими ж самими сутностями (пацієнт, маючи одного лікуючого лікаря, може мати також кілька лікарів-консультантів; лікар може бути лікуючим лікарем декількох пацієнтів і може одночасно консультиувати кілька інших пацієнтів);
- тринарний зв'язок: наприклад, лікар може призначити кілька пацієнтів на кілька аналізів, аналіз може бути призначений декількома лікарями будь-яким пацієнтам і пацієнт може бути призначений на кілька аналізів декількома лікарями (рис. 17, 18);

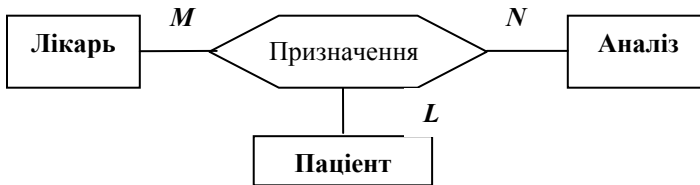


Рис. 17. Приклад тринарного зв'язку між сутностями

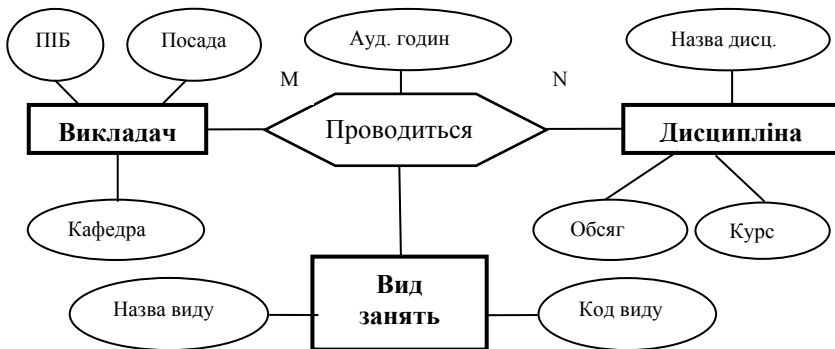


Рис. 18. Приклад тринарного зв'язку між сутностями з їх властивостями

- рекурсивний зв'язок. Наприклад, предмети (компоненти), що входять до складу виробів. Предметами є як самі вироби, так і вузли й деталі. Останні входять до складу вузлів або виробу, вузли нижнього рівня структурного дерева входять до складу вузлів більш високого рівня або виробу і т. д. (рис. 19). Інший рекурсивний зв'язок відображає склад предметів, тобто з яких інших предметів (вузлів,

деталей) складається певний предмет. Обидва рекурсивних зв'язки мають тип 1:N;

- зв'язки більш високих порядків, семантика (зміст) яких іноді дуже складна.

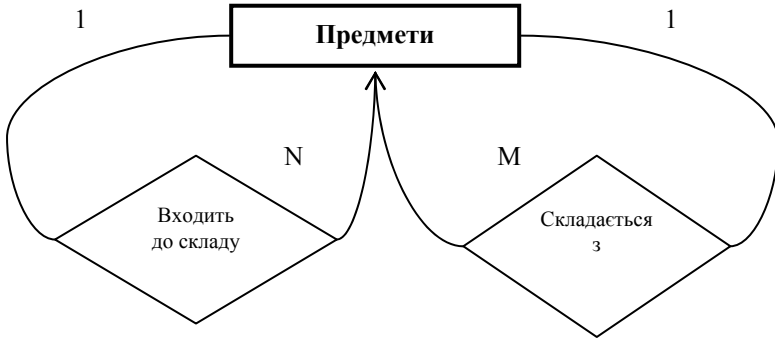


Рис. 19. Приклад рекурсивних зв'язків

Для виявлення зв'язків між сутностями необхідно, як мінімум, визначити самі сутності. Але це не проста задача, тому що в різних предметних сферах той самий об'єкт може бути як сутністю, так і атрибутом. Наприклад, на рис. 18 «Кафедра» є властивістю сутності «Викладачі», але якщо при опису предметної сфери говориться не тільки про назву кафедри, а про її код, керівника кафедри та факультет, до складу якого вона входить (тобто говориться про її властивості), то «Кафедра» вже виступає як сутність (рис. 20).

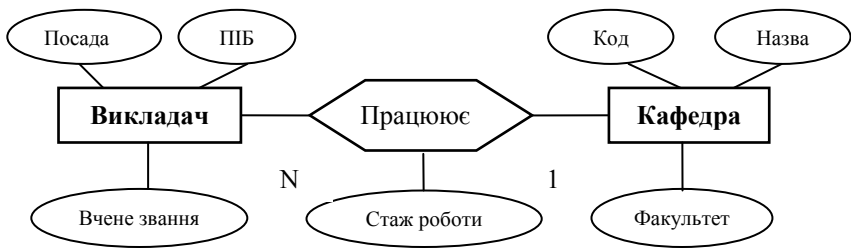


Рис. 20. Приклад ERD, де інформаційний об'єкт **Кафедра** є сутністю

Відмітимо, що вище було представлено не всі елементи нотації Чена. Так, не розкривалося поняття слабких сутностей, слабких зв'язків, супертипів, підтипів, тощо. Крім того, варто відмітити, що,

крім нотації Чена, є й інші нотації (способи представлення складових ER-діаграм, наприклад, нотація Баркера [2], засоби мови UML (Unified Model Language – уніфікована мова моделювання), нотація за стандартом IDEF1X [6] тощо.

Наскрізний приклад проектування бази даних ІСПП. Інфологічне моделювання. Визначення зв'язків

Виходячи із визначених сутностей та із концептуальної моделі, одержимо наступну схему зв'язків сутностей (рис. 21).

Кінець фрагмента наскрізного прикладу (визначення зв'язків у ERD)

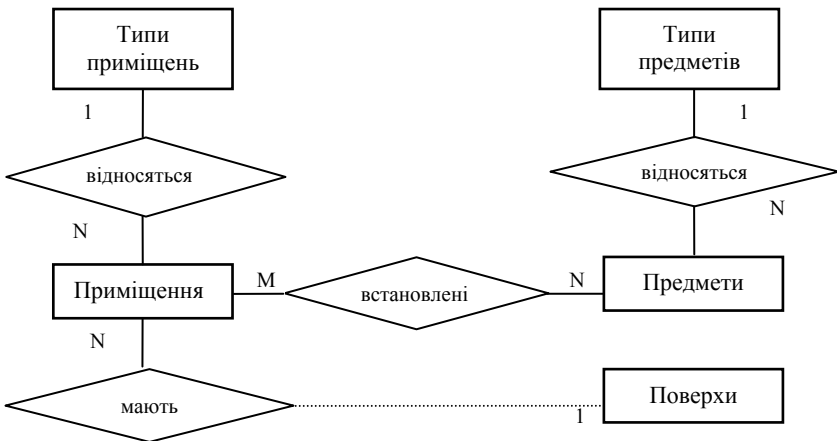


Рис. 21. Схема «Сутності – зв'язки» бази даних про предмети і приміщення

Останнім кроком моделювання є ідентифікація атрибутів.

Атрибут – будь-яка характеристика сутності, значима для розглянутої предметної сфери і призначена для кваліфікації, ідентифікації, класифікації, кількісної характеристики або вираження стану сутності. Атрибут представляє тип характеристик або властивостей, асоційованих з множиною реальних або абстрактних об'єктів (людей, місць, подій, станів, ідей, параметрів, предметів і т. д.). Екземпляр атрибута – це визначена характеристика окремого елемента множини. Екземпляр атрибута визначається типом

характеристики і її значенням, названим значенням атрибута. У ER-моделі атрибути асоціюються з конкретними сутностями. Таким чином, екземпляр сутності повинний володіти єдиним визначеним значенням для асоційованого атрибута.

Атрибут може бути або *обов'язковим*, або *необов'язковим*. Обов'язковість означає, що атрибут не може приймати невизначених значень (null values). Атрибут може бути або *описовим* (тобто звичайним дескриптором сутності), або *входить до складу* унікального ідентифікатора (потенційного ключа).

Унікальний ідентифікатор – це атрибут або сукупність атрибутів та/або зв'язків, призначена для унікальної ідентифікації кожного екземпляра даного типу сутності. У випадку повної ідентифікації кожен екземпляр даного типу сутності цілком ідентифікується своїми власними ключовими атрибутами, у протилежному випадку в його ідентифікації беруть участь також атрибути іншої сутності – сутності – батька.

Кожен атрибут ідентифікується унікальним ім'ям, що виражається граматичним оборотом іменника, який описує характеристику, що представляється атрибутом. Атрибути зображуються, як вже наводилося, у виді овалів, причому кожний атрибут займає окремий овал. Атрибути, що визначають первинний ключ, виділяються, наприклад, жирним шрифтом.

Кожна сутність повинна володіти хоча б одним потенційним ключем. *Потенційний ключ* сутності – це один або кілька атрибутів, чий значення однозначно визначають кожен екземпляр сутності. При існуванні декількох потенційних ключів один з них позначається як *первинний ключ*, а інші – як *альтернативні* або *вторинні* ключі.

Наскрізний приклад проектування бази даних ІСПП. Інфологічне моделювання. Визначення атрибутів та ключів

Аналізуючи концептуальну модель (таблиці 3 і 4), визначаємо атрибути сутностей та доповнюємо побудовану раніше схему зв'язків сутностей (рис. 21) цими атрибутами і одночасно визначаємо ключові атрибути (рис. 22).

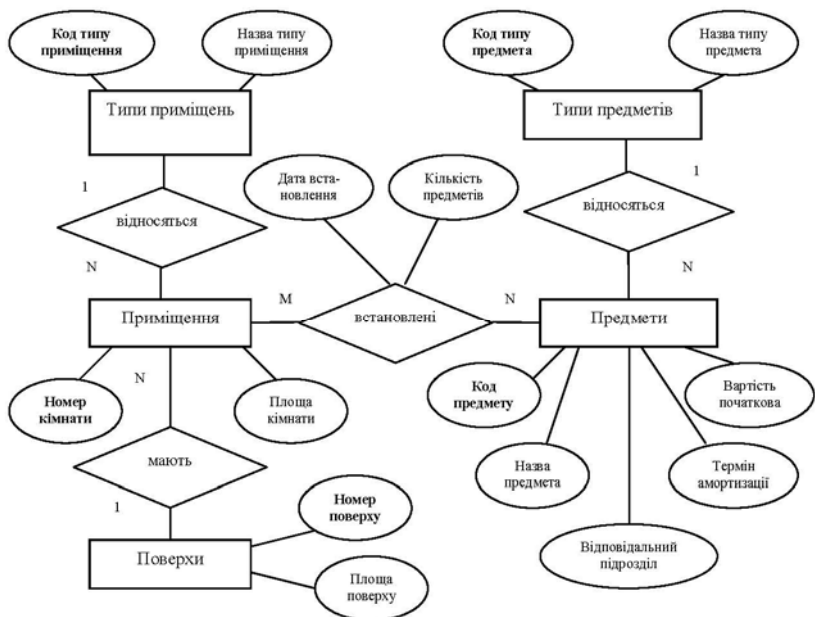


Рис. 22. Повна схема «Сутності – зв’язки» бази даних про предмети і приміщення

Відмітимо, що розглянута вище предметна сфера «Приміщення-предмети» не має рекурсивних зв’язків.

Кінець фрагмента наскрізного прикладу (визначення атрибутів).

3.6. Даталогічна модель бази даних. Схеми функціональних залежностей. Нормалізація відношень

3.6.1. Перетворення інфологічної моделі в даталогічну модель

Даталогічна модель (ДЛМ) являє собою структуру даних: структурні одиниці даних та їх елементи і зв’язки між елементами даних незалежно від їх змісту та середовища зберігання, а особливості обраної СКБД враховуються у фізичній моделі. Таким чином, ДЛМ є подальшою формалізацією інфологічної моделі і являє собою по суті модель даних. Як відомо, існують три основних типи моделей даних: *реляційна*, *ієрархічна* та *мережна*. В даних матеріалах розглядається *реляційна модель* даних (РМД), тобто даталогічна інформаційна

модель представляє сукупність відношень (реляцій), вміст яких змінюється в часі. Тому на першому етапі на основі отриманої схеми «сутність – зв'язок» виконується перехід до реляційної моделі даних. При цьому слід дотримуватися таких правил:

- якщо в ERD між сутностями були встановлені зв'язки 1:1, то в РМД кожна сутність перетворюється у таблицю, а властивості сутностей – в атрибути відповідних таблиць. При цьому може постати питання про з'єднання двох таблиць в одну. На прийняття рішення може вплинути, зокрема, організаційна структура об'єкта інформатизації.

- якщо в ERD між сутностями були встановлені зв'язки 1:N, то в РМД кожна сутність перетворюється у таблицю, а властивості сутностей – в атрибути відповідних таблиць. Додатково до таблиці з боку N, тобто до підпорядкованої таблиці додається ще один атрибут(и) – первинний ключ головної таблиці (таблиці з боку «1»), який у підпорядкованій таблиці буде зовнішнім ключем;

- якщо в ERD між сутностями були встановлені зв'язки M:N, то в РМД також кожна сутність перетворюється у таблицю, а властивості сутностей – в атрибути відповідних таблиць, але додатково створюється ще одна, так звана зв'язувальна таблиця, первинний ключ якої складається із первинних ключів базових таблиць. При цьому, якщо зв'язок має ще й певні властивості, то вони становляться атрибутами зв'язувальної таблиці. Зв'язувальна таблиця може зв'язувати не тільки дві (бінарний зв'язок), але й більше таблиць (m-арний зв'язок).

Що стосується атрибутів таблиць, то виконуються:

- уточнення типів даних та їх властивостей, можливо з урахуванням особливостей вже обраної СКБД;
- уточнення первинних ключів;
- визначення зовнішніх ключів;
- визначення певних обмежень на значення атрибутів або правил, що задають такі обмеження;
- визначення масок уведення даних;
- прийняття рішень щодо класифікації та кодування значень окремих атрибутів, що, як правило, тягне за собою включення до бази даних додаткових таблиць-довідників, що відображають прийняті рішення щодо системи класифікації й кодування (СКК).

Примітка. При виборі СКК треба намагатися використовувати чинні державні або галузеві класифікатори, наприклад, «Державний класифікатор професій», «Галузевий класифікатор рахунків у бухгалтерському обліку» та інші. Але, коли база даних та побудована

на її основі інформаційна система мають локальний характер, то й СКК може бути локальною.

**Наскрізний приклад проектування бази даних ІСПП.
Даталогічна модель**

Виходячи із правил перетворення ERD у даталогічну, остання для бази даних «Приміщення предмети» буде виглядати таким чином (рис. 23).



Рис. 23. Таблиці та схема взаємозв'язків даталогічної моделі

Як бачимо, при уведенні зв'язувальної таблиці зв'язки типу M:N стали відсутніми. Для позначення типу зв'язку 1:N тут використано направлену лінію, де стрілка відповідає стороні «N». Крім того, в нотацію додані такі речі, як позначення зовнішніх ключів та складених первинних ключів. Зовнішні ключі позначені символом «&», сполучення символів «&#» означає, що даний атрибут є, з одного боку, зовнішнім ключем, а з іншого – часткою складеного первинного ключа зв'язувальної таблиці. Символом «*» позначено альтернативні ключі.

Далі на цьому етапі приймаємо рішення щодо ідентифікаторів таблиць і атрибутів, а також типів даних та інших характеристик атрибутів. Ці рішення зведемо в таблицю (табл. 6). В таблиці прийняті такі позначення ключів: ПК – первинний ключ; ВК – вторинний (альтернативний) ключ; ЗК – зовнішній ключ; СПК – складений первинний ключ (альтернативою таким позначенням можуть бути їх англійські позначення: PK – Primary key; SK – Secondary key; FK – Foreign key; CPK – Composite primary key).

Кінець фрагмента наскрізного прикладу (Даталогічна модель)

Таблиця 6

Склад та характеристики атрибутів таблиць ДІМ

Сутність	Назва атрибуту	Ключ: ПК, ВК, ЗК, ЗК+СПК	Обов'язкове значення	Тип даних	Обмеження
Таблиця FLOORS – «ПОВЕРХИ»					
NbFl	Номер поверху (числовий)	ПК	Так	Ціле число	{1,2,3,4,5}
NmFl	Номер поверху (літерний)	ВК	Так	Текст	До 10 символів
SqFl	Загальна площа кімнат на поверсі		Так	Дійсне число	Від 100,0 до 400,0
Таблиця ROOMS – «КІМНАТИ»					
NbRm	Номер кімнати	ПК	Так	Ціле число	Від N01 до N15, де N-номер поверху
CdTR	Код типу кімнати	ЗК до TPRM		Текст	До 2 символів
SqRm	Площа кімнати		Так	Дійсне число	Від 7,0 до 180,0
NbFl	Номер поверху (числовий)	ЗК до FLOORS	Так	Ціле число	{1,2,3,4,5}
Таблиця ITEMS – «ПРЕДМЕТИ»					
CdPr	Код предмета	ПК	Так	Ціле число	Від 1 і більше
NmPr	Назва предмета	ВК	Так	Текст	До 50 символів
CdTP	Код типу предмета	ЗК до TPIT	Так	Текст	До 2 символів

Закінчення табл. 6

BgCs	Початкова вартість предмета		Так	Дійсне число	Більше 0.0
TrUs	Термін амортизації		Так	Дійсне число	Більше 0.0
Таблиця TPRM – «ТИП КІМНАТ»					
CdTR	Код типу кімнати	ПК	Так	Текст	До 2 символів
NmTR	Назва типу кімнат	ВК	Так	Текст	До 50 символів
Таблиця TRIT – «ТИПИ ПРЕДМЕТІВ»					
CdTP	Код типу предмету	ПК	Так	Текст	До 2 символів
NmTP	Назва типу предмету	ВК	Так	Текст	До 50 символів
Таблиця IT RM – «КІМНАТИ ПРЕДМЕТІ»					
NbRm	Номер кімнати	СПК, ЗК до ROOMS	Так	Ціле число	Від N01 до N15, де N-номер поверху
CdPr	Код предмета	СПК, ЗК до ITEMS	Так	Ціле число	Від 1 і більше
DtVs	Дата встановлення предметів		Так	Дата (рік, місяць)	01.01.2000 <Date <Поточної
QtyP	Кількість предметів у кімнаті		Так	Ціле число	Більше 0

Наступним етапом робіт зі створення даталогічної моделі є нормалізація бази даних, методика проведення якої наведено у наступному пункті.

3.6.2. Принципи нормалізації. Основні властивості нормальних форм

За класичним підходом процес проектування реляційної БД виконується в термінах реляційної моделі даних засобом послідовних наближень до задовільного набору схем відношень. Початковим є подання предметної сфери у вигляді одного або декількох відношень, і на кожному етапі проектування виробляється деякий набір схем відношень, які мають кращі властивості. Процес проектування являє собою процес нормалізації схем відношень, причому кожна наступна нормальна форма має кращі властивості, ніж попередня.

Кожній нормальній формі відповідає деякий певний набір обмежень, і відношення знаходиться в деякій нормальній формі, якщо їй задовольняє властивий набір обмежень.

У теорії реляційних БД, як правило, виділяють таку послідовність нормальних форм:

- перша нормальна форма – 1НФ (1NF);
- друга нормальна форма – 2НФ (2NF);
- третя нормальна форма – 3НФ (3NF);
- нормальна форма Бойса-Кодда – БКНФ (BCNF);
- четверта нормальна форма – 4НФ (4NF);
- п'ята нормальна форма, або нормальна форма проєкції-сполучення – 5НФ або ПС/НФ (5NF або PJ/NF).

Основні властивості нормальних форм:

- кожна наступна нормальна форма в деякому розумінні поліпшує властивості попередньої;
- у разі переходу до наступної нормальної форми властивості попередніх нормальних форм зберігаються.

В основу класичного процесу проектування покладено засіб нормалізації, що спирається на декомпозицію (на основі проєкції) відношення, яке знаходиться у попередній нормальній формі, у два або більше відношень, які задовольняють вимоги наступної нормальної форми.

Найважливіші на практиці нормальні форми відношень ґрунтуються на фундаментальному в теорії реляційних БД понятті функціональної залежності. Для подальшого викладення необхідно ввести ряд визначень.

Визначення 1. У відношенні R атрибут Y функціонально залежить від атрибута X (X і Y можуть бути складеними атрибутами, тобто реально складатися з декількох атомарних атрибутів) тільки в тому випадку, якщо кожному значенню X відповідає тільки одне значення Y : $R.X \rightarrow R.Y$.

Визначення 2. Функціональну залежність $R.X \rightarrow R.Y$ називають *повною*, якщо атрибут Y не залежить функціонально від будь-якої точної

підмножину X (точною підмножиною множини X називають будь-яку її підмножину, яка не збігається з множиною X).

Визначення 3. Функціональну залежність $R.X \rightarrow R.Y$ називають *транзитивною*, якщо існує такий атрибут Z , що існує в функціональній залежності $R.X \rightarrow R.Z$ і $R.Z \rightarrow R.Y$.

Визначення 4. *Неключовим атрибутом* називають будь-який атрибут відношення, що не входить до складу первинного ключа.

Визначення 5. *Можливим ключем* відношення називають його атомарний або складений атрибут, значення якого повністю функціонально визначають значення всіх інших атрибутів відношення.

Визначення 6. Два або більше атрибутів називають *взаємно незалежними*, якщо жоден з них не є функціонально залежним від інших атрибутів.

3.6.3. Перша нормальна форма

Будь-яке відношення знаходиться у першій нормальній формі, тобто відношення R знаходиться у першій нормальній формі (1NF) тоді і лише тоді, коли воно має такі властивості:

- атомарність значень атрибутів;
- відсутність кортежів-дублікатів;
- відсутність атрибутів-дублікатів.

При цьому вимоги щодо упорядкованості кортежів та упорядкованості атрибутів не є обов'язковими.

3.6.4. Друга нормальна форма

Розглянемо такий приклад схеми відношення:

СПІВРОБІТНИКИ-ВІДДІЛИ-ПРОЕКТИ(СПІВР_НОМЕР, СПІВР_ЗАРП, ВІД_НОМЕР, ПРО_НОМЕР, СПІВР_ЗАВД). Хай, наприклад, таблиця, що відповідає схемі цього відношення, має такий вміст.

СПІВР_НОМЕР	СПІВР_ЗАРП	ВІД_НОМЕР	ПРО_НОМЕР	СПІВР_ЗАВД
E001	100	D01	P01	T001
E002	100	D01	P01	T001
E003	100	D01	P02	T001
E004	100	D01	P02	T002
E005	200	D02	P01	T001
E006	200	D02	P01	T002
E007	200	D02	P02	T001
E008	200	D02	P02	T002

Первинний ключ: <СПІВР_НОМЕР, ПРО_НОМЕР>

Функціональні залежності (ФЗ): СПІВР_НОМЕР → СПІВР_ЗАРП;
СПІВР_НОМЕР → ВІД_НОМЕР; ВІД_НОМЕР → СПІВР_ЗАРП;
СПІВР_НОМЕР, ПРО_НОМЕР → СПІВР_ЗАВД .

Наведені ФЗ можна представити такою схемою (рис. 24).



Рис. 24. Схема функціональних залежностей відношення СПІВРОБІТНИКИ–ВІДДІЛИ–ПРОЕКТИ

Як видно, хоча первинним ключем є складений атрибут СПІВР_НОМЕР, ПРО_НОМЕР, атрибути СПІВР_ЗАРП і ВІД_НОМЕР функціонально залежать від частини первинного ключа, тобто атрибута СПІВР_НОМЕР. У результаті, не можна вставити у відношення СПІВРОБІТНИКИ–ВІДДІЛИ–ПРОЕКТИ кортеж, який описує співробітника, котрий ще не виконує жодного проекту (первинний ключ не може містити невизначене значення). Під час вилучення кортежу не тільки руйнується зв'язок даного співробітника з даним проектом, але втрачається інформація про те, що він працює в деякому відділі. У разі переведення співробітника в інший відділ, необхідно модифікувати всі кортежі, які описують цього співробітника, інакше отримаємо непогоджений результат. Такі неприємні явища називають *аномаліями схеми відношення*, їх усувають шляхом нормалізації.

Визначення 7. Відношення R знаходиться у другій нормальній формі (2NF) тоді і лише тоді, коли знаходиться в 1NF, і кожний неключовий атрибут повністю залежить від первинного ключа.

Для зведення відношення R до другої нормальної форми можна провести наступну декомпозицію відношення СПІВРОБІТНИКИ–ВІДДІЛИ–ПРОЕКТИ у два відношення СПІВРОБІТНИКИ–ВІДДІЛИ і СПІВРОБІТНИКИ–ПРОЕКТИ.

Вміст відношення СПІВРОБІТНИКИ–ВІДДІЛИ (СПІВР_НОМЕР, СПІВР_ЗАРП, ВІД_НОМЕР):

СПІВР_НОМЕР	СПІВР_ЗАРП	ВІД_НОМЕР
E001	100	D01
E002	100	D01
E003	100	D01
E004	100	D01
E005	200	D02
E006	200	D02
E007	200	D02
E008	200	D02

Первинний ключ: <СПІВР_НОМЕР >

Функціональні залежності: СПІВР_НОМЕР → СПІВР_ЗАРП;
СПІВР_НОМЕР → ВІД_НОМЕР; ВІД_НОМЕР → СПІВР_ЗАРП

Вміст відношення СПІВРОБІТНИКИ–ПРОЕКТИ (СПІВР_НОМЕР, ПРО_НОМЕР, СПІВР_ЗАВД):

СПІВР_НОМЕР	ПРО_НОМЕР	СПІВР_ЗАВД
E001	P01	T001
E002	P01	T001
E003	P02	T001
E004	P02	T002
E005	P01	T001
E006	P01	T002
E007	P02	T001
E008	P02	T002

Первинний ключ: <СПІВР_НОМЕР, ПРО_НОМЕР>

Функціональна залежність:

СПІВР_НОМЕР, ПРО_НОМЕР → СПІВР_ЗАВД

Схеми ФЗ цих відношень наведено на рис. 25.



Рис. 25. Схема функціональних залежностей відношень СПІВРОБІТНИКИ–ПРОЕКТИ та СПІВРОБІТНИКИ–ВІДДІЛИ

Кожне з цих двох відношень знаходиться в 2NF, і в них усунені відзначені вище аномалії.

3.6.5. Третя нормальна форма

Розглянемо ще раз відношення співробітники-відділи, яке знаходиться у 2NF. Відзначимо, що функціональна залежність СПІВР_НОМЕР→СПІВР_ЗАРП є транзитивною, тобто вона є наслідком (у математичному розумінні) функціональних залежностей СПІВР_НОМЕР→ ВІД_НОМЕР та ВІД_НОМЕР→СПІВР_ЗАРП. Інакше кажучи, заробітна плата співробітника насправді є характеристикою не співробітника, а відділу, в якому він працює (це не дуже природне припущення, але достатнє для прикладу).

У результаті, не можна занести у БД інформацію, що характеризує заробітну плату відділу, доки в цьому відділі не з'явиться хоча б один співробітник (первинний ключ не може містити невизначене значення). У разі вилучення кортежу, що описує останнього співробітника даного відділу, втрачається інформація про заробітну плату відділу. Для того, щоб за погодженням змінити заробітну плату відділу, необхідно заздалегідь знайти всі кортежі, що описують співробітників цього відділу. Тобто у відношенні СПІВРОБІТНИКИ–ВІДДІЛИ й досі існують аномалії, їх можна усунути шляхом подальшої нормалізації.

Визначення 8. Відношення R знаходиться у третій нормальній формі (3NF) в тому і тільки в тому випадку, якщо воно знаходиться в 2NF і кожний неключовий атрибут нетранзитивно залежить від первинного ключа.

Еквівалентним альтернативним визначенням третьої нормальної форми є визначення 9.

Визначення 9. Відношення R знаходиться у третій нормальній формі (3NF) в тому і тільки в тому випадку, якщо всі неключові атрибути R взаємно незалежні і повністю залежать від первинного ключа.

Виконаємо декомпозицію відношення СПІВРОБІТНИКИ–ВІДДІЛИ у два відношення: СПІВРОБІТНИКИ і ВІДДІЛИ.

Відношення СПІВРОБІТНИКИ(СПІВР_НОМЕР, ВІД_НОМЕР):

СПІВР_НОМЕР	ВІД_НОМЕР
E001	D01
E002	D01
E003	D01
E004	D01
E005	D02
E006	D02
E007	D02
E008	D02

Первинний ключ: <СПІВР_НОМЕР>

Функціональна залежність: СПІВР_НОМЕР → ВІД_НОМЕР

Відношення ВІДДІЛИ(ВІД_НОМЕР, СПІВР_ЗАРП):

ВІД_НОМЕР	СПІВР_ЗАРП
D01	100
D02	200

Первинний ключ: <ВІД_НОМЕР>

Функціональна залежність: ВІД_НОМЕР → СПІВР_ЗАРП

Кожне з цих двох відношень знаходиться в 3NF і не має описаних вище аномалій. Схема ФЗ для отриманої в результаті нормалізації бази даних наведено на рис. 26.

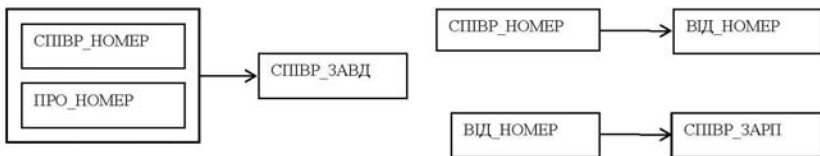


Рис. 26. Схема функціональних залежностей відношень СПІВРОБІТНИКИ–ПРОЕКТИ, СПІВРОБІТНИКИ та ВІДДІЛИ

На практиці третя нормальна форма схем відношень є достатньою у більшості випадків, і зведенням до третьої нормальної форми процес проектування реляційної БД, як правило, закінчується. Однак інколи доцільно продовжити процес нормалізації.

3.6.6. Нормальна форма Бойса-Кодда

На жаль, навіть відношення у ЗНФ можуть мати аномалії. Це може бути, коли для відношення мають місце умови:

- 1) відношення має два або більше альтернативні (потенційні) ключі;
- 2) альтернативні ключі є складеними;
- 3) складені ключі перекриваються (тобто мають спільний атрибут або атрибути).

У цьому разі визначення третьої нормальної форми удосконалюється для запобігання аномалій). Розглянемо такий приклад схеми відношення КОНСУЛЬТАНТ (IdSTUD, COURSE, TEACHER), де:

IdSTUD – код студента;

COURSE – дисципліна (курс, предмет);

TEACHER – викладач.

Хай умови до цього відношення будуть такими:

1. Студент може обирати одну або кілька дисциплін;
2. Консультантами з одного й того ж предмета можуть бути кілька викладачів;
3. Кожний викладач може бути консультантом тільки з одного предмета.

Крім того, будемо вважати, прізвища у всіх викладачів різні. Приклад вмісту відношення КОНСУЛЬТАНТ наведено на рис. 27.

IdSTUD	COURSE	TEACHER
100	Математика	Тихта
150	Програмування	Дибенко
200	Математика	Кейфура
250	Математика	Тихта
300	Схемотехніка	Колобуто
300	Програмування	Бадрак

Рис. 27. Вміст відношення КОНСУЛЬТАНТ

Оскільки студенти можуть спеціалізуватися за декількома предметами, атрибут IdSTUD не визначає атрибут COURSE. Крім того, оскільки студент може мати декілька консультантів, то атрибут IdSTUD не визначає й атрибут TEACHER. Таким чином, атрибут IdSTUD сам по собі не може бути ключем.

Сполучення <IdSTUD,COURSE> визначає атрибут TEACHER, а сполучення <IdSTUD,TEACHER> визначає атрибут COURSE. Таким чином, можливі (потенційні) ключі відношення КОНСУЛЬТАНТ – це <IdSTUD,COURSE> та <IdSTUD,TEACHER>.

Якщо первинним ключем обрати <IdSTUD,COURSE>, то функціональні залежності будуть такими: <IdSTUD,COURSE>→TEACHER та TEACHER→COURSE.

Якщо первинним ключем обрати <IdSTUD,TEACHER>, то функціональні залежності будуть такими: <IdSTUD,TEACHER>→COURSE та COURSE→TEACHER.

Схеми ФЗ для обох випадків наведено на рис. 28.

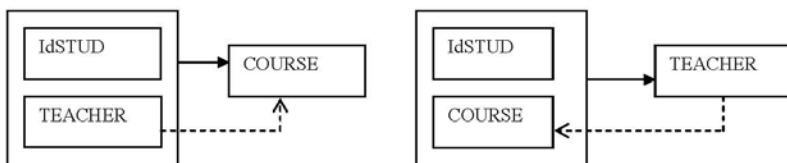


Рис. 28. Схеми ФЗ при різних первинних ключах

За визначенням, відношення КОНСУЛЬТАНТ знаходиться у 1НФ. Воно також знаходиться у 2НФ, оскільки не має неключових атрибутів (кожний атрибут є частиною, як мінімум, одного ключа). Воно ж знаходиться і в 3НФ, тому що не має транзитивних залежностей. Тим не менш, воно має певні аномалії незалежно від того, який з можливих ключів обраний як первинний ключ. Це є наслідком того факту, що:

або одночасно певний атрибут відношення функціонально залежить як від складеного ключа, так і від частини цього ключа (ліва схема на рис. 28);

або існують функціональні залежності атрибутів складеного ключа відношення від неключового атрибута (права схема на рис. 28).

Хай студента з номером 300 відраховано з університету. При видаленні цього рядка буде втрачено інформацію про те, що викладач Бадрак є консультантом з дисципліни «Програмування». Це аномалія видалення. Далі, ми не зможемо додати в базу даних інформацію про нового викладача дисципліни «Фізика» Хрунова доти, поки до нього не запишеться хоча б один із студентів. Це аномалія додавання.

Ситуації, що подібні вищенаведеним, привели до визначення *нормальної форми Бойса-Кодда* – БКНФ (*Boyce-Codd normal form – BK/NF*). На відміну від 2НФ і 3НФ, ця нормальна форма визначається

не через попередні нормальні форми, а через таке поняття, як детермінант.

Визначення 10. *Детермінантом* називають будь-яке сполучення атрибутів, від якого функціонально повністю залежить деякий інший атрибут. Іншими словами – це ліва частина в схемі ФЗ.

Так, із схеми ФЗ на рис. 28 видно, що у відношенні КОНСУЛЬТАНТ детермінант TEACHER не є потенційним ключем відношення.

Визначення 11. Відношення R знаходиться у нормальній формі Бойса-Кода в тому і тільки в тому випадку, якщо кожний детермінант є можливим ключем.

З цього визначення випливає, що в цій формі кожний з неключових атрибутів повністю визначається первинним ключем і не залежить від його частини (у разі складеного первинного ключа).

Зауваження. Легко помітити, що якщо у відношенні є тільки один можливий ключ (і він є первинним ключем), то це визначення стає еквівалентним визначенню третьої нормальної форми.

Для приведення відношення до БКНФ також застосовують операцію «проєкція», результати якої наведено на рис. 29.

STUDENT-TEACHER TEACHER- COURSE

IdSTUD	TEACHER	TEACHER	COURSE
100	Тихта	Тихта	Математика
150	Дибенко	Дибенко	Програмування
200	Кейфура	Кейфура	Математика
250	Тихта	Колобуто	Схемотехніка
300	Колобуто	Бадрак	Програмування
300	Бадрак		

Рис. 29. Відношення в БКНФ

3.6.7. Четверта нормальна форма

Повернемося до бази даних про проекти, завдання у проектах та співробітників, що виконують ці проекти. Припустимо, що будь-який співробітник, що бере участь у проекті, виконує всі завдання, передбачені цим проектом. Розглянемо приклад наступної схеми відношення. Відношення ПРОЕКТИ (ПРО_НОМЕР, ПРО_СПІВ, ПРО_ЗАВД):

ПРО НОМЕР	ПРО СПІВ	ПРО ЗАВД
P01	E001	T001
P01	E001	T002
P01	E002	T001
P01	E002	T002
P02	E003	T001
P02	E003	T002
P02	E004	T001
P02	E004	T002

Відношення ПРОЕКТИ містить номери проектів, для кожного проекту – список співробітників, які можуть виконувати проект, і список завдань, що передбачає проект. Співробітники можуть брати участь у декількох проектах, і різні проекти можуть містити однакові номери завдань.

Кожний кортеж відношення зв'яже деякий проект зі співробітником, що бере участь у цьому проекті, і завданням, яке співробітник виконає у рамках проекту. Враховуючи умови, сформульовані вище, єдиним можливим ключем відношення є складений атрибут ПРО_НОМЕР, ПРО_СПІВ, ПРО_ЗАВД, і немає жодних інших детермінантів. Отже, відношення ПРОЕКТИ знаходиться у BCNF. Але при цьому воно має недоліки: якщо, наприклад, деякий співробітник приєднується до проекту, необхідно вставити у відношення ПРОЕКТИ стільки кортежів, скільки завдань у ньому передбачено.

Визначення 12. У відношенні $R(A,B,C)$ існує багатозначна залежність, яку будемо тут позначати як $R.A \twoheadrightarrow R.B$, тоді і лише тоді, коли множина значень B , яка відповідає парі значень A і C , залежить тільки від A і не залежить від C .

У відношенні ПРОЕКТИ існують дві багатозначні залежності:
 ПРО_НОМЕР \twoheadrightarrow ПРО_СПІВ

ПРО_НОМЕР \twoheadrightarrow ПРО_ЗАВД

Р. Фейджин (R. Fagin) показав [3,5,7], що у загальному випадку у відношенні $R(A,B,C)$ існує багатозначна залежність $R.A \twoheadrightarrow R.B$ лише в тому випадку, коли існує багатозначна залежність $R.A \twoheadrightarrow R.C$. Отже, багатозначні залежності завжди створюють зв'язані пари, що позначають виразом $A \twoheadrightarrow B|C$.

Подальша нормалізація відношень, подібних до відношення ПРОЕКТИ, ґрунтується на *теоремі Фейджина*:

Відношення $R(A,B,C)$ можна спроекувати без втрат у відношення $R_1(A,B)$ і $R_2(A,C)$ лише тоді, коли існує багатозначна залежність $A \twoheadrightarrow B|C$.

Під проектуванням без втрат розуміємо такий засіб декомпозиції відношення, за якого вхідне відношення повністю і без надлишковості відновлюється шляхом природного сполучення отриманих відношень.

Визначення 13. Відношення R знаходиться в четвертій нормальній формі (4NF) лише в тому разі, коли за умови існування багатозначної залежності $A \twoheadrightarrow B$ всі інші атрибути R функціонально залежать від A.

Інакше кажучи, четверта нормальна форма не містить багатозначної залежності між атрибутами.

У прикладі, що наводився вище, можна зробити декомпозицію відношення ПРОЕКТИ у два відношення: ПРОЕКТИ-СПІВРОБІТНИКИ і ПРОЕКТИ-ЗАВДАННЯ.

Відношення ПРОЕКТИ-СПІВРОБІТНИКИ (ПРО_НОМЕР, ПРО_СПІВ):

ПРО НОМЕР	ПРО СПІВ
P01	E001
P01	E002
P02	E003
P02	E004

Відношення ПРОЕКТИ-ЗАВДАННЯ (ПРО_НОМЕР, ПРО_ЗАВД):

ПРО НОМЕР	ПРО ЗАВД
P01	T001
P01	T002
P02	T001
P02	T002

Обидва відношення знаходяться в 4NF.

3.6.8. П'ята нормальна форма

В усіх розглянутих до цього моменту нормалізаціях проводилась декомпозиція одного відношення у два. Інколи це зробити не вдається, але можлива декомпозиція у більшу кількість відношень, кожне з яких має кращі властивості.

У якості приклада розглянемо відношення SPW(IdS,IdP,IdW):

IdS – код постачальника;

IdP – код предмета;

IdW – код складу.

Відношення SPW відображає постачання предметів P_i до складу W_j постачальником S_k.

Приклад вмісту таблиці SPW наведено нижче.

IdS	IdP	IdW
S1	P1	W2
S1	P2	W1

Первинним ключем цього відношення є повна сукупність його атрибутів. У відношенні немає функціональних і багатозначних залежностей, тому відношення знаходиться в 4НФ. Однак у ньому можуть існувати аномалії.

Хай у наведеному вище прикладі умовами постачання є вимога, щоб кожний з постачальників доставляв усі предмети, що є на складі. Припустимо, що постачальник S2 доставив предмет P1 до складу W1, що вимагає від власника таблиці бази даних додати кортеж <S2,P1,W1> (рис. 30a).

IdS	IdP	IdW
S1	P1	W2
S1	P2	W1
S2	P1	W1

a)

IdS	IdP	IdW
S1	P1	W2
S1	P2	W1
S2	P1	W1
S1	P1	W1

b)

Рис. 30. Приклад залежності сполучення, що приводить до аномалії додавання

Але, додавання цього кортежу означає, що до складу W1, крім предмета P2, ще постачається предмет P1, а постачальник S1 доставляє до цього складу тільки предмет P2. Тому одночасно з уведенням кортежу <S2,P1,W1> треба увести и кортеж <S1,P1,W1>. Це аномалія додавання. Новий вміст таблиці наведено на рис. 7.13b.

Тепер проаналізуємо операції видалення. Якщо видаляти кортеж <S1,P1,W1>, то проблем не виникає: відомості про предмети, що доставляються до складів, залишаться: до складу W2 доставляється всього один предмет P1, а до W1 – предмети P1 і P2, і постачальник S1 їх доставляє. Але, коли ми будемо вилучати кортеж <S2,P1,W1>, то це є наслідком того, що до складу W1 уже більше не треба постачати предмет P1, тому одночасно з цим кортежем треба вилучити й кортеж <S1,P1,W1>. Це аномалія видалення.

Наведених аномалій можна уникнути шляхом декомпозиції відношення SPW у три відношення. Взагалі для будь-якого

відношення R із n атрибутів можна побудувати n проєкцій по кожному з цих атрибутів. Сполучення отриманих відношень (усіх або деяких) являє собою нове відношення. Таке відношення може або збігатися з первинним (тобто відновлюється без втрат), або ні.

Визначення 14. Відношення $R(X, Y, \dots, Z)$ задовольняє залежності сполучення $*(X, Y, \dots, Z)$ в тому і тільки в тому випадку, коли R відновлюється без втрат шляхом сполучення своїх проєкцій на X, Y, \dots, Z .

Інакше кажучи, відношення залежить від сполучення своїх проєкцій.

Визначення 15. П'ята нормальна форма. Відношення R знаходиться у п'ятій нормальній формі (нормальній формі проєкції-сполучення – PJ/NF) в тому і тільки в тому випадку, коли будь-яка залежність сполучення в R створюється з проєкцій за ключовими атрибутами, які в сукупності є можливим ключем відношення.

Виконаємо три проєкції відношення SPW(IdS, IdP, IdW), отримаємо три нових відношення: SP, PW та SW (рис. 31).

IdS	IdP
S1	P1
S1	P2
S2	P1

IdP	IdW
P1	W2
P2	W1
P1	W1

IdS	IdW
S1	W2
S1	W1
S2	W1

Рис. 31. Три проєкції відношення SPW

Нові таблиці відповідають усім вимогам 5НФ і попередніх нормальних форм, але потрібно перевірити, чи буде відновлена початкова таблиця після операції сполучення. Нижче на рис. 32а наведено таблицю SPW1, яка є результатом операції $SP \text{ join } PW \text{ on } SP.IdP = PW.IdP$. Як бачимо, це сполучення не відновлює початкову таблицю SPW.

IdS	IdP	IdW
S1	P1	W2
S1	P2	W1
S2	P1	W1
S2	P1	W2
S1	P1	W1

IdS	IdP	IdW
S1	P1	W2
S1	P2	W1
S2	P1	W1
S1	P1	W1

a)

b)

Рис. 32. Результати послідовності операцій відновлення відношення SPW

Якщо останню таблицю з'єднати з третьою проекцією SW, тобто $SPW1 \text{ join } SW \text{ on } SPW1.IdS=SW.IdS \text{ and } SPW1.IdW=SW.IdW$, то отримаємо таблицю SPW2 (рис. 32b).

Як бачимо, таблиця SPW2 відновлює таблицю SPW.

П'ята нормальна форма – це остання нормальна форма, яку можна отримати шляхом декомпозиції з 4НФ. Її умови достатньо нетривіальні, і на практиці 5НФ майже не зустрічається, але такі випадки можуть траплятися. Відзначимо, що залежність сполучення є узагальненням як багатозначної, так і функціональної залежностей.

Наскрізний приклад проектування бази даних ІСПП. Даталогічна модель. Функціональні залежності. Нормалізація відношень бази даних

Оскільки вище були виконані роботи зі створення ЛІМ, у тому числі була побудована ER-діаграма, яка за певними правилами була перетворена в даталогічну модель, то можна очікувати, що отримані відношення будуть знаходитися у 2-й і навіть у 3-й нормальній формі. Але для перевірки ступеня нормалізованості побудуємо схеми функціональних залежностей для кожного із відношень (рис. 33).

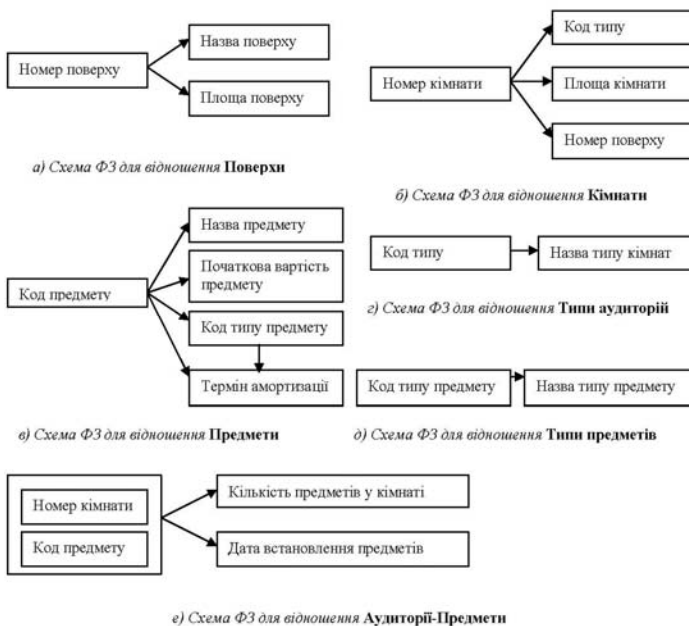


Рис. 33. Схеми функціональних залежностей отриманої ДЛІМ

Як видно із схем ФЗ, усі відношення відповідають вимогам 2НФ, тому що усі не ключові атрибути знаходяться у функціональній залежності від ключових. Вимогам 3НФ відповідають усі відношення за виключенням відношення «Предмети», тому що там, у процесі подальшої роботи із бухгалтерією, з'ясувалося, що термін амортизації залежить від типу предмета, тому цей факт знайшов відображення у наявності функціональної залежності не ключового атрибута «Термін амортизації» від іншого неключового атрибута «Код типу предмета», тобто існує транзитивна залежність неключового атрибута «Термін амортизації» від ключового атрибута «Код предмета» через неключовий атрибут «Код типу предмета». Для приведення відношення **Предмети** до 3НФ виконуємо проєкцію без втрат цього відношення на два: **Предмети** (Код предмета, Назва предмета, Код типу предмета) та **Норми_амортизації** (Код типу предмета, Термін амортизації). З урахуванням виконаної операції схема ФЗ, що представлена на рис. 30в, перетвориться у нові дві схеми ФЗ (рис. 34).



Рис. 34. Схеми функціональних залежностей після вилучення транзитивної залежності у відношенні **Предмети**

Таким чином, тепер усі відношення відповідають вимогам 3НФ. Але тепер дві таблиці: «*Типи предметів*» і «*Норми амортизації*» мають загальний ключ «*Код типу предмета*», тому два названих відношення можуть бути об'єднані в одне **Типи_предметів** (Код типу предмета, Назва типу предмета, Термін амортизації), а відношення **Норми амортизації** вилучити із бази даних. Схема функціональних залежностей для нового відношення **Типи_предметів** показано на рис. 35.

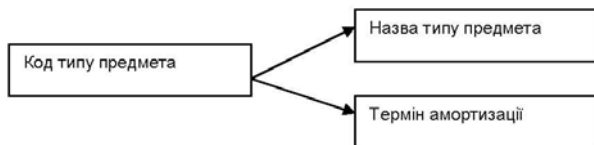


Рис. 35. Кінцевий варіант схеми ФЗ для відношення **Типи_предметів**

Розглянемо, чи відповідають вони вимогам нормальних форм більш високого порядку.

Предметом дослідження на невідповідність НФБК можуть бути тільки відношення, що мають складений ключ. Таким відношенням у даній базі даних є тільки відношення **Кімнати_Предмети**. Але це відношення не має альтернативних ключів, тому воно, зокрема, і вся база даних знаходиться в НФБК.

Предметом дослідження на невідповідність 4НФ можуть бути тільки відношення, де існують багатозначні залежності. В даній базі даних вони відсутні, тому вся база даних знаходиться в 4НФ.

Предметом дослідження на невідповідність 5НФ можуть бути тільки відношення, де є складений ключ із трьох і більше атрибутів. Оскільки серед відношень цієї бази даних такі відсутні, тому вся база даних знаходиться і в 5НФ.

З урахуванням результатів нормалізації наведемо кінцевий варіант даталогічної моделі у вигляді схеми взаємозв'язків між таблицями (рис. 36) та переліку атрибутів таблиць з їх характеристиками (табл. 7). Останні є уточненням рис. 23 і табл. 6 відповідно.

Кінець фрагмента наскрізного прикладу (Даталогічна модель. Функціональні залежності. Нормалізація відношень бази даних)



Рис. 36. Таблиці та схема взаємозв'язків кінцевого варіанта даталогічної моделі

Таблиця 7

**Склад та характеристики атрибутів таблиць
кінцевого варіанта ДЛМ**

Сутність	Назва атрибуту	Ключ: ПК, ВК, ЗК, ЗК+СПК	Обов'язкове значення	Тип даних	Обмеження
Таблиця FLOORS – «ПОВЕРХИ»					
NbFl	Номер поверху (числовий)	ПК	Так	Ціле число	{1,2,3,4,5}
NmFl	Номер поверху (літерний)	ВК	Так	Текст	До 10 символів
SqFl	Загальна площа кімнат на поверсі		Так	Дійсне число	Від 100,0 до 400,0
Таблиця ROOMS – «КІМНАТИ»					
NbRm	Номер кімнати	ПК	Так	Ціле число	Від N01 до N15, де N-номер поверху
CdTR	Код типу кімнати	ЗК до TPRM		Текст	До 2 символів
SqRm	Площа кімнати		Так	Дійсне число	Від 7,0 до 180,0
NbFl	Номер поверху (числовий)	ЗК до FLOORS	Так	Ціле число	{1,2,3,4,5}
Таблиця ITEMS – «ПРЕДМЕТИ»					
CdPr	Код предмета	ПК	Так	Ціле число	Від 1 і більше
NmPr	Назва предмета	ВК	Так	Текст	До 50 символів
CdTP	Код типу предмета	ЗК до TPIT	Так	Текст	До 2 символів
VgCs	Початкова вартість предмета		Так	Дійсне число	Більше 0.0

Таблиця TPRM – «ТИП КІМНАТ»					
CdTR	Код типу кімнати	ПК	Так	Текст	До 2 символів
NmTR	Назва типу кімнат	ВК	Так	Текст	До 50 символів
Таблиця TRIT – «ТИПИ ПРЕДМЕТІВ»					
CdTP	Код типу предмета	ПК	Так	Текст	До 2 символів
NmTP	Назва типу предмета	ВК	Так	Текст	До 50 символів
TrUs	Термін амортизації		Так	Дійсне число	Більше 1.0
Таблиця IT RM – «КІМНАТИ ПРЕДМЕТІ»					
NbRm	Номер кімнати	СПК, ЗК до ROOMS	Так	Ціле число	Від N01 до N15, де N-номер поверху
CdPr	Код предмета	СПК, ЗК до ITEMS	Так	Ціле число	Від 1 і більше
QtyP	Кількість предметів у кімнаті		Так	Ціле число	Більше 0
DtVs	Дата встановлення предметів		Так	Дата (рік, місяць)	01.01.2000 <Date <Поточної

Таким чином, послідовність робіт при створенні даталогічної інформаційної моделі на основі використання реляційної моделі даних може бути наступною:

- перетворення ІЛМ у реляційну модель даних (РМД);
- аналіз отриманої РМД на функціональну повноту (відповідність вимогам користувачів/замовника);
- побудова схем функціональних залежностей між атрибутами для кожної з таблиць РМД;
- встановлення рівня нормалізованості бази даних та подальша її нормалізація (якщо вона можлива);
- уточнення первинних, вторинних (потенційних) та зовнішніх ключів для нормалізованої РБД;

- уточнення обмежень цілісності бази даних;
- графічне представлення даталогічної моделі у вигляді таблиць із зазначенням їх атрибутів та обраних первинних і зовнішніх ключів, та зв'язків між таблицями;
- табличне представлення атрибутів таблиць та їх характеристик: назва, сутність (якщо потрібно), тип даних, ключовий чи не ключовий, тип ключа (для ключових атрибутів), обмеження або правила перевірки, обов'язковість значення, тощо.

Якщо структура даних, що обрана в ДЛМ, не може бути реалізована на рівні фізичної моделі, то коректують ДЛМ, що може привести до необхідності коректування ІЛМ, а можливо і КМ.

3.7. Фізична модель бази даних

Фізична модель визначає розміщення даних у зовнішній пам'яті. Вона ще називається внутрішньою моделлю системи і форма її представлення залежить від обраної СКБД. Якщо обрана така СКБД, яка підтримує реляційну модель даних, то треба таблиці разом з атрибутами і зв'язки між таблицями перенести в середовище СКБД з врахуванням вимог до відповідних об'єктів БД. Так, ідентифікатори (імена) таблиць і полів мають задовольняти вимогам СКБД, типи даних, розміри полів, обмеження теж мають бути приведені у відповідність до прийнятих у даній СКБД.

Далі визначаються стратегії індексування, а також взаємозв'язки між таблицями, первинні та зовнішні ключі на основі визначених у даталогічній моделі та врахуванням способів їх завдання в обраній СКБД.

На етапі фізичного проектування слід приділити особливу увагу забезпеченню цілісності БД. У СКБД цілісність даних забезпечується обмеженнями цілісності, тобто набором правил, що встановлюють допустимість даних та зв'язків між ними. Наводиться контрольний приклад заповненої БД з врахуванням встановлених обмежень цілісності.

Таким чином, послідовність робіт при створенні фізичної моделі на основі реляційної моделі даних може бути наступною:

- аналіз та вибір СКБД;
- розробка структури (фізичної моделі) бази даних засобами вибраної СКБД з урахуванням типів даних та обмежень цілісності атрибутів;
- розробка схеми взаємозв'язків бази даних засобами вибраної СКБД з урахуванням обмежень посилальної цілісності;
- розробка контрольного приклада заповнення бази даних.

Кінцевим результатом роботи є представлення таблиць та зв'язків між ними, що описані в даталогічній моделі даних, у середовищі обраної СКБД. При цьому мають бути визначені (запрограмовані мовою опису даних) обмеження цілісності як для атрибутів, так и посилальна цілісність.

Наскрізний приклад проектування бази даних ІСПП. Фізична модель бази даних

На етапі проектування бази даних на фізичному рівні треба, у першу чергу, визначитися з СКБД. Оскільки обсяг бази даних незначний, а підрозділи, що будуть її використовувати, розміщені у межах одного корпусу ВНЗ, то найбільш прийнятним рішенням буде вибір «настільної» СКБД Access та розміщення її на сервері, до якого будуть мати доступ підрозділи-користувачі. Питання розмежування доступу слід вирішити засобами операційної системи.

Створимо базу даних ITM_RMS та її таблиці згідно з даталогічною моделлю (табл. 7). Головне вікно бази даних наведено на рис. 37.

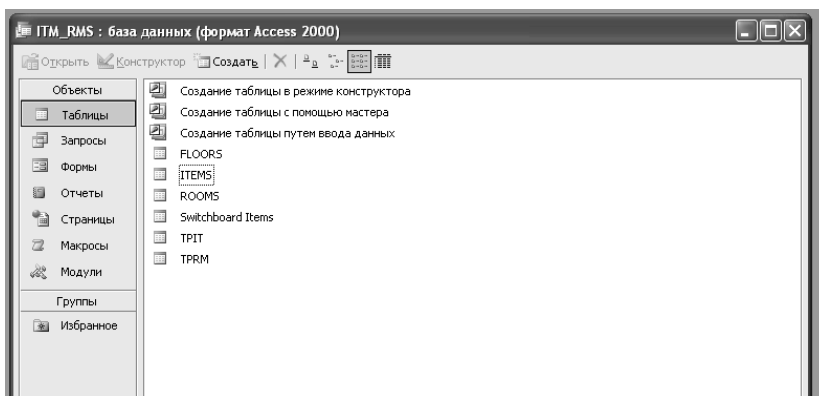


Рис. 37. Головне вікно бази даних: Таблиці

Засобами СКБД визначасмо ключі та інші властивості атрибутів у відповідності до табл. 7. На рис. 38-43 наведено вигляд таблиць та у режимі конструктора, при цьому показано властивості тих атрибутів, що потребують їх встановлення (визначення).

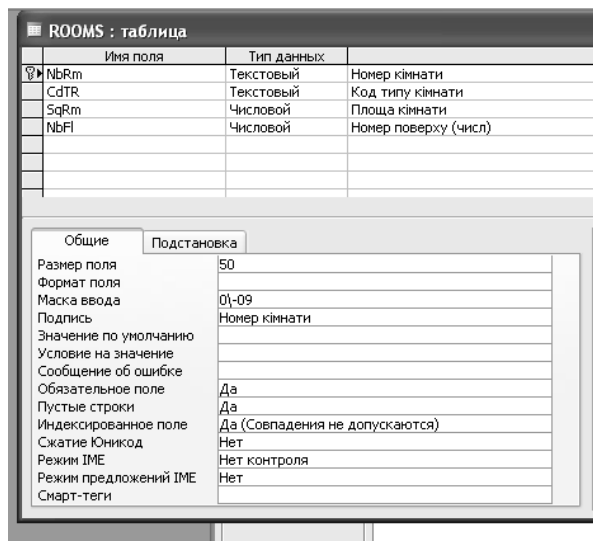


Рис. 38

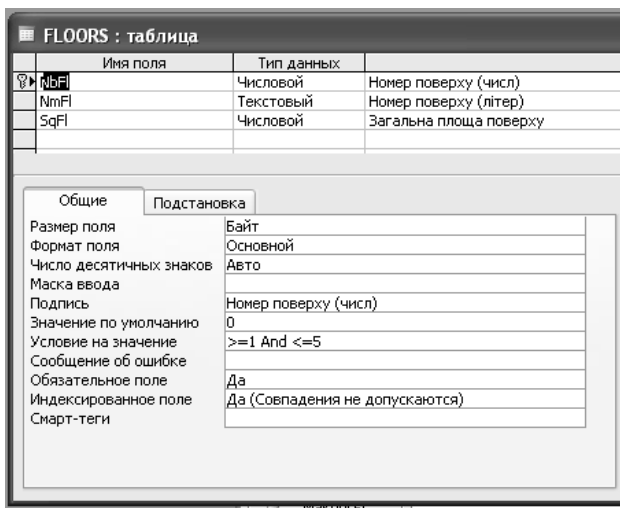


Рис. 39

TPRM : таблица

	Имя поля	Тип данных	
🔍	CdTR	Текстовый	Код типу кімнати
	NmTR	Текстовый	Назва типу кімнати

Общие Подстановка

Размер поля	2
Формат поля	
Маска ввода	
Подпись	Код типу кімнати
Значение по умолчанию	
Условие на значение	
Сообщение об ошибке	
Обязательное поле	Да
Пустые строки	Да
Индексированное поле	Да (Совпадения не допускаются)
Сжатие Юникод	Да
Режим IME	Нет контроля
Режим предложений IME	Нет
Смарт-теги	

Рис. 40

ITEMS : таблица

	Имя поля	Тип данных	
🔍	CdPr	Счетчик	Код предмету
	NmPr	Текстовый	Назва предмету
▶	CdTP	Числовой	Код типу предмету
	BgCs	Числовой	Початкова вартість предмету
	TrUs	Числовой	Термін амортизації

Общие Подстановка

Размер поля	Длинное целое
Формат поля	
Число десятичных знаков	Авто
Маска ввода	
Подпись	Код типу предмету
Значение по умолчанию	1
Условие на значение	≠0 And <15
Сообщение об ошибке	
Обязательное поле	Нет
Индексированное поле	Нет
Смарт-теги	

Рис. 41

TRIT : таблица

	Имя поля	Тип данных	
?	CdTR	Числовой	Код типу предмету
	NmTR	Текстовый	Назва типу предмету

Общие Подстановка

Размер поля	Длинное целое
Формат поля	
Число десятичных знаков	Авто
Маска ввода	
Подпись	Код типу предмету
Значение по умолчанию	0
Условие на значение	
Сообщение об ошибке	
Обязательное поле	Нет
Индексированное поле	Да (Совпадения не допускаются)
Смарт-теги	

Рис. 42

IT_RM : таблица

	Имя поля	Тип данных	
?	NbRm	Текстовый	Номер кімнати
?	CdPr	Числовой	Код предмету
	QtyP	Числовой	Кількість предметів у кімнаті
	DtVs	Дата/время	Дата встановлення предметів

Общие Подстановка

Размер поля	5
Формат поля	
Маска ввода	0 -09
Подпись	
Значение по умолчанию	
Условие на значение	
Сообщение об ошибке	
Обязательное поле	Да
Пустые строки	Да
Индексированное поле	Да (Допускаются совпадения)
Сжатие Юникод	Нет
Режим IME	Нет контроля
Режим предложений IME	Нет
Смарт-теги	

Рис. 43.

На рис. 44 представлено схему взаємозв'язків таблиць бази даних.

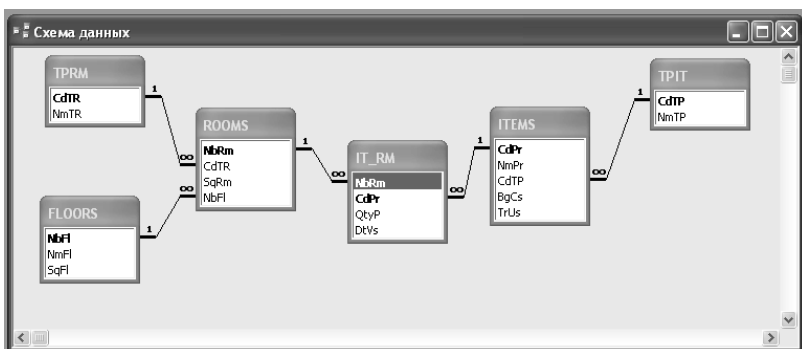


Рис. 44. Схема взаємозв'язків таблиць бази даних

Як видно на схемі, цілісність бази даних на рівні таблиць забезпечено наявністю первинних ключів у кожній таблиці атрибуту виділені жирним шрифтом), а на рівні зв'язків – посилальною цілісністю (на схемі вона відображена символами '1' та '∞' на кінцях зв'язків).

На рис. 45-50 представлено контрольний приклад вмісту таблиць бази даних.

Кінець фрагмента наскрізного прикладу (фізична модель бази даних)

ROOMS			
Номер кімнати	Код типу кімнати	Площа кімнати	Номер поверху (числ)
1-1	лабораторія	35	2
1-2	кабінет	25	1
1-3	кабінет	26	1
1-4	лабораторія	36	1
1-5	лекційна аудиторія	80	1
1-6	кабінет	50	1
1-7	семінарська аудиторія	40	4
1-8	лабораторія	38	1
2-1	семінарська аудиторія	28	2
2-2	лекційна аудиторія	55	2

ROOMS			
Номер кімнати	Код типу кімнати	Площа кімнати	Номер поверху (числ)
2-3	лекційна аудиторія	60	2
2-4	комп'ютерний клас	44	2
2-5	лабораторія	55	2
2-6	комп'ютерний клас	50	2
3-1	лабораторія	22	3
3-2	кабінет	30	3
3-3	лабораторія	34	3
3-4	кабінет	40	3
3-5	семінарська аудиторія	54	3
3-6	кабінет	36	3
3-7	лабораторія	24	3
3-8	комп'ютерний клас	50	4
3-9	кабінет	50	3
4-1	комп'ютерний клас	32	4
4-2	лабораторія	23	4
4-3	семінарська аудиторія	33	4
4-5	кабінет	46	4
4-6	кабінет	35	4
4-7	комп'ютерний клас	44	4

Рис. 45

FLOORS		
Номер поверху (числ)	Номер поверху (літер)	Загальна площа поверху
1	перший	295
2	другий	299
3	третій	290
4	четвертий	303
5	п'ятий	101

Рис. 46

TPRM

Код типу кімнати	Назва типу кімнати
1	лекційна аудиторія
2	лабораторія
3	кабінет
4	комп'ютерний клас
5	семінарська аудиторія

Рис. 47

ІТЕМС				
Код предмета	Назва предмета	Код типу предмета	Початкова вартість предмета	Термін амортизації
1	Стіл аудиторний на два місця	1	50,4	5
2	Парта	1	40,56	5
3	Пристрій для нагріву води	2	150	10
4	Комп'ютер Pentium-4	3	3000	3
5	Стіл двухтумбовий	4	320	10
6	Дошки	1	300	10

Рис. 48

ТРІТ	
Код типу предмета	Назва типу предмета
1	Столи, стільці, дошки, шафи для аудиторій
2	Лабораторне устаткування
3	Комп'ютерна техніка
4	Офісні меблі

Рис. 49

IT_RM			
Номер кімнати	Код предмета	Кількість предметів у кімнаті	Дата встановлення предметів
1-1	1	50	01.09.2002
1-1	2	10	20.10.2002
1-1	3	10	12.10.1998
1-1	4	10	01.01.2004
1-1	5	3	12.10.2003
1-2	2	16	01.02.2003
1-2	3	15	01.06.2001
1-2	4	1	01.10.2002
2-1	1	5	15.09.2003
2-1	2	30	10.03.2002
2-2	3	20	01.01.2003
2-2	4	5	20.04.2003
3-1	1	8	01.07.2002
3-1	2	40	01.01.2003
3-2	1	60	01.01.2002
4-1	2	55	25.09.2003
4-1	3	7	01.11.2001
4-1	4	5	30.05.2000

Рис. 50

3.8. Опис реалізації проекту інформаційної системи на основі створеної бази даних

Опис реалізації проекту інформаційної системи включає, крім обґрунтування обраної СКБД, опис реалізації об'єктів бази даних, зокрема (якщо обрано СКБД Access), запити, форми, звіти, макроси й модулі. Якщо програмний код займає великий обсяг (більше 2 сторінок), то він виноситься в додатки. В додатки може бути винесені й результати виконання запитів і звітів, якщо вони займають більше сторінки. Перелік реалізованих об'єктів бази даних має витікати із концептуальної моделі, зокрема із діаграм потоків даних (DFD). Із останньої або виходячи із зручності користування має бути

запропонована структура меню, яке реалізується засобами обраної СКБД. Зокрема, якщо обрано СКБД Access, то воно може бути реалізовано як головна кнопкова форма.

Таким чином, послідовність робіт при реалізації функціональної частини проекту може бути наступною:

- створення меню користувача;
- розробка форм для введення інформації та наповнення БД;
- розробка запитів засобами СКБД, у тому числі засобами мови SQL;
- розробка форм для користувачів;
- розробка звітів для користувачів;
- розробка макросів: гарячих клавіш та макросів подій до кнопок;
- розробка програмних модулів до кнопок.

Примітка. Якщо студентом для реалізації БД та інформаційної системи на її основі обрано СКБД, що підтримує архітектуру «Клієнт-сервер», (робота підвищеної складності), то у цьому розділі має бути наведено:

1) обґрунтування вибору технології реалізації бази даних багаторівневої архітектури «клієнт-сервер»;

2) опис реалізації клієнтських застосувань:

- опис призначення та функцій застосування;
- опис інтерфейсу розробленої системи, функціональне призначення основних опцій меню;
- особливості розробки клієнтських застосувань у вибраній технології реалізації бази даних багаторівневої архітектури «клієнт-сервер»;
- графічні зображення екранних форм;
- генеровані звіти;

3) опис реалізації сервера застосувань:

- перелік використаних компонентів;
- графічне зображення вікна сервера застосувань з усіма компонентами;
- опис метаданих, що зберігаються на сервері СКБД.
- script-файл формування бази даних з обов'язковим використанням доменів для опису однотипних стовпців таблиць БД, обмежень цілісності, значеннями за замовчуванням;
- тексти збережених процедур;
- тексти тригерів;
- тексти представлень;

- тексти правил;
- тексти SQL-запитів.

**Фрагмент наскрізного прикладу проектування бази даних ІСПП.
Опис реалізації функціональної частини з боку клієнта**

Розробку функціональної частини інформаційної системи «Приміщення і предмети» в даному прикладі обмежимо розробкою АРМ робітника господарчого відділу (АРМ ГВ). За основу проекту меню візьмемо ієрархію потоків даних, що наведена вище в табл. 5, але з врахуванням виділення інформаційних об'єктів бази даних «Приміщення» і «Предмети». Ієрархія пунктів меню наведена в табл. 8.

Таблиця 8

Структура і зміст пунктів меню

№	Назва пункту меню
1	Приміщення
1.1	Ведення інформації про приміщення
1.1.1	Додати інформацію про приміщення
1.1.2	Оновити інформацію про приміщення
1.1.3	Форма для видалення приміщення із аудиторного фонду з перевіркою, чи залишкова площа на поверсі не стане меншою за 100 кв.м. Якщо «так», то відмовити у видаленні
1.1.4	Повернення на рівень вище
1.2	Отримання інформації про приміщення
1.2.1	Запити з інформацією про приміщення
1.2.1.1	Приміщення, в яких встановлено лабораторне устаткування
1.2.1.2	Приміщення, які не є лекційними аудиторіями і в яких не встановлено жодного комп'ютера
1.2.1.3	На якому поверху більш за все лабораторій і скільки їх?
1.2.1.4	Предмети, термін амортизації яких закінчується в поточному році?
2.2.1.5	Повернення на рівень вище
1.2.2	Форми з інформацією про приміщення
1.2.2.1	Вкладена форма з кнопкою: 1-й рівень форми: кімнати та їх характеристики; 2-й рівень форми: предмети та їх характеристики; кнопка: виділяє предмети, що встановлені у поточному році.

1.2.2.2	Вкладена форма з кнопками: 1-й рівень форми: поверхи та їх характеристики; 2-й рівень форми: кімнати та їх характеристики; 1-ша кнопка: відкриває створений запит п. 1.2.1.3; 2-га кнопка: підраховує кількість комп'ютерів на поверсі.
1.2.2.3	Графік: площа приміщень по поверхах
1.2.2.4	Повернення на рівень вище
1.2.3	Звіти з інформацією про приміщення
1.2.3.1	Приміщення та предмети в них, що мають найраніший термін встановлення
1.2.3.2	Поверхи та залишкова вартість встановлених предметів
1.2.3.3	Повернення на рівень вище
1.3	Повернення до головного меню
2	Предмети
2.1	Ведення інформації про предмети
2.1.1	Форму для уведення інформації про встановлення нових предметів у кімнату з кнопкою, яка перевіряє, чи не порушено правил: у лекційну аудиторію не можна встановлювати комп'ютери, а в лабораторії – дошки.
2.1.2	Оновити інформацію про предмети
2.1.3	Видалити інформації про предмети
2.1.4	Повернення на рівень вище
2.2	Отримання інформації про предмети
2.2.1	Запити з інформацією про предмети
2.2.1.1	Кількість комп'ютерів по поверхах
2.2.1.2	Кількість стільців на j-му поверсі
2.2.1.3	Які предмети є в аудиторії 1-2, такі, яких немає в аудиторії 2-2?
2.2.1.4	Отримати інформацію: які предмети на 3-му поверху мають початкову вартість, таку що більше хоча б одного предмета на 2-му поверху?
2.2.1.5	Повернення на рівень вище
2.2.2	Форми з інформацією про предмети
2.2.2.1	Вкладена форма з кнопками: 1-й рівень форми: типи предметів; 2-й рівень форми: предмети та їх характеристики; 1-ша кнопка: відкриває створений запит 2.2.1.1; 2-га кнопка: підраховує сумарну залишкову вартість предметів для даного типу предметів.

2.2.2.2	Вкладена форма з кнопкою: 1-й рівень форми: предмети та їх характеристики; 2-й рівень форми: кімнати, в яких встановлені ці предмети кнопка: підраховує дати закінчення термінів амортизації предметів по кімнатах
2.2.2.3	Графік: кількість парт по поверхах
2.2.1.4	Повернення на рівень вище
2.2.3	Звіти з інформацією про предмети
2.2.3.1	Предмети, термін амортизації яких вже закінчився, та їх місцезнаходження
2.2.3.2	Нарахування сум амортизації предметів по приміщеннях
2.2.3.3	Повернення на рівень вище
2.3	Повернення до головного меню
3	Вихід із системи

Меню користувача в середовищі Access зручно реалізувати за допомогою головної кнопкової форми (Switchboard). Така форма меню зручна і в користуванні. Нижче на рис. 51-54 показано екранне зображення сторінки головної кнопкової форми та декількох підлеглих сторінок.

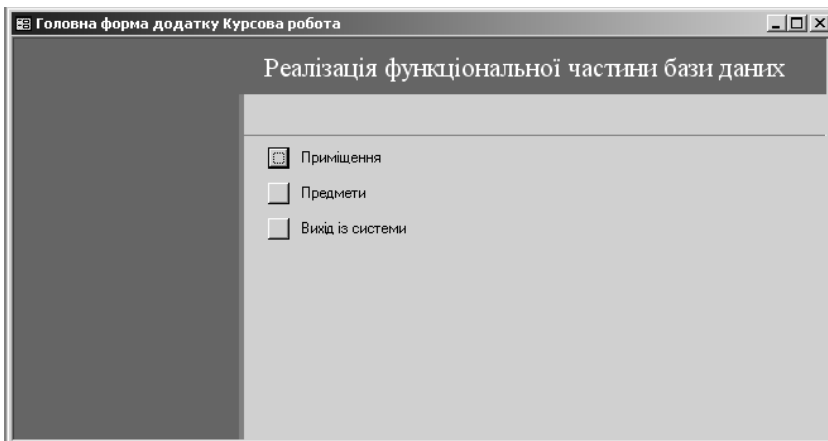


Рис. 51. Головна форма додатка

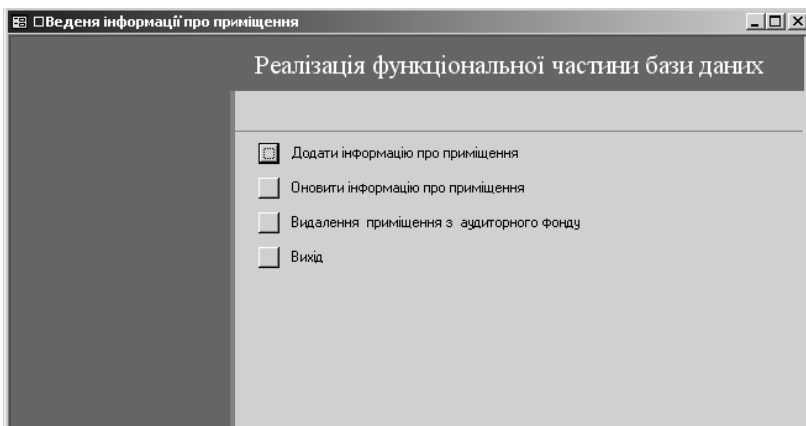


Рис. 52. Уведення інформації про приміщення

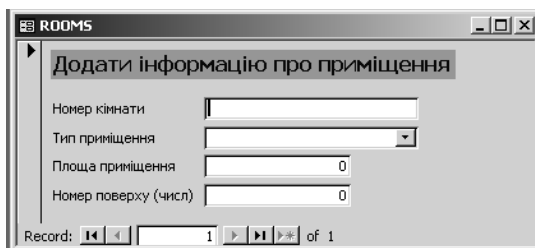


Рис. 53. Додавання інформації про приміщення

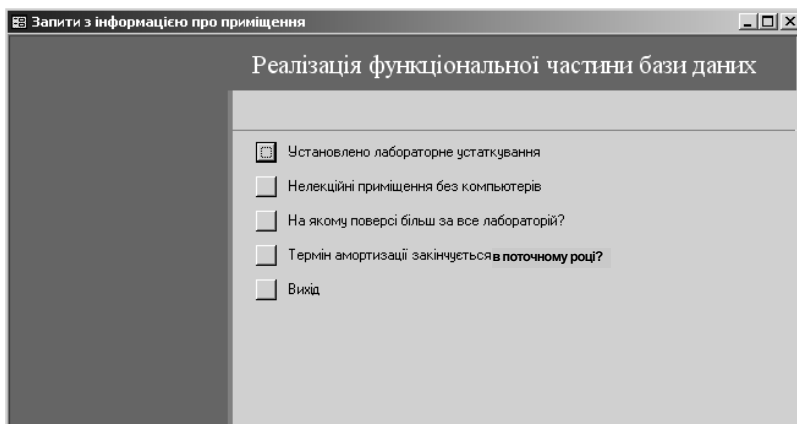


Рис. 54. Запити про інформацію щодо приміщень

Далі наведемо приклади реалізації запитів, форм, звітів, макросів і модулів. На рис. 55 наведено бланк запиту 1.2.1.2 таблиці 8 мовою QBE, а на рис. 56 – результат запиту.

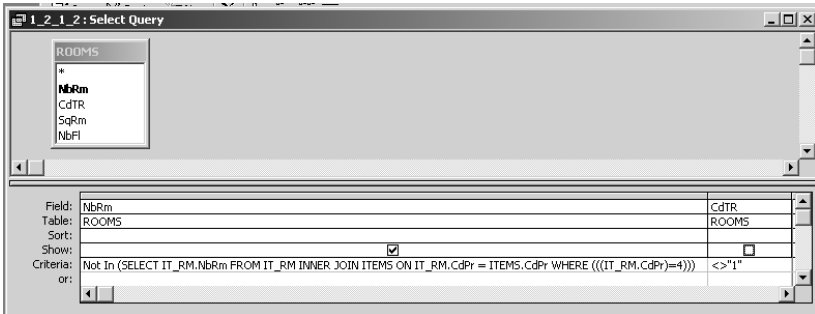


Рис. 55. Бланк запиту 1.2.1.2 мовою QBE

	Номер кімнати	Код типу кімнати
▶	2-1	лабораторія
	2-4	комп'ютерний клас
	2-5	лабораторія
	2-6	комп'ютерний клас
	3-1	лабораторія
	3-2	кабінет
	3-3	лабораторія
	3-4	кабінет
	3-5	семінарська аудиторія
	3-6	кабінет
	3-7	лабораторія
	3-8	комп'ютерний клас
	3-9	кабінет
	4-2	лабораторія
	4-3	семінарська аудиторія
	4-5	кабінет
	4-6	кабінет
	4-7	комп'ютерний клас
	1-2	кабінет
	1-3	кабінет
	1-4	лабораторія
	1-6	кабінет
	1-7	семінарська аудиторія
	1-8	лабораторія

Рис. 56. Результат виконання запиту

На рис. 57 наведено текст запиту 2.2.1.3 мовою SQL, а на рис. 58 – результат запиту

```

SELECT DISTINCT IT_RM.CdPr, ITEMS.NmPr
FROM IT_RM INNER JOIN ITEMS ON IT_RM.CdPr = ITEMS.CdPr
WHERE (((IT_RM.CdPr) Not In (SELECT IT_RM.CdPr FROM IT_RM
WHERE (((IT_RM.NbRm)="2-1")))) And (IT_RM.CdPr) In (SELECT
IT_RM.CdPr FROM IT_RM WHERE (((IT_RM.NbRm)="1-2"))));

```

Рис. 57. Бланк запиту 2.2.1.3 мовою SQL

	CdPr	Назва предмету
▶	6	Пристрій для нагріву води
	4	Комп'ютер Pentium-4

Рис. 58. Результат виконання запиту

На рис. 59 наведено екранне зображення форми п. 2.2.2.1 (табл. 8).

CdPr	NmPr	BgCs	TrUs
▶ 1	Стіл аудиторний на	50,4	5
2	Парта	40,56	5
6	Дошки	300	10
* nber)		0	0

Рис. 59. Екранне зображення форми п. 2.2.2.1 таблиці 8

На формі п.2.2.2.1 реалізовано дві кнопки. Дія другої кнопки «підрахунок сумарної залишкової вартості предметів для даного типу предметів реалізовано макросом, текст якого наведено нижче (рис. 60).

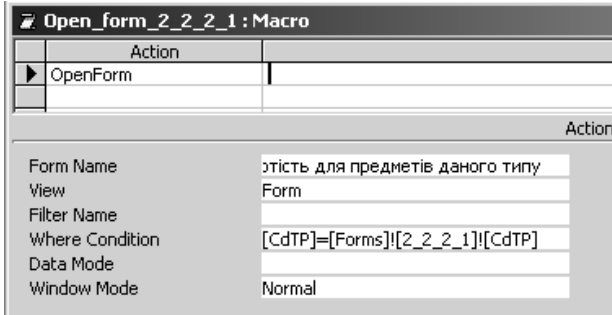


Рис. 60. Зображення макросу в режимі конструктора

На рис. 61 наведено екранне зображення форми п.1.2.2.2.

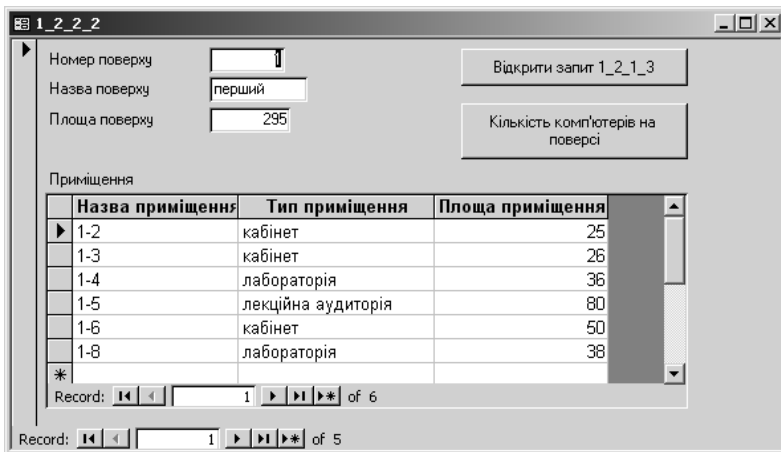


Рис. 61. Екранне зображення форми п.1.2.2.2

На формі п.1.2.2.2 реалізовано дві кнопки. Дія другої кнопки «підрахунок кількості комп'ютерів на поверсі» реалізовано модулем мовою VBA, текст якого наведено нижче.

Процедура відкриття форми, в якій виконується обчислення кількості комп'ютерів на поверсі.

```
Private Sub Command15_Click()
On Error GoTo Err_Command15_Click
Dim stDocName As String
Dim stLinkCriteria As String
stDocName = ChrW(1050) & ChrW(1110) & ChrW(1083)
& ChrW(1100) & ChrW(1082) & ChrW(1110) & ChrW(1089)
& ChrW(1090) & ChrW(1100) & ChrW(32) & ChrW(1082)
& ChrW(1086) & ChrW(1084) & ChrW(1087) & ChrW(39)
& ChrW(1102) & ChrW(1090) & ChrW(1077) & ChrW(1088)
& ChrW(1110) & ChrW(1074)
stLinkCriteria = "[Номер поверху]=" & Me![NbFl]
DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Command15_Click:
Exit Sub
```

На рис. 62 наведено звіт про п. 2.2.3.2.

Нарахування сум амортизації по приміщенням

<i>NbRm</i>	<i>NmPr</i>	<i>Амортизація</i>
<i>1-1</i>		
	Стіл двохтумбовий	64
	Пристрій для нагріву води	105
	Перта	24,32
	Комп'ютер Pentium-4	1000
	Стіл аудиторний на два місяця	30,23
	Сума амортизації	1223,55
<i>1-2</i>		
	Перта	16,21
	Пристрій для нагріву води	60
	Комп'ютер Pentium-4	3000
	Сума амортизації	3076,21
<i>2-1</i>		
	Стіл аудиторний на два місяця	20,15
	Перта	24,32

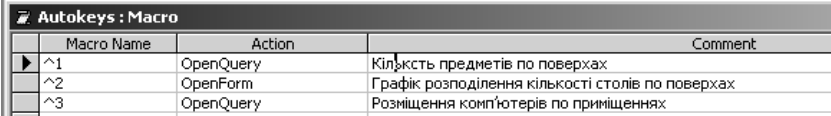
Рис. 62. Звіт для п. 2.2.3.2 (табл. 8)

Крім того, для зручності були запрограмовані гарячі клавіші:

Клавіші	Дія
^1	Кількість предметів по поверхах
^2	Графік розподілення кількості столів по поверхах
^3	Розміщення комп'ютерів по приміщеннях

Команди реалізації гарячих клавіш макросу Autokeys наведено нижче (рис. 63).

Кінець фрагменту наскрізного прикладу (Опис реалізації функціональної частини)



The screenshot shows a window titled "Autokeys : Macro" containing a table with three columns: "Macro Name", "Action", and "Comment". The table lists three macros: ^1 with action "OpenQuery" and comment "Кількість предметів по поверхах"; ^2 with action "OpenForm" and comment "Графік розподілення кількості столів по поверхах"; and ^3 with action "OpenQuery" and comment "Розміщення комп'ютерів по приміщеннях".

Macro Name	Action	Comment
^1	OpenQuery	Кількість предметів по поверхах
^2	OpenForm	Графік розподілення кількості столів по поверхах
^3	OpenQuery	Розміщення комп'ютерів по приміщеннях

Рис. 63. Реалізація гарячих клавіш за допомогою макросу Autokeys

3.9. Висновки

Наводяться висновки стосовно ефективності роботи системи та пропозиції щодо практичного впровадження розробленої БД.

3.10. Додатки

Додатки містять таблиці, форми документів, тексти програм, копії екранних форм тощо, які винесені із основної частини з-за їх великого обсягу або довідкового характеру.