

**Чернігівський національний педагогічний університет
імені Т.Г.Шевченка**

Горошко Ю.В.

**ТЕОРІЯ І МЕТОДИКА РОБОТИ
З БАЗАМИ ДАНИХ**

Рекомендовано до друку
вченою радою Чернігівського національного
педагогічного університету імені Т.Г.Шевченка
від 25 лютого 2015 р., протокол №7

Чернігів, 2015

УДК 378.016:004.65 (075.8)
ББК 397р30
Г 70

Горошко Ю.В.

Г 70 ТЕОРІЯ І МЕТОДИКА РОБОТИ З БАЗАМИ ДАНИХ

Чернігів, 2015, - 78 с.

В посібнику викладено теоретичні засади обробки баз даних, розглянуто інформаційні моделі, пов'язані з базами даних.

Наведено особливості використання вільно поширюваної СУБД LibreOffice Base до обробки баз даних. Запропоновано цикл лабораторних робіт.

Посібник призначений для студентів, що вивчають інформатику у педагогічному університеті.

Рецензенти:

Торубара Олексій Миколаєвич - доктор педагогічних наук, професор, декан технологічного факультету Чернігівського національного педагогічного університету імені Т.Г.Шевченка.

Покришень Дмитро Анатолійович – кандидат педагогічних наук, доцент, завідувач кафедри інформатики та інформаційно-комунікаційних технологій в освіті Чернігівського обласного інституту післядипломної педагогічної освіти імені К.Д.Ушинського.

Ткач Юлія Миколаївна – кандидат педагогічних наук, доцент, завідувач кафедри математичного моделювання та інформатики Чернігівського національного технологічного університету.

(с) Горошко Ю.В., 2015

(с) ЧНПУ імені Т.Г.Шевченка

ЗМІСТ

1. Вступ.....	4
2. Сучасні тенденції у використанні баз даних та моделей даних.....	5
3. Ієрархічна модель даних.....	9
4. Мережева модель даних.....	10
5. Реляційна модель даних. Модель даних SQL.....	11
6. Моделі концептуального подання даних.....	19
7. Побудова баз даних в СУБД LibreOffice Base.....	21
8. Лабораторна робота “Створення і редагування таблиць в СУБД LibreOffice Base”.....	35
9. Лабораторна робота “Реалізація запитів в СУБД LibreOffice Base”.....	52
10. Лабораторна робота «Створення звітів в LibreOffice Base».....	69
11. Список використаних джерел.....	78

1. Вступ

В Указі Президента України "Про Національну доктрину розвитку освіти" відмічено, що "Пріоритетом розвитку освіти є впровадження сучасних інформаційно-комунікаційних технологій, що забезпечує дальше удосконалення навчально-виховного процесу, доступність та ефективність освіти, підготовку молодого покоління до життєдіяльності в інформаційному суспільстві" [6].

На даний момент намітилася тенденція широкого використання баз даних практично в усіх сферах людської діяльності. Таке широке використання баз даних, заснованих на різних моделях даних, у найрізноманітніших галузях інформаційних технологій є основою для поглибленого вивчення питань, пов'язаних з базами даних, і насамперед питань щодо моделей даних, у курсі інформатики в педагогічному університеті.

Оволодіння ефективними прийомами опрацювання даних на основі їх формалізації і структуризації за допомогою інформаційного моделювання стає конче необхідним для спеціалістів найрізноманітніших галузей, а тим більше для майбутніх учителів інформатики.

Практична частина даного посібника орієнтована на використання СУБД LibreOffice Base. Такий вибір зумовлений тим, що дана СУБД є складовою частиною пакету офісних програм LibreOffice, який є кросплатформенним, а також вільно поширюваним. В той же час це потужний засіб обробки баз даних.

Посібник призначений для студентів фізико-математичних факультетів педагогічних університетів, але він може бути корисним і для студентів інших спеціальностей педагогічних університетів, що вивчають інформатику, а також для вчителів інформатики.

2. Сучасні тенденції у використанні баз даних та моделей даних

Як відомо, кількість всеможливих інформаційних матеріалів і повідомлень, що циркулюють в інформаційному просторі, зростає небаченими темпами. Відповідно виникає потреба у надійному зберіганні цих даних, а також у ефективному маніпулюванні ними. Всім цим потребам відповідають бази даних, оскільки для них накопичено багато напрацювань щодо підтримки цілісності даних, швидкого доступу до них, розділення прав доступу до даних між користувачами різного рівня.

Поряд з широким використанням баз даних у традиційних сферах, наприклад банківській, управлінській, можна спостерігати і застосування баз даних у тих галузях, де вони раніше не використовувалися.

Так, вже досить давно спостерігаються тенденції щодо переходу від зберігання даних у окремих файлах до зберігання їх у базах даних. Наведемо кілька прикладів. На початку розвитку мережі Інтернет та служб WWW дані в основному зберігалися в html-документах та окремих файлах інших типів (наприклад, графічних). Зараз такий підхід виправданий тільки для невеликих, любительських сайтів, де даних мало і оновлюються вони досить рідко. У всіх інших випадках дані сайту зберігаються у базі даних.

В операційній системі Windows більшість параметрів її роботи зберігаються у спеціальній базі даних, основаній на ієрархічній моделі – “Реєстр” (Register).

Іншим прикладом переходу до використання баз даних для більш зручного маніпулювання даними є сучасні підходи до роботи з музичними матеріалами. Раніше музичні програвачі (такі як дуже популярний свого часу WinAmp і багато його аналогів) орієнтувались в основному на програвання музичних файлів із папки, вказаної користувачем. Тобто користувач повинен був сам створити папку для зберігання музичних файлів, скопіювати у цю папку відповідні музичні файли, а потім вказати на музичному програвачеві цю папку для того, щоб почати програвати ці файли. Такий спосіб використання музичних даних був дуже незручним, наприклад, якщо користувач хотів відтворити всі музичні дані конкретного виконавця або

музичні дані у вибраному жанрі, то примітивний механізм списків програвання (playlists) не дуже зручний для цього. В сучасних музичних програвачах (наприклад, iTunes, Windows Media Player, Real Player, програвачі багатьох сучасних мобільних телефонів та смартфонів та інші) для зберігання музичних файлів створюються спеціальні бази даних, використання яких дає змогу легко сортувати та групувати музичні дані відповідно до спеціальних даних (музичних тегів), таких як прізвище композитора, прізвище виконавця, альбом, жанр і інші, що записані у музичних файлах.

Схожа тенденція спостерігається і щодо зберігання та використання фотографій та інших графічних файлів. При роботі сучасних фотопереглядачів, таких як, наприклад, iPhoto, Picasa фотографії зберігаються у базах даних, що надає можливість легко відібрати фотографії за певну дату, або зроблені за допомогою певної моделі фотокамери, або ті, на яких є зображення певної людини, а якщо у фотокамері передбачено можливість геотегінгу (запису у файл фотографії даних про географічні координати зйомки), то і відібрати фотографії, зроблені у певному районі земної кулі.

Найбільш масштабним прикладом переходу до зберігання і маніпулювання даними за допомогою баз даних є надання властивостей бази даних файлової системі. Першим кроком у цьому напрямі була BeOS – операційна система для настільних персональних комп'ютерів, розроблена компанією Be. Важливою особливістю цієї операційної системи була 64-бітна файлова система BeFS з можливістю журналювання. Особливістю цієї файлової системи є і покращений підхід до ідентифікації файлів даних. Крім традиційної прив'язки розширення файлу до програми, за якою опрацьовується файл, в BeOS використовується ідентифікація, заснована на так званих MIME-типах, достатньо стандартизований спосіб ідентифікації файлів і зв'язку їх з програмами. Використання цих розширених файлових атрибутів, що індексуються, наближають функціональність цієї файлової

системи до реляційних баз даних, тобто баз даних, основаних на реляційній моделі [9], [5].

У сучасних операційних системах, таких як MAC OS X, Windows XP SP3 і старших вміст дисків автоматично індексується у фоновому режимі. В процесі цієї індексації у спеціальну базу даних заносяться як імена файлів, так і зміст більшості з них. Це дозволяє проводити миттєвий пошук за іменами папок та файлів, їх змістом, за поштою, адресною книгою, календарем тощо. У операційній системі MAC OS X результат пошуку можна, наприклад, розмістити у “розумній” папці (Smart Folder), і вона буде автоматично оновлюватися при додаванні або вилученні будь-яких даних на комп’ютері. “Розумна” папка містить файли, що згруповані за певним критерієм, а не ті, що фізично знаходяться поруч. Таким чином, один і той самий файл може бути зафіксований в кількох розумних папках і, крім того, фізично знаходитися в іншому місці. Отже зникає потреба дублювати, переміщувати та оновлювати файли, це буде зроблено автоматично [10].

У корпорації Microsoft триває розробка файлової системи WinFS.

Суттю WinFS є так звана модель структурованих даних, тобто механізм, за яким постійно опрацьовуються (адмініструються і структуруються) певні сутності (у дослівному перекладі «предмети» – items). В Items використовуються описові елементи, що виходять за поняття файлу. Слід розуміти, що ці описові елементи не зберігаються у файлі, а повністю належать і управляються WinFS. Отже, фізична структура файлів на рівні файлової системи NTFS не зазнає яких-небудь змін. У даній схемі як сутність можна реєструвати не лише файли, але і, наприклад, контакти, улюблені посилання в Інтернеті, листи, дати, бренди, дані про виробників тощо, тобто будь-які атрибути, встановлені за замовчуванням або призначені користувачем.

З точки зору користувачів, використання items знімає необхідність у визначенні фізичного місця розташування файлів. Замість цього у Windows дані організуються, залежно від їх вмісту, у віртуальні папки. При пошуку даних призначений для користувача критерій типу «Всі фотографії з

відпусток за останні два роки» (атрибути «тип файлу», «звідки» і «за який період часу») тепер замінює відомості про формат файлу, автора і розташування [11].

Таким чином, робота користувача з файловою системою наближається до роботи з базою даних.

Таке широке використання баз даних, заснованих на різних моделях даних, у найрізноманітніших галузях інформаційних технологій є основою для поглибленого вивчення питань, пов'язаних з базами даних, і насамперед питань щодо моделей даних, у курсі інформатики в педагогічному університеті.

Розглянемо питання, пов'язані з моделюванням даних в базах даних.

Модель даних – це деяка абстракція, в якій знаходять своє відображення найбільш важливі аспекти об'єктів визначеної предметної галузі, а другорядні – ігноруються. Модель даних можна розглядати як деяку цільову модель предметної галузі. У моделі даних розрізняють три головні складові:

- 1) структурна частина, за якою визначаються правила породження допустимих для даної системи управління базами даних (СУБД) видів структур даних;
- 2) управляюча частина, за якою визначаються можливі операції над такими структурами;
- 3) класи обмежень цілісності даних, які можуть бути реалізовані засобами цієї системи [1].

В структурній частині визначається, чим логічно є база даних, через класи обмежень цілісності визначаються засоби опису коректних станів бази даних, а через управляючу частину визначаються способи переходу між станами бази даних і способи отримання даних із бази даних [4]. Управляюча частина містить специфікацію однієї або кількох мов, призначених для створення запитів до бази даних.

Вперше поняття моделі даних було введено Едгаром Коддом для опису реляційного підходу до організації бази даних, але найбільш розповсюджена трактовка цього поняття належить Кристоферу Дейту [2].

Кожна СУБД будується на основі певної моделі даних, наприклад, в основі ієрархічних СУБД лежить ієрархічна модель даних, в основі мережевих СУБД – мережева модель даних, в основі реляційних СУБД – реляційна модель даних. Розглянемо більш детально ці моделі даних.

3. Ієрархічна модель даних

Ієрархічна модель даних подається як дерево, у вузлах якого розміщено дані, а зв'язки між даними подано гілками дерева. Вузли дерева розподілено за рівнями. Кожен вузол нижчого рівня зв'язаний тільки з одним вузлом вищого. Дерево має тільки одну вершину (яку називають коренем), не підпорядковану ніяким іншим вершинам. Кожен вузол має своє ім'я (ідентифікатор). Вузли дерева є записами, що складаються з певної кількості полів, кожне з яких характеризується іменем та типом даних, що там зберігаються. У дереві є записи різних типів, що мають різні назви і складаються з різних полів.

Ієрархічна база даних може складатися з кількох дерев, у найпростішому випадку з одного дерева.

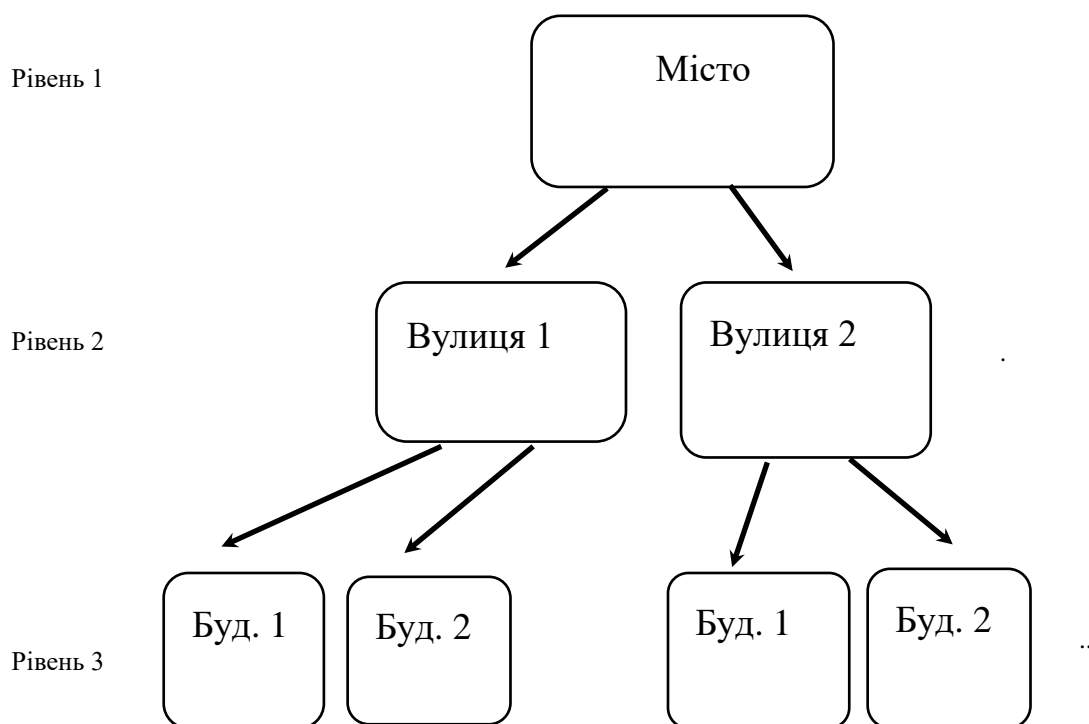


Рис. 1. Ієрархічна модель для подання даних про жителів міста.

Наприклад, застосуємо ієрархічну модель для подання даних про жителів міста (модель схематична, для кожного типу записів набір полів не визначено) (Рис. 1):

Для ієрархічної бази даних визначається порядок обходу дерева: згори-вниз, зліва-направо.

На основі дерева (дерев) моделі будують екземпляри дерев з конкретними даними.

Прикладами маніпуляції у ієрархічній базі даних можуть бути:

- 1) знайти вказаний екземпляр;
- 2) перейти від одного екземпляра до іншого;
- 3) перейти від екземпляра до його нащадка (у наведеному прикладі перейти від вулиці до першого будинку на цій вулиці);
- 4) перейти від одного запису до іншого запису в порядку обходу ієрархії (наприклад, від будинку 1 до будинку 2);
- 5) вставити новий запис у вказану позицію;
- 6) вилучити поточний запис.

В ієрархічній моделі даних автоматично підтримується цілісність посилань між предками та нащадками за основним правилом: ніякий нащадок не може існувати без свого батька.

4. Мережева модель даних

Мережева модель щодо організації даних є розширенням ієрархічної моделі. Вона має ті самі основні складові: вузол, рівень, зв'язки, але якщо в ієрархічній моделі нащадок повинен мати тільки одного батька, в мережевій моделі у нащадка може бути довільна кількість батьків, тобто можуть існувати зв'язки між вузлами будь-яких рівнів. У вузлах розміщено записи, що складаються з полів.

Мережева база даних складається з набору записів і набору зв'язків між цими записами. Зв'язок визначається для батька і нащадка і складається з одного екземпляра типу запису батька і впорядкованого набору екземплярів типу запису нащадка. Для прикладу застосуємо мережеву модель для

зображення зв'язків викладачів ВНЗ з групами, в яких вони працюють (Рис. 2):

Наведемо деякі основні типи операцій маніпулювання даними у мережевій базі даних:

- знайти конкретний запис у наборі однотипних записів (наприклад, викладача з прізвищем Петренко);
- перейти від батька до першого нащадка за певним зв'язком (наприклад до першої групи, у якій працює Петренко);
- перейти до наступного нащадка за певним зв'язком (наприклад до наступної групи, у якій працює Петренко);
- створити новий запис;
- модифікувати існуючий запис;
- включити у зв'язок;
- виключити із зв'язку;
- переставити в інший зв'язок.

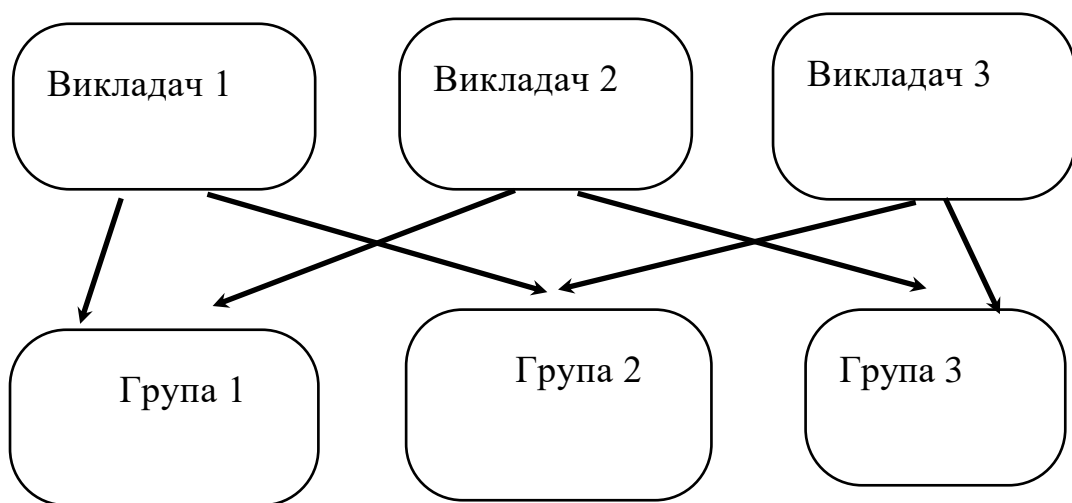


Рис. 2. Мережева модель для зображення зв'язків викладачів ВНЗ з групами, в яких вони працюють.

5. Реляційна модель даних. Модель даних SQL

Основні ідеї реляційної моделі були запропоновані Едгаром Коддом у 1969 р. Подальший розвиток реляційної моделі пов'язаний з фірмою IBM. Там була розроблена мова маніпулювання даними SQL (Structured Query

Language – мова структурованих запитів), яка базується на реляційній моделі даних. Вдосконаленням реляційної моделі у 90-х роках 20-го століття займалися Кристофен Дейт та Х'ю Дарвен [3].

При розробці реляційної моделі використовувалася математична теорія відношень. Відношенням R над множинами C_1, C_2, \dots, C_k називають підмножину декартового добутку цих множин. Множини C_1, C_2, \dots, C_k називають доменами. Кортеж – це елемент згаданого декартового добутку, тобто набір з k значень, взятих з відповідних доменів. Оскільки термін „відношення” походить від англійського „relation”, ця модель і має назву “реляційна”.

Якщо подати відношення таблицею, то кортежі будуть її рядками. Стовпці такої таблиці називають атрибутами. Кожен атрибут містить значення з відповідного домену. Відношення і атрибути повинні бути іменованими.

Терміни, запропоновані Коддом, широко використовуються в теоретичних дослідженнях, пов'язаних з реляційною моделлю даних, але при розробці реляційних СУБД використовується не реляційна модель Кодда у “чистому” вигляді, а модель даних SQL, яка базується на реляційній моделі, але не тотожна з нею. SQL-база даних подається як набір таблиць, кожна з яких в будь-який момент часу містить певну мультимножину (множину, що допускає повторення елементів) рядків, що відповідають заголовку таблиці. В цьому полягає перша відмінність моделі даних SQL від реляційної моделі даних. Другою суттєвою відмінністю є те, що для таблиці підтримується порядок стовпців, що відповідає порядку їх визначення. Таким чином, таблиця в SQL-моделі – це не зовсім відношення з реляційної моделі [7]. Оскільки вивчення СУБД у педагогічному ВНЗ має яскраво виражену прикладну спрямованість, доцільно реляційну модель подавати оглядово, а більш детально вивчати модель даних SQL, співставивши два набори термінів за допомогою наступної таблиці:

Співставлення термінів реляційної моделі та SQL-моделі.

Терміни за Коддом	Терміни СУБД
Відношення	Таблиця
Кортеж	Запис
Атрибут	Поле
Домен	Тип значень поля

Надалі будемо користуватися термінологією, що застосовується в моделі даних SQL, і відповідно трактувати термін “реляційний”, як це і робиться в документації до реальних СУБД.

Таким чином, база даних в моделі даних SQL – це сукупність іменованих таблиць. Кожен рядок таблиці називається записом і містить дані про один об’єкт предметної галузі, наприклад, кожен запис таблиці “Студент” містить персональні дані про одного студента. Кожен запис складається з одного і того самого набору полів, що є характеристиками запису, наприклад у таблиці “Студент” полями можуть бути номер залікової книжки, прізвище, дата народження тощо. Кожне поле характеризується ім’ям та типом даних. Ім’я поля зображується у заголовку відповідного стовпця таблиці.

Кожна таблиця повинна мати первинний ключ (Primary Key) – поле, або сукупність полів таблиці (мінімальну), що однозначно характеризують кожен запис таблиці, тобто всі значення ключа є гарантовано різними. Під мінімальністю первинного ключа треба розуміти наступне: якщо з множини полів первинного ключа вилучити хоча б одне поле, то у таблиці можуть з’явитися однакові значення ключа.

У таблиці може бути кілька варіантів вибору первинного ключа, всі ці можливі первинні ключі називають потенційними ключами (Candidate Key). Наприклад, у таблиці студентів потенційними ключами є “номер студентського квитка” та “прізвище, ім’я, по-батькові”, з них первинним

ключем краще вибрати “номер студентського квитка”, як такий, що містить більш короткі значення.

Взагалі, всі дані у реляційній базі даних можна тримати в одній таблиці, але це дуже неефективно внаслідок значного дублювання даних. Наприклад, якщо тримати в одній таблиці як персональні дані студента, так і дані про отримані ним упродовж навчання оцінки, то для кожної нової оцінки потрібно буде продублювати персональні дані студента. Тому логічно для зберігання таких даних створити дві таблиці: одну з персональними даними студентів, іншу – з даними про отримані ними оцінки.

Для встановлення зв'язку між такими таблицями в реляційній моделі визначено поняття зовнішнього ключа. Зовнішній ключ (Foreign Key) – це сукупність полів таблиці, значення яких є одночасно і значеннями первинного або потенційного ключа іншої таблиці. Якщо таблиці “Студенти” та “Оцінки” мають наступну структуру (знак “+” вказує, що поле входить у ключ):

Таблиця 2.

Таблиця "Студенти"

Ключ	Поле
+	Номер студентського квитка
	Прізвище ім'я по-батькові
	Адреса
	Дата народження
	Група

Таблиця 3.

Таблиця "Оцінки"

Ключ	Поле
+	Номер студентського квитка

Ключ	Поле
+	Предмет
+	Дата іспиту
	Оцінка

то поле “Номер студентського квитка” у таблиці “Оцінки” є зовнішнім ключем для зв’язування з таблицею “Студенти”.

Поле “Група” у таблиці “Студенти” є зовнішнім ключем для зв’язування з таблицею “Групи”, що може мати наступну структуру:

Таблиця 4.

Таблиця "Групи"

Ключ	Поле
+	Група
	Спеціальність
	Куратор

Підтримка цілісності в реляційній моделі містить такі складові:

- 1) структурна цілісність;
- 2) посилальна цілісність;
- 3) обмеження на реальні дані.

Поняття структурної цілісності перш за все пов’язане з первинним ключем, а саме з вимогою його обов’язкової наявності в кожній таблиці. Як вже говорилося вище, це призводить до відсутності дублікатів первинного ключа в таблиці. Наприклад, якщо в таблиці “Студенти” первинним ключем є “Номер студентського квитка”, стає неможливим ввести один і той самий номер квитка для різних студентів.

Посилальна цілісність вже стосується пов’язаних таблиць, тому її ще називають обмеженням зовнішнього ключа. Суть посилальної цілісності полягає в наступному:

1) в підлеглий таблиці немає записів, для яких не існує відповідних записів в основній таблиці;

2) кожен запис в підлеглий таблиці має тільки один батьківський запис в основній таблиці.

Наприклад, посилальна цілісність для таблиць “Студенти” та “Оцінки” полягає в тому, що в таблиці “Оцінки” не може бути записів з таким номером студентського квитка, якого немає в таблиці “Студенти”.

Перед поясненням посилальної цілісності доцільно запропонувати студентам зміст таблиць “Студенти” та “Оцінки” з порушенням посилальної цілісності, наприклад такі:

Таблиця 5.

Таблиця "Студенти"

Номер квитка	ПІБ	Адреса	Дата народження	Група
1	Іваненко Іван Іванович	вул. Шевченка,13, кв. 1	01.01.1990	11
2	Петренко Петро Петрович	вул. Щорса,14, кв. 12	02.02.1990	11

Таблиця 6.

Оцінки

Номер квитка	Предмет	Дата іспиту	Оцінка
1	Інформатика	14.01.2009	Добре
2	Інформатика	14.01.2009	Задовільно
3	Інформатика	14.01.2009	Добре

і запропонувати їм визначити, чи є помилка в даних, а якщо є, запропонувати вказати її.

Обмеження на реальні дані вимагають, щоб значення належали певному діапазону значень або задовольняли певне арифметичне співвідношення між значеннями констант та інших полів. Наприклад, значення поля “Дата

народження” гарантовано не може відноситися до 19-го століття, значення поля “Група” на даному факультеті повинне бути у межах від 11 до 69 тощо.

До підтримки цілісності також можна віднести правила, за якими повинні вилучатися та оновлюватися дані у зв’язаних таблицях.

Для вилучення даних можливі такі варіанти:

1. **RESTRICT:** заборонено вилучати запис із батьківської таблиці, якщо є його нащадки в підлеглий таблиці;
2. **CASCADE:** при вилученні запису в батьківській таблиці вилучаються всі його нащадки в підлеглий таблиці;
3. **SET NULL:** при вилученні запису з батьківської таблиці у всіх записах-нащадках підлеглої таблиці зовнішнім ключам надається значення NULL (невизначене значення);
4. **SET DEFAULT:** аналогічно до попереднього, але замість невизначеного значення відбувається надання значення, встановленого за замовчуванням.

Для оновлення даних можливі такі варіанти:

1. **RESTRICT:** не можна змінювати первинний ключ у запису батьківської таблиці, якщо в підлеглий таблиці цей запис має нащадків;
2. **CASCADE:** при зміні первинного ключа в запису батьківської таблиці автоматично така ж зміна відбувається у всіх записах-нащадках підлеглої таблиці;
3. **SET NULL:** при зміні первинного ключа в запису батьківської таблиці автоматично у всіх записах-нащадках підлеглої таблиці зовнішній ключ замінюється значенням NULL;
4. **SET DEFAULT:** при зміні первинного ключа в запису батьківської таблиці автоматично у всіх записах-нащадках підлеглої таблиці зовнішній ключ замінюється значенням за замовчуванням.

Щодо маніпулювання реляційними даними, то оскільки реляційні дані являють собою множини, до них застосовні звичайні теоретико-множинні операції: об’єднання, переріз, віднімання, декартовий добуток.

Кодд розширив цей стандартний набір операцій і запропонував у якості засобу для маніпулювання реляційними базами даних спеціальний набір операцій, які було названо алгеброю Кодда. В алгебрі Кодда десять операцій: об'єднання, перетин, віднімання, декартовий добуток, перейменування, проекція, обмеження, об'єднання за умовою, ділення і надання значення [3].

При об'єднанні двох таблиць однакової структури створюється таблиця, що містить всі записи, які входять хоча б в одну з таблиць.

Результатом перетину двох таблиць з однаковою структурою є таблиця, що містить всі записи, які входять до обох таблиць.

Різницею двох таблиць з однаковою структурою є таблиця, яка містить ті записи першої таблиці, яких немає в другій.

Декартовим добутком двох таблиць, що не мають однакових полів, є таблиця, записи якої утворюються шляхом об'єднання записів першої та другої таблиць.

Операція перейменування призначена для зміни імен полів, це дає змогу виконувати попередні операції над таблицями, у яких не співпадають імена відповідних полів, але співпадають їх типи, а також виконувати операцію декартового добутку над таблицями, що мають поля з однаковими іменами.

Результатом обмеження таблиці за певною умовою є таблиця, що містить тільки ті записи, які задовольняють цю умову.

Результатом проекції таблиці на вказану підмножину її полів є таблиця, що містить тільки ті поля, що вказані у проекції.

При об'єднанні за умовою двох таблиць створюється таблиця, записи якої утворюються шляхом об'єднання записів першої і другої таблиць і задовольняють цю умову.

Операція надання значення призначена зберегти результат обчислення реляційного виразу в існуючій таблиці бази даних.

В реальних реляційних СУБД використання реляційної алгебри для маніпулювання реляційними даними практично не застосовується, а широкого застосування набули мови запитів, що базуються на моделі SQL. Реляційне числення ґрунтується на використанні предикатів, що мають

задовольнити потрібні кортежі або домени. В мовах запитів, таких як SQL, QBE (Query By Example) реалізуються не тільки функції відповідної теорії, але і використовуються певні додаткові операції, наприклад, арифметичні операції, операції друкування тощо [7].

Реляційна модель і її нащадок модель SQL виявилися досить ефективними для подання даних для великої кількості задач з різних предметних галузей і тому покладені в основу функціонування більшості сучасних СУБД, які відповідно отримали загальну назву реляційних СУБД.

6. Моделі концептуального подання даних

Оволодіння студентами знаннями про реляційну (SQL) модель ще не є гарантією успішного розв'язування практичних задач, пов'язаних з базами даних, оскільки в таких задачах умова сформульована в термінах предметної галузі, а не реляційної моделі. Для розв'язування цієї проблеми необхідно познайомити студентів з моделями концептуального проектування баз даних. Розрізняють два головних підходи до моделювання даних при концептуальному проектуванні:

- 1) семантичні моделі;
- 2) об'єктні моделі [1].

В семантичних моделях головна увага приділяється структурі даних. Найбільш поширеною є модель "сутність – зв'язок" (ER-модель). Моделювання даних в цій моделі відображає логічну структуру даних.

В об'єктних моделях головна увага приділяється "поведінці" об'єктів даних і засобам маніпуляції даними. Головне поняття таких моделей – об'єкт, що характеризується станом і „поведінкою”.

Розглянемо більш детально ER-модель. Вперше поняття ER-моделі запровадив П.Чен. Підхід П.Чена дозволив перевести концептуальне проектування в практичну площину проектування бази даних [1]. В ER-моделі оперують сутностями, атрибутами та зв'язками.

Сутність – це абстракція реального об'єкту предметної галузі. Оскільки сутність відповідає цілому класу однотипних об'єктів предметної галузі, то

передбачається, що існує багато екземплярів даної сутності. Атрибутом називають поіменовану характеристику сутності. Зв'язок – це засіб, за допомогою якого подаються відношення між сутностями.

Прикладами сутностей можуть бути сутності ВИКЛАДАЧ, СТУДЕНТ, ОЦІНКА. Атрибутами сутності ВИКЛАДАЧ можуть бути, наприклад, ТАБЕЛЬНИЙ НОМЕР, ПРИЗВИЩЕ, НАУКОВИЙ СТУПІНЬ, КАФЕДРА. набір атрибутів, що однозначно характеризує кожен конкретний екземпляр сутності, називають ключовим.

Між сутностями встановлюються зв'язки, що відповідають реальним зв'язкам між відповідними об'єктами предметної галузі. Зв'язки бувають бінарні (між двома сутностями); тернарні (між трьома сутностями); N-арні (між N сутностями) та рекурсивні (зв'язок сутності самої з собою).

Найбільш поширеними є бінарні зв'язки, які поділяють на:

- 1) 1:1 (один до одного);
- 2) 1:M (один до багатьох);
- 3) N:M (багато до багатьох).

Наприклад, зв'язок “один до одного” встановлюють між сутностями “куратор” та “група”, оскільки у кожній студентській групі є тільки один куратор.

Зв'язок “один до багатьох” встановлюють між сутностями “студент” та “оцінка”, оскільки кожен студент упродовж навчання отримує багато оцінок.

Зв'язок “багато до багатьох” встановлюють між сутностями “студент” та “викладач”, оскільки один студент навчається у багатьох викладачів, і, в свою чергу, один викладач навчає багатьох студентів.

Взагалі, ER-модель досить складна і у достатньо повному вигляді її доцільно вивчати у курсі “Системи управління базами даних” на спеціальності “Інформатика” педагогічного університету [8].

Наступним етапом розв'язування задачі засобами СУБД є логічне проектування бази даних. Для реляційних СУБД логічне проектування полягає в тому, що на основі ER-моделі створюють реляційну схему, визначають кількість і структуру таблиць, створюють відповідні запити до

БД, визначають типи звітних документів і відповідно формують звіти в СУБД, створюють форми для організації інтерфейсу з користувачем, в тому числі для перегляду, введення та редагування даних. Коректність логічної моделі перевіряють і, якщо необхідно, вдосконалюють за допомогою правил нормалізації, що дозволяє впевнитися в логічній цілісності даних у моделі і позбутися надмірності даних. Побудована таким чином концептуальна модель реалізується у конкретній СУБД.

7. Побудова баз даних в СУБД LibreOffice Base

У Чернігівському педагогічному університеті на кафедрі інформатики в якості навчальної СУБД обрано СУБД LibreOffice Base. Такий вибір зумовлений тим, що дана СУБД є складовою частиною пакету офісних програм LibreOffice, який є кросплатформенним (є реалізації для всіх популярних операційних систем: сімейства ОС Microsoft Windows, сімейства ОС Linux, ОС Mac OS X), а також вільно поширюваним. СУБД LibreOffice Base з цього пакету є потужною реляційною СУБД, використання якої дає змогу створювати складні бази даних і використовувати мову SQL для формування запитів. В актуальній 4-й версії запити на вибірку можна створювати і за допомогою мови QBE, але для запитів на оновлення, додавання та вилучення даних потрібно обов'язково застосовувати мову SQL, причому реалізовано це таким чином, що мінімізується можливість необдуманого багаторазового запуску таких запитів. Такий спосіб реалізації запитів на модифікацію бази даних більш вдалий, ніж наприклад в СУБД MS ACCESS, де запити на модифікацію можуть запускатися недосвідченими студентами багатократно без будь-яких обмежень, і практика роботи з СУБД MS ACCESS показує, що це часто призводить до спотворення даних.

Для подальшого формування у студентів компетентностей щодо аналізу предметної галузі задачі, побудови інформаційної моделі (в даному випадку спочатку ER-моделі, а на її основі SQL-моделі), та дослідження отриманої моделі студентам пропонується цикл з трьох лабораторних робіт. У першій роботі студенти повинні на основі умови задачі у термінах предметної галузі

розробити адекватну модель даних, визначивши сутності предметної галузі, їх властивості та зв'язки між ними; на основі цієї моделі даних створити відповідні таблиці, визначити ключі цих таблиць, встановити зв'язки між таблицями, створити форми для введення і редагування даних з визначенням обмежень на значення у полях. Заповнити таблиці даними.

У другій лабораторній роботі студенти повинні розробити запити до бази даних з попередньої лабораторної роботи. Умови запитів сформульовані у термінах предметної галузі. Запити на вибірку даних та обчислення над даними пропонується виконувати за допомогою мови QBE (у термінах СУБД LibreOffice Base – у дизайнері запитів), а запити на модифікацію даних повинні бути описані мовою SQL.

У третій лабораторній роботі студенти повинні розробити звіти (з використанням обчислювальних полів) до бази даних, розробленої у першій лабораторній роботі.

Завдання у лабораторних роботах містять 12 варіантів, таким чином кожен студент підгрупи виконує власний варіант. Умови лабораторних робіт наведено у параграфах 9, 10, 11.

Наведемо етапи розв'язування (побудова моделі даних, бази даних, форм, запитів, звітів) задачі, подібної до тих, що є в умовах лабораторних робіт.

Задача 1.

Створити базу даних FIRMA, в якій зберігатимуться дані про співробітників деякої фірми і зроблені їм виплати заробітної плати упродовж деякого періоду (не більше однієї виплати в день).

У базі даних мають зберігатися такі відомості про кожного співробітника:

- табельний номер;
- прізвище, ім'я, по-батькові;
- дата народження;
- домашня адреса;
- посада у фірмі;
- стаж роботи;

- дата отримання заробітної плати;
- отримана сума заробітної плати.

Розв'язування задачі почнемо з побудови ER-моделі даних, тобто моделі “Сутність-Зв’язок”. З умови задачі можна виділити наступні сутності предметної галузі: “Співробітник фірми” та “Виплата”. Ці сутності пов’язані відношенням “Один до багатьох”, оскільки кожен співробітник фірми отримує багато виплат. Тепер визначимо атрибути кожної з сутностей. За умовою задачі сутність “Співробітник фірми” має атрибути “Табельний номер”, “Прізвище ім’я по-батькові (ПІБ)”, “Дата народження”, “Адреса” (домашня адреса), “Посада” та “Стаж”, а сутність “Виплата” має атрибути “Дата виплати” (коли зроблена виплата) та “Сума” (розмір виплати). Первинним ключем сутності “Співробітник фірми” є атрибут “Табельний номер”, а сутності “Виплата” – “Дата виплати” (оскільки за умовою задачі не можна робити більше однієї виплати за день).

Далі необхідно перетворити отриману ER-модель в реляційну (SQL) модель. Для кожної сутності створюється відповідна таблиця, полями якої стають атрибути сутності, а первинний ключ сутності стає первинним ключем таблиці. Отримаємо таблиці, які назвемо “Співробітники” та “Виплати”.

Для встановлення зв’язків “один до багатьох” між таблицями потрібно у кожному таблицю, що відповідає підлеглій сутності, додати набір полів, що є первинним ключем таблиці, відповідної основній сутності. Цей набір полів стає зовнішнім ключем. Оскільки сутність “Виплата” є підлеглою до сутності “Співробітник фірми”, у таблицю “Виплати” потрібно додати поле “Табельний номер” із таблиці “Співробітники”.

Таблиця 7.

Таблиця "Співробітники"

Співробітники
Табельний номер
ПІБ
Дата народження

Співробітники
Адреса
Посада
Стаж

Таким чином отримано наступні таблиці (Табл. 7, 8) (ключові поля позначено напівжирним шрифтом):

Ці таблиці пов'язані відношенням “один до багатьох” за полем “Табельний номер”.

Таблиця 8.

Таблиця "Виплати"

Виплати
Табельний номер
Дата виплати
Сума

Далі потрібно перевірити утворену модель даних за допомогою правил нормалізації. Результатом процесу нормалізації є отримання такої моделі даних, в якій таблиці мають просту і регулярну структуру. В створеній моделі даних можуть бути аномалії – такі ситуації, що призводять до протиріч у базі даних, або ускладнюють опрацювання даних. Нормалізація надає можливість позбутися таких аномалій. Одним з видів аномалій є аномалія модифікації. Вона виникає, коли зміну значень якогось поля потрібно проводити у багатьох записах. Це може призвести до порушення цілісності бази даних (у разі неправильного введення нового значення поля в одному з записів). У розробленій моделі даних така аномалія виникає, коли потрібно змінити назву посади, оскільки таку зміну потрібно виконати для всіх співробітників, які обіймають цю посаду. Для ліквідації цієї аномалії створимо нову таблицю “Штатний розпис”, яка міститиме поля “Номер посади” та “Назва посади”. Тоді поле “Посада” у таблиці “Співробітники”

відповідатиме полю “Номер посади” у таблиці “Штатний розпис” і повинно мати такий самий тип даних.

Таблиця 9.

Таблиця "Штатний розпис"

Штатний розпис
Номер посади
Назва посади

Для формування у студентів компетентностей щодо розробки запитів в конкретній СУБД студентам необхідно запропонувати такі завдання, які б максимально охоплювали всі види запитів: запити на вибірку даних (в тому числі з використанням параметрів), запити на обчислення, в тому числі і групові, запити на модифікацію даних (зміну існуючих даних, додавання нових, вилучення існуючих). Важливо, щоб студенти мали компетентності щодо створення запитів як мовою QBE, так і мовою SQL.

Наведемо кілька прикладів створення запитів до бази даних, про яку йшла мова у задачі 1, в СУБД LibreOffice Base.

Запити на вибірку даних і виконання обчислень

Для задання умов щодо даних числових, грошових типів і типу дата/час застосовуються оператори порівняння $>$, $<$, $>=$, $<=$, $=$ (= не пишеться).

Для задання кон'юнкції (“і”) застосовується оператор *And*, для заперечення (“не”) – *Not*, для диз'юнкції (“або”) – *Or*, складові диз'юнкції можуть також записуватись в різних рядках бланку запиту.

Константи типу дата/час обмежуються позначками #, наприклад #10/10/02#

Для задання умов щодо текстових полів можна застосовувати символи шаблонів * і ?, які мають загальноприйняте значення (відповідно довільна кількість довільних символів і довільний єдиний символ). Якщо потрібний повний збіг з умовою відбору, відповідний текст записується без символів

шаблонів. При вказуванні шаблону слід записати оператор *Like* (подібний), а потім шаблон, взятий в одинарні лапки. Наприклад, умова *Like* ‘*бухгалтер’ є критерієм відбору таких рядків, як, бухгалтер, головний бухгалтер, старший бухгалтер тощо.

Логічна константа *EMPTY* означає відсутність значення у полі.

В СУБД LibreOffice Base передбачено можливість формувати умови відбору даних в процесі виконання запиту. Така властивість запитів реалізується завдяки використанню параметрів. Щоб створити запит з параметром, слід у бланку запиту в рядку *Критерій* вказати не константу певного типу, з якою будуть порівнюватись дані, а ім'я змінної, перед яким поставити двокрапку. При виконанні такого запиту на екран виводиться вікно з назвою *Введення параметра*, в якому є поле для введення значення параметра. Після введення значення можна побачити відповідь на запит.

Можна вказати у бланку запиту кілька параметрів у різних полях. Тоді при виконанні запиту слід послідовно вказати значення всіх параметрів.

Для створення обчислюваного поля слід у бланку запиту в рядку *Поле* ввести вираз, за яким обчислюються значення у створюваному полі. Він може містити знаки математичних операцій, дужки, стандартні функції LibreOffice Base, імена полів таблиці, до якої створюється запит, взяті в подвійні лапки. Ім'я нового поля слід ввести в рядок бланку запиту *Псевдонім*.

Для обчислення підсумкових значень необхідно включити у бланк запиту рядок *Функція* (для цього слід натиснути кнопку *Функції* на панелі інструментів *Дизайн* вікна програми).

В СУБД LibreOffice Base передбачено виконання таких групових операцій над даними:

Group – операція призначена для групування даних у тому полі, в якому вона встановлена (відповідні записи виводяться підряд);

Сума (Sum) – знаходження суми значень у відповідному полі;

Середнє значення (Average) – знаходження середнього арифметичного (в Конструкторі запитів для виконання цієї операції слід ввести відповідний вираз;

Мінімум (Min) – знаходження мінімального значення у полі;

Максимум (Max) – знаходження максимального значення у полі;

Кількість (Count) – знаходження кількості записів.

Приклад 1. За допомогою мови QBE створити запит для формування даних (ПІБ, Дата виплати, Сума) про розміри всіх виплат співробітнику Петренку, зроблених з початку поточного (2010) року.

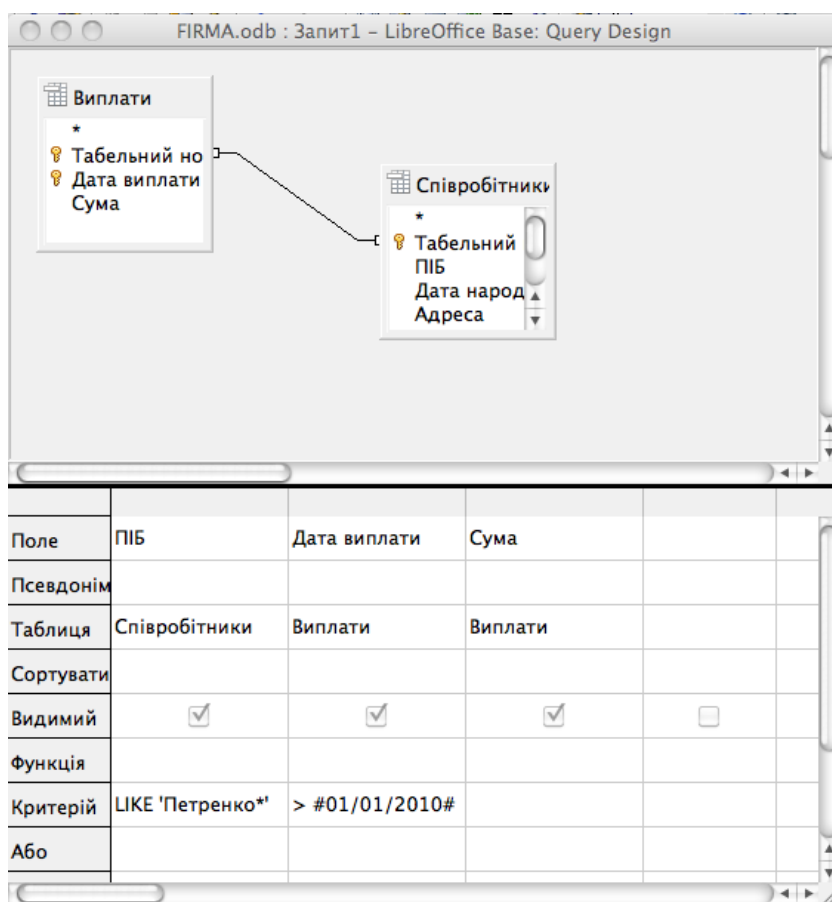


Рис. 3. Запит до Прикладу 1.

Запит мовою SQL матиме наступний вигляд:

```
SELECT "Співробітники"."ПІБ", "Виплати"."Дата виплати",  
"Виплати"."Сума" FROM "Виплати", "Співробітники" WHERE  
"Виплати"."Табельний номер" = "Співробітники"."Табельний
```

номер" AND "Співробітники"."ПІБ" LIKE 'Петренко%' AND
 "Виплати"."Дата виплати" > {D '2010-01-01' }

Запит мовою QBE матиме вигляд, як на Рис.3.

Приклад 2. Використовуючи дані таблиць “Співробітники” і “Виплати”, обчислити суми, відраховані у фонд соціального страхування (1%) з виплат усім співробітникам за 2009 рік. Вивести у запиті такі відомості: ПІБ, Сума, Податок.

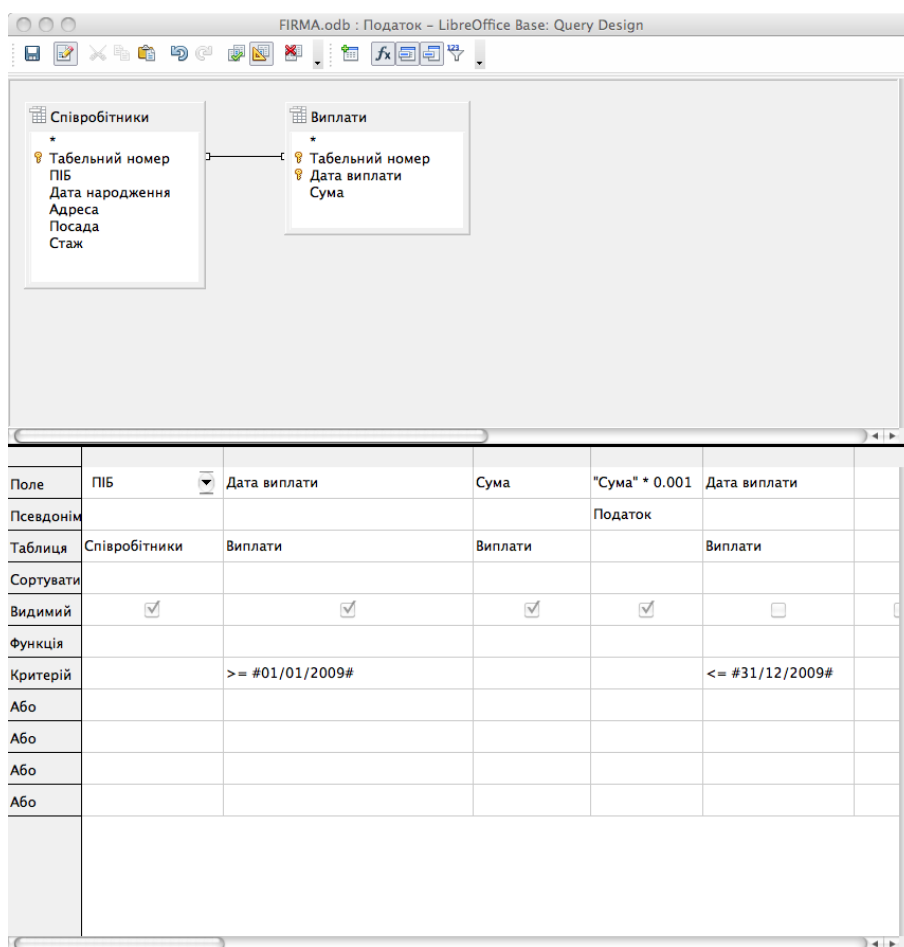


Рис. 4. Запит до Прикладу 2.

Запит мовою SQL матиме наступний вигляд:

```
SELECT      "Співробітники"."ПІБ",      "Виплати"."Дата виплати",
           "Виплати"."Сума", "Сума" * 0.001 AS "Податок" FROM
           "Виплати", "Співробітники" WHERE "Виплати"."Табельний
номер"    =    "Співробітники"."Табельний номер" AND
           "Виплати"."Дата виплати" >= {D '2009-01-01' } AND
```

"Виплати"."Дата виплати" <= {D '2009-12-31' }

Запит мовою QBE матиме вигляд, як на Рис.4.

Приклад 3. Використовуючи дані таблиць “Співробітники” і “Виплати”, обчислити кількість виплат, зроблених кожному співробітнику упродовж другого півріччя 2009 року. У відповіді вивести поля ПІБ, Кількість виплат у 2 півр. 2009 р.

Запит мовою SQL матиме наступний вигляд:

```
SELECT "Співробітники"."ПІБ", COUNT( "Виплати"."Сума" ) AS
"Кількість виплат у 2 півр. 2009 р." FROM "Виплати",
"Співробітники" WHERE "Виплати"."Табельний номер" =
"Співробітники"."Табельний номер" AND "Виплати"."Дата
виплати" >= {D '2009-07-01' } AND "Виплати"."Дата виплати" <=
{D '2009-12-31' } GROUP BY "Співробітники"."ПІБ"
```

Запит мовою QBE матиме вигляд, як на Рис. 5.

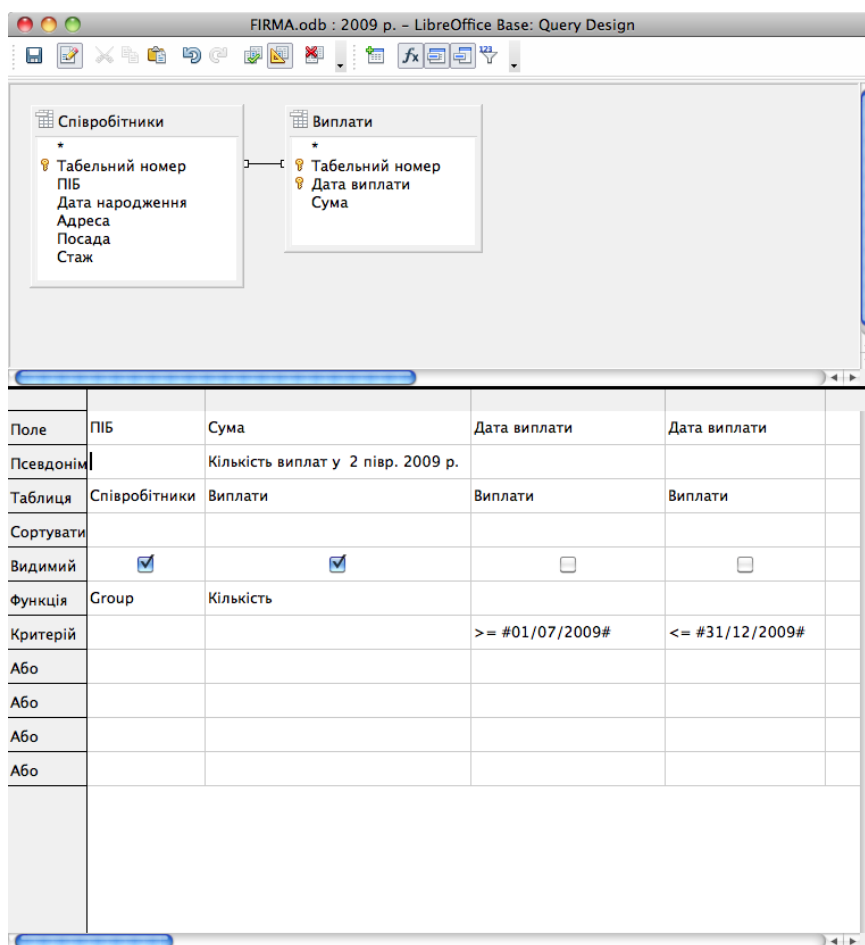


Рис. 5. Запит до Прикладу 3.

Запити на модифікацію даних

В СУБД LibreOffice Base передбачено можливість створювати запити, результатами яких є не нові таблиці, а зміни, внесені у вихідні таблиці бази даних. Такі запити створюються за допомогою засобів мови SQL. Після запуску запиту на виконання в LibreOffice Base не виводиться попередження про можливі зміни даних. Виконання запиту призводить до внесення незворотних змін у таблицю. Тому виконувати такі запити слід помірковано і зважено. Після виконання запитів на модифікацію їх результат переглядають, відкривши відповідну таблицю на закладці *Таблиці* вікна бази даних.

При створенні запитів на зміну таблиць слід перейти на закладку *Таблиці* вікна бази даних і вибрати команду меню *Сервіс – SQL*. У допоміжному вікні *Виконати інструкцію SQL* в поле *Виконувана команда* слід ввести текст запиту і натиснути кнопку *Виконати*. Повідомлення про результат виконання команди (успішне виконання або знайдені помилки) виводиться в нижній частині вікна в полі *Стан*. Після завершення роботи з запитом слід натиснути кнопку *Закрити*. Текст SQL-запитів у LibreOffice Base не зберігається. Тому для повторного запуску або збереження запиту в певному звіті текст запиту слід скопіювати до буфера обміну для подальшого використання.

Використовуючи запити на оновлення, можна змінювати значення окремих полів вихідної таблиці. Для полів, що підлягають зміні, можуть бути вказані певні умови.

Найпростіша структура запиту на оновлення:

```
UPDATE "Ім'я таблиці"  
SET "Ім'я поля"="нове значення"  
WHERE <умова>
```

<умова> – це вираз, в якому можуть використовуватись імена полів, константи, знаки математичних операцій і відношень, логічні оператори.

Текстові константи беруться в одинарні лапки, наприклад: *'текст'*

Константи типу „Дата” записуються у вигляді *{d 'рррр-мм-дд'}*, наприклад:

{d '2010-12-01'} означає 1 грудня 2010 року.

За допомогою запитів на вилучення даних користувач може вилучати з таблиці записи, у яких значення певних полів відповідають вказаним умовам.

Найпростіша структура запиту на вилучення:

```
DELETE FROM "Ім'я таблиці"  
WHERE <умова>
```

Запити на додавання дозволяють поповнювати таблицю даними з іншої таблиці або вказаними безпосередньо у тексті запиту.

Найпростіша структура запиту на додавання:

```
INSERT INTO "Ім'я таблиці"  
VALUES(список значень полів)
```

Приклад 4. Замінити у таблиці “Виплати” всі входження дати 1 травня 2010 року на 11 травня 2010 року.

Виконання. У допоміжному вікні *Виконати інструкцію SQL* в поле *Виконувана команда* ввести такий текст запиту:

```
update "Виплати"  
set "Дата виплати"={d '2010-05-10'}  
where "Дата виплати"={d '2010-05-01'}
```

Приклад 5. Вилучити з таблиці “Виплати” відомості про виплати, зроблені директору.

Виконання. У допоміжному вікні *Виконати інструкцію SQL* в поле *Виконувана команда* ввести такий текст запиту:

```
delete from "Виплати"  
  where "Табельний номер" in  
  (select "Табельний номер" from "Співробітники"  
  where "Посада" in  
  (select "Номер посади" from "Штатний розпис"  
  where "Назва посади"='Директор'))
```

Приклад 6. Додати до таблиці “Співробітники” всі анкетні відомості про нового співробітника: Табельний номер – 7777, ПІБ – Ткач О.О., Дата народження – 10.10.1980, Адреса – Проспект Перемоги, 33/22, Посада – 3, Стаж – 10.

Виконання. У допоміжному вікні *Виконати інструкцію SQL* в поле *Виконувана команда* ввести такий текст запити:

```
insert into "Співробітники"  
  values(7777,'Ткач О.О.',  
  {d '1980-10-10'},'Просп Перемоги, 33/22',3,10)
```

Приклад 7. Збільшити на один рік стаж всіх співробітників, які працюють на посаді інженера і чий стаж більше одного року.

Виконання. У допоміжному вікні *Виконати інструкцію SQL* в поле *Виконувана команда* ввести такий текст запити:

```
update "Співробітники"  
  set "Стаж"="Стаж"+1  
  where "Стаж">1 and "Посада" in  
  (select "Номер посади" from "Штатний розпис"  
  where "Назва посади"='Інженер')
```

Приклад 8. Подвоїти всі виплати директору фірми.

Виконання. У допоміжному вікні *Виконати інструкцію SQL* в поле *Виконувана команда* ввести такий текст запити:

```
update "Виплати"  
set "Сума"="Сума"*2  
where "Табельний номер" in  
(select "Табельний номер" from "Співробітники"  
where "Посада" in  
(select "Номер посади" from "Штатний розпис"  
where "Назва посади"='Директор'))
```

Приклад 9. Вилучити з таблиці “Співробітники” всі відомості про інженерів, стаж роботи яких більше 20 років.

Виконання. У допоміжному вікні *Виконати інструкцію SQL* в поле *Виконувана команда* ввести такий текст запити:

```
delete from "Співробітники"  
where "Стаж">20 and "Посада" in  
(select "Номер посади" from "Штатний розпис"  
where "Посада" in  
(select "Номер посади" from "Штатний розпис"  
where "Назва посади"='Інженер'))
```

Приклад 10. Вилучити відомості про виплати бухгалтерам, стаж яких більше 10 років.

Виконання. У допоміжному вікні *Виконати інструкцію SQL* в поле *Виконувана команда* ввести такий текст запити:

```
delete from "Виплати"  
where "Табельний номер" in  
(select "Табельний номер"
```

*from "Співробітники"
where "Стаж">10 and
"Посада" in
(select "Номер посади" from "Штатний розпис"
where "Назва посади"='Бухгалтер'))*

Приклад 11. Додати до таблиці “Співробітники” окремі анкетні відомості (Табельний номер, ПІБ, Посада) про нового співробітника.

Виконання. У допоміжному вікні *Виконати інструкцію SQL* в поле *Виконувана команда* ввести такий текст запити:

*insert into "Співробітники" ("Код", "ПІБ", "Посада")
values(8888, 'Шевченко А.А.', 1)*

8. Лабораторна робота “Створення і редагування таблиць в СУБД LibreOffice Base”

Практичне завдання

Варіант 1

1. Створити базу даних РЕЛІГІЯ, яка буде містити відомості про розподіл віруючих різних релігій на континентах. В базі даних мають міститися дані про континенти: назву континенту, площу континенту, чисельність населення континенту, дату, на яку подано дані про континент; дані про релігії: назву релігії, % віруючих у світі, століття заснування релігійної течії; дані про розподіл релігій на різних континентах: % віруючих на кожному окремому континенті, кількість храмів на кожному окремому континенті, дати, на які було отримано дані про кількість храмів та % віруючих на континенті.
2. Проаналізувати необхідні дані, розробити структуру бази даних, визначити необхідні таблиці, поля та їх типи (для зв'язування створених таблиць ввести в них додаткові ключові поля). Узгодити розроблену структуру з викладачем.
3. Увійти до LibreOffice Base. Створити в заново створеній папці базу даних та таблиці в ній.
4. Для створених таблиць бази даних створити схему даних із забезпеченням цілісності даних, в якій пов'язати між собою відповідні поля створених таблиць.
5. Для кожної таблиці створити форму для внесення до неї даних. Встановити обмеження на поля форм:
 - поля, які будуть зв'язуючими для таблиці, мають контролюватися підстановкою з відповідної головної таблиці;
 - площа континенту вводиться в км², за замовчуванням встановлюється 10000;
 - населення континенту вводиться в тис. осіб. та знаходиться в межах від 0 до $9 \cdot 10^7$;

- кількість храмів вводиться в тис. та знаходиться в межах від 0 до $9 \cdot 10^4$;
 - поля % віруючих у світі та на континенті знаходяться в межах від 0 до 100, за замовчуванням 1;
 - дати знаходяться в межах від 01.01.2000 до 31.12 поточного року.
6. Заповнити таблиці даними принаймні про 5 континентів, 5 релігій та 20 записів про різні релігії на різних континентах.
 7. Створити дві підпорядковані форми, лише для перегляду даних. Зовнішній вигляд субформи – “*як аркуш даних*”. Одна форма має містити поля для введення даних про континент та релігії на континентах. Друга форма має містити поля для введення відомостей про релігію та її поширення на континентах.

Варіант 2

1. Створити базу даних ВІДПОЧИНОК, в якій будуть міститися дані про оформлення туристичних поїздок різними туристичними агенціями. В базі даних мають міститися відомості про туристичні агенції: назву, адресу знаходження, прізвище директора, дату ліцензування, статутний капітал; відомості про тури: назва туру, місце відпочинку за даним туром, необхідність візи для відпочинку, кількість днів відпочинку; дані про продаж путівок різними агенціями на різні тури: кількість проданих путівок певним туристичним агентством на деякий тур, дати від’їзду на відпочинок за проданими путівками, ціну за одну продану путівку певною агенцією.
2. Проаналізувати необхідні дані, розробити структуру бази даних, визначити необхідні таблиці, поля та їх типи. Узгодити розроблену структуру з викладачем.
3. Увійти до LibreOffice Base. Створити в заново створеній папці базу даних.
4. Для створених таблиць бази даних створити із забезпеченням цілісності даних схему даних, в якій пов’язати між собою відповідні поля створених таблиць.

5. Для кожної таблиці створити форму для внесення до неї даних.

Встановити обмеження на поля форм:

- поля, які будуть зв'язуючими для таблиць та призначені для введення однойменних даних з іншої таблиці, мають контролюватися підстановкою з відповідної головної таблиці;
- дата заснування агенції знаходиться в межах від 01.01.1947 до 31.12.2010;
- статутний капітал вводиться в тис. грн. та знаходиться в межах від 1 до 9000, за замовчуванням встановлюється 100;
- поле назви туру контролюється з наперед заданого списку;
- поле місця відпочинку контролюється з наперед заданого списку;
- кількість днів відпочинку знаходиться в межах від 1 до 60, за замовчуванням 7;
- кількість проданих путівок знаходиться в межах від 1 до 10, за замовчуванням 1;
- дата від'їзду на відпочинок знаходиться в межах від 01.01.2005 до 31.10 поточного року;
- ціна за одну путівку вводиться в грн. та знаходиться в межах від 250 до 5000.

6. Заповнити таблиці даними принаймні про 5 туристичних агенцій, 5 турів та 20 записів про продаж туристичних путівок на різні тури.

7. Створити дві підпорядковані форми, лише для перегляду даних. Зовнішній вигляд субформи – *„як аркуш даних”*. В одній формі мають міститися поля для введення даних про туристичну агенцію та продаж путівок на різні тури. В другій формі мають міститися поля для введення даних про тур та продаж путівок на нього різними агенціями.

Варіант 3

1. Створити базу даних АКЦІОНЕРИ, яка буде містити дані про придбання акцій різних підприємств акціонерами. В базі даних мають міститися відомості про акціонерів: прізвище, ім'я, по батькові акціонера, адресу

проживання, величину надбавки, документ, за яким зареєстровано акціонера; дані про підприємства: назву підприємства, дату його заснування, прізвище, ім'я, по батькові директора, адресу розміщення, номер телефону; дані про купівлю різними акціонерами акцій різних підприємств: дату придбання певним акціонером акцій деякого підприємства, номер випуску придбаних акцій, кількість придбаних акцій.

2. Проаналізувати необхідні дані, розробити структуру бази даних, визначити необхідні таблиці, поля та їх типи. Узгодити розроблену структуру з викладачем.
3. Увійти до LibreOffice Base. Створити в заново створеній папці базу даних.
4. Для створених таблиць бази даних створити схему даних із забезпеченням цілісності даних, в якій пов'язати між собою відповідні поля створених таблиць.
5. Для кожної таблиці створити форму для внесення до неї даних.
Встановити обмеження на поля форм:

- поля, які будуть зв'язуючими для таблиць та призначені для введення однойменних даних з іншої таблиці, мають контролюватися підстановкою з відповідної головної таблиці;
- поле документу контролюється за даними з наперед заданого списку;
- надбавка до виплат вводиться в % та знаходиться в межах від 0 до 25;
- дата заснування підприємства знаходиться в межах від 01.01.1995 до 30.10.2005;
- дата придбання акцій знаходиться в межах від 01.01.2005 до 31.01 поточного року;
- номер випуску акцій є цілим числом, знаходиться в межах від 0 до 10, за замовчуванням 1;
- кількість придбаних акцій вводиться в шт. та знаходиться в межах від 0 до 10000, за замовчуванням 50.

6. Заповнити таблиці даними принаймні про 5 акціонерів, 5 підприємств та 20 записів про купівлю акцій підприємств акціонерами.
7. Створити дві підпорядковані форми, лише для перегляду даних. Зовнішній вигляд субформи – *“як аркуш даних”*. В одній формі мають міститися поля для даних про акціонера та те, акції яких підприємств і в якій кількості він купував. В другій формі мають міститися поля для даних про підприємство та акціонерів, які купували його акції.

Варіант 4

1. Створити базу даних ВІДОМІСТЬ, яка буде містити відомості про оцінки студентів з різних дисциплін. В базі даних мають міститися дані про студентів: прізвище студента, дата його народження, спеціальність, номер групи, відмітка про навчання за контрактом чи за держзамовленням; дані про дисципліну: назва дисципліни, прізвища лектора та викладача, який веде практичні (лабораторні) роботи, загальну кількість годин на дисципліну; дані про складання різними студентами різних екзаменів: дату складання студентом певного предмету, семестр, за який було складено екзамен, кількість годин, виділених на дисципліну в цьому семестрі, бал, отриманий студентом на екзамені.
2. Проаналізувати необхідні дані, розробити структуру бази даних, визначити необхідні таблиці, поля та їх типи. Узгодити розроблену структуру з викладачем.
3. Увійти до LibreOffice Base. Створити в заново створеній папці базу даних.
4. Для створених таблиць бази даних створити схему даних, із забезпеченням цілісності даних, в якій пов'язати між собою відповідні поля створених таблиць.
5. Для кожної таблиці створити форму для внесення до неї даних. Встановити обмеження на поля форм:
 - поля, які будуть зв'язуючими для таблиць та призначені для введення однойменних даних з іншої таблиці, мають контролюватися підстановкою з відповідної головної таблиці;

- поле спеціальності контролюється за даними з наперед заданого списку;
- поле прізвищ викладачів контролюється за даними з наперед заданого списку;
- дата народження знаходиться в межах від 01.01.1970 до 31.12.2000;
- відмітка про контракт містить значення „так”, якщо студент навчається за контрактом;
- кількість годин на дисципліну загалом вводиться в год. та знаходиться в межах від 0 до 500, за замовчуванням 120;
- кількість годин на дисципліну за семестр вводиться в год. та знаходиться в межах від 0 до 64, за замовчуванням 60;
- дата складання іспиту знаходиться в межах від 01.01.1988 до 01.01 поточного року;
- отримані бали знаходиться в межах від 60 до 100, за замовчуванням встановлюється 60.

6. Заповнити таблиці даними принаймні про 5 студентів, 5 предметів та 20 записів про складання іспитів студентами.

7. Створити дві підпорядковані форми, лише для перегляду даних. Зовнішній вигляд субформи – *“як аркуш даних”*. В одній формі мають міститися дані про студента та оцінки, які він отримував на екзаменах з різних дисциплін. В другій формі мають міститися відомості про дисципліну та те, які студенти і як складали екзамени з неї.

Варіант 5

1. Створити базу даних РЕАЛІЗАЦІЯ, в якій будуть міститися дані про продаж товарів різними магазинами. В базі даних мають міститися відомості про магазини: назва магазину, прізвище його власника, адреса розміщення, дата заснування; відомості про товар: назва товару, виробника, характеристика упаковки, кількість одиниць в упаковці; дані

про продаж різних товарів різними магазинами: дата продажу товару, кількість проданих одиниць та ціна за одиницю.

2. Проаналізувати необхідні дані, розробити структуру бази даних, визначити необхідні таблиці, поля та їх типи. Узгодити розроблену структуру з викладачем.
3. Увійти до LibreOffice Base. Створити в заново створеній папці базу даних.
4. Для створених таблиць бази даних створити схему даних, із забезпеченням цілісності даних, в якій пов'язати між собою відповідні поля створених таблиць.
5. Для кожної таблиці створити форму для внесення до неї даних. Встановити обмеження на поля форм:

- поля, які будуть зв'язуючими для таблиць та призначені для введення однойменних даних з іншої таблиці, мають контролюватися підстановкою з відповідної таблиці;
- дата заснування магазину знаходиться в межах від 01.01.1960 до 31.12.1990;
- поле упаковки містить тип упаковки (коробка, пляшка, мішок, ...) та контролюється за даними з наперед заданого списку;
- дата продажу товару знаходиться в межах від 01.01.2005 до 31.12 поточного року;
- кількість проданого товару вводиться в шт. та знаходиться в межах від 0 до 1000, за замовчуванням 2;
- вартість за одиницю вводиться в грн. та знаходиться в межах від 1 до 20000.

6. Заповнити таблиці даними принаймні про 5 магазинів, 5 товарів та 20 записів про продаж товарів в магазинах.
7. Створити дві підпорядковані форми, лише для перегляду даних. Зовнішній вигляд субформи – “як аркуш даних”. В одній формі мають міститися поля для введення даних про магазин, товари, які були продані в ньому (з вказуванням кількості). В другій формі мають міститися поля з

даними про товар та про те, в яких магазинах і в якій кількості вони були продані.

Варіант 6

1. Створити базу даних КІНОТЕАТР, яка буде містити відомості про надходження фільмів. В базі даних мають міститися дані про фільми: назва фільму, студію, яка його випустила, його тривалість, прізвище режисера, дата прем'єри; відомості про постачальника: назва фірми постачальника, країна, в якій її засновано, її статутний капітал; дані про постачання різних фільмів різними постачальниками: дата постачання фільму, мова перекладу фільму, кількість днів прокату.
2. Проаналізувати необхідні дані, розробити структуру бази даних, визначити необхідні таблиці, поля та їх типи. Узгодити розроблену структуру з викладачем.
3. Увійти до LibreOffice Base. Створити в заново створеній папці базу даних.
4. Для створених таблиць бази даних створити схему даних, із забезпеченням цілісності даних, в якій пов'язати між собою відповідні поля створених таблиць.
5. Для кожної таблиці створити форму для внесення до неї даних. Встановити обмеження на поля форм:
 - поля, які будуть зв'язуючими для таблиць та призначені для введення однойменних даних з іншої таблиці, мають контролюватися підстановкою з відповідної таблиці;
 - тривалість фільму вводиться в хв. та знаходиться в межах від 45 до 300, за замовчуванням 120;
 - дата прем'єри знаходиться в межах від 01.01.2005 до 20.12 поточного року;
 - статутний капітал постачальника вводиться в млн. грн. та знаходиться в межах від 0 до 500, за замовчуванням 1;
 - кількість днів прокату знаходиться в межах від 1 до 20, за замовчуванням 6;

- поле мови перекладу фільму контролюється за даними з наперед заданого списку .
6. Заповнити таблиці даними принаймні про 5 фільмів, 5 дистриб'юторів та 20 записів про отримання фільмів у прокат.
 7. Створити дві підпорядковані форми, лише для перегляду даних. Зовнішній вигляд субформи – “як аркуш даних”. В одній формі мають міститися поля з даними про фільм та те, які постачальники на яких умовах їх поставляли. В другій формі мають міститися поля з даними про дистриб'ютора та фільми, які він надавав в прокат кінотеатру.

Варіант 7

1. Створити базу даних БІБЛІОТЕКА, в якій будуть міститися відомості про отримання студентами книг у бібліотеці. В базі даних мають міститися дані про студента: номер залікової книжки, прізвище, дата його народження, факультет, спеціальність, група; відомості про книги: назва та прізвище автора книги, рік її випуску, назва видавництва, кількість сторінок; дані про використання книг студентами: дата отримання книги певним студентом, термін, на який було взято книгу.
2. Проаналізувати необхідні дані, розробити структуру бази даних, визначити необхідні таблиці, поля та їх типи. Узгодити розроблену структуру з викладачем.
3. Увійти до LibreOffice Base. Створити в заново створеній папці базу даних.
4. Для створених таблиць бази даних створити схему даних, із забезпеченням цілісності даних, в якій пов'язати між собою відповідні поля створених таблиць.
5. Для кожної таблиці створити форму для внесення до неї даних. Встановити обмеження на поля форм:
 - поля, які будуть зв'язуючими для таблиць та призначені для введення однойменних даних з іншої таблиці, мають контролюватися підстановкою з відповідної таблиці;

- дата народження студента знаходиться в межах від 01.01.1970 до 31.12.2000;
 - номер групи знаходиться в межах від 10 до 70;
 - рік випуску книги знаходиться в межах від 1900 до 2010;
 - кількість сторінок книги знаходиться в межах від 10 до 1200, за замовчуванням 363;
 - поле назви видавництва контролюється за даними з наперед заданого списку;
 - дата отримання книги знаходиться в межах від 01.01.2000 до 31.12 поточного року;
 - термін, на який видано книгу, вводиться в днях та знаходиться в межах від 1 до 365, за замовчуванням 31.
6. Заповнити таблиці даними принаймні про 5 студентів, 5 книг та 20 записів про отримання книг студентами.
7. Створити дві підпорядковані форми, лише для перегляду даних. Зовнішній вигляд субформи – “*як аркуш даних*”. В одній формі мають міститися поля з даними про студента та книги, які він брав в бібліотеці, і дату отримання. В другій формі мають міститися поля з даними про книги і те, який студент і коли їх брав.

Варіант 8

1. Створити базу даних АПАРТАМЕНТИ, в якій будуть міститися дані про апартаменти та їх кількість в різних готелях на різних поверхах. В базі даних мають міститися відомості про готель: назва готелю, країна його знаходження, дата заснування, кількість поверхів, кількість зірок; відомості про апартаменти: тип апартаментів, загальна площа, кількість спальних місць, кількість спалень; дані про кількість різних апартаментів в окремих готелях на окремих поверхах: поверх розміщення апартаментів, кількість відповідних апартаментів на поверсі, дату прибирання апартаментів.

2. Проаналізувати необхідні дані, розробити структуру бази даних, визначити необхідні таблиці, поля та їх типи. Узгодити розроблену структуру з викладачем.
3. Увійти до LibreOffice Base. Створити в заново створеній папці базу даних.
4. Для створених таблиць бази даних створити схему даних, із забезпеченням цілісності даних, в якій пов'язати між собою відповідні поля створених таблиць.
5. Для кожної таблиці створити форму для внесення до неї даних. Встановити обмеження на поля форм:
 - поля, які будуть зв'язуючими для таблиць та призначені для введення однойменних даних з іншої таблиці, мають контролюватися підстановкою з відповідної таблиці;
 - дата заснування готелю знаходиться в межах від 01.01.1970 до 31.12.2000;
 - кількість поверхів є ціле число та знаходиться межах від 10 до 70;
 - кількість зірок готелю є ціле число та знаходиться межах від 1 до 6, за замовчуванням 2;
 - поле типу апартаментів контролюється за даними з наперед заданого списку; дата прибирання знаходиться в межах від 01.01.1970 до 20.12 поточного року;
 - загальна площа апартаментів вводиться в м² та знаходиться в межах від 3 до 150, за замовчуванням 20.
6. Заповнити таблиці даними принаймні про 5 готелів, 5 типів апартаментів та 20 записів про наявні апартаменти в різних готелях.
7. Створити дві підпорядковані форми лише для перегляду даних. Зовнішній вигляд субформи – *“як аркуш даних”*. В одній формі мають міститися поля з даними про готель та наявні в ньому типи апартаментів на різних поверхах. В другій формі мають міститися поля з даними про апартаменти та про те, в яких готелях і в якій кількості вони наявні.

Варіант 9

1. Створити базу даних ПОСТАЧАННЯ, в якій будуть міститися відомості про постачання різних страв різними закладами харчування. В базі даних мають міститися дані про заклади харчування: назва закладу, дата його ліцензування, прізвище, ім'я, по батькові директора, адреса знаходження; дані про страви: назва страви, тип страви, вага порції, час її приготування; дані про оформлення заявок на постачання різних страв з різних закладів харчування: дата доставки, орієнтовні години постачання, ціна порції з постачанням.
2. Проаналізувати необхідні дані, розробити структуру бази даних, визначити необхідні таблиці, поля та їх типи. Узгодити розроблену структуру з викладачем.
3. Увійти до LibreOffice Base. Створити в заново створеній папці базу даних.
4. Для створених таблиць бази даних створити схему даних, із забезпеченням цілісності даних, в якій пов'язати між собою відповідні поля створених таблиць.
5. Для кожної таблиці створити форму для внесення до неї даних. Встановити обмеження на поля форм:
 - поля, які будуть зв'язуючими для таблиць та призначені для введення однойменних даних з іншої таблиці, мають контролюватися підстановкою з відповідної таблиці;
 - дата ліцензування знаходиться в межах від 01.01.1983 до 31.12.2010;
 - поле типу страви контролюється за даними з наперед заданого списку;
 - вага порції вводиться в гр. та знаходиться в межах від 50 до 500, за замовчуванням 200;
 - час приготування вводиться в хв. та знаходиться в межах від 10 до 120, за замовчуванням 25;
 - дата постачання знаходиться в межах від 01.01.2008 до 31.12 поточного року;

- ціна порції вводиться в грн., за замовчуванням 5,50;
 - орієнтовні години постачання вводяться у вигляді «10:30-11:15».
6. Заповнити таблиці даними принаймні про 5 закладів харчування, 5 страв та 20 записів про оформлення заявок на постачання.
 7. Створити дві підпорядковані форми лише для перегляду даних. Зовнішній вигляд субформи – *“як аркуш даних”*. В одній формі мають міститися поля з даними про заклад харчування та оформлення заявок на постачання страв. В другій формі мають міститися поля з даними про страви та про те, які заклади харчування їх постачали.

Варіант 10

1. Створити базу даних КУРСИ, в якій будуть міститися відомості про вчителів та предмети, яких вони навчають у різних класах. В базі даних мають міститися дані про вчителя: прізвище, ім'я, по батькові, його категорія та стаж, дата народження; дані про клас: напрямок навчання, номер класу, кількість учнів у класі; дані про проведення вчителями занять в різних класах: назва предмету, навантаження з предмету тижневе та на весь курс, дата початку занять.
2. Проаналізувати необхідні дані, розробити структуру БД, визначити необхідні таблиці, поля та їх типи. Узгодити розроблену структуру з викладачем.
3. Увійти до LibreOffice Base. Створити в заново створеній папці базу даних.
4. Для створених таблиць бази даних створити схему даних із забезпеченням цілісності даних, в якій пов'язати між собою відповідні поля створених таблиць.
5. Для кожної таблиці створити форму для внесення до неї даних. Встановити обмеження на поля форм:
 - поля, які будуть зв'язуючими для таблиць та призначені для введення однойменних даних з іншої таблиці, мають контролюватися підстановкою з відповідної таблиці;

- поле категорія контролюється з наперед заданого списку (наприклад перша, друга, вища, вчитель методист);
 - стаж вчителя вводиться в роках та знаходиться в межах від 1 до 40, за замовчуванням 5;
 - дата народження вчителя знаходиться в межах від 01.01.1940 до 31.12.2000;
 - номер класу є цілим числом в межах від 10 до 90;
 - тижневе навантаження з предмету вводиться в год. та знаходиться в межах від 2 до 8, за замовчуванням 2;
 - навантаження з предмету на курс вводиться в год. та знаходиться в межах від 20 до 100, за замовчуванням 34;
 - дата початку занять знаходиться в межах від 01.01.2000 до 31.12 поточного року.
6. Заповнити таблиці даними принаймні про 5 вчителів, 5 груп учнів та 20 записів про проведення занять вчителями з деяких предметів.
7. Створити дві підпорядковані форми лише для перегляду даних. Зовнішній вигляд субформи – “*як аркуш даних*”. В одній формі мають міститися поля з даними про вчителя та те, в яких класах яких предметів він навчає. В другій формі мають міститися поля з даними про класи учнів та про те, який вчитель навчає їх певних предметів.

Варіант 11

1. Створити базу даних ЗООПАРК, яка буде містити дані про тварин та їх годування різними кормами. В базі даних мають міститися дані про тварин: назва тварини, її прізвисько, сімейство, дата народження, вага; дані про корм: назва корму, назва виробника, одиниця виміру корму, ціна за одиницю виміру; дані про годування певної тварини певним кормом: дата годування тварини, кількість згодованих одиниць корму.
2. Проаналізувати необхідні дані, розробити структуру бази даних, визначити необхідні таблиці, поля та їх типи. Узгодити розроблену структуру з викладачем.

3. Увійти до LibreOffice Base. Створити в заново створеній папці базу даних.
4. Для створених таблиць бази даних створити схему даних, із забезпеченням цілісності даних, в якій пов'язати між собою відповідні поля створених таблиць.
5. Для кожної таблиці створити форму для внесення до неї даних. Встановити обмеження на поля форм:
 - поля, які будуть зв'язуючими для таблиць та призначені для введення однойменних даних з іншої таблиці, мають контролюватися підстановкою з відповідної таблиці;
 - поле „сімейство тварини” контролюється з наперед заданого списку;
 - дата народження знаходиться в межах від 01.01.1940 до 31.12.2009;
 - вага тварини вводиться в кг. та знаходиться в межах від 1 до 10000, за замовчуванням 120;
 - ціна за одиницю корму вводиться в грн. та знаходиться в межах від 10 до 5000;
 - дата годування тварини знаходиться в межах від 01.01.2001 до 31.12 поточного року;
 - кількість згодованих одиниць корму є цілим числом.
6. Заповнити таблиці даними принаймні про 5 тварин, 5 видів корму та 20 записів про годування тварин.
7. Створити дві підпорядковані форми лише для перегляду даних. Зовнішній вигляд субформи – *“як аркуш даних”*. В одній формі мають міститися поля з даними про тварину та її годування різними кормами. В другій формі мають міститися поля з даними про корм та про те, яких тварин ним годували.

Варіант 12

1. Створити базу даних ПАКУВАННЯ, в якій будуть міститися відомості про пакування різних продуктів в різні упаковки. В базі даних мають

міститися дані про упаковки: тип упаковки, відмітка про герметичність, вага упаковки; дані про продукти: назва продукту, дата його виготовлення; дані про пакування певного продукту в певну упаковку: дата пакування, назва фірми пакувальника, термін придатності продукту, вага продукту в упаковці.

2. Проаналізувати необхідні дані, розробити структуру бази даних, визначити необхідні таблиці, поля та їх типи. Узгодити розроблену структуру з викладачем.
3. Увійти до LibreOffice Base. Створити в заново створеній папці базу даних.
4. Для створених таблиць бази даних створити схему даних, із забезпеченням цілісності даних, в якій пов'язати між собою відповідні поля створених таблиць.
5. Для кожної таблиці створити форму для внесення до неї даних. Встановити обмеження на поля форм:
 - поля, які будуть зв'язуючими для таблиць та призначені для введення однойменних даних з іншої таблиці, мають контролюватися підстановкою з відповідної таблиці;
 - поле тип упаковки контролюється з наперед заданого списку (наприклад: пляшка, картонна коробка, поліетилен, ...);
 - дата виготовлення продукту знаходиться в межах від 01.01.2000 до 31.12.2010;
 - дата пакування продукту знаходиться в межах від 01.01.2008 до 30.12 поточного року;
 - термін придатності продукту вводиться в днях та знаходиться в межах від 1 до 300, за замовчуванням 31;
 - вага упаковки та продукту в упаковці вводиться в гр. та знаходиться в межах від 0 до 10000.
6. Заповнити таблиці даними принаймні про 5 типів упаковок, 5 продуктів та 20 записів про упаковку продуктів.
7. Створити дві підпорядковані форми, лише для перегляду даних. Зовнішній вигляд субформи – “як аркуш даних”. В одній формі мають

міститися поля з даними про упаковку та про те, які продукти були упаковані в неї. В другій формі мають міститися поля з даними про продукти, та в які упаковки вони були упаковані.

9. Лабораторна робота “Реалізація запитів в СУБД LibreOffice Base”

Практичне завдання

Реалізувати наступні запити до бази даних, створеної в лабораторній роботі “Створення і редагування таблиць в СУБД LIBREOFFICE BASE”

ВАРІАНТ 1

1. Вивести дані про континенти, населення на яких менше 500 тисяч. У відповідь включити дані про назву, площу та кількість населення на континенті.
2. Вивести дані про релігії, засновані раніше 5-го століття, а % віруючих у світі більший 50. У відповідь включити дані про назву релігії, % віруючих у світі та століття заснування.
3. Вивести дані про континенти та релігії, кількість храмів у яких знаходиться в межах від 3 до 9 тис., а населення континенту не менше 500 тисяч. У відповідь включити дані про назву континенту, назву релігії та кількість храмів на континенті.
4. Вивести дані про релігії, що поширені на певному континенті (назва континенту вводиться з клавіатури під час формування запиту). У відповідь включити дані про назву континенту, назви релігій.
5. Вивести дані про континенти, що були отримані в минулому році.
6. Вивести дані про континенти, у відповідь включити дані про назву континенту, його площу, чисельність населення та густоту населення (кількість людей на 1 м²).
7. Підрахувати загальну кількість храмів на всіх континентах з площею більше 600 тис. км². Результат помістити в поле „кількість храмів”.
8. Підрахувати кількість храмів, що знаходяться на кожному континенті. У відповідь включити дані про назву континенту та поле „кількість храмів”.

9. Вивести дані про континент, на якому % віруючих найбільший. У відповідь включити дані про назву континенту, назву релігії та % віруючих на континенті.
10. Підрахувати загальну кількість релігій на континентах, назви яких надані у відповідь на запит 9. У відповідь включити дані про назву континенту, площу, чисельність населення континенту та назву релігії.
11. Збільшити значення поля „населення” для континенту Африка на 1%.
12. Для тих релігій, що мають % віруючих у світі менший за 20%, зменшити значення поля % віруючих на континенті на 1%.
13. Занести дані про нову релігію, всі дані ввести довільні.
14. Занести дані про наявність релігії, дані про яку внесені до бази даних у запиті 13, в запис, де назвою континенту є Австралія, датою, на яку подані дані про континент, є дата виконання запиту, всі інші дані ввести довільні.
15. Вилучити відомості, введені про релігії на континентах упродовж останнього тижня.

ВАРІАНТ 2

1. Вибрати туристичні агенції зі статутним капіталом більше 500 тис. грн. У відповідь включити дані про назву туристичної агенції, прізвище, ім'я, по батькові директора, статутний капітал.
2. Вивести дані про тури тривалістю більше 15 днів, для яких не потрібні візи. У відповідь включити дані про назву туру, місце відпочинку, кількість днів відпочинку та необхідність візи.
3. Вивести дані про тури на Мальдіви, на які продавали путівки в кількості більше 7 на один тур. У відповідь включити дані про назву туру, назву агенції та кількість проданих путівок.
4. Вивести дані про тури, які надає певна туристична агенція (назва туристичної агенції вводиться з клавіатури під час формування

- запиту). У відповідь включити дані про назву туристичної агенції та назву туру.
5. Вивести дані про туристичні агенції, які мають ліцензію не менше 10 років.
 6. Вивести дані про продаж путівок, у відповідь включити дані про дату від'їзду, кількість проданих путівок, ціну за 1 путівку та вартість всіх проданих путівок (кількість*вартість).
 7. Підрахувати кількість турів, де не потребуються візи. Результат помістити в поле „кількість безвізових турів”.
 8. Визначити дані про агенції, які продавали найдешевші путівки. У відповідь включити дані про назву агенції, назву туру та вартість однієї путівки.
 9. Підрахувати, яку кількість путівок продала кожна агенція, дані про які отримано в запиті 8. У відповідь включити дані про назву агенції та поле „кількість путівок”.
 10. Зменшити на 2 кількість днів відпочинку на тури, де потребуються візи.
 11. Агенціям, які продавали більше ніж 7 путівок, збільшити статутний капітал на 10%.
 12. Занести дані про новий тур, всі дані ввести довільні.
 13. Занести дані про продаж агенціями зі статутним капіталом менше 100 тис. грн. путівок на тур, дані про який внесено до бази даних в результаті виконання запиту 12, дату від'їзду внести на 1 день раніше дати виконання запиту, всі інші дані ввести довільні.
 14. Вилучити дані про продаж путівок, дата від'їзду яких була упродовж останніх 10 днів.

ВАРІАНТ 3

1. Вибрати дані про акціонерів, що мають надбавку більше 10%. У відповідь включити дані про прізвище, адресу та розмір надбавки акціонера.

2. Вибрати підприємства, засновані в Чернігові до 01.01.1990 року. У відповідь включити дані про назву підприємства, його адресу та дату заснування.
3. Вибрати дані про акціонерів з нульовою надбавкою, які купували акції з кодом 3. У відповідь включити дані про прізвище акціонера, назву підприємства, код та кількість придбаних акцій.
4. Вивести дані про акціонерів, які володіють акціями певного випуску (номер випуску акцій вводиться з клавіатури під час формування). У відповідь включити дані про номер випуску акцій, прізвище акціонера та дату його народження.
5. Вивести дані про підприємства, засновані в минулому місяці.
6. Для кожного акціонера підрахувати вартості пакетів акцій (вважаючи що ціна однієї акції становить 10 грн.). У відповідь включити дані про код акції, дату придбання, кількість придбаних акцій та поле „вартість пакету”.
7. Підрахувати загальну кількість акцій, придбаних акціонерами, які мають надбавку більше 10%. Результат помістити в поле „кількість акцій з надбавкою”.
8. Підрахувати загальну кількість акцій, придбаних кожним акціонером. У відповідь включити дані про прізвище, адресу акціонера та поле „кількість акцій”.
9. Вибрати дані про акціонерів з найбільшою кількістю одноразово придбаних акцій. У відповідь включити дані про прізвище акціонера, кількість придбаних акцій та назву підприємства, акції якого було придбано.
10. Підрахувати загальну кількість акцій різних кодів у акціонерів, прізвища яких отримані у відповідь на запит 9. У відповідь включити дані про код акцій та поле „кількість акцій”.
11. Збільшити надбавку акціонерів, що реєструвалися за паспортом, на 2%.

12. Всім акціонерам з надбавкою, меншою за 20%, подвоїти кількість акцій з кодом 3.
13. Занести дані про нове підприємство, всі дані ввести довільні.
14. Занести дані про купівлю акцій у підприємства, відомості про яке були внесені в результаті виконання запиту 13, акціонерами, що мають нульову надбавку, в поле „дата придбання акцій” внести дату виконання запиту, всі інші дані ввести довільні.
15. Вилучити відомості про купівлю акцій за останні 3 дні.

ВАРІАНТ 4

1. Вибрати дисципліни з кількістю годин на дисципліну більше 100. У відповідь включити дані про назву дисципліни, прізвище лектора та кількість годин на предмет.
2. Вибрати дані про прізвища студентів, які навчаються за контрактом на 2-му курсі. У відповідь включити дані про прізвище, спеціальність, номер групи та дату народження студента.
3. Вибрати дані про студентів-контрактників, які склали іспит на відмінно (від 90 до 100 балів). У відповідь включити дані про прізвище студента, назву предмету та отримані бали.
4. Вивести дані про студентів, які склали екзамен з вказаного предмету (назва предмету вводиться з клавіатури під час виконання запиту). У відповідь включити дані про назву предмету, прізвище студента, номер групи, спеціальність та дату народження студента.
5. Вивести дані про студентів, яким виповнилося 19 років.
6. Підрахувати, скільки відведено занять на кожну дисципліну, враховуючи, що одне заняття триває 2 години. У відповідь включити дані про назву предмету, кількість годин на предмет та поле „кількість занять”.
7. Підрахувати кількість отриманих п'ятірок (від 90 до 100 балів) всіма студентами-контрактниками. Результат помістити в поле „кількість п'ятірок”.

8. Підрахувати кількість іспитів, складених кожним студентом. У відповідь включити прізвище студента та поле „кількість іспитів”.
9. Вибрати дані про студентів, які склали іспити останніми. У відповідь включити дані про прізвище студента, дату та назву предмету, з якого складено іспит.
10. Підрахувати загальну кількість годин за всі семестри для предметів, з яких склалися іспити, для кожного студента, дані про яких були отримані в результаті виконання запиту 9. У відповідь включити дані про назву предмета та поле „кількість годин”.
11. Збільшити загальну кількість годин з інформатики на 10.
12. Всім студентам, які склали іспити з загальною кількістю годин більше 400, кількість набраних балів збільшити на 1.
13. Занести дані про новий предмет, всі дані ввести довільні.
14. Занести дані про складання іспиту з предмету, дані про який були введені в результаті виконання запиту 13, всіма студентами-контрактниками. Дату складання ввести на 1 день раніше за дату виконання запиту, всі інші дані ввести довільні.
15. Вилучити відомості про іспити, складені за останній місяць.

ВАРІАНТ 5

1. Вибрати дані про товари з кількістю в упаковці більшою 10. У відповідь включити відомості про назву товару, виробника, упаковку та кількість одиниць товару в упаковці.
2. Вивести дані про магазини, які були засновані до 01.01.1980 та знаходяться в Києві. У відповідь включити дані про назву магазину, адресу розміщення та дату заснування.
3. Вивести дані про магазини, в яких продавалися товари з ціною за одиницю від 1000 до 5000 грн. У відповідь включити дані про назву магазину, назву товару, дату та ціну продажу.
4. Вивести дані про товари, продані в певному магазині (назва магазину вводиться з клавіатури під час формування запиту). У відповідь

- включити відомості про назву магазину, назву товару, назву виробника та упаковку.
5. Вивести дані про магазини, які почали функціонувати в позаминулому місяці.
 6. Для кожного запису продажу вивести вартість продажу (добуток кількості на ціну товару). У відповідь включити дані про дату, кількість, ціну продажу та поле „вартість”.
 7. Підрахувати продану кількість товарів, для яких кількість одиниць в упаковці є меншою за 4. Результат помістити в поле „продана кількість”.
 8. Підрахувати кількість одиниць кожного проданого товару. У відповідь включити дані про назву товару, назву виробника та поле „продано товару”.
 9. Вибрати відомості про товари, кількість продажу яких максимальна. У відповідь включити дані про назву товару, назву магазину, в якому його було продано, та максимальну кількість продажу.
 10. Підрахувати загальну кількість магазинів, в яких було продано товари, відомості про які отримані в результаті виконання запиту 9. У відповідь включити дані про назву товару та поле „кількість магазинів”.
 11. Для товарів, кількість яких в упаковці є більшою за 10, збільшити цю кількість на 1.
 12. На всі товари, упаковані в пляшки, збільшити ціну продажу за одиницю на 10 коп.
 13. Занести дані про новий магазин, всі дані ввести довільні.
 14. Занести дані про продаж у магазині, дані про який введені в результаті виконання запиту 13, товарів, кількість яких в упаковці становить 5 шт. В поле „дата продажу” ввести дату виконання запиту, всі інші дані ввести довільні.
 15. Вилучити відомості про продаж товарів за останній місяць.

ВАРІАНТ 6

1. Вивести дані про фільми, тривалість яких перевищує 115 хвилин. У відповідь включити дані про назву фільму, прізвище режисера, студію, рік випуску та тривалість фільму.
2. Вивести дані про постачальників зі статутним капіталом більше 100 млн. грн. та заснованих в США. У відповідь включити дані про назву фірми постачальника, країну, де заснована ця фірма, та її статутний капітал.
3. Вивести дані про фільми, які постачалися менше, ніж на 10 днів, та тривалістю, меншою 120 хв. У відповідь включити дані про назву фільму, назву фірми постачальника та мову перекладу фільму.
4. Вивести дані про постачальників, які постачали деякий фільм (назва фільму вводиться з клавіатури під час формування запиту). У відповідь включити дані про назву фільму, назву фірми постачальника та її статутний капітал.
5. Вивести дані про фільми, прем'єра яких була в минулому році.
6. Підрахувати час тривалості фільмів в годинах. У відповідь включити дані про назву фільму, режисера, студію та поле „тривалість в годинах”.
7. Підрахувати кількість фільмів, які надавали фірми постачальники зі статутним капіталом менше 200 млн. грн. Результат помістити в поле „кількість фільмів”.
8. Підрахувати, яка кількість фільмів надавалась кожним постачальником. У відповідь включити дані про назву фірми та поле „кількість фільмів”.
9. Вибрати дані про фільми, які мають максимальну кількість днів прокату. У відповідь включити відомості про назву фільму, назву фірми постачальника та кількість днів прокату.
10. Підрахувати, скількома постачальниками було надано фільми, відомості про які були отримані в результаті виконання запиту 9. У відповідь включити дані про назву фільму та поле „кількість”.

11. Для фільмів, перекладених українською мовою, збільшити тривалість прокату на два тижні.
12. Для всіх фірм постачальників з України збільшити статутний капітал на 1 млн. грн.
13. Занести дані про нову фірму постачальника, всі дані ввести довільні.
14. Занести дані про надання фірмою, відомості про яку були внесені в результаті виконання запиту 13, фільму тривалістю до 100 хв. В поле „дата постачання” ввести дату, на 1 день меншу за дату виконання запиту, всі інші дані ввести довільні.
15. Вилучити відомості, які стосуються останніх 5 днів, про надання фільмів у прокат.

ВАРІАНТ 7

1. Вивести дані про студентів спеціальності МІ. У відповідь включити дані про прізвище студента, його факультет, спеціальність, курс та дату народження.
2. Вивести дані про книги з кількістю сторінок більше 500, що були випущені в 2010 році. У відповідь включити дані про назву книги, автора, рік випуску та кількість сторінок.
3. Вивести дані про книги з кількістю сторінок менше 300, що бралися на термін від 10 до 15 днів. У відповідь включити дані про назву книги, кількість її сторінок, прізвище студента, дату отримання книги та кількість днів, на яку її було отримано.
4. Визначити, які книги брав в бібліотеці певний студент (прізвище студента вводиться з клавіатури під час формування запиту). У відповідь включити дані про прізвище студента, назву та рік випуску взятої книги.
5. Вивести всі дані про студентів, яким у минулому році виповнилося 19 років.

6. Визначити дати повернення отриманих книжок. У відповідь включити дані про дату отримання книги, термін використання та поле „дата повернення”.
7. Підрахувати, яку кількість книг брали студенти першого курсу. Результат помістити в поле „кількість прочитаних книг”.
8. Підрахувати, яку кількість книг прочитали студенти кожної окремої спеціальності. У відповідь включити дані про спеціальність та поле „кількість прочитаних книг”.
9. Вивести дані про книги, термін використання яких був мінімальний. У відповідь включити дані про назву книги, прізвище студента та кількість днів, на яку її брали.
10. Підрахувати, скільки студентів брали книги, відомості про які були отримані в результаті виконання запиту 9. У відповідь включити поле „кількість”.
11. Змінити рік видання на наступний для одної з тих книг, які написав О.В.Погорелов.
12. Для всіх книг, що були видані до 1950 року, зменшити на 1 день термін, на який ці книги були отримані.
13. Занести дані про нову книгу, всі дані ввести довільні.
14. Занести дані про отримання студентами 1-го курсу книги, дані про яку були внесені в результаті виконання запиту 13. В поле „дата отримання книги” внести дату виконання запиту, всі інші дані ввести довільні.
15. Вилучити відомості про отримання студентами книг упродовж поточного місяця.

ВАРІАНТ 8

1. Вивести дані про готелі України. У відповідь включити дані про назву готелю, країну його розміщення, кількість поверхів та кількість вічок.

2. Вивести дані про апартаменти, в яких не менше 2-х спалень та 3-х спальних місць. У відповідь включити дані про тип апартаментів, їх площу, кількість спалень та спальних місць.
3. Вивести дані про готелі, в яких на 3-му поверсі принаймні 4 апартаменти одного типу. У відповідь включити дані про назву готелю, тип апартаментів та кількість апартаментів.
4. Вивести дані про апартаменти в певному готелі (назва готелю вводиться з клавіатури під час формування запиту). У відповідь включити дані про назву готелю, тип апартаментів та кількість спалень в них.
5. Вивести всі дані про готелі, які використовуються не менше ніж 25 років.
6. Вивести дані про апартаменти, у відповідь включити дані про тип апартаментів, їх площу, кількість спалень, кількість спальних місць та середню площу кожної спальні (вважати, що всі спальні апартаменту мають однаковий розмір).
7. Підрахувати кількість апартаментів типу люкс в усіх готелях Франції. Результат помістити в поле „кількість люксів”.
8. Підрахувати загальну кількість апартаментів в кожному готелі. У відповідь включити дані про назву готелю та поле „кількість апартаментів”.
9. Вивести дані про апартаменти, які містяться на найвищому поверсі. У відповідь включити дані про тип апартаментів, назву готелю та поверх.
10. Підрахувати загальну кількість готелів, де наявні апартаменти, відомості про які отримані в результаті виконання запиту 9. У відповідь включити дані про тип апартаментів, їх площу та поле „кількість”.
11. В апартаментах, площа яких перевищує 150 м^2 , збільшити кількість спальних місць на 1.

12. Збільшити на одиницю кількість кожного типу апартаментів, які містяться на других поверхах двозіркових та тризіркових готелів.
13. Занести дані про новий тип апартаментів, всі дані ввести довільні.
14. Для апартаментів, дані про які внесено в результаті виконання запиту 13, занести дані про прибирання. В поле „дата прибирання” ввести дату, на 1 день раніше, ніж дата виконання запиту. Всі інші дані ввести довільні.
15. Вилучити відомості про апартаменти, які придбалися упродовж останніх 3-х днів.

ВАРІАНТ 9

1. Вивести дані про страви, вага яких менше 250 гр. У відповідь включити дані про назву страви, вагу порції та ціну.
2. Вивести дані про салати, приготування яких менше 20 хвилин, а ціна не менше 15 грн. У відповідь включити дані про назву та тип страви, вагу порції та час приготування.
3. Вивести дані про заклади харчування, які отримали ліцензію до 01.01.2000 і мають в меню страви, дорожчі за 100 грн. У відповідь включити дані про назву закладу, дату ліцензування, назву страви та її ціну.
4. Вивести список назв страв, що пропонуються певним закладом (назва закладу вводиться з клавіатури під час формування запиту). У відповідь включити дані про назву закладу, назву страви та прізвище директора.
5. Вивести дані про заклади, які отримали ліцензії в позаминулому році.
6. Вивести дані про страви, у відповідь включити дані про назву та тип страви, час приготування в хвилинах та годинах.
7. Підрахувати, скільки закладів харчування пропонують страви, дорожчі за 450 грн. Результат помістити в поле „кількість дорогих страв”.

8. Підрахувати, яку кількість страв пропонує кожний заклад. У відповідь включити дані про назву закладу, його адресу та поле „кількість страв”.
9. Вивести дані про страви з найменшою ціною. У відповідь включити дані про назву страви, назву закладу харчування та ціну страви.
10. Підрахувати, яка кількість закладів пропонує страви, дані про які були отримані в результаті виконання запиту 9.
11. Зменшити ціну десертів кожного закладу на 1 грн.
12. На всі страви, час приготування яких більше 60 хв., зменшити ціну на 5%.
13. Занести дані про новий заклад харчування, всі дані ввести довільні.
14. Занести дані про замовлення в закладі, дані про який були внесені в результаті виконання запиту 13, страв з часом приготування менше 15 хв. В поле „дата постачання” ввести дату виконання запиту, всі інші дані ввести довільні.
15. Вилучити відомості про замовлення за останні 5 днів.

ВАРІАНТ 10

1. Вивести дані про класи, в яких кількість учнів більше 18. У відповідь включити дані про напрям навчання класу, його номер та кількість учнів.
2. Вивести дані про вчителів першої категорії, зі стажем більше 5 років. У відповідь включити дані про прізвище, адресу, дату народження, стаж та категорію вчителя.
3. Вивести дані про вчителів 3-ї категорії, які навчають предметів з кількістю тижневих годин від 4 до 8. У відповідь включити дані про прізвище вчителя, його стаж та категорію, напрям навчання класу, назву предмету та кількість тижневих годин.
4. Вивести дані про класи, в яких проводить заняття певний вчитель (прізвище вчителя вводиться з клавіатури під час формування

- запиту). У відповідь включити дані про прізвище вчителя, напрям навчання та номер класу.
5. Вивести дані про вчителів, яким виповнилося 40 років на дату виконання запиту.
 6. Визначити кількість робочих тижнів для кожного предмету, виходячи з загальної та щотижневої кількості годин. У відповідь включити дані про предмет, кількість загальних і тижневих годин та поле „кількість робочих тижнів”.
 7. Підрахувати загальну кількість годин у класах з кількістю учнів, меншою, ніж 10. Результат помістити в поле „кількість годин”.
 8. Підрахувати загальну кількість годин занять, які проводить кожен вчитель. У відповідь включити дані про прізвище вчителя, стаж і категорію вчителя та поле „разом годин”.
 9. Вивести дані про вчителів, які навчають предметів з найбільшою кількістю годин занять. У відповідь включити дані про прізвище вчителя, напрям навчання класу та кількість загальних годин.
 10. Підрахувати кількість класів, в яких заняття ведуть вчителі, дані про яких отримано в запиті 9. У відповідь включити дані про прізвище вчителя та поле „кількість класів”.
 11. Збільшити на 1 категорію вчителів, які працюють більше 30 років та мають категорію вище 2-ї.
 12. Збільшити кількість загальних годин на 2 для предметів, заняття з яких проводять вчителі 2-ї категорії.
 13. Занести дані про новий клас, всі дані ввести довільні.
 14. Занести дані про заняття з деякого предмету в класі, дані про який були внесені в результат виконання запиту 13. Заняття мають проводитися вчителями зі стажем більше 35 років, в поле „дата початку занять” ввести дату виконання запиту, всі інші дані занести на власний розсуд.
 15. Вилучити дані про предмети, вивчати які почали упродовж останніх 10 днів.

ВАРІАНТ 11

1. Вивести дані про корми, ціна за одиницю яких більше 500 грн. У відповідь включити назву корму, одиниці виміру та ціну за одиницю.
2. Вивести дані про тварин, які народилися до 01.01.1979 та вагою не менше 100 кг. У відповідь включити дані про назву та сімейство тварини, її дату народження та вагу.
3. Вивести дані про корми, одноразова норма видавання яких більша за 10. У відповідь включити дані про назву корму, назву тварини, дату годування та кількість згодованого.
4. Вивести список тварин, яких годували певним кормом (назва корму вводиться з клавіатури під час формування запиту). У відповідь включити дані про назву та сімейство тварини, назву та виробника корму.
5. Вивести дані про тварин, яким виповнилося 10 років.
6. Вивести дані про тварин, у відповідь включити дані про назву і сімейство тварини, дату її народження, дату годування та вагу тварини в тонах.
7. Підрахувати загальну кількість згодованого корму тваринам вагою менше 100 кг. Результат помістити в поле „кількість згодованого”.
8. Підрахувати, яку кількість тварин годують кожним кормом. У відповідь включити дані про назву корму та його виробника і поле „Кількість тварин”.
9. Вивести дані про тварин, яких годували останніми. У відповідь включити назву тварини, назву корму та дату годування.
10. Підрахувати, якою кількістю кормів годують кожну тварину, дані про яку отримані у відповіді на запит 9. У відповідь включити дані про назву тварини та поле „кількість кормів”.
11. Збільшити дані про вагу кожного зі слонів на 10 кг.
12. Для всіх кормів з ціною за одиницю, меншою 50 грн., збільшити кількість згодованого на 10%.

13. Занести дані про новий корм, всі дані ввести довільні.
14. Занести дані про годування тварин вагою більше 5000 кг. кормом, дані про який були внесені в результаті виконання запиту 13. Дата годування має бути на 1 день раніше за дату виконання запиту, всі інші дані ввести довільні.
15. Вилучити дані про годування тварин, яке відбувалось упродовж останніх трьох днів.

ВАРІАНТ 12

1. Вивести дані про герметичні упаковки. У відповідь включити дані про тип упаковки, її герметичність та вагу.
2. Вивести дані про макарони, виготовлені після 01.01.2011. У відповідь включити дані про назву продукту, тип упаковки та дату виготовлення.
3. Вивести дані про упаковки, в які упаковані продукти з кількістю днів придатності більше 30. У відповідь включити дані про тип упаковки та її вагу, назву продукту та термін придатності.
4. Вивести дані про продукти, які упаковуються в певний тип упаковки (тип упаковки вводиться з клавіатури під час формування запиту). У відповідь включити дані про назву продукту, тип та вагу упаковки.
5. Вивести дані про продукти, виготовлені в позаминулому році.
6. Вивести дані про пакування продуктів, у відповідь включити дані про дату пакування, термін придатності, фірму-пакувальника, вагу упаковки, вагу продукту в упаковці та вагу упакованого продукту.
7. Підрахувати, в яку кількість упаковок упаковано хліб. Результат помістити в поле „кількість упаковок хліба”.
8. Підрахувати, в яку кількість упаковок упаковано кожний продукт. У відповідь включити дані про назву продукту та поле „кількість упаковок”.

9. Вивести дані про упаковки з найбільшим терміном придатності запованих в них продуктів. У відповідь включити дані про тип упаковки, назву продукту та термін придатності.
10. Підрахувати, яка кількість продуктів упакована в упаковки, дані про які отримані у відповідь на запит 9. У відповідь включити дані про тип упаковки та поле „кількість продуктів”.
11. Збільшити вагу герметичних упаковок на 10 г.
12. Збільшити на 2 дні термін придатності всіх продуктів, упакованих в герметичні упаковки.
13. Занести дані про новий продукт, всі дані ввести довільні.
14. Занести дані про вчорашнє пакування в герметичні упаковки товару, дані про який внесено в результаті виконання запиту 13. Дата пакування має бути на 1 день раніше за дату виконання запиту, всі інші дані ввести довільні.
15. Вилучити відомості про пакування товарів за останні 7 днів.

10. Лабораторна робота «Створення звітів в LibreOffice Base»

Практичне завдання

Використовуючи базу даних, створену в попередніх лабораторних роботах, виконати наступні завдання:

ВАРІАНТ 1

1. Створити звіт із даними про континенти, включивши до нього дані про назву континенту, його площу та населення. Встановити наступні елементи в звіті:
 - відсортувати дані в порядку зростання кількості населення;
 - дані розмістити окремими рядками;
2. Створити звіт про релігії, включивши в нього дані про назву релігії, її розповсюдження в світі та століття заснування. Встановити наступні елементи в звіті:
 - відсортувати дані в порядку зростання за кількістю віруючих у світі;
 - дані розмістити окремими стовпчиками;
3. Створити звіт про релігії, розповсюджені на кожному з континентів, включивши в нього назву континенту, назву релігії, кількість храмів та розповсюдження релігії на континенті. Встановити наступні елементи в звіті:
 - згрупувати дані за назвою континенту;
 - в межах групи відсортувати дані в порядку спадання за кількістю храмів;
 - дані в межах групи розмістити окремими стовпчиками;
 - підрахувати загальну кількість храмів за кожною групою та за звітом в цілому.

ВАРІАНТ 2

1. Створити звіт про туристичні агенції, включивши в нього дані про назву агенції, його адресу, статутний капітал, прізвище директора та

дату ліцензування. Встановити наступні елементи в звіті:

- відсортувати дані в порядку зростання статутного капіталу;
- дані розмістити окремими рядками;

2. Створити звіт про тури, включивши до нього дані про назву туру, місце відпочинку, необхідність візи та кількість днів відпочинку.

Встановити наступні елементи в звіті:

- відсортувати дані за кількістю днів відпочинку;
- дані розмістити окремими стовпчиками;

3. Створити звіт про продаж путівок, включивши до нього дані про назву туристичної агенції, назву туру, дату від'їзду, кількість путівок.

Встановити наступні елементи в звіті:

- згрупувати дані за назвою туру;
- в межах групи відсортувати дані за датою від'їзду;
- дані в межах групи розмістити окремими стовпчиками;
- підрахувати загальну кількість проданих путівок за кожною групою та за звітом в цілому.

ВАРІАНТ 3

1. Створити звіт про акціонерів, включивши до нього дані про прізвище акціонера, його адресу проживання та надбавку. Встановити наступні елементи в звіті:

- відсортувати дані в порядку спадання надбавки;
- дані розмістити окремими рядками.

2. Створити звіт про підприємства, включивши до нього дані про назву, прізвище директора, адресу підприємства та дату його заснування.

Встановити наступні елементи в звіті:

- відсортувати дані в порядку зростання дати заснування;
- дані розмістити окремими стовпчиками.

3. Створити звіт про придбання акцій, включивши до нього дані про прізвище акціонера, назву підприємства, дату, кількість та номери придбаних акцій. Встановити наступні елементи в звіті:

- згрупувати дані за підприємствами;
- відсортувати дані в групах за прізвищем акціонера в алфавітному порядку;
- дані в межах групи розмістити окремими стовпчиками;
- підрахувати загальну кількість придбаних акцій за кожною групою та за звітом в цілому.

ВАРІАНТ 4

1. Створити звіт про студентів, включивши до нього дані про прізвище студента, дату його народження, спеціальність, номер групи. Встановити наступні елементи в звіті:
 - відсортувати дані в алфавітному порядку;
 - дані розмістити окремими рядками.
2. Створити звіт про дисципліни, включивши до нього дані про назву дисципліни, лектора та кількість годин на дисципліну. Встановити наступні елементи в звіті:
 - відсортувати дані за прізвищами лекторів в порядку зростання;
 - дані розмістити окремими стовпчиками;
3. Створити звіт про отримані оцінки студентами, включивши до нього дані про прізвище студента, назву дисципліни, дату складання іспиту, отриману оцінку та кількість годин, відведену на навчання предмета. Встановити наступні елементи в звіті:
 - згрупувати дані за прізвищем студента;
 - в межах групи відсортувати дані за датами складання іспиту в порядку зростання;
 - дані в межах групи розмістити окремими стовпчиками;
 - підрахувати загальну кількість годин за кожною групою та за звітом в цілому.

ВАРІАНТ 5

1. Створити звіт про наявність товару, включивши до нього дані про

назву товару, виробника, упаковку та кількість одиниць товару в упаковці. Встановити наступні елементи в звіті:

- відсортувати дані за назвами виробників в алфавітному порядку;
- дані розмістити окремими рядками.

2. Створити звіт про магазини, включивши до нього дані про назву магазину, прізвище власника, адресу та дату відкриття. Встановити наступні елементи в звіті:

- відсортувати дані за датою відкриття в порядку зростання;
- дані розмістити окремими стовпчиками;

3. Створити звіт про продаж товарів, включивши до нього дані про назву товару, назву магазину, дату, кількість та ціну товару. Встановити наступні елементи в звіті:

- згрупувати дані за назвою магазину;
- в межах групи відсортувати дані за датами продажу в порядку зростання;
- дані в межах групи розмістити окремими стовпчиками;
- підрахувати загальну кількість проданих товарів за кожною групою та за звітом в цілому.

ВАРІАНТ 6

1. Створити звіт про фільми, включивши до нього дані про назву фільму, режисера, студію, дату прем'єри та тривалість. Встановити наступні елементи в звіті:

- дані відсортувати за назвами студій в алфавітному порядку;
- дані розмістити окремими рядками;

2. Створити звіт про постачальників, включивши до нього дані про назву фірми, країну, в якій її засновано, та статутний капітал. Встановити наступні елементи в звіті:

- відсортувати дані за розміром статутного капіталу в порядку зростання;
- дані розмістити окремими стовпчиками;

3. Створити звіт про постачання фільмів. Включити до звіту дані про назву фільму, назву фірми постачальника, дату надання фільму в прокат, мову перекладу фільму та кількість днів прокату. Встановити наступні елементи в звіті:
- згрупувати дані за назвою фільму;
 - в межах групи відсортувати в порядку спадання кількості днів прокату;
 - дані в межах групи розмістити окремими стовпчиками;
 - підрахувати, яка кількість постачальників надавала фільми за кожною групою та за звітом в цілому.

ВАРІАНТ 7

1. Створити звіт про студентів, включивши до нього дані про прізвище студента, його факультет, групу, спеціальність та дату народження. Встановити наступні елементи в звіті:
- відсортувати дані за прізвищами студентів в алфавітному порядку;
 - дані розмістити окремими рядками.
2. Створити звіт про книги, включивши до нього дані про назву книги, рік випуску, видавництво та кількість сторінок. Встановити наступні елементи в звіті:
- відсортувати дані за назвами книг в алфавітному порядку;
 - дані розмістити окремими стовпчиками.
3. Створити звіт про користування бібліотекою. Включити до звіту дані про прізвище студента, назву книги, дату її отримання та термін повернення. Встановити наступні елементи в звіті:
- згрупувати дані за назвами книг;
 - в межах групи відсортувати прізвища авторів в алфавітному порядку;
 - дані в межах групи розмістити окремими стовпчиками;

- підрахувати, яка кількість студентів брали книги в межах кожної групи та за звітом в цілому.

ВАРІАНТ 8

1. Створити звіт про готелі, включивши до нього дані про назву готелю, країну його розміщення, дату заснування, кількість поверхів та кількість зірок. Встановити наступні елементи в звіті:
 - відсортувати дані за спаданням кількості зірок;
 - дані розмістити окремими рядками.
2. Створити звіт про апартаменти, включивши до нього дані про тип апартаментів, площу, кількість спалень та кількість спальних місць. Встановити наступні елементи в звіті:
 - відсортувати дані за зростанням площі апартаментів;
 - дані розмістити окремими стовпчиками.
3. Створити звіт про наявність апартаментів в готелях. Включити до звіту дані про назву готелю, тип апартаментів, поверх розміщення, кількість апартаментів на поверсі. Встановити наступні елементи в звіті:
 - згрупувати дані за типом апартаментів;
 - в межах групи відсортувати дані за поверхом розміщення;
 - дані в межах групи розмістити окремими стовпчиками;
 - підрахувати загальну кількість апартаментів в межах кожної групи та за звітом в цілому.

ВАРІАНТ 9

1. Створити звіт про заклади харчування, включивши до нього дані про назву закладу, дату ліцензування та прізвище, ім'я, по батькові директора. Встановити наступні елементи в звіті:
 - відсортувати дані за прізвищами директорів в алфавітному порядку;
 - дані розмістити окремими рядками;

2. Створити звіт про страви, включивши до нього дані про назву, тип, вагу та час приготування страви. Встановити наступні елементи в звіті:
 - відсортувати дані за назвами страв в алфавітному порядку;
 - дані розмістити окремими стовпчиками;
3. Створити звіт про замовлення страв. Включити до звіту дані про назву закладу харчування, назву страви, дату постачання та ціну. Встановити наступні елементи в звіті:
 - згрупувати дані за назвою закладу;
 - в межах групи відсортувати дані за датами постачання в порядку спадання;
 - дані в межах групи розмістити окремими стовпчиками;
 - підрахувати загальну ціну за кожною групою та за звітом в цілому.

ВАРІАНТ 10

1. Створити звіт про вчителів, включивши до нього дані про прізвище, адресу, стаж та категорію вчителів. Встановити наступні елементи в звіті:
 - відсортувати дані за прізвищами в алфавітному порядку;
 - дані розмістити окремими рядками.
2. Створити звіт про групи, включивши до нього дані про напрям навчання, номер та кількість учнів у класі. Встановити наступні елементи в звіті:
 - відсортувати дані в порядку спадання за кількістю учнів;
 - дані розмістити окремими стовпчиками.
3. Створити звіт про проведення занять в різних групах. Включити до звіту прізвище вчителя, напрям навчання, назву предмета, загальну та тижневу кількість годин. Встановити наступні елементи в звіті:
 - згрупувати дані за прізвищами вчителів;

- в межах групи відсортувати дані в порядку спадання кількості тижневих годин;
- дані в межах класу розмістити окремими стовпчиками;
- підрахувати загальну кількість годин за кожним класом та за звітом в цілому.

ВАРІАНТ 11

1. Створити звіт про тварин, включивши до нього дані про назву тварини, її сімейство, дату народження та вагу. Встановити наступні елементи в звіті:
 - відсортувати дані за вагою в порядку зростання;
 - дані розмістити окремими рядками.
2. Створити звіт про корми, включивши до нього назву корму, виробника, одиниці вимірювання та ціну за одиницю товару. Встановити наступні елементи в звіті:
 - відсортувати дані за назвами кормів в алфавітному порядку;
 - дані розмістити окремими стовпчиками.
3. Створити звіт про годування тварин. Включити до звіту дані про назву тварини, назву корму, дату годування та кількість згодованого. Встановити наступні елементи в звіті:
 - згрупувати дані за назвами кормів в алфавітному порядку;
 - в межах групи відсортувати в порядку спадання дати годування;
 - дані в межах групи розмістити окремими стовпчиками;
 - підрахувати загальну кількість згодованого за кожною групою та за звітом в цілому.

ВАРІАНТ 12

1. Створити звіт про упаковки, включивши до нього дані про тип упаковки, її герметичність та вагу. Встановити наступні елементи в звіті:
 - відсортувати дані за вагою упаковок в порядку зростання;

- дані розмістити окремими рядками.
2. Створити звіт про продукти, включивши до нього назву продукту та дату його виготовлення. Встановити наступні елементи в звіті:
- відсортувати дані за назвами продуктів в алфавітному порядку;
 - дані розмістити окремими стовпчиками.
3. Створити звіт про пакування продуктів. Включити до звіту дані про назву продукту, тип упаковки, назву фірми пакувальника, дату пакування, термін придатності. Встановити наступні елементи в звіті:
- згрупувати дані за назвою продукту;
 - в межах групи відсортувати дані в порядку спадання дати пакування;
 - дані в межах групи розмістити окремими стовпчиками;
 - підрахувати, в яку кількість упаковок упаковані продукти за кожною групою та за звітом в цілому.

11. Список використаних джерел

1. Гайна Г. А. Основы проектирования баз данных : навч. посіб. / Г. А. Гайна. – К.: Кондор, 2008. – 200 с.
2. Дейт К. Введение в системы баз данных / К. Дейт. – 8-е изд. – М.; СПб.: Вильямс, 2005. – 440 с.
3. Кузнецов С. Д. Базы данных : вводный курс. – Режим доступа : http://www.citforum.ru/database/advanced_intro/7.shtml – 5.09.2012 р.
4. Модель данных. – Режим доступа : http://ru.wikipedia.org/wiki/Модель_данных – 5.09.2012 р.
5. Операционная система BeOS (БиОС). Особенности версии 5.0. – Режим доступа : http://avs-info.ru/os/os_beos/beos.htmlhttp://avs-info.ru/os/os_beos/beos.html – 08.12.2009 р.
6. Про Національну доктрину розвитку освіти : Указ Президента України // Законодавчі акти України з питань освіти. – К.: Парламент. вид-во, 2004.
7. Рамський Ю. С. Проективання і опрацювання баз даних : посіб. для вчителів / Ю. С. Рамський, Г. Ю. Цибко. – К., 2003. – 136 с.
8. Цибко Г.Ю. Підвищення рівня теоретичної підготовки з інформатики на фізико-математичних факультетах педагогічних вузів : дис. ... канд. пед. наук : 13.00.02 / Г. Ю. Цибко ; НПУ ім. М.П. Драгоманова. – Київ, 1998. – 200 с.
9. Be File System : [материал из Википедии – свободной энциклопедии]. – Режим доступа : <http://ru.wikipedia.org/wiki/BeFS> – 5.09.2012 р.
10. Spotlight: быстрый поиск чего угодно и где угодно. – Режим доступа : <http://iland.com.ua/articles/spotlight-osxtiger-review/>
11. WinFS : [материал из Википедии – свободной энциклопедии]. – Режим доступа : <http://ru.wikipedia.org/wiki/WinFS>