

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КРЕМЕНЧУЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ МИХАЙЛА ОСТРОГРАДСЬКОГО



МЕТОДИЧНІ ВКАЗІВКИ
ЩОДО ВИКОНАННЯ ЛАБОРАТОРНИХ РОБІТ
З НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
«ПРОЕКТУВАННЯ КАДАСТРОВИХ БАЗ ДАНИХ»
ДЛЯ СТУДЕНТІВ УСІХ ФОРМ НАВЧАННЯ
ЗІ СПЕЦІАЛЬНОСТІ 193 – «ГЕОДЕЗІЯ ТА ЗЕМЛЕУСТРІЙ»
ОСВІТНЬОГО СТУПЕНЯ «БАКАЛАВР»

КРЕМЕНЧУК 2019

Методичні вказівки щодо виконання лабораторних робіт з навчальної дисципліни «Проектування кадастрових баз даних» для студентів усіх форм навчання зі спеціальності 193 – «Геодезія та землеустрій» освітнього ступеня «Бакалавр»

Укладач к. т. н., доц. В. І. Козарь

Рецензент к. б. н., доц. Н. П. Гальченко

Кафедра геодезії, землевпорядкування та кадастру

Затверджено методичною радою Кременчуцького національного університету імені Михайла Остроградського

Протокол № ____ від _____ 2019 р.

Голова методичної ради _____ проф. В. В. Костін

ЗМІСТ

Вступ.....	4
1 Перелік тем лабораторних робіт.....	6
Лабораторна робота № 1 Реляційна модель даних.....	6
Лабораторна робота № 2 Зв'язування таблиць.....	8
Лабораторна робота № 3 Операції реляційної алгебри.....	10
Лабораторна робота № 4 Мова запитів за зразком QBE.....	14
Лабораторна робота № 5 Структурована мова запитів SQL.....	16
Лабораторна робота № 6 Створення бази даних Microsoft Access.....	19
Лабораторна робота № 7 Створення таблиць бази даних Microsoft Access.....	20
Лабораторна робота № 8 Робота з даними в режимі конструктора й таблиці СКБД Microsoft Access.....	23
Лабораторна робота № 9 Визначення ключів, індексів та зв'язків таблиць бази даних Microsoft Access.....	24
Лабораторна робота № 10 Залежності між атрибутами.....	27
Лабораторна робота № 11 Метод нормальних форм.....	29
Лабораторна робота № 12 Побудова ER-моделей під час проектування бази даних методом «Сутність-зв'язок».....	32
Лабораторна робота № 13 Формування відношень на підставі діаграм ER-типу.....	36
Лабораторна робота № 14 Пошук, відбір і сортування даних.....	39
Лабораторна робота № 15 Робота із запитамі.....	41
Лабораторна робота № 16 Робота зі звітами.....	43
Лабораторна робота № 17 Робота з екранними формами.....	45
Лабораторна робота № 18 Робота з мультимедійними даними та гіперпосиланнями.....	47
2 Критерії оцінювання знань студентів.....	50
Список літератури.....	51

ВСТУП

Сучасні кадастрові системи – це складні інформаційні системи, що розв’язують різноманітні завдання на всіх адміністративно-територіальних рівнях. Основною складовою кадастрових інформаційних систем є банки інформації та геоінформації. Одним із основних компонентів банку інформації є база чи декілька баз даних і вивчення принципів і методів їх проектування. Набуття умінь і навичок роботи з системами керування базами даних є важливими складовими підготовки бакалаврів з геодезії та землеустрою. Тому до навчального плану підготовки фахівців зі спеціальності «Геодезія та землеустрій» було включено навчальну дисципліну «Проектування кадастрових баз даних».

Методичні вказівки визначають мету, зміст, вимоги до оформлення та порядок захисту лабораторних робіт з навчальної дисципліни «Проектування кадастрових баз даних». Основна мета лабораторних робіт полягає у виробленні у студентів навичок розробки кадастрових баз даних і використання поширених програмних продуктів для їх створення.

Більшість робіт виконуються з використанням обчислювальної техніки, відповідних програмних засобів та іншого приладдя. Студент повинен підготуватися до кожної лабораторної роботи, користуючись підручниками, конспектом лекцій, методичними вказівками, електронною довідкою використовуваного програмного продукту тощо. Перед початком кожного заняття виконується контроль підготовленості студентів до лабораторної роботи. Звітні матеріали з кожної лабораторної роботи оформлюють у вигляді файлів у форматі відповідних програмних продуктів. Під час захисту лабораторних робіт студент повинен відповісти на контрольні питання, що стосуються теоретичних положень і практичних дій під час виконання лабораторної роботи.

Після вивчення навчальної дисципліни студент повинен

знати:

- основні поняття банків інформації та геоінформації;
- основні положення теорії баз даних;
- основні поняття реляційних моделей даних;
- методи проектування банків геоінформації;
- мови запитів;
- основні вимоги до проектування кадастрових баз даних;

уміти:

- використовувати сучасні програмні продукти для створення баз даних, зокрема, Microsoft Access;
- визначати зв'язки між таблицями бази даних;
- виконувати операції над відношеннями бази даних;
- використовувати різні методи проектування баз даних;
- використовувати мови запитів і формувати запити до баз даних;
- формувати звіти на підставі інформації баз даних.

1 ПЕРЕЛІК ТЕМ ЛАБОРАТОРНИХ РОБІТ

Лабораторна робота № 1

Тема. Реляційна модель даних

Мета роботи: навчитись визначати характеристики основних елементів реляційної моделі даних

Апаратно-програмне забезпечення та вихідні дані: комп'ютер, встановлена програма Microsoft Word, приклади відношень баз даних.

Короткі теоретичні відомості

База даних (БД) являє собою сукупність спеціальним чином організованих даних, що зберігаються в пам'яті обчислювальної системи та відображають стан об'єктів і їхніх взаємозв'язків у предметній області. Предметна область – це всі предмети, явища й абстрактні поняття, що складають підґрунтя загального розуміння поданої інформації. Комплекс мовних і програмних засобів, призначений для створення, ведення та спільного використання БД кількома користувачами називають системою керування базами даних (СКБД).

Логічну структуру даних, що зберігаються в базі даних, називають моделлю подання даних. Прийнято розрізняти ієрархічні, мережеві, реляційні, постреляційні, багатомірні, об'єктно-орієнтовані й інші моделі подання даних. Найпоширенішою є реляційна модель даних.

Найважливішим елементом реляційної моделі є відношення – двовимірна таблиця особливого виду. Список імен усіх атрибутів називають схемою відношення (заголовком відношення). У будь-якому відношенні можна визначити атрибут, який однозначно ідентифікує кожний його кортеж. Такий атрибут називають ключем відношення. Основними характеристиками відношення є ступінь і потужність. Ступінь (арність) відношення – це число його атрибутів. Потужність відношення (кардинальне число) – це число його кортежів.

Звичайно, більшість об'єктів і явищ реального світу мають так звану «просторову» складову. Такі дані називають просторовими.

Просторові (географічні) дані – це дані про просторові об'єкти, які містять відомості про їх місцезнаходження, властивості, просторові та непросторові атрибути. Вони складаються з двох взаємопов'язаних частин:

- непозиційна складова – тематичний зміст даних, атрибутивні дані про об'єкт;

- позиційна складова – опис просторового положення об'єкта, тополого-геометричні дані про об'єкт.

Атрибутивні дані є типовими для реляційних СКБД. У сучасних СКБД можуть використовувати такі основні типи даних: числовий, символний (алфавітно-цифровий, текстовий), дата, грошовий (значення валют), символний змінної довжини (для зберігання текстової інформації великої довжини), двійковий (для зберігання графічних об'єктів, аудіо- та відеоінформації, просторової та іншої спеціальної інформації), гіперпосилання (для зберігання посилань на різноманітні ресурси, що знаходяться поза базою даних).

Просторове положення об'єктів можна подати чотирма типами даних: точка (описує образ 0-вимірного об'єкта, наприклад, міста на карті світу), крива (описує образ 1-вимірного об'єкта, наприклад, річки на карті світу, і часто подається за допомогою ліній заданих точками), полігон (описує образ 2-вимірного об'єкта, наприклад країни на карті світу, і часто подається за допомогою багатокутників заданих лініями або точками), геометричний набір (описує образи складних об'єктів, наприклад множина нафтових свердловин, група островів тощо і, в свою чергу, поділяються на три види: складена точка, складена крива і складений полігон). Такі дані в реляційних СКБД можуть бути подані через окремі таблиці.

Порядок виконання лабораторної роботи

1. Відкрити файл вправ «Р_1» (місце розташування – папка Навчальна\Навчальні дані\ПКБД) і ознайомитися з покроковими інструкціями виконання роботи.

2. Визначити типи даних у прикладах відношень баз даних.

3. Визначити основні елементи реляційної моделі даних у прикладах відношень баз даних.

4. Зберегти результати у власній папці як документ Microsoft Word.

Звітні матеріали: файл формату Microsoft Word, збережений у власній папці.

Контрольні запитання

1. Що називається банком даних, і які компоненти він містить?
2. Що називається базою даних?
3. Що називається системою керування базами даних?
4. Назвіть поширені моделі даних.
5. Охарактеризуйте основні типи даних.
6. Дайте визначення реляційної моделі даних і назвіть її складові.
7. Охарактеризуйте основні складові реляційної моделі даних.
8. Назвіть вимоги, за умови виконання яких таблицю можна вважати відношенням.
9. Що таке ступінь відношення?
10. Що таке потужність відношення?

Лабораторна робота № 2

Тема. Зв'язування таблиць

Мета роботи: навчитися визначати відповідність записів таблиць бази даних і встановлювати вид зв'язку.

Апаратно-програмне забезпечення та вихідні дані: комп'ютер, встановлена програма Microsoft Word, приклади відношень баз даних.

Короткі теоретичні відомості

Під час проектування реальних БД інформацію звичайно розміщують у декількох таблицях, які при цьому зв'язані семантикою інформації. У реляційних СКБД для вказування зв'язків між таблицями виконують операцію їх зв'язування. Під час зв'язування двох таблиць виділяють основну та додаткову таблицю. Логічне зв'язування таблиць виконується за допомогою ключа зв'язку. Цей ключ може складатися з одного чи декількох полів, які називають полями зв'язку.

Сутність зв'язування полягає у встановленні відповідності полів зв'язку основної та додаткової таблиці. Для зв'язування таблиць найчастіше використовують ключові поля. Залежно від того, як визначені поля зв'язку основної та додаткової таблиці, між двома таблицями можуть бути встановлені такі види зв'язку: один-до-одного (1:1); один-до-багатьох (1:M); багато-до-багатьох (M:1); багато-до-багатьох (M:N).

Зв'язок 1:1 має місце у випадку, коли одному запису основної таблиці відповідає один запис додаткової таблиці.

Зв'язок 1:M має місце у випадку, коли одному запису основної таблиці відповідає один чи декілька записів додаткової таблиці.

Зв'язок M:1 має місце у випадку, коли одному чи декільком записам основної таблиці ставиться у відповідність один запис додаткової таблиці.

Зв'язок M:N має місце у випадку, коли декільком записам основної таблиці відповідає декілька записів додаткової таблиці.

Порядок виконання лабораторної роботи

1. Відкрити файл вправ «Р_2» (місце розташування – папка Навчальна\Навчальні дані\ПКБД) і ознайомитися з покроковими інструкціями виконання роботи.

2. Дати характеристику зв'язку між прикладами відношень баз даних визначивши:

- які поля є зв'язувальними;
- які дані зберігаються в таблицях;

- які записи основної таблиці відповідають записам додаткової і навпаки;
- вид зв'язку;
- які псевдозаписи утворюються після зв'язування таблиць.

3. Зберегти результати у власній папці як документ Microsoft Word.

Звітні матеріали: файл формату Microsoft Word, збережений у власній папці.

Контрольні запитання

1. З якою метою встановлюються зв'язки між таблицями?
2. Що розуміють під ключем зв'язку?
3. Які види зв'язків можуть встановлюватись між двома таблицями?
4. Охарактеризуйте зв'язок типу 1:1.
5. Охарактеризуйте зв'язок типу 1:M.
6. Охарактеризуйте зв'язок типу M:1.
7. Охарактеризуйте зв'язок типу M:N.

Лабораторна робота № 3

Тема. Операції реляційної алгебри

Мета роботи: навчитися визначати результати виконання основних операцій реляційної алгебри, які виконуються над відношеннями бази даних.

Апаратно-програмне забезпечення та вихідні дані: комп'ютер, встановлена програма Microsoft Word, приклади відношень баз даних.

Короткі теоретичні відомості

У реляційній моделі будь-яке подання даних зводиться до сукупності двовимірних таблиць особливого виду, відомих у математиці як відношення (англ. relation). До відношень можна застосовувати систему операцій, які дозволяють отримувати нову інформацію на підставі тієї, що зберігається. Операції реляційної алгебри виконуються над одним або двома відношеннями. Результатом виконання операцій також є відношення. Мови реляційної алгебри є процедурними (такими, що задають правила виконання запиту), оскільки порядок отримання результуючого відношення визначається шляхом

вказування послідовності виконання реляційних операторів. Оператори складаються з операндів, у ролі яких виступають відношення, і реляційних операцій. Відношення, які беруть участь у операціях, повинні бути сумісними за структурою, що означає сумісність імен атрибутів і типів даних відповідних доменів.

Операції реляційної алгебри можна розділити на дві групи: базові теоретико-множинні операції та спеціальні реляційні операції.

Перша група операцій містить класичні операції теорії множин: об'єднання, перетинання, різниця і добуток.

Операція **об'єднання** – виконується над двома односхемними таблицями-відношеннями. Результатом об'єднання є побудована за тією самою схемою таблиця-відношення, що містить усі кортежі першої таблиці-відношення та всі кортежі другої таблиці-відношення. При цьому кортежі-дублікати в підсумковій таблиці усуваються.

Операція **віднімання** – виконується над двома односхемними таблицями-відношеннями. Результатом є таблиця-відношення, побудована за тією самою схемою, яка містить тільки ті кортежі першої таблиці-відношення, яких немає в складі кортежів другої таблиці-відношення.

Операція **перетинання** – виконується над двома односхемними таблицями-відношеннями. Результатом є таблиця-відношення, побудована за тією самою схемою, яка містить тільки ті кортежі першої таблиці-відношення, що входять також до складу кортежів другої таблиці-відношення.

Операція **добуток** – виконується над таблицями-відношеннями з різними схемами. Результатом є таблиця-відношення, схема якої включає всі поля першої та всі поля другої таблиць. Кортежі результуючої таблиці утворюються шляхом послідовного зчеплення кожного кортежу першої таблиці-відношення з кожним кортежем другої таблиці-відношення. Кількість кортежів результуючої таблиці відповідно дорівнює добутку кількості кортежів першої таблиці на кількість кортежів другої таблиці.

До спеціальних реляційних операцій належать вибірка, проекція, ділення та з'єднання.

Операція **вибірка** (горизонтальна підмножина) – виконується над однією таблицею-відношенням. Результатом є таблиця-відношення тієї самої схеми, що містить підмножину кортежів вихідної таблиці-відношення, які задовольняють умову вибірки;

Операція **проекція** (вертикальна підмножина)– виконується над однією таблицею-відношенням. Результатом є нова таблиця-відношення, схема якої містить тільки деяку підмножину полів вихідної таблиці-відношення. Кожному кортежу вихідної таблиці відповідає кортеж підсумкової таблиці, утворений відповідними значеннями по полях, що ввійшли в підсумкову таблицю-відношення. При цьому в підсумковій таблиці кортежі-дублікати усуваються й тому потужність підсумкової таблиці (кількість кортежів) може дорівнювати або бути меншою, ніж у вихідній;

Операція **ділення** – виконується над двома таблицями-відношеннями, перша з яких називається діленням, а друга дільником. При цьому схема таблиці-дільника повинна складатися з підмножини полів таблиці діленого. Схема результуючої таблиці-відношення містить тільки ті поля таблиці-діленого, яких немає у таблиці-дільнику. Кортежі підсумкової таблиці-відношення утворюються на підставі кортежів першої таблиці (діленого) за значеннями полів, що ввійшли в підсумкову таблицю за умови того, що якщо взяти добуток підсумкової таблиці-відношення й другої таблиці-відношення (дільника), то утворяться відповідні кортежі першої таблиці (діленого);

Операція **З'єднання** – виконується над таблицями-відношеннями з різними схемами. У кожній таблиці-відношенні виділяється поле, по якому буде здійснюватися з'єднання. При цьому обидва поля повинні бути визначені на тому самому домені. Схема результуючої таблиці-відношення містить усі поля першої таблиці та всі поля другої таблиці (як у добутку). Кортежі підсумкової таблиці-відношення утворюються шляхом зчеплення кожного

кортежу з першої таблиці з тими кортежами другої таблиці, значення яких по полю зчеплення однакові.

Окрім цих операцій, існують додаткові операції реляційної алгебри (перейменування, присвоєння, розширення, видалення, оновлення тощо), які дозволяють змінювати імена атрибутів, додавати нові атрибути до відношень, змінювати значення у полях відношень і т. ін. Ці операції аналогічні командам мов програмування.

Очевидно, що розглянуті операції дозволяють виділяти з відношень (таблиць) їх підмножини, за необхідності знову об'єднувати ці підмножини у єдине відношення, оновлювати зміст відношень і подавати їх у різному виді.

Порядок виконання лабораторної роботи

1. Відкрити файл вправ «Р_3» (місце розташування – папка Навчальна\Навчальні дані\ПКБД) і ознайомитися з покроковими інструкціями виконання роботи.

2. Виконати базові та спеціальні реляційні операції над прикладами відношень баз даних.

3. Зберегти результати у власній папці як документ Microsoft Word.

Звітні матеріали: файл формату Microsoft Word, збережений у власній папці.

Контрольні запитання

1. Назвіть основні операції, що можуть виконуватися над відношеннями.
2. Що є теоретичним підґрунтям виконання операцій над відношеннями?
3. Опишіть суть реляційної алгебри.
4. Що є результатом об'єднання відношень?
5. Що є результатом віднімання відношень?
6. Що є результатом перетинання відношень?
7. Що є результатом добутку відношень?
8. Що є результатом вибірки з відношення?
9. Що є результатом проекції відношення?
10. Що є результатом ділення відношень?

11. Що є результатом з'єднання відношень?

Лабораторна робота № 4

Тема. Мова запитів за зразком QBE

Мета роботи: навчитися створювати запити за допомогою мови QBE.

Апаратно-програмне забезпечення та вихідні дані: комп'ютер, встановлена програма Microsoft Word, приклади відношень баз даних.

Короткі теоретичні відомості

Запит являє собою спеціальним чином описану вимогу, яка визначає склад виконуваних над базою даних операцій вибору, видалення або модифікації даних, що зберігаються. Однією з мов описання запитів під час їх підготовки за допомогою СКБД є мова QBE (Query By Example) – мова запитів за зразком. Ця мова передбачає візуальне формування запитів.

Теоретичним підґрунтям мови QBE є реляційне числення зі змінними доменами. Мова QBE дозволяє задавати складні запити до БД шляхом заповнення форми, що пропонується СКБД. Це забезпечує високу наочність і не вимагає задавання алгоритму виконання операцій – достатньо описати зразок очікуваного результату. Мовою QBE можна задавати однотабличні та багатотабличні запити. Результатом виконання запиту є нова таблиця або оновлена вихідна таблиця.

Шаблон для запиту має вигляд таблиці, ім'я та назва полів якої співпадають з іменем та назвами полів відповідної вихідної таблиці. У сучасних СКБД, наприклад, в Access і Visual FoxPro, більшість дій з підготовки запитів за допомогою мови QBE виконуються візуально за допомогою «миші».

Вибірка, вставка, видалення та модифікація даних можуть виконуватися відповідно до умов, що задаються за допомогою логічних виразів. Розрахунки над даними задаються за допомогою арифметичних виразів і створюють у таблицях-відповідях нові поля, що розраховуються.

Для описання змінних в умовах відбору записів використовуються елементи прикладу, які відіграють роль ідентифікатора змінної та задаються

символьно-цифровою послідовністю. Елементи прикладу вказуються обов'язково під час запису логічних умов та зв'язування таблиць у запитах. У шаблонах запиту вони виділяються підкреслюванням.

За допомогою запитів мовою QBE можна виконати такі операції: вибірка даних, розрахунки над даними, вставка нових записів, видалення записів, модифікація даних.

Для визначення необхідності внесення до таблиці-відповіді того чи іншого поля використовується конструкція «P.», що означає «друкувати». Порожні колонки з шаблону запиту можна видаляти. Для вставки, видалення та модифікації даних відповідно використовуються конструкції – «I», «D», «U».

Порядок виконання лабораторної роботи

1. Відкрити файл вправ «P_4» (місце розташування – папка Навчальна\Навчальні дані\ПКБД) і ознайомитися з покроковими інструкціями виконання роботи.

2. Сформувані на мові QBE запити до бази даних, яка містить приклади відношень.

3. Зберегти результати у власній папці як документ Microsoft Word.

Звітні матеріали: файл формату Microsoft Word, збережений у власній папці.

Контрольні запитання

1. Що являє собою запит?
2. Охарактеризуйте мову запитів QBE.
3. Що називається елементом прикладу?
4. Як виконується впорядкування даних у результуючих таблицях?
5. Як задається умова вибору даних?
6. Які конструкції використовуються для виведення, вставки, видалення та модифікації даних?

Лабораторна робота № 5

Тема. Структурована мова запитів SQL

Мета роботи: навчитися створювати запити за допомогою мови SQL.

Апаратно-програмне забезпечення та вихідні дані: комп'ютер, встановлена програма Microsoft Word, приклади відношень баз даних.

Короткі теоретичні відомості

Окрім мови QBE (див. лабораторну роботу № 4), для описання запитів використовується мова SQL (Structured Query Language) – структурована мова запитів. Використання мови SQL передбачає програмування запитів.

Структурована мова запитів SQL базується на реляційному численні зі змінними кортежами. мова SQL призначена для виконання операцій над таблицями (створення, видалення, зміна структури) і над даними таблиць (вибірка, зміна, додавання та видалення), а також деяких супутніх операцій. мова SQL є непроцедурною мовою та не містить операторів керування, організації підпрограм, введення-виведення і т. ін. Тому SQL автономно не використовується і, звичайно, підвантажена у середовище вбудованої мови програмування СКБД (наприклад, FoxPro у СКБД Visual FoxPro чи Visual Basic for Applications у СКБД Microsoft Access).

Основним призначенням мови SQL є підготовка і виконання запитів. У результаті вибірки даних із однієї чи декількох таблиць може бути отримана множина записів, що називається поданням. Подання по суті є таблицею, яка формується в результаті виконання запиту. Тобто, воно є різновидністю запиту, що зберігається. За одними й тими самими таблицями можна побудувати декілька поданнів.

Оператори мови SQL можна умовно розділити на дві підмови: мову визначення даних (Data Definition Language – DDL) і мову маніпулювання даними (Data Manipulation Language – DML).

Під час описання формату операторів прийнято використовувати такі позначення:

- конструкція, яка є обов'язковою, береться в кутові дужки < >;

- конструкція, яка є необов'язковою, береться у квадратні дужки [];
- конструкція, яка передбачає обов'язковий вибір одного із запропонованих операндів, береться у фігурні дужки {};
- конструкція, що розділяє операнди, один з яких потрібно вибрати, позначається за допомогою риски між ними |;

Формат та особливості основних операторів детально описані у спеціальній літературі. Розглянемо формат запису на прикладі найважливішого оператора мови SQL – оператора вибірки записів. Він має формат вигляду:

`SELECT [ALL|DISTINCT]`

`<список даних>`

`FROM <список таблиць>`

`[WHERE <умова вибірки>]`

`[GROUP BY <ім'я стовпчика>[,<ім'я стовпчика>] ...]`

`[HAVING <умова пошуку>]`

`[ORDER BY <специфікація>[,<специфікація>] ...]`

Він дозволяє виконувати вибірку та розрахунки над даними однієї чи декількох таблиць. Результатом виконання оператора є таблиця-відповідь, яка може мати (ALL), або не мати (DISTINCT) рядків, що повторюються. За замовчанням до таблиці вносяться всі рядки. У відборі даних беруть участь записи однієї чи декількох таблиць, перерахованих у списку операнда FROM.

Список даних може містити імена стовпчиків, які беруть участь у запиті, а також виразів над ними (можна використовувати імена стовпчиків, знаки арифметичних операцій, константи, круглі дужки). Під час використання у списках даних імен стовпчиків декількох таблиць для позначення належності стовпчика певній таблиці беруть конструкцію вигляду <ім'я таблиці>.<ім'я стовпчика>.

Операнд WHERE задає умови, яким повинні відповідати записи у результуючій таблиці. Вираз <умова вибірки> є логічним. Його елементами можуть бути імена стовпчиків, операції порівняння, арифметичні операції, логічні зв'язки, дужки, спеціальні функції та ін.

Операнд GROUP BY дозволяє виділити у результуючій множині записів групи. Групою є записи зі значеннями, що збігаються у стовпчиках, перерахованих за ключовими словами GROUP BY.

Операнд HAVING діє спільно з операндом GROUP BY і використовується для додаткового відбору записів під час визначення груп. Правила запису <умова пошуку> аналогічні правилам формування <умова вибірки> операнда WHERE.

Операнд ORDER BY задає порядок упорядкування результуючої множини. Звичайно кожна <специфікація> являє собою пару вигляду: <ім'я стовпчика>[ASC|DESC] і дозволяє впорядковувати значення в стовпчику за зростанням (ASC) або за спаданням (DESC).

Оператор може мати й інші складніші синтаксичні конструкції.

Порядок виконання лабораторної роботи

1. Відкрити файл вправ «P_5» (місце розташування – папка Навчальна\Навчальні дані\ПКБД) і ознайомитися з покроковими інструкціями виконання роботи.

2. Застосовуючи оператори мови SQL виконати операції над прикладами відношень бази даних.

3. Зберегти результати у власній папці як документ Microsoft Word.

Звітні матеріали: файл формату Microsoft Word, збережений у власній папці.

Контрольні запитання

1. Охарактеризуйте мову запитів SQL.
2. Що називається поданням?
3. Як розділяються оператори мови SQL?
4. Опишіть формат і особливості оператора створення таблиці.
5. Опишіть формат і особливості оператора зміни структури таблиці.
6. Опишіть формат і особливості оператора видалення таблиці.
7. Опишіть формат і особливості оператора створення подання.
8. Опишіть формат і особливості оператора видалення подання.

9. Опишіть формат і особливості оператора вибірки записів.
10. Опишіть формат і особливості оператора зміни записів.
11. Опишіть формат і особливості оператора вставки нових записів.
12. Опишіть формат і особливості оператора видалення записів.

Лабораторна робота № 6

Тема. Створення бази даних Microsoft Access

Мета роботи: навчитися створювати бази даних різними способами у середовищі СКБД Microsoft Access.

Апаратно-програмне забезпечення та вихідні дані: комп'ютер, встановлена програма Microsoft Access.

Короткі теоретичні відомості

База даних – це набір відомостей, які належать до визначених тем чи завдань, таких як відстеження замовлень користувачів, облік нерухомості тощо.

Microsoft Access дозволяє керувати всіма відомостями з одного файлу бази даних. Під час створення чи відкривання файлу Microsoft Access, відкривається вікно бази даних. Вікно бази даних є центром керування файлами Access. У цьому вікні користувачі створюють і використовують будь-які об'єкти бази даних чи проекту Access.

Вікно бази даних, як і вікно будь-якої програми, має такі атрибути: рядок заголовка з кнопками системного меню і керування вікном; рядок меню; панель інструментів; статусний рядок.

У Microsoft Access підтримуються два способи створення бази даних. Є можливість створити на підставі шаблону за допомогою майстра готову до використання базу даних з усіма необхідними таблицями, формами і звітами. Це найпростіший спосіб створення бази даних. Є також можливість створити порожню базу даних, а потім додати до неї таблиці, форми, звіти та інші об'єкти. Такий спосіб є найбільш гнучким, але потребує окремого визначення кожного елемента бази даних. В обох випадках залишається можливість у будь-який час змінити та розширити створену базу даних.

Порядок виконання лабораторної роботи

1. Відкрити файл вправ «Р_6» (місце розташування – папка Навчальна\Навчальні дані\ПКБД) і ознайомитися з покроковими інструкціями виконання роботи.

2. Ознайомитися з інтерфейсом та порядком створення баз даних у Microsoft Access.

3. Запустити Microsoft Access.

4. Виконати завдання, наведені у файлі «Р_6».

5. Зберегти результати у власній папці як файл Microsoft Access.

Звітні матеріали: файл формату Microsoft Access, збережений у власній папці.

Контрольні запитання

1. Які завдання можна розв'язувати за допомогою СКБД?
2. Яке призначення мають основні команди меню Файл Microsoft Access?
3. Яке призначення мають основні команди меню Правка Microsoft Access?
4. Яке призначення мають основні команди меню Вид Microsoft Access?
5. Яке призначення мають основні команди меню Вікно Microsoft Access?
6. Які способи створення баз даних є у СКБД Microsoft Access?
7. Опишіть алгоритм створення баз даних за допомогою майстра.
8. Опишіть алгоритм створення баз даних за допомогою конструктора.

Лабораторна робота № 7

Тема. Створення таблиць бази даних Microsoft Access

Мета роботи: навчитися створювати таблиці бази даних різними способами у середовищі СКБД Microsoft Access.

Апаратно-програмне забезпечення та вихідні дані: комп'ютер, встановлена програма Microsoft Access.

Короткі теоретичні відомості

Для зберігання даних у файлі бази даних Microsoft Access використовуються таблиці. Зазвичай таблиця містить набір даних з конкретної теми (наприклад, земельні ділянки чи землевласники). Використання окремої таблиці для кожної теми означає, що відповідні дані збережені тільки один раз, що робить базу даних ефективнішою та зменшує кількість помилок під час уведення даних.

Таблиця бази даних може бути відкрита в режимі конструктора або в режимі таблиці. У режимі конструктора відображається макет об'єкта бази даних. У цьому режимі можна створити нову таблицю або додати, видалити чи настроїти поля наявної таблиці. В режимі таблиці дані виводяться у форматі рядків і стовпчиків. У цьому режимі виконується зміна значень полів, додавання чи видалення даних, пошук даних. У Microsoft Access можна створити порожню таблицю або наповнену даними, використовуючи інформацію із інших джерел.

Microsoft Access дозволяє створити порожню таблицю одним із трьох способів: використання майстра таблиць, використання конструктора таблиць і в режимі введення даних. Незалежно від способу, що використовується для створення таблиці, завжди є можливість використати режим конструктора (після виділення таблиці у вікні бази даних) для зміни макету таблиці, наприклад, для додавання нових полів, зміни типу даних, розміру полів та ін.

Для створення таблиць на підставі зовнішніх даних у Microsoft Access використовується імпорт або зв'язування даних із таблиць інших баз даних Microsoft Access, а також даних з інших програм та файлів інших форматів, наприклад, Microsoft Excel, dBASE, Microsoft FoxPro або Paradox.

Під час імпортування даних створюється їх копія у новій таблиці поточної бази даних Microsoft Access. Вихідна таблиця чи файл при цьому не змінюються. Зв'язування дозволяє зберігати дані в поточному форматі й використовувати їх. Це дозволяє читати і в більшості випадків оновлювати дані у зовнішньому джерелі даних без їх імпорту. Формат зовнішніх джерел даних

при цьому не змінюється, тому файли зовнішніх джерел можна продовжувати використовувати у програмі, якою вони були створені.

Порядок виконання лабораторної роботи

1. Відкрити файл вправ «Р_7» (місце розташування – папка Навчальна\Навчальні дані\ПКБД) і ознайомитися з покроковими інструкціями виконання роботи.

2. Ознайомитися з основними поняттями та порядком створення таблиць у Microsoft Access.

3. Запустити Microsoft Access.

4. Виконати завдання, наведені у документі Microsoft Word «Р_7».

5. Зберегти результати у власній папці як файл Microsoft Access.

Звітні матеріали: файл формату Microsoft Access, збережений у власній папці у власній папці.

Контрольні запитання

1. Які способи створення таблиць є у СКБД Microsoft Access?

2. Що являє собою імпорт таблиць у СКБД Microsoft Access?

3. Що являє собою зв'язування таблиць у СКБД Microsoft Access?

4. Що враховується під час вибору типу даних у полях таблиці Microsoft Access?

5. Назвіть основні типи даних у СКБД Microsoft Access, і де вони використовуються.

6. Опишіть порядок створення таблиць за допомогою майстра таблиць.

7. Опишіть порядок створення таблиць за допомогою конструктора таблиць.

8. Опишіть порядок створення таблиць шляхом введення даних.

9. Опишіть порядок імпортування чи зв'язування таблиць.

Лабораторна робота № 8




Тема. Робота з даними в режимі конструктора й таблиці СКБД Microsoft Access


Мета роботи: навчитися вводити, змінювати та видаляти дані у таблицях бази даних різними способами у середовищі СКБД Microsoft Access.



Апаратно-програмне забезпечення та вихідні дані: комп'ютер, встановлена програма Microsoft Access.


Короткі теоретичні відомості


Зміна значень полів, додавання, видалення та пошук даних виконуються у режимі таблиці. У цьому режимі доступні всі необхідні інструменти для роботи з даними.

Для копіювання чи переміщення вибраного тексту, полів, записів використовуються кнопки **Вирізати**, **Копіювати** та **Вставити**   , розміщені на панелі інструментів.

Для перевірки орфографії у виділених текстових полях чи полях МЕМО використовується кнопка **Орфографія**  на панелі інструментів.

Для друкування та перегляду об'єкта перед друком використовуються кнопки **Друк** та **Попередній перегляд**  .

Для додавання записів використовується кнопка **Новий запис**  на панелі інструментів, під час натиснення якої в кінці таблиці з'являється незаповнений рядок.

Для видалення записів необхідно вибрати будь-яке поле запису таблиці та натиснути кнопку **Видалити запис**  на панелі інструментів.

Порядок виконання лабораторної роботи

1. Відкрити файл вправ «Р_8» (місце розташування – папка Навчальна\Навчальні дані\ПКБД) і ознайомитися з покроковими інструкціями виконання роботи.

2. Ознайомитися з основними засобами обробки даних у таблицях Microsoft Access.

3. Запустити Microsoft Access.

4. Виконати завдання, наведені у файлі вправ «Р_8».
5. Зберегти результати у власній папці як файл Microsoft Access.

Звітні матеріали: файл формату Microsoft Access, збережений у власній папці.

Контрольні запитання

1. Як виконують редагування, додавання та видалення записів у режимі таблиці Microsoft Access?
2. Як виконують перевірку орфографії, друкування записів у режимі таблиці Microsoft Access?
3. Як можна змінювати розміри полів, переміщувати стовпчики і рядки таблиці у режимі таблиці Microsoft Access?
4. Що являє собою фіксування стовпчиків таблиць у СКБД Microsoft Access?
5. Як перейменовують стовпчики у режимі таблиці Microsoft Access?
6. Як виконується збереження записів таблиці Microsoft Access?

Лабораторна робота № 9

Тема. Визначення ключів, індексів та зв'язків таблиць бази даних Microsoft Access

Мета роботи: навчитися встановлювати зв'язки, визначати ключові поля та індекси таблиць бази даних у середовищі СКБД Microsoft Access.

Апаратно-програмне забезпечення та вихідні дані: комп'ютер, встановлена програма Microsoft Access.

Короткі теоретичні відомості

Ключем таблиці бази даних називають одне чи декілька полів, які однозначно ідентифікують кожний запис. Якщо для таблиці визначені ключові поля, то Microsoft Access попереджає дублювання значень або пропуски під час введення даних.

У Microsoft Access виділяють три типи ключових полів: лічильник, простий ключ і складений ключ. Задавання як ключового поля лічильника є

найпростішим. Під час додавання кожного запису до таблиці в поле лічильника автоматично вноситься порядковий номер запису. Поле таблиці можна визначити як простий ключ, якщо воно містить унікальні значення, наприклад, коди чи інвентарні номери. У випадках, коли неможливо гарантувати унікальність значень кожного поля, існує можливість створити ключ, що складається з декількох полів. Прикладом може слугувати база даних, у якій використовуються один основний та декілька допоміжних інвентарних номерів.

Для можливості створення запитів і звітів, у яких виводяться дані з декількох таблиць, необхідно встановити зв'язки між таблицями. Зв'язок між таблицями встановлюється у разі збігу значень у полях різних таблиць. У більшості випадків з ключовим полем однієї таблиці зв'язується відповідне їй поле в іншій таблиці, яке називають зовнішнім ключем. Зв'язувані поля не обов'язково повинні мати однакові імена, але повинні мати однакові типи даних і зміст одного типу. Окрім цього, зв'язувані поля числового типу повинні мати однакові значення властивості «Розмір поля».

Microsoft Access дозволяє встановлювати такі типи зв'язків:

– «один-до-одного», при якому запис до таблиці А може мати не більше одного зв'язаного запису у таблиці В, а запис у таблиці В може мати не більше одного зв'язаного запису в таблиці А;

– «один-до-багатьох», при якому кожному запису в таблиці А може відповідати декілька записів у таблиці В, а запис у таблиці В не може мати більше ніж один відповідний йому запис у таблиці А;

– «багато-до-багатьох», при якому одному запису в таблиці А може відповідати декілька записів у таблиці В, а одному запису в таблиці В може відповідати декілька записів у таблиці А.

Найчастіше використовується зв'язок типу «один-до-багатьох». Зв'язок типу «один-до-одного» використовують рідко, оскільки такі дані можуть бути поміщені в одну таблицю. Його застосовують переважно для розділення широких таблиць з метою захисту даних, а також для збереження даних, що належать до підмножини значень головної таблиці. Зв'язок типу «багато-до-

багатьох» реалізується тільки за допомогою третьої таблиці, ключ якої складається найменшою мірою з двох полів, які є полями зовнішнього ключа у вихідних таблицях.

Для прискорення сортування та пошуку даних можна використати індекси. У таблицях Microsoft Access можна створити індекси на підставі одного чи декількох полів. Ключові поля таблиці індексуються автоматично. Складений індекс створюється, якщо буде часто здійснюватися сортування чи пошук двох і більше полів одночасно. При цьому спочатку здійснюється сортування за першим полем, а якщо у ньому містяться записи зі значеннями, що повторюються, то сортування здійснюється за другим полем, визначеним для даного індексу і т. д.

Порядок виконання лабораторної роботи

1. Відкрити файл вправ «Р_9» (місце розташування – папка Навчальна\Навчальні дані\ПКБД) і ознайомитися з покроковими інструкціями виконання роботи.

2. Ознайомитися з основними засобами визначення ключів, індексів та зв'язків таблиць бази даних Microsoft Access.

3. Запустити Microsoft Access.

4. Виконати завдання, наведені у файлі вправ « Р_9».

5. Зберегти результати у власній папці як файл Microsoft Access.

Звітні матеріали: файл формату Microsoft Access, збережений у власній папці.

Контрольні запитання

1. З якою метою визначають ключові поля в таблицях Microsoft Access?

2. Які типи ключових полів можна визначити у Microsoft Access?

3. Що називається полем лічильника?

4. Що являє собою простий ключ?

5. Що називається складеним ключем?

6. З якою метою зв'язують між собою таблиці бази даних Microsoft Access?

7. Які типи зв'язків можна встановити у Microsoft Access?
8. Що являє собою зв'язок типу «один-до-одного»?
9. Що являє собою зв'язок типу «один-до-багатьох»?
10. Що являє собою зв'язок типу «багато-до-багатьох»?
11. З якою метою виконують індексування таблиць бази даних Microsoft Access?
12. Які типи індексів можна визначити у Microsoft Access?

Лабораторна робота № 10

Тема. Залежності між атрибутами

Мета роботи: навчитися виявляти залежності між атрибутами відношення бази даних.

Апаратно-програмне забезпечення та вихідні дані: комп'ютер, встановлена програма Microsoft Word, приклади відношень баз даних.

Короткі теоретичні відомості

Одним із фундаментальних понять у теорії реляційних баз даних є поняття залежності між атрибутами відношення. Розрізняють такі основні види залежностей між атрибутами: функціональні, транзитивні й багатозначні. Базовим є поняття функціональної залежності, а на його підставі формулюються визначення решти видів залежностей.

Атрибут В функціонально залежить від атрибута А, якщо кожному значенню А відповідає не більше, ніж одне значення В. Математично функціональна залежність В від А визначається записом $A \rightarrow B$. Це означає, що в усіх кортежах з однаковим значенням атрибута А атрибут В буде мати також одне і те саме значення.

Атрибути А і В можуть бути складеними. Отже, у відношенні може існувати часткова або повна залежності атрибута В від атрибута А. Частковою залежністю (частковою функціональною залежністю) називають залежність неключового атрибута В від частини складеного ключа А. Альтернативним

варіантом є повна функціональна залежність, за якої неключовий атрибут В залежить від усіх атрибутів, що належать до складеного ключа А.

Якщо існує функціональна залежність виду $A \rightarrow B$ і $B \rightarrow A$, то між А і В є взаємно однозначна відповідність, або функціональна взаємозалежність. Наявність функціональної взаємозалежності між атрибутами А і В позначається як $A \leftrightarrow B$ або $B \leftrightarrow A$.

Між атрибутами можуть мати місце транзитивна і багатозначна залежності.

Атрибут С залежить від атрибута А транзитивно (існує транзитивна функціональна залежність), якщо для атрибутів А, В, С виконуються умови $A \rightarrow B$ і $B \rightarrow C$, а зворотна залежність відсутня.

Атрибут В багатозначно залежить від А, якщо кожному значенню А відповідає множина значень В, ніяк не зв'язаних з іншими атрибутами з цього відношення. Наявність багатозначної залежності атрибута В від атрибута А позначається може позначатися так: $A \rightarrow \rightarrow B$, $A \Rightarrow B$.

У відношеннях можуть зустрічатися також і взаємно незалежні атрибути. Два чи більше атрибутів називаються взаємно незалежними, якщо жоден з цих атрибутів не є функціонально залежним від інших атрибутів. Відсутність залежності атрибута а від атрибута В можна позначити так: $A \not\rightarrow B$. Випадок, коли $A \not\rightarrow B$ і $B \not\rightarrow A$, можна позначити $A \not\rightarrow B$.

Для кожного відношення існує визначена множина функціональних залежностей. На практиці можуть зустрічатися досить незвичайні випадки функціональних залежностей, відшукування яких є досить складним завданням. Основний спосіб визначення наявності функціональних залежностей – це уважний аналіз семантики атрибутів. Окрім цього, деякі функціональні залежності можуть змінюватися з часом.

Порядок виконання лабораторної роботи

1. Відкрити файл вправ «Р_10» (місце розташування – папка Навчальна\Навчальні дані\ПКБД) і ознайомитися з покроковими інструкціями виконання роботи.

2. Визначити залежності між атрибутами у прикладах відношень баз даних.

3. Зберегти результати у власній папці як документ Microsoft Word.

Звітні матеріали: файл формату Microsoft Word, збережений у власній папці.

Контрольні запитання

1. Які види залежностей між атрибутами можна визначити у відношеннях?

2. Що таке функціональна залежність?

3. Що таке часткова функціональна залежність?

4. Що таке повна функціональна залежність?

5. Що таке функціональна взаємозалежність?

6. Що таке транзитивна функціональна залежність?

7. Що таке багатозначна функціональна залежність?

8. Що таке взаємно незалежні атрибути?

Лабораторна робота № 11

Тема. Метод нормальних форм

Мета роботи: навчитися виконувати нормалізацію відношень бази даних.

Апаратно-програмне забезпечення та вихідні дані: комп'ютер, встановлена програма Microsoft Word, приклади відношень баз даних.

Короткі теоретичні відомості

Метод нормальних форм є класичним методом проектування БД. З використанням цього методу проектування всі дані вносять до однієї чи декількох таблиць, яким надається форма відношень бази даних. Наступним кроком проектування є нормалізація відношень. Це поступовий процес розкладання початкових відношень БД на інші, менші та простіші відношення.

Прийнято виділяти таку послідовність нормальних форм:

– перша нормальна форма (1НФ);

– друга нормальна форма (2НФ);

- третя нормальна форма (3НФ);
- нормальна форма Бойса-Кодда (БКНФ);
- четверта нормальна форма (4НФ);
- п'ята нормальна форма (5НФ);

Будь-якій нормальній формі відповідає деякий визначений набір обмежень. Кожна наступна нормальна форма в деякому розумінні краще попередньої і з переходом від попередньої до наступної нормальної форми властивості попередніх нормальних форм зберігаються.

Відношення знаходиться в 1НФ, якщо всі його атрибути є простими (мають єдине значення). Оскільки вимога першої нормальної форми є базовою вимогою класичної реляційної моделі даних, то вихідний набір відношень уже відповідає цій вимозі. Іншими словами, кожна таблиця в реляційній БД задовольняє умову, відповідно до якої в позиції на перетині кожного рядка і стовпця таблиці завжди знаходиться єдине атомарне значення, і ніколи не може бути множини таких значень. Тобто ненормалізовані таблиці (таблиці, що містять повторювані групи), навіть не допускаються в реляційній БД. Отже, будь-яке вихідне відношення знаходиться в першій нормальній формі.

Відношення знаходиться в 2НФ, якщо воно знаходиться в 1НФ і кожен його неключовий атрибут функціонально повно залежить від первинного ключа. Перехід до 2НФ дозволяє позбутися явної надлишковості даних. Для переведення відношення в 2НФ необхідно, використовуючи операцію проєкції, розкласти його на декілька відношень так:

- побудувати проєкцію без атрибутів, що знаходяться в частковій функціональній залежності від первинного ключа;
- побудувати проєкції на частини первинного ключа й атрибути, що залежать від цих частин.

Відношення знаходиться в 3НФ, якщо воно знаходиться в 2НФ і кожен його неключовий атрибут нетранзитивно залежить від первинного ключа. Тобто, відношення знаходиться в 3НФ, якщо всі неключові атрибути відношення взаємно незалежні та повністю залежать від первинного ключа.

Транзитивні залежності викликають надлишкове дублювання інформації у відношенні. Для переведення відношення в ЗНФ необхідно ліквідувати транзитивні залежності, використовуючи операцію проєкції на атрибути, що є причиною транзитивних залежностей. При цьому отримують кілька відношень, кожне з яких знаходиться в ЗНФ.

Відношення знаходиться в нормальній формі Бойса-Кодда (НФБК), якщо воно знаходиться в ЗНФ, і в ньому відсутні залежності ключів від неключових атрибутів, тобто, якщо будь-який атрибут, від якого функціонально залежить деякий інший атрибут, є ключем. Перехід до БКНФ здійснюється завдяки ліквідації функціональної залежності частини складеного ключа від неключового атрибута. При цьому вихідне відношення розкладається на два відношення, які знаходяться у БКНФ, і з'єднання яких за визначеним атрибутом дасть вихідне відношення.

На практиці побудова схем відношень, які знаходяться в ЗНФ, у більшості випадків є достатньою і приведенням до них процес проектування реляційної БД завершується. Але якщо у відношенні є залежність атрибутів складеного ключа від неключових атрибутів, то переходять до підсиленої ЗНФ, яку називають нормальною формою Бойса-Кодда.

Порядок виконання лабораторної роботи

1. Відкрити файл вправ «Р_11» (місце розташування – папка Навчальна\Навчальні дані\ПКБД) і ознайомитися з покроковими інструкціями виконання роботи.
2. Сформулювати вихідне відношення бази даних.
3. Виконати приведення відношення до ЗНФ.
4. Зберегти результати у власній папці як документ Microsoft Word.

Звітні матеріали: файл формату Microsoft Word, збережений у власній папці.

Контрольні запитання

1. Які відношення бази даних називаються вихідними?
2. Охарактеризуйте нормальні форми.

3. У чому суть процесу нормалізації відношень?
4. Як виконується нормалізація відношень?
5. Дайте визначення першої нормальної форми.
6. Дайте визначення другої нормальної форми.
7. Дайте визначення третьої нормальної форми.
8. Дайте визначення підсиленої третьої нормальної форми.

Лабораторна робота № 12

Тема. Побудова ER-моделей під час проектування бази даних методом «Сутність-зв'язок»

Мета роботи: навчитися визначити ступінь зв'язку та клас належності сутностей і побудувати діаграму ER-типу.

Апаратно-програмне забезпечення та вихідні дані: комп'ютер, встановлена програма Microsoft Word, приклади відношень баз даних.

Короткі теоретичні відомості

Метод «Сутність-зв'язок» називають також методом «ER-діаграм». ER – аббревіатура від слів Entity (сутність) і Relation (зв'язок). У рамках методу предметна область поділяється на сутності, які характеризуються своїми атрибутами і взаємодіють за допомогою зв'язків. Основними поняттями методу «Сутність-зв'язок» є сутність, атрибут сутності, ключ сутності, зв'язок між сутностями, ступінь зв'язку, клас належності екземплярів сутності, діаграми ER-екземплярів, діаграми ER-типу.

Сутність являє собою об'єкт, інформація про який зберігається в базі даних. Це об'єкти, які можуть незалежно існувати фізично і концептуально. Екземпляри сутності відрізняються один від одного й однозначно ідентифікуються. Назвами сутностей є, зазвичай іменники, наприклад, ВУЛИЦЯ, БУДІВЛЯ, ВЛАСНИК.

Сутності описуються атрибутами. Атрибут являє собою певну властивість сутності. Це поняття аналогічне поняттю атрибута у відношенні.

Наприклад, атрибутами сутності ВУЛИЦЯ можуть бути її НАЗВА, ДОВЖИНА, ШИРИНА, ТИП ПОКРИТТЯ, НОМЕР_ЛІНІЇ, ВИСОТА та ін.

У просторових базах даних розрізняють однозначні та багатозначні атрибути. Наприклад, НАЗВА – однозначний атрибут сутності ВУЛИЦЯ, тому що одна вулиця не може мати дві чи більше назв. У той же час сутність вулиця має атрибут НОМЕР_ЛІНІЇ, який використовується для описання геометричного місця розташування екземплярів сутності – окремих вулиць. Будь-яка вулиця може займати в просторі два різні геометричні місця (наприклад, поділитися на дві частини іншою дорогою чи річкою) і атрибут НОМЕР_ЛІНІЇ стає багатозначним. Припустимо, що потрібно зберігати інформацію про висоту вулиці над рівнем моря. Оскільки ВИСОТА може змінювати своє значення в межах одного екземпляра сутності ВУЛИЦЯ, цей атрибут також потрібно моделювати як багатозначний. Багатозначними є всі атрибути, які описують геометричні властивості об'єктів, тобто позиційну складову просторових (географічних) даних.

Ключ сутності – атрибут або набір атрибутів, який використовується для однозначної ідентифікації екземпляра сутності. Як видно з визначення, поняття ключа сутності аналогічне поняттю ключа відношення.

Зв'язок двох чи більше сутностей передбачає залежність між атрибутами цих сутностей. За допомогою зв'язків сутності взаємодіють чи з'єднуються між собою. Назва зв'язку зазвичай зображується дієсловом. Наприклад, ВУЛИЦЯ *МАЄ У СКЛАДІ* БУДІВЛІ, *ВЛАСНИК МАЄ ПРАВО НА* БУДІВЛЮ тощо.

Характеристиками зв'язку між сутностями є ступінь зв'язку і клас належності. Ступінь зв'язку між двома сутностями може бути таких видів: 1:1, 1:М, М:N. Клас належності може бути обов'язковим і необов'язковим. Клас належності сутностей є обов'язковим, якщо всі екземпляри цієї сутності обов'язково беруть участь у розглядуваному зв'язку. В іншому випадку клас належності сутності є необов'язковим.

З метою підвищення зручності проектування для зображення сутностей, екземплярів сутностей і зв'язків між ними використовуються такі графічні засоби, як діаграми ER-екземплярів і діаграми ER-типу (ER-діаграми).

Діаграми ER-екземплярів показують, які конкретно екземпляри різних сутностей пов'язані між собою (рис. 12.1).

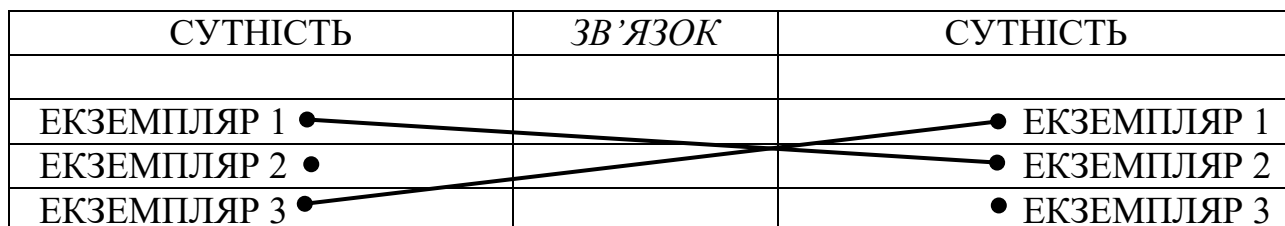


Рис. 12.1 – Діаграми ER-екземплярів

Діаграми ER-типу показують, які сутності зв'язані між собою. Також на них відображаються ступінь зв'язку сутностей і клас їх належності. На ER-діаграмах сутності зображуються прямокутниками, атрибути – овалами, назви зв'язків – ромбами або шестикутниками, які розміщуються на лініях, що зв'язують окремі сутності між собою. Обов'язкова участь екземплярів певної сутності у зв'язку позначається блоком з крапкою всередині, який є суміжним з блоком цієї сутності. За необов'язкової участі екземплярів сутності у зв'язку додатковий блок до прямокутника сутності не добудовується, а точка розміщується на лінії зв'язку. Символи на лінії зв'язку біля ромба, яким його позначено, вказують на ступінь зв'язку. Ключові атрибути позначають підкреслюванням, а багатозначні – подвійним овалом (рис. 12.2).

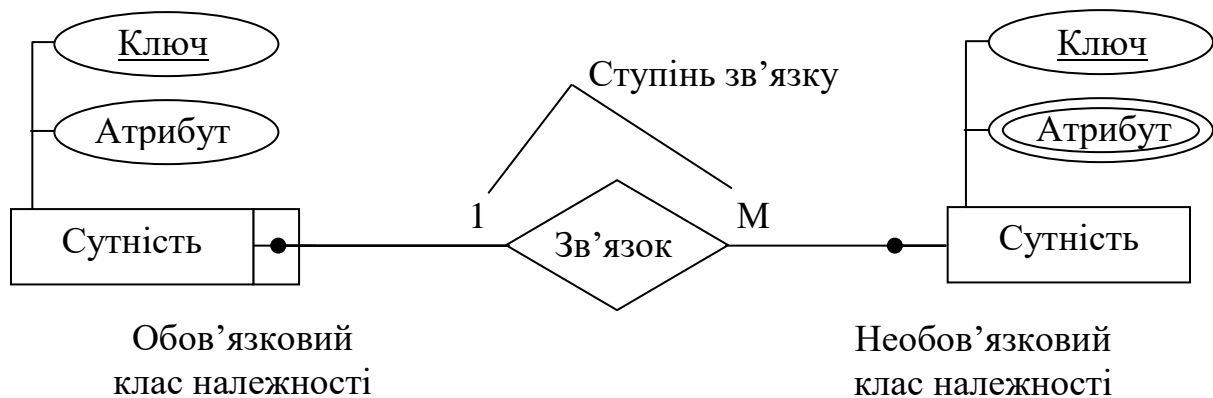


Рис. 12.2 – Елементи ER-діаграм

Ступінь зв'язку є характеристикою зв'язку між сутностями, яка може бути таких типів: 1:1, 1:М, М:1, М:N. Клас належності (КН) може бути обов'язковим і необов'язковим. Клас належності сутностей є обов'язковим, якщо всі екземпляри цієї сутності обов'язково беруть участь у розглядуваному зв'язку. В іншому випадку клас належності сутності є необов'язковим.

Порядок виконання лабораторної роботи

1. Відкрити файл вправ «Р_12» (місце розташування – папка Навчальна\Навчальні дані\ПКБД) і ознайомитися з покроковими інструкціями виконання роботи.

2. На підставі аналізу діаграми ER-екземплярів визначити ступінь зв'язку та клас належності сутностей.

3. Побудувати діаграму ER-типу.

4. Зберегти результати у власній папці як документ Microsoft Word.

Звітні матеріали: файл формату Microsoft Word, збережений у власній папці.

Контрольні запитання

1. Назвіть основні поняття методу «Сутність-зв'язок».
2. Що називається сутністю?
3. Що називається атрибутом?
4. Що називається ключем сутності?
5. Що називається зв'язком сутності?

6. Які графічні засоби використовуються для підвищення зручності проектування за допомогою методу «Сутність-зв'язок»?
7. Що показують діаграми ER-екземплярів?
8. Що показують діаграми ER-типу?
9. Коли клас належності сутності є обов'язковим, а коли необов'язковим?
10. Якими засобами зображають сутності, атрибути, зв'язки, ступінь зв'язку, клас належності на діаграмах ER-типу?

Лабораторна робота № 13

Тема. Формування відношень на підставі діаграм ER-типу

Мета роботи: навчитися формувати структуру відношень бази даних за розробленою ER-моделлю.

Апаратно-програмне забезпечення та вихідні дані: комп'ютер, встановлена програма Microsoft Word, приклади відношень баз даних.

Короткі теоретичні відомості

У процесі проектування баз даних після виявлення сутностей, зв'язків між ними, визначення ключа для кожної сутності та побудови діаграм ER-типу приступають до формування набору відношень. Цей етап є одним із вузлових етапів процесу проектування. Він виконується на підставі аналізу ER-діаграм і ряду правил.

Правила формування відношень ураховують ступінь зв'язку між сутностями, клас належності екземплярів сутності та вид атрибутів. Під час формування відношень просторових баз даних на підставі діаграм ER-типу використовують вісім правил.

Правило 1. Якщо ступінь зв'язку 1:1 і клас належності обох сутностей обов'язковий, то формується одне відношення. Первинним ключем цього відношення може бути ключ будь-якої з двох сутностей.

Правило 2. Якщо ступінь зв'язку 1:1 і клас належності однієї сутності обов'язковий, а другої – необов'язковий, то під кожен із сутностей формується по одному відношенню з первинними ключами, які є ключами відповідних

сутностей. Далі до відношення, сутність якого має обов'язковий клас належності, додається як атрибут ключ сутності з необов'язковим класом належності.

Правило 3. Якщо ступінь зв'язку 1:1 і клас належності обох сутностей необов'язковий, то необхідно використовувати три відношення. Два відношення відповідають зв'язуванню сутностям, ключі яких є первинними ключами цих відношень. А третє відношення є зв'язуючим між першими двома. Воно обов'язково містить ключові атрибути обох сутностей і може містити інші атрибути, що характеризують цей зв'язок. Ключ зв'язуючого відношення є складеним із ключових атрибутів відношень, які воно зв'язує.

Правило 4. Якщо ступінь зв'язку 1:М і клас належності М-зв'язної сутності обов'язковий, то під кожен із сутностей формується по одному відношенню з первинними ключами, які є ключами відповідних сутностей. Далі до відношення, що відповідає М-зв'язній сутності додається як атрибут ключ 1-зв'язної сутності.

Правило 5. Якщо ступінь зв'язку 1:М і клас належності М-зв'язної сутності необов'язковий, то необхідно формувати три відношення. Два відношення відповідають зв'язуванню сутностям, ключі яких є первинними ключами цих відношень. А третє відношення є зв'язувальним між першими двома. Воно обов'язково містить ключові атрибути обох сутностей і може містити інші атрибути, що характеризують цей зв'язок. Ключ зв'язувального відношення є складеним із ключових атрибутів відношень, які воно зв'язує.

Правило 6. Якщо ступінь зв'язку М:М, то незалежно від класу належності формується три відношення. Два відношення відповідають зв'язуванню сутностям, ключі яких є первинними ключами цих відношень. А третє відношення є зв'язувальним між першими двома. Воно обов'язково містить ключові атрибути обох сутностей і може містити інші атрибути, що характеризують цей зв'язок. Ключ зв'язувального відношення є складеним із ключових атрибутів відношень, які воно зв'язує.

Правило 7. Для багатозначних атрибутів створюються окремі відношення. Кожне таке відношення має два стовпчики: стовпчик, який відповідає багатозначному атрибуту, та стовпчик, який відповідає ключу сутності, що має цей багатозначний атрибут. Разом багатозначний атрибут і ключ сутності утворюють ключ нового відношення.

Правило 8. Просторові (геометричні) дані (точка, крива та полігон) подаються як окремі відношення. Такого самого підходу вимагає атрибут висота, який описує просторове положення об'єкта відносно рівня моря. Кожному із цих атрибутів відповідає одне відношення: точка, лінія, багатокутник і висота (рис. 13.1).



Рис. 13.1 – Схема подання точок, ліній, багатокутників і висоти

Порядок виконання лабораторної роботи

1. Відкрити файл вправ «Р_13» (місце розташування – папка Навчальна\Навчальні дані\ПКБД) і ознайомитися з покроковими інструкціями виконання роботи.
2. На підставі аналізу вихідних даних визначити ступінь зв'язку та клас належності сутностей.
3. Визначити правило, яке необхідно застосовувати у розглядуваному випадку.

4. Зобразити графічно схеми сформованих відношень та зв'язок між ними.

5. Зберегти результати у власній папці як документ Microsoft Word.

Звітні матеріали: файл формату Microsoft Word, збережений у власній папці.

Контрольні запитання

1. Що враховують правила формування відношень?
2. Сформулюйте 1-ше правило формування відношень.
3. Сформулюйте 2-ге правило формування відношень.
4. Сформулюйте 3-тє правило формування відношень.
5. Сформулюйте 4-те правило формування відношень.
6. Сформулюйте 5-те правило формування відношень.
7. Сформулюйте 6-те правило формування відношень.

Лабораторна робота № 14

Тема. Пошук, відбір і сортування даних

Мета роботи: навчитися знаходити необхідні дані та виконувати сортування записів у таблицях бази даних.

Апаратно-програмне забезпечення та вихідні дані: комп'ютер, встановлена програма Microsoft Access.

Короткі теоретичні відомості

У Microsoft Access можна виконувати пошук даних за номером запису, за значенням і за зразком (за допомогою підстановочних знаків). Підстановочні знаки використовуються як прототипи для інших символів під час зазначення зразка пошуку, коли відома тільки частина значення або необхідно знайти значення, що починається з конкретної букви чи відповідає визначеному шаблону.

Для перегляду тільки визначених записів таблиці найчастіше використовується фільтрація. Фільтр – це набір умов, які використовуються для відбору множини записів або для сортування записів таблиці. У Microsoft

Access існують чотири способи відбору записів за допомогою фільтрів: фільтр за виділеним фрагментом, звичайний фільтр, поле «Фільтр для» і розширений фільтр.

Рядки в таблиці можна також упорядкувати згідно із змістом одного чи декількох стовпчиків. Виконання впорядкування записів для таблиці допускається також і після виконання фільтрації. Під час виконання впорядкування записів є можливість виконати просте впорядкування, згідно з яким записи впорядковуються за зростанням, або за зменшенням. Під час зазначення порядку впорядкування записів у вікні розширеного фільтру є можливість виконати складне впорядкування, за якого допускається за одними полями впорядкування за зростанням, а за іншими – за зменшенням.

Порядок виконання лабораторної роботи

1. Відкрити файл вправ «P_14» (місце розташування – папка Навчальна\Навчальні дані\ПКБД) і ознайомитися з покроковими інструкціями виконання роботи.

2. Ознайомитися з основними засобами пошуку, фільтрування та впорядкування даних у таблицях Microsoft Access.

3. Запустити Microsoft Access.

4. Виконати завдання, наведені файл вправ «P_14».

5. Зберегти результати у власній папці як файл Microsoft Access.

Звітні матеріали: файл формату Microsoft Access, збережений у власній папці.

Контрольні запитання

1. Як можна виконати пошук даних у Microsoft Access?
2. У яких випадках застосовують підстановочні знаки?
3. Які підстановочні знаки застосовують для пошуку даних у Microsoft Access?
4. Що називається фільтром у СКБД Microsoft Access?
5. Які існують способи відбору даних за допомогою фільтрів у СКБД Microsoft Access?
6. Як можна впорядкувати дані у таблиці Microsoft Access?

7. Опишіть процес пошуку даних за номером запису.
8. Опишіть процес пошуку даних за зразком.
9. Опишіть процес відбору даних за допомогою «Фільтру за виділенням».
10. Опишіть процес відбору даних за допомогою звичайного фільтру.
11. Опишіть процес відбору даних за допомогою розширеного фільтру.
12. Опишіть процес відбору даних за допомогою поля «Фільтр для».
13. Опишіть процес сортування записів.
14. Опишіть процес редагування фільтрів.

Лабораторна робота № 15

Тема. Робота із запитам

Мета роботи: навчитись створювати запити до бази даних Microsoft Access.

Апаратно-програмне забезпечення та вихідні дані: комп'ютер, встановлена програма Microsoft Access.

Короткі теоретичні відомості

Переглядати, аналізувати та змінювати дані з декількох таблиць бази даних можна за допомогою запитів. Найчастіше використовується запит на вибірку, за якого дані, що задовольняють умови відбору, вибираються з однієї чи декількох таблиць і виводяться у визначеному порядку. Окрім цього, у Microsoft Access можна створювати запит на заміну, який за одну операцію вносить зміни у декілька записів.

Для створення запитів використовується майстер або конструктор запитів. Майстер запитів автоматично виконує основні дії залежно від відповідей користувача на поставлені запитання. У режимі конструктора всі параметри запиту користувач визначає самостійно.

Запити зручно використовувати для об'єднання чи виконання операцій над даними, що розміщені в декількох таблицях. Під час додавання до запиту декількох таблиць необхідно впевнитися, що вони з'єднані між собою. Під час виконання запиту Microsoft Access виконує перевірку на наявність в об'єднаних

полях відповідних значень. За наявності відповідних записів вони об'єднуються і відображаються в результатах запиту як один запис, а в іншому випадку в результатах запиту нічого не відображається.

У запиті можна виконати деякі розрахунки, наприклад, знайти суму чи середнє за значеннями одного поля, перемножити значення двох полів тощо. Вирази, які визначають поля, що розраховуються, створюються за допомогою майстра простих записів або для згрупованих записів вводяться користувачем у рядок «Групова операція» бланка запиту, у якому допускається вибір статистичних функцій для розрахунків. За допомогою виразу, який вводиться до порожньої комірки «Поле» на бланку запитів, можна створити поле, що розраховується.

Запити дозволяють виводити на екран результати розрахунків таких типів:

- розрахунок суми, розрахунок середнього, підрахунок кількості значень, визначення мінімального чи максимального значення, розрахунок середньоквадратичного відхилення та дисперсії;
- операції за допомогою виразів, які визначаються користувачем над числовими та символічними значеннями, або над значеннями дат;
- визначення умов відбору записів у запиті або для визначення запитів, над якими виконуються операції;
- для оновлення даних у запиті на оновлення.

Порядок виконання лабораторної роботи

1. Відкрити файл вправ «P_15» (місце розташування – папка Навчальна\Навчальні дані\ПКБД) і ознайомитися з покроковими інструкціями виконання роботи.
2. Ознайомитися з основними поняттями та порядком створення запитів до бази даних Microsoft Access.
3. Запустити Microsoft Access.
4. Виконати завдання, наведені файл вправ «P_15».
5. Зберегти результати у власній папці як файл Microsoft Access.

Звітні матеріали: файл формату Microsoft Access, збережений у власній папці.

Контрольні запитання

1. З якою метою використовуються запити у базах даних?
2. Які типи запитів можна створювати за допомогою Microsoft Access?
3. Які способи створення запитів є в СКБД Microsoft Access?
4. Які типи об'єднання таблиць існують у СКБД Microsoft Access?
5. Які типи розрахунків можна виконувати за допомогою запитів у СКБД Microsoft Access?
6. Опишіть процес створення запиту на основі фільтра.
7. Опишіть процес створення запиту за допомогою майстра.
8. Опишіть процес створення запиту без допомоги майстра.
9. Як можна змінити структуру запиту в СКБД Microsoft Access?

Лабораторна робота № 16

Тема. Робота зі звітами

Мета роботи: навчитись створювати звіти за допомогою СКБД Microsoft Access.

Апаратно-програмне забезпечення та вихідні дані: комп'ютер, встановлена програма Microsoft Access.

Короткі теоретичні відомості

Звіт – це гнучкий і ефективний засіб для організації даних під час їх роздрукування. За допомогою звітів можна вивести необхідні відомості у тому вигляді, у якому вимагається. У звіті можна додати емблему чи рисунок, виконати сортування записів, вивести сумарне значення, зобразити дані на діаграмі тощо. Більшість відомостей у звіті вибирається з базових таблиці чи запиту, які є джерелом даних для звіту.

Для створення зв'язку між звітом і його вихідними даними використовуються елементи керування. Елементи керування – це об'єкти звіту, які слугують для виведення даних на екран, виконання розрахунків, або

оформлення звіту. У Microsoft Access елементи керування можуть бути зв'язаними, вільними або розраховуваними. Зв'язаний елемент керування приєднаний до поля базової таблиці чи запиту і використовується для відображення значень з полів бази даних. Для вільного елемента керування джерела даних не існує і вони використовуються для виведення на екран тексту, ліній, прямокутників і рисунків. Для розраховуваних елементів керування як джерело даних використовується вираз, у якому можуть бути використані дані з полів базової таблиці чи запиту, а також дані іншого елемента керування звіту.

Звіти можна відкривати у режимах конструктора та перегляду. Можливі декілька варіантів створення звіту. Найпростішим варіантом є створення звіту за допомогою команди **Автозвіт**. При цьому створюються нескладні звіти, що містять усі поля джерела даних (таблиці чи запиту). Швидко створити звіт можна також у режимі **Майстра**. З використанням **Майстра** можна вибрати як джерело даних довільне число таблиць (запитів), додавши до звіту потрібні поля. Режим **Конструктора** є найпотужнішим, але і найбільш трудомістким засобом розробки звіту. Для скорочення часу розробки доцільно спочатку використати один із способів, описаних вище, а за допомогою конструктора допрацювати й змінити звіт.

Порядок виконання лабораторної роботи

1. Відкрити файл вправ «Р_16» (місце розташування – папка Навчальна\Навчальні дані\ПКБД) і ознайомитися з покроковими інструкціями виконання роботи.
2. Ознайомитися з основними поняттями та порядком створення звітів до бази даних Microsoft Access.
3. Запустити Microsoft Access.
4. Виконати завдання, наведені файл вправ «Р_16».
5. Зберегти результати у власній папці як файл Microsoft Access.

Звітні матеріали: файл формату Microsoft Access, збережений у власній папці.

Контрольні запитання

1. Що називають звітом у СКБД Microsoft Access?
2. Що називають елементами керування у СКБД Microsoft Access?
3. Які типи елементів керування є в СКБД Microsoft Access?
4. Які способи створення звітів є в СКБД Microsoft Access?
5. Опишіть процес створення звіту за допомогою команди Автозвіт.
6. Опишіть процес створення звіту за допомогою майстра.
7. Опишіть процес створення звіту за допомогою конструктора.

Лабораторна робота № 17

Тема. Робота з екранними формами

Мета роботи: навчитись створювати екранні форми СКБД Microsoft Access.

Апаратно-програмне забезпечення та вихідні дані: комп'ютер, встановлена програма Microsoft Access.

Короткі теоретичні відомості

Форма являє собою об'єкт бази даних, який зазвичай використовується для відображення даних у базі даних. Форму можна також використовувати як кнопку форму, що відкриває інші форми або звіти бази даних, а також як користувальницьке діалогове вікно для введення даних.

Більшість форм є приєднаними до однієї або декількох таблиць і запитів з бази даних. Джерелом даних форми є поля в базових таблицях і запитах. Приєднана форма отримує дані з базового джерела записів. Дані надходять із полів у базовому джерелі записів. Інші відомості, що виводяться у формі, такі як заголовок, дата й номери сторінок, зберігаються в макеті форми. Графічні елементи, такі як лінії й прямокутники, також зберігаються в макеті форми. Зв'язок між формою і її джерелом записів створюється за допомогою графічних об'єктів, які називають елементами керування. Найчастіше використовуваним типом елементів керування є поле. У полях відображаються дані з таблиці й вводяться дані до таблиці.

Форми можна відкривати у різних режимах: конструктора, форми, таблиці та ін. Можливі декілька варіантів створення форми. Найпростішим варіантом створення форми є застосування команди **Автоформа**. При цьому створюються нескладні форми, що містять усі поля джерела даних (таблиці чи запити). З використанням **Майстра** можна вибрати як джерело даних довільне число таблиць (запитів) додавши до форми потрібні поля. Майстер виводить на екран питання й створює форму на підставі відповідей користувача. Режим **Конструктора** є найпотужнішим, але і найбільш трудомістким засобом розробки форм.

Порядок виконання лабораторної роботи

1. Відкрити файл вправ «Р_17» (місце розташування – папка Навчальна\Навчальні дані\ПКБД) і ознайомитися з покроковими інструкціями виконання роботи.
2. Ознайомитися з основними поняттями та порядком створення екранних форм у Microsoft Access.
3. Запустити Microsoft Access.
4. Виконати завдання, наведені файл вправ «Р_17».
5. Зберегти результати у власній папці як файл Microsoft Access.

Звітні матеріали: файл формату Microsoft Access, збережений у власній папці.

Контрольні запитання

1. З якою метою використовуються форми у базах даних Microsoft Access?
2. Що є джерелом даних для форм Microsoft Access?
3. Які здійснюється зв'язок між формою та її джерелом записів?
4. Які варіанти створення форм є в Microsoft Access?
5. Які області містить форма в режимі конструктора?
6. Які вікна використовуються під час розробки форм у режимі конструктора?
7. Які інструменти містить панель елементів Microsoft Access?
8. Опишіть процес створення форми за допомогою команди Автоформа.
9. Опишіть процес створення форми за допомогою майстра.

10. Опишіть процес створення форми за допомогою конструктора.

11. Як можна змінити структуру форми в СКБД Microsoft Access?

Лабораторна робота № 18

Тема. Робота з мультимедійними даними та гіперпосиланнями

Мета роботи: навчитись вставляти чи встановлювати зв'язок між OLE-об'єктом і полем таблиці та створювати гіперпосилання в СКБД Microsoft Access.

Апаратно-програмне забезпечення та вихідні дані: комп'ютер, встановлена програма Microsoft Access.

Короткі теоретичні відомості

Мультимедійні дані – це дані різного походження (звукові, відео, графічні, текстові) з різними ефектами відображення на екрані. Для зберігання таких даних у базі необхідно в структурі таблиць мати поля типу **Поле об'єкта OLE**. OLE (Object Linking and Embedding – зв'язування й вбудовування об'єктів) – технологія обміну й спільного використання даних програмами, розроблена фірмою Microsoft. Вона дозволяє вбудовувати або зв'язувати об'єкт, створений однією програмою, у документ, створений іншою програмою.

Працювати з мультимедійними даними можна в режимі таблиці або в режимі форми (звіту). Другий варіант надає більші можливості. Окрім цього, мультимедійні дані можуть зберігатися як вбудовані або як зв'язані об'єкти. Вбудований об'єкт зберігається у файлі бази даних, що робить його доступним у будь-який момент часу. Зі зміною такого об'єкта в таблиці (формі, звіті), він змінюється у файлі бази даних. Під час зв'язування об'єкт можна переглядати й змінювати у таблиці (формі, звіті), але ці зміни зберігаються у файлі об'єкта, а не файлі бази даних. Файл об'єкта можна змінювати і незалежно за допомогою інших програм. Зв'язування корисне під час роботи з великими файлами, які небажано включати до файлу бази даних, а також з файлами, що використовуються декількома таблицями (формами, звітами). У разі зміни

місця розташування файлів зв'язаних об'єктів, необхідно заново встановлювати зв'язок з ними.

Серед можливих типів полів таблиць бази даних Microsoft Access є також тип даних – **Гіперпосилання**. Він дозволяє зберігати у полі прості або складні посилання на файли чи документи, що знаходяться поза базою даних. Гіперпосилання є покажчиком з одного об'єкта на інший. Найчастіше гіперпосилання вказує на веб-сторінку, але також може вказувати на рисунок, адресу електронної пошти, файл (наприклад, файл мультимедіа або документ Microsoft Office) або програму. Гіперпосилання можуть містити URL-адреси або мережевий маршрут у форматі UNC до файлу на сервері локальної мережі чи на диску локального комп'ютера.

Поле гіперпосилання є текстовим. Адреса гіперпосилання може містити до чотирьох компонентів, розділених знаком #: описання (необов'язкове), основна адреса гіперпосилання, додаткова адреса (необов'язкове) і підказка (необов'язкове). Описання являє собою текст, що відображається у полі таблиці (елементі управління). Адреса гіперпосилання – це URL-адреса або UNC-адреса. Додаткова адреса задає іменованій об'єкт усередині файла (наприклад, діапазон комірок у робочому аркуші Microsoft Excel). Підказка – це текст, що з'являється при утриманні покажчика миші на гіперпосиланні.

URL-адреса – адреса, що використовується для пошуку ресурсу в Internet. Адреса URL зазвичай починається з імені протоколу, за яким іде назва організації, що володіє вузлом (суфікс позначає тип організації). Наприклад, адреса <http://www.yale.edu/> містить такі відомості:

- *http*: даний веб-сервер використовує протокол HTTP;
- *www*: даний вузол знаходиться в Internet;
- *edu*: даний веб-вузол належить освітній установі.

UNC-адреса – повне ім'я ресурсу в мережі. Стандартний формат запису шляху (UNC) – це формат запису імен файлів й інших ресурсів, який починається з двох знаків зворотної косої риски (\), що вказують на розміщення ресурсу на мережевому комп'ютері. Імена UNC використовують синтаксис \\ Ім'я

Сервера \ Ім'я спільного ресурсу. UNC-ім'я каталогу або файлу може також містити шлях каталогу після імені спільного ресурсу. При цьому використовується такий синтаксис: \\ Ім'я Сервера \ Ім'я спільного ресурсу \ Каталог \ Ім'я файлу.

У просторових базах даних мультимедійні файли та гіперпосилання можуть використовуватися для зберігання графічної інформації про об'єкти (наприклад, плани ділянок, будівель тощо).

Порядок виконання лабораторної роботи

1. Відкрити файл вправ «Р_18» (місце розташування – папка Навчальна\Навчальні дані\ПКБД) і ознайомитися з покроковими інструкціями виконання роботи.

2. Ознайомитися з основними поняттями та порядком вставляння чи встановлення зв'язку між OLE-об'єктом і полем таблиці, а також порядком створення гіперпосилання в СКБД Microsoft Access.

3. Запустити Microsoft Access.

4. Виконати завдання, наведені файл вправ «Р_18».

5. Зберегти результати у власній папці як файл Microsoft Access.

Звітні матеріали: файл формату Microsoft Access, збережений у власній папці.

Контрольні запитання

1. Що називають звітом у СКБД Microsoft Access?
2. Що називають елементами керування у СКБД Microsoft Access?
3. Які типи елементів керування є в СКБД Microsoft Access?
4. Які способи створення звітів є в СКБД Microsoft Access?
5. Опишіть процес створення звіту за допомогою команди Автозвіт.
6. Опишіть процес створення звіту за допомогою майстра.
7. Опишіть процес створення звіту за допомогою конструктора.

2 КРИТЕРІЇ ОЦІНЮВАННЯ ЗНАНЬ СТУДЕНТІВ

Контроль знань здійснюється згідно з вимогами «Положення про проведення поточного та семестрового контролю» КрНУ імені Михайла Остроградського. Кількість балів, які студент може отримати за кожну лабораторну роботу, наведена у таблиці 2.1.

Таблиця 2.1 – Кількість балів за окремими лабораторними роботами

№ пор.	Назва теми	Кількість годин
1	Реляційна модель даних	1
2	Зв'язування таблиць	1
3	Операції реляційної алгебри	2
4	Мова запитів за зразком QBE	1
5	Структурована мова запитів SQL	1
6	Створення бази даних Microsoft Access	1
7	Створення таблиць бази даних Microsoft Access	1
8	Робота з даними в режимі конструктора й таблиці СКБД Microsoft Access	1
9	Визначення ключів, індексів та зв'язків таблиць бази даних Microsoft Access	1
10	Залежності між атрибутами	1
11	Метод нормальних форм	1
12	Побудова ER-моделей під час проектування бази даних методом «Сутність-зв'язок»	2
13	Формування відношень на підставі діаграм ER-типу	1
14	Пошук, відбір і сортування даних	1
15	Робота із запитами	1
16	Робота зі звітами	1
17	Робота з екранними формами	1
18	Робота з мультимедійними даними та гіперпосиланнями	1
	Усього:	20

Максимальна кількість балів ставиться, якщо робота виконана своєчасно, а студент під час захисту виявив досконале володіння матеріалами роботи, дав кваліфіковані відповіді на задані йому питання, навів приклади. Якщо робота виконана несвоєчасно або студент частково володіє вивченим матеріалом, кількість балів зменшується залежно від ступеня володіння матеріалом.

СПИСОК ЛИТЕРАТУРИ

1. Дейт К. Дж. Введение в системы баз данных / пер. с англ. 6-е изд. Москва: Издательский дом «Вильямс», 2001. 1072 с.
2. Ульман Дж. Основы систем баз данных. Москва: Финансы и статистика, 1983. 334 с.
3. Сеннов А. Access 2010. Учебный курс. Санкт-Петербург: Питер, 2010. 288 с.
4. Хомоненко А. Д., Цыганков В. М., Мальцев М. Г. Базы данных: учебник для высших учебных заведений ; под. ред. проф. А. Д. Хомоненко. Санкт-Петербург: Корона принт, 2004. 736 с.
5. Одиночкина С. В. Разработка баз данных в Microsoft Access 2010. Санкт-Петербург: НИУ ИТМО, 2012. 83 с.
6. Бекаревич Ю. Б., Пушкина Н. В. Самоучитель Access 2010. Санкт-Петербург: БХВ – Петербург, 2011. 432 с.

Методичні вказівки щодо виконання лабораторних робіт з навчальної дисципліни «Проектування кадастрових баз даних» для студентів усіх форм навчання зі спеціальності 193 – «Геодезія та землеустрій» освітнього ступеня «Бакалавр»

Укладач к. т. н., доц. В. І. Козарь

Відповідальний за випуск зав. кафедри геодезії, землевпорядкування та кадастру В. В. Артамонов

Підп. до др. _____. Формат 60×84 1/16. Папір тип. Друк ризографія.
Ум. друк. арк. _____. Наклад _____ прим. Зам. № _____. Безкоштовно.

Видавничий відділ
Кременчуцького національного університету
імені Михайла Остроградського
вул. Першотравнева, 20, м. Кременчук, 39600