

СИСТЕМА УПРАВЛІННЯ БАЗАМИ ДАНИХ.

ПЛАН.

1. Основні відомості про базу даних, класифікація.
2. Системи управління базами даних.
3. СУБД Access 7.0, коротка характеристика.
4. Порядок створення даних в середовищі Access.

1.

Термін **база** вживається як синонім „склад” або „сховище”, де є чітка внутрішня система розміщення об’єктів зберігання.

База даних-це єдине централізоване сховищу даних певної предметної області (школи, вузу, підприємства) до якої мають доступ багато програм. Прикладом баз даних можуть бути: записна книжка, бібліотечний каталог, груповий журнал, різноманітні журнали обліку і т.д.

При використанні баз даних на ПК усувається ряд недоліків, пов’язаних з надмірністю та дублюванням різних даних.

Наприклад прізвища студентів та їх середній бал використовуються як для призначення стипендії так і для написання листів батькам про незадовільний стан справ студента.

Таким чином, в базах даних дані розробляються і зберігаються, як єдине ціле в інтересах розв’язування всіх задач предметної області.

Бази даних мають складну внутрішню структуру і складаються з записів. Записом є інформація про один елемент(підручник, деталь, студент). В запис може входити різна інформація, наприклад, прізвище, ім’я та по-батькові, дата народження, домашня адреса, номер залікової книжки і т.д.

Інформація заноситься в інформаційні поля. Вони можуть бути різної довжини та різні по типу.(цифрова, текстова, символна, лічильник, дата і т.д.). Характер зв’язку між записами та іншими елементами бази даних визначає **тип організації баз даних.**

Розрізняються три типи організації баз даних:

- Ієрархічний;
- Мережний;
- Реляційний.

Ієрархічна база даних має дерево видну структуру,

Мережева-є така база, в якій один елемент пов’язаний з іншим, або з сукупністю інших.

Для персональних комп’ютерів використовується переважно реляційна база даних.

Концепцію реляційної бази даних розробив Е.Ф. Кодд в 1970 році. В основі цієї бази лежить математичне поняття відношень. Відношення представляються у вигляді двомірних таблиць. Рядок таблиці відповідає у файлі даних, а стовпчик- полю, а в теорії рядок –кортеж, стовпчик-атрибут.

**ТАБЛИЦЯ-РЯДОК- СТОВПЧИК
ФАЙЛ- ЗАПИС -ПОЛЕ**

ВІДНОШЕННЯ- КОРТЕЖ-АТИБУТ

Реляційна структура баз даних має такі переваги:

- Простота розуміння і роботи з базою даних;
- Розглянуту базу простіше розвивати і доповнювати;
- Допускається максимальна гнучкість при обробці не передбачуваних, або випадкових запитів;
- Спощується контроль секретності даних;
- Спрощується фізична організація даних.
-

2. Система управління базами даних

Функцію забезпечення незалежності даних від програм реалізує система управління базами даних. (СУБД). Відомо що процес розв'язування задач на комп'ютері, обробка заданих алгоритмах.

СУБД – це спеціальний пакет програми який забезпечує створення, супроводження та використання баз даних багатьма користувачами.

В склад пакета програм **СУБД** входять такі програми:

1. Ядро системи, яке забезпечує зберігання і пошук в інформації бази даних.

2. Програма відображення форм для вводу – виводу даних.

3. Генератор звітів.

4. Засоби взаємодії ядра пакета і програм обробки текстів.

Будь - яка база даних створюється спочатку на папері, потім вводиться в комп'ютер, але починається з визначення, переліку змісту і типу даних, що використовується. Іншими словами визначаються імена полів, тип даних і величина – це є структура бази даних.

Наступний етап-введення даних в пам'ять комп'ютера, з одночасною перевіркою їх вірності.

На стані супроводження і використання баз даних, тобто на стані їх експлуатації, СУБД розв'язує задачі доповнення, редагування та інші операції.

Редагування містить такі операції:

1.Видалення застарілих записів.

2.Заміну одних записів іншими.

3. Вставку в певні місця нових записів.

4. Доповнення новими записами.

5. Додавання нових полів, заміна їх імен та ін.

Інформація видається користувачеві у відповідності з запитом, що надійшов до нього, видача на екран, або принтер у різноманітній, зручній наочній формах. Це може бути у вигляді таблиць, діаграм, графіків.

СУД забезпечує роботу доступу кількох користувачів, виконує: сортування, різноманітні підрахунки, пошук потрібних запитів.

У разі збоїв, чи раптового зникнення електроенергії СУБД забезпечує цілісність бази даних.

Крім поняття бази даних, на сьогодні використовується поняття банку даних.

Під банком даних слід розуміти базу даних та СУБД.

Банк даних має відповідати слідуючим вимогам:

- **Мати можливість оновлення, поповнення і розширення бази даних,**
- **Забезпечити високу надійність зберігання інформації,**
- **Мати запити, що забезпечують БД від несанкціонованого доступу.**

3. СУБД Access 7.0, коротка характеристика.

Вказана програма входить до пакету Microsoft office. Використовується русифікована версія, яка дозволяє розв'язати широке коло задач користувачів без програмування.

Запуск програми здійснюється з ОС Windows, аналогічно тому, як запускається будь-яка інша програма.

Після її завантаження на екрані –вікно, в якому справа розміщена інформація по створенню та роботі з базами даних.

Одна з основних переваг системи Access полягає в тому, що вона має прості і зручні способи обробки таблиць в одній базі даних.

Таблиця - це місце зберігання даних, вона є основним об'єктом бази даних.

Перші покоління СУБД давали змогу в одному файлі зберігати одну таблицю, для спільного використання таблиць застосовувались спеціальні програми (утиліти). Пізніше почали використовуватись команди.

Зберігання таблиць в одній базі даних має цілий ряд переваг:

- **Можливість розподілу даних по таблицях згідно з змістом даних;**
- **Табличне (модульне) подання даних дозволяє модернізувати кожну таблицю окремо;**
- **Редагування даних в одному місці приводить до їх автоматичного редагування в іншому місці.**

У системі **Access** є різні способи управління даними: система має панелі інструментів, контекстне меню, комбінації клавіш і т.д.

Access має значну кількість програм, які отримали назву «майстри», майстер таблиць, майстер форм, майстер звітів, майстер запитів, майстер діаграм. Для зручності робота з майстрами поділяється на покрокове виконання.

Система **Access** має зручні способи копіювання рядків, полів та форматів однієї таблиці.

Таблиці однієї бази даних можна переслати до іншої бази даних.

Можна отримувати нові копії таблиць та їх структур.

Програма має розвинуту систему запитів, дозволяє отримувати різні відомості з таблиць.

За допомогою майстра форм на екрані видається по бажанню користувача бажана форма таблиці.

В системі є три стандартні форми.

В системі широко використовується поняття звіт. Звіт багато в чому схожий на форму і його використовують при виведенні документів на друк. Різниця – формати звітів відповідають стандартним форматом паперу.

Система **Access** має потужну довідникову систему.

Система має можливість переносу таблиць в середовище для розрахунків та в середовище **Word** для зміни форматування таблиць.

4. Створення бази даних.

На початковому етапі доцільно побудувати на папері структуру таблиці.

Варіанти створення:

- Виконати команду **Новая база данных** результаті на екрані Файл новой базы данных)
- Набирається ім'я файлу бази даних і натискується кнопка-**Создать**.

(в результаті вікно Базы данных, в якому є головне меню, панель інструментів, а в вікні таблиці перераховані способи створення:

1. **Создание таблицы в режиме конструктора;**
2. **Создание таблицы с помощью мастера;**
3. **Создание таблицы путем ввода данных;**

Виконуємо команду №1 двічі клацнувши мишею.

(на екрані з'явиться вікно, в якому необхідно ввести назви полів та визначитись з типом даних).

Після заповнення всієї таблиці необхідно виділити ключове поле.

Після цієї команди закривається вікно і підтверджується збереження структури таблиці.

Для формування бази даних необхідно виконати команду-**Открыть базу данных**.

В вікні необхідно відкрити ту таблицю, в яку будуть вводиться дані.

Заповнення даними створених структур таблиць може здійснюватись різними способами.(якщо виникає потреба зміни структури таблиці, то необхідно виконати команду **Конструктор** і ввести необхідні корективи.

Робота з базою даних Access

1. Робота з таблицями
2. Створення запитів

Таблиця - це набір записів, описуючих дані одного виду. Ключовими словами в останньому реченні є дані одного виду. Всі дані записи в одній таблиці мають подібні дані. Інформація про бейсболістів може бути в одній

таблиці, а інформація про які-небудь обчислення – в іншій. Але в одній таблиці не може знаходитись і те, і інше. Комбінація інформації цих двох видів - це щось нове, але це буде працювати в Access 97.

База даних Access - це набір однотипних даних із визначеною предметною областю. База даних включає таблиці, запити, звіти, форми і інші структурні елементи, які допомагають створити Access 97 для роботи з вашими даними. Замість того, щоб зберігати всі ці елементи окремо, кожен у своєму файлі на магнітному диску, де вони можуть стати недоступними, бути пропущеними чи видаленими помилково, вони групуються і зберігаються в одному файлі.

У Microsoft Access існує два способи створення таблиці. Для введення власних даних можна створити порожню таблицю. Можна також створити таблицю, використовуючи вже існуючі дані з іншого джерела.

У Microsoft Access існує чотири способи створення порожньої таблиці.

- Використання майстра баз даних для створення всієї бази даних, що містить всі необхідні звіти, таблиці і форми, за одну операцію. Майстер баз даних створює нову базу даних, його не можна використовувати для додавання нових таблиць, форм, звітів у вже існуючу базу даних.

- Майстер таблиць дозволяє вибрати поля для даної таблиці з безлічі визначених раніше таблиць, таких як ділові контакти, список особистого майна чи рецепти.

- Введення даних безпосередньо в порожню таблицю в режимі таблиці. При збереженні нової таблиці в Microsoft Access дані аналізуються і кожному полю привласнюється необхідний тип даних і формат.

- Визначення всіх параметрів макета таблиці в режимі конструктора.

Незалежно від методу, застосованого для створення таблиці, завжди є можливість використовувати режим конструктора для подальшої зміни макета таблиці, наприклад, для додавання нових полів, встановлення значень за замовчуванням чи для створення масок введення.

Для створення бази даних за допомогою майстра:

1. При запуску Microsoft Access відкриється діалогове вікно, у якому пропонується створити нову базу даних чи відкрити існуючу. Якщо з'являється це діалогове вікно, виберіть параметр Запуск майстра, а потім натисніть кнопку ОК.

Якщо база даних вже була відкрита чи якщо було закрито перше діалогове вікно, натисніть кнопку **Создать базу данных** на панелі інструментів.

2. Встановіть покажчик на потрібний значок шаблону бази даних і двічі натисніть кнопку миші.

3. Вкажіть ім'я і каталог для створюваної бази даних.

4. Щоб почати створення нової бази даних, натисніть кнопку ОК.

Для створення таблиці за допомогою майстра таблиць:

1. Переключіться у вікно бази даних. Для переключення з іншого вікна у вікно бази даних натисніть клавішу F11.

2. На вкладці **Таблица** натисніть кнопку **Создать**.

3. Двічі клацніть елемент **Мастер таблиць**.

4. Слідуйте інструкціям, видаваним у діалогових вікнах майстра таблиць.

У разі потреби по закінченні роботи з майстром таблиць змінити чи розширити отриману таблицю можна в режимі конструктора.

Для створення таблиць шляхом введення даних у таблицю:

1. Переключіться у вікно бази даних. Для переключення з іншого вікна у вікно бази даних натисніть клавішу F11.

2. На вкладці **Таблиця** натисніть кнопку **Создать**.

3. Двічі клацніть елемент **Режим таблиць**. На екрані з'явиться порожня таблиця, що складається з 20 стовпців і 30 рядків. За замовчуванням задаються наступні імена стовпців: «Поле1», «Поле2» і т.д.

4. Для перейменування кожного стовпця двічі клацніть назву стовпця, введіть ім'я, слідуючи умовам про імена об'єктів Microsoft Access, і натисніть клавішу ENTER.

5. Якщо таблиця повинна містити більш 20 стовпців, то можна додати додаткові. Для цього натисніть кнопку миші праворуч від стовпця, поруч з яким необхідно розмістити новий, і в меню **Вставка** виберіть команду **Столбик**.

6. Введіть дані в таблицю.

Вводити в кожен стовпець дані визначеного типу (у таблицях Microsoft Access кожен стовпець називають полем). Наприклад, вводити в один стовпець прізвища, а в інший імена. При введенні чисел, значень дат і часу необхідно використовувати погоджені формати, щоб Microsoft Access могла автоматично визначити відповідний тип даних і формат відображення стовпця. Усі стовпці з незміненими іменами, залишені порожніми, будуть вилучені при збереженні таблиці.

7. Закінчивши введення даних в потрібні стовпці, натисніть кнопку **Сохранить** на панелі інструментів, щоб зберегти таблицю.

8. При збереженні таблиці виводиться запрошення створити ключове поле. Якщо дані, що можуть однозначно ідентифікувати кожен запис, такі як інвентарні номери чи коди, ще не були введені, то рекомендується натиснути кнопку **Да**. Якщо дані, що можуть однозначно ідентифікувати кожен запис, введені, то це поле можна визначити як ключове.

Тип даних кожного поля (стовпця) визначається типом введених даних. Якщо надалі прийдеться змінити визначення поле, наприклад, задати інший тип даних чи визначити умову на значення, ці дії можна виконати в режимі конструктора.

Крім перейменування і вставки стовпців у будь-який час, як до, так і після збереження нової таблиці, допускається видалення стовпців чи зміна їхнього положення в таблиці.

Для швидкого створення таблиці в режимі конструктора:

1. Переключіться у вікно бази даних. Для переключення з іншого вікна у вікно бази даних натисніть клавішу F11.

2. На вкладці **Таблиця** натисніть кнопку **Создать**.

3. Двічі клацніть елемент **Конструктор**.
4. Визначіть в таблиці кожне поле.
5. Визначіть ключові поля до збереження таблиці.

Рекомендується визначати ключові поля, хоча це зовсім не обов'язково. Якщо вони не були визначені, то при збереженні таблиці видається питання, чи потрібно їх створювати.

6. Для збереження таблиці натисніть кнопку **Сохранить** на панелі інструментів, а потім введіть ім'я таблиці.

Для створення нової таблиці з використанням існуючих даних:

У Microsoft Access існує два способи для створення таблиць на основі вже існуючих даних.

- Дані можна зв'язати чи імпортувати з іншої бази даних Microsoft Access чи з безлічі форматів файлів інших програм.

- Створити таблицю, засновану на даних поточної таблиці, можна за допомогою запиту на створення таблиці. Наприклад, запити на створення таблиць можна використовувати для архівації старих записів, копіювання таблиць, вибору груп записів для експорту в іншу базу даних чи для використання як основу для звітів, що відбивають дані у визначений момент часу.

Для створення нової таблиці в запиті на створення таблиці:

1. Створіть запит, вибравши таблиці чи запити, що містять записи, які варто помістити в нову таблицю.

2. В режимі конструктора запиту натисніть стрілку поруч із кнопкою **Тип запроса** на панелі інструментів і виберіть команду **Создание таблицы**. На екрані з'явиться діалогове вікно **Создание таблицы**.

3. В поле **Имя таблицы** введіть ім'я таблиці, яку необхідно створити чи замінити.

4. Виберіть параметр **У текущей базе данных**, щоб помістити таблицю у відкриту в даний момент базу даних, чи виберіть параметр **В другой базе данных** і введіть ім'я бази даних, у яку потрібно помістити нову таблицю. У разі потреби вкажіть шлях.

5. Натисніть кнопку ОК.

6. Перемістіть поля, що повинна містити нова таблиця, зі списку полів у бланк запиту.

7. Для полів, переміщених у бланк запиту, в клітині **Условие отбора** введіть умову.

8. Для перегляду нової таблиці до її створення на панелі інструментів натисніть кнопку **Вид**. Щоб повернутися в режим конструктора запиту і внести необхідні зміни чи запустити запит, на панелі інструментів натисніть кнопку **Вид**.

9. Для створення нової таблиці натисніть кнопку **Запуск** на панелі інструментів.

- Щоб зупинити виконання запиту, натисніть клавіші CTRL+BREAK.

- Встановлені у вихідній таблиці властивості полів і ключові поля не переходять до даних у новій таблиці.

1. Створення запитів

Запит - означає запитання, яке ви ставити Access про вміст таблиці або бази даних. Без запитів не можливо швидко знайти потрібні дані.

При створенні звіту вкажіть наступне:

Поля, по яких ви хочете вести пошук. Якщо ви хочете знайти всі записи про клієнтів з прізвищем Сміт, котрі мають поштовий індекс 20210, потрібно шукати в полях «Фамилия» та «Почтовый индекс».

Які поля ви хотіли б вивести в результаті виконаного запиту. Не обов'язково виводити всі поля. Необхідно вказати тільки ті, які потрібно вивести в якості відповіді на запит.

За бажанням можна вказати порядок в якому Access виведе знайдені записи.

Створення простого запиту.

За допомогою простого запиту можна зробити набагато більше, ніж при використанні директиви Find (Поиск).

В даний момент має бути завантажена база даних Caveat і її вікно має бути виведене на монітор. Клацніть по ярлику Query (Запрос), зліва в вікні бази даних. Для створення першого запиту клацніть по кнопці New (Создать) в верхній частині вікна бази даних. При цьому відкривається діалогове вікно режиму New Query (Новый запрос). Але поки що ви не будете використовувати Query Wizards (Мастеров по запросам). Перший запит буде дуже простим для того, щоб для його створення залучати майстра. Так що клацніть по кнопці New Query (Новый запрос).

За допомогою майстрів по звітам можна створити наступні типи запитів.

Crosstabs, в якому відповіді виводяться у вигляді таблиці, в якій значення одного поля показані в рядку, а другий - в стовпці; Find Duplicates, котрий знаходить таблицю записів, яка утримує такі ж дані; Find Unmatched Queryt яка знаходить таблицю, не задовільняючу певну умову, і Archive Query, котра вибирає записи з таблиці і копіює їх в нову таблицю.

Наступним кроком буде вказівка Access, в якій таблиці потрібно шукати, з'явиться діалогове вікно Add Table (Добавление таблицы).

Частини вікна Query (Запрос - выборка).

В верхній частині вікна виведені назви таблиці (із списком полів), які використовуються в запиті. При створенні запиту за допомогою миші потрібні поля із списком перемістіть в нижню частину вікна. Нижня частина вікна вміщує 4 рядки:

Field (Поле). Місце розташування списку полів, необхідного для виконання запиту. Якщо ви починаєте пошук в даному полі, потрібно заповнити рядок Criteria (Условие отбора). Якщо ж цього не зробити, поле буде використане в запиті, але його значення буде виводитись у відповіді.

Sort (Сортировка). В цьому рядку ви вказуєте Access відсортувати результати пошуку в алфавітному порядку (зверху вниз чи навпаки). Або не сортувати зовсім.

Criteria (Условие отбора). В цьому рядку потрібно вказати, що ви хочете шукати в даному полі. Якщо цей рядок буде не заповнений, поле буде брати участь у пошуку, хоча його ім'я може бути у відповіді.

Вибір потрібних полів.

Для того, щоб використати поля які будуть використані в запиті потрібно зробити наступне:

1. «Витягніть» поле «Номер» в таблиці і розмістіть його в першому рядку вікна форматування запиту.
2. Клацніть в вікні на перетині першого рядка і другого стовпця для переміщення в це місце курсору.
3. Клацнувши двічі в полі «Дата продажу» в списку поля таблиці «Продажа». Це поле з'явиться в місці де розміщений в даний час курсор.
4. Перемістіть поле «Название» в перший рядок і третій стовпець того ж рядка.
5. Виведіть умови запиту.

Виконання та збереження запиту.

На панелі інструментів розміщена кнопка з зображенням знаку оклику. Ця кнопка означає виконання запиту. Клацніть по ній - Access знайде всі потрібні записи, які задовільняють умови, вказані у запиті.

Access виводить відповідь на запит у вигляді міні-таблиці, складеної із стовпців та рядків. В цій таблиці (dynaset) ви побачите тільки відповідні поля і відповідь яка задовільняє запит.

Відповідь на запит в Access називають dynaset. Коли ви отримаєте відповідь на запит в інших системах управління баз даних, таких як Paradoxs або dBASE, то ця відповідь вміщує копію даних з вашої таблиці. В Access, dynaset включає реальні, «живі» дані таблиці. Якщо ви щось зміните у dynaset, то це автоматично призведе до відповідних змін і в самій таблиці.

Перше dynaset дає більше можливостей, ніж відповідь на запит в інших системах управління базою даних. Друге - ви повинні уважно і обережно виконувати будь-які операції з даними dynaset, оскільки це не просто копія табличних даних.

Останнім кроком в процесі роботи з запитом є його збереження.

Відкрийте меню File (Файл) і виберіть директиву Save Query (Сохранение запроса). При цьому з'явиться вікно Save As (Сохранение). Воно виявиться для вас знайомим, тому що аналогічне вікно ви використовували при створенні форм і таблиць.

В якості імені запиту введіть з клавіатури назву і клацніть по командній кнопці ОК. Потім закрийте вікно Query, клацнувши двічі по системній кнопці управління вікном його лівому верхньому куті. Access повинна повернутись до вікна бази даних.

Повторне виконання запиту.

Access зберігає виконані запити, котрі можна потім використовувати. Для цього необхідно клацнути два рази по імені запиту в вікні бази даних і dynaset з'явиться як раніше.

Міні-таблицю dynaset можна вивести на друк як і іншу таблицю, вибравши директиву Print (Печать) з меню File (Файл), коли dynaset знаходиться на екрані.

ЗВ'ЯЗКИ МІЖ ТАБЛИЦЯМИ.

Види зв'язків

Як правило, у базі даних MS Access інформацію не зберігають в одній таблиці, а подають її у вигляді декількох таблиць. Наприклад, інформацію про замовлені клієнтами товари доцільно зберігати у двох таблицях "Товари" та "Клієнти", які між собою зв'язані певним чином. Розглянемо види зв'язків між таблицями.

1. *Зв'язок "один до одного".* У цьому випадку одному запису із таблиці А відповідає один запис із таблиці В і навпаки одному запису із таблиці В відповідає один запис із таблиці А. Цей тип зв'язку використовують рідко, оскільки такі дані можуть бути розміщені в одній таблиці. Зв'язок "один до одного" використовується для розбиття дуже широких таблиць, для відділення частини таблиці із міркувань захисту інформації від несанкціонованого доступу, а також у випадку, коли друга таблиця є підмножиною записів для першої. Такий зв'язок створюється між ключовими полями двох таблиць.
2. *Зв'язок "один до багатьох".* У цьому випадку кожному запису із таблиці А може відповідати декілька записів із таблиці В і одному запису із таблиці В – лише один запис із таблиці А. Цей вид зв'язку використовується найчастіше. Наприклад, такий зв'язок може бути встановлений між таблицями "Постачальники" і "Товари", оскільки один постачальник може поставити на склад багато найменувань товарів, але партія одного і того ж товару поставляється лише одним постачальником. Зв'язок "один до багатьох" створюється між ключовим полем таблиці на стороні "один" і зовнішнім ключем на стороні "багато".
3. *Зв'язок "багато до багатьох".* У цьому випадку одному запису із таблиці А може відповідати декілька записів із таблиці В і навпаки, одному запису із таблиці В – декілька записів із таблиці А. Такий зв'язок реалізується за допомогою третьої таблиці, зовнішній ключ якої складається по крайній мірі із двох полів, які є ключовими для таблиць А і В. Наприклад, вищезгадані таблиці "Товари" та "Клієнти" можуть бути зв'язані за допомогою третьої таблиці "Замовлення".

Створення зв'язків між таблицями:

1. Відкрити вікно "Схема даних" (вибр. "Сервис", "Схема данных").
2. Додати таблиці, що мають бути зв'язаними ("Связи", "Добавить таблицу").
3. Лівою клавішею миші перетягти ключове поле однієї таблиці на ключове поле або поле зовнішнього ключа іншої. Щоб зв'язок був функціональним, відмітити галочкою поле "Забезпечення цілісності даних".
4. Закрити вікно "Схема даних" із збереженням внесених змін.

Цілісність даних.

Зв'язки між таблицями забезпечують так звану *"цілісність даних"*. Це означає, що не можна вилучити запис із основної таблиці, якщо вона містить зв'язані записи в інших таблицях. Наприклад, при наявності зв'язку між таблицями "Постачальники" і "Товари" Access не дозволить вилучити записи про тих постачальників, які поставили на склад товари, тобто ніколи не виникне ситуація, що товар виявився бракованим, а інформація про постачальника відсутня.

Крім цього, наявність зв'язку між двома таблицями виключає можливість появи "сиріт". Це означає, що при наявності зв'язку "один до багатьох" між двома таблицями у таблиці на стороні "багато" не може бути записів не зв'язаних із записами таблиці на стороні "один". Наприклад, якщо ми маємо дві таблиці "Клієнти" і "Кредити", зв'язані співвідношенням "один до багатьох", то у таблицю "Кредити" не можна ввести запис, у якому було б відсутнє посилання (код) на клієнта.

Контрольні запитання

1. Які способи введення даних до таблиць вам відомі?
2. Як переміщатися між різними комірками таблиці?
3. Як переключитися з режиму таблиці в режим конструктора таблиць і навпаки?
4. Де розташована панель переходу і для чого вона потрібна?
5. За допомогою якої команди меню виконується імпорт даних із зовнішнього джер
6. Як імпортувати електронну таблицю Excel у базу даних?
7. На якому етапі імпорту таблиці створюється ключове поле?
8. Для чого потрібен звіт?
9. Наведіть послідовність дій під час створення звіту.

Пов'язування даних

Відношення між таблицями Розглянемо можливі відношення між таблицями бази даних. Вони б> таких типів: "один до одного", "один до багатьох", "багато до одного" і "б до багатьох". Найпоширенішим у таблицях реляційних баз даних є відної "один до багатьох".



Відношення "один до багатьох" означає, що одному запису і відповідають кілька записів в іншій таблиці.

Розглянемо створені таблиці для бази даних «Фірма «Подорож». Турис фірма продає путівки, причому однакові путівки вона може продати кі клієнтам. Тому одному запису в таблиці «Путівки» можуть відповідати де* записів у таблиці «Замовлення» (мал. 61.4 і мал. 61.7, б). Наприклад, **noirji** путівки з кодом 2 замовлені клієнтами 4 і 5.

Сторона "один" у відношенні "один до багатьох" називається *голо* таблицею. Сторона "багато" у цьому самому відношенні називається *зв'я* таблицею. У прикладі зв'язаних таблиць «Путівки» і «Замовлення» ми що ключове поле головної таблиці зв'язується з відповідним полем зв'яї таблиці.

Тип відношень "один до одного" трапляється в таблицях баз даних ніж "один до багатьох".

У "наведених нами раніше таблицях немає відношення «один до Щоб проілюструвати це відношення, ми могли б додатково до таблиці

створити таблицю «Банківські дані», у якій містилися б банківські реквізити клієнтів. Наприклад, у полі «Рахунок» було б зазначено номери розрахункових рахунків клієнтів. Оскільки рахунок кожного клієнта унікальний, між полем «Код клієнта» таблиці «Клієнти» і полем «Рахунок» таблиці «Банківські дані» існувало б відношення «один до одного».

Встановлення зв'язків між таблицями

Отже, між полями таблиць можуть існувати певні відношення. Однак щодо них програма Access нічого не знає, поки ми не визначимо зв'язки між таблицями за допомогою передбаченої для цього процедури. Розглянемо встановлення зв'язків на прикладі таблиць «Путівки», «Клієнти» і «Замовлення».

- Починаючи зв'язування таблиць, переконайтеся, що всі таблиці і форми закриті. Потім переключіться у вікно бази даних, натиснувши клавішу F11.
- Клацніть по кнопці Схема даних eg на панелі інструментів вікна Access - у відповідь відкриється однойменне діалогове вікно.

Водночас з вікномСхема даних з'явиться діалог Добавление таблицы Отже, між полями таблиць можуть існувати певні відношення. Однак щодо них програма Access нічого не знає, поки ми не визначимо зв'язки між таб-

лицями за допомогою передбаченої для цього процедури. Розглянемо встановлення зв'язків на прикладі таблиць «Путівки», «Клієнти» і «Замовлення».

- Починаючи зв'язування таблиць, переконайтеся, що всі таблиці і форми закриті. Потім переключіться у вікно бази даних, натиснувши клавішу F11.
- Клацніть по кнопці Схема даних eg на панелі інструментів вікна Access -у відповідь відкриється однойменне діалогове вікно.
- Водночас з вікном Схема даних з'явиться діалог Добавление таблицы (мал. 62.1). Однак цей діалог може і не з'явитися, якщо раніше були створені зв'язки між таблицями (вікно Схема даних не порожнє). У цьому разі потрібно очистити вікно від зв'язків, клацнувши по кнопці Очистить макет. Потім натисніть кнопку Добавить таблицу на панелі інструментів.
- За умовчанням в діалозі Добавление таблицы буде відкрита вкладка Таблиць). Виділіть у списку таблицю «Замовлення» і натисніть кнопку Добавить. У вікні Схема даних з'явиться список полів таблиці «Замовлення» (мал. 62.2).
- Аналогічно виведіть списки полів таблиць «Клієнти» і «Путівки» у вікно Схема даних, після чого закрийте діалог Добавление таблицы.
- Для зв'язування полів «Код путівки» оберіть це поле у головній таблиці «Путівки» і перетягніть його мишею до зв'язаної таблиці «Замовлення». Зазначимо, що напрямок перетягування поля завжди повинен бути «від головної таблиці до зв'язаної»

завжди повинен бути «від головної таблиці

Макет зв'язків

Зручним для вас способом (наприклад, щоб лінії зв'язків не перетиналися розмістіть списки полів у вікні Схема даних. Списки можна перетягуваї захопивши мишею заголовок списку. Розташування списків у вікні називаєть *макетом зв'язків*. Після цього можете закрити вікно Схема даних. Програш виведе на екран запит щодо того, чи потрібно зберігати макет зв'язків Клацніть по кнопці Да. Якщо ви відповісте Нет, то збережуться лише створені зв'язки, а не компоновання

Зверніть увагу, що в розглянутих випадках ми зв'язували ключове поле ізначене напівжирним шрифтом у списку полів) головної таблиці з відповідним йому полем зв'язаної таблиці. Поле зв'язаної таблиці називають *полем зов-^Шинього ключа*.

Для видалення будь-якого зв'язку не потрібно знову створювати макет зв'яз-в. Виділіть зв'язок клацанням миші й натисніть клавішу Delete. Після появи Впиту на видалення зв'язку клацніть по кнопці Да.

Щонтрольні запитання

Щ, Які відношення між таблицями вам відомі?

І. Що таке головна і зв'язана таблиці у відношенні «один до багатьох»? III.

Опишіть процедуру створення зв'язків між таблицями.

f. Як виглядатиме лінія зв'язку у вікні Схема даних, якщо не установити перемикач Обеспечение целостности даних? , Що таке макет зв'язків? Як

видалити зв'язок між таблицями? І. Що таке ключове поле? . Що таке поле зовнішнього ключа?

Сортування, фільтрація і пошук даних.

1. прийоми сортування записів;
2. задання фільтрів даних;
3. запуск процедури пошуку.

Сортування записів.

При введенні даних у таблиці або форми записи розташовуються у порядку введення. Це не завжди зручно при перегляді введеної інформації. Бажано згрупувати й упорядкувати інформацію, щоб вона була змістовною й у ній було легко кінтуватися. Наприклад, у таблиці «Замовлення» доречно виділити товар з 'найбільшою купівельною спроможністю, помістивши відповідні записи на ючаток таблиці, а в таблиці «Клієнти» доцільно розподілити клієнтів за [регіонами. Цього легко досягти сортуванням записів, яке виконується у такий (спосіб. і*

Відкрийте таблицю і перейдіть у Режим таблиці.

- Активізуйте поле таблиці, за яким відбуватиметься сортування.
- Клацніть по кнопці д| (сортування за зростанням) або по кнопці (сортування за спаданням).

Виконайте вправу: відсортувати таблицю «Клієнти» за полем «Адреси», а таблицю «Путівки» - за полем «Ціна». Коли ви збережете відсортований об'єкт (таблицю або форму) у режимі таблиці, порядок сортування також автоматично збережеться, тобто сортування буде повторно застосовано при новому відкритті об'єкта.

Описані прийоми сортування можна застосовувати до таблиць, а також до форм і запитів (останні розглядатимуться нижче). Сортування в цих об'єктах виконується аналогічно.

Прості фільтри даних.

Сортування даних дозволяє упорядковувати їх, однак воно не скор ПАКОСТІ записів у таблиці, які доводиться переглядати користувачу. Зру* засобами для перегляду записів є *фільтри*.

Припустимо, що вам потрібно переглянути відомості про путівки до Відкрийте таблицю «Путівки» у режимі таблиці (мал. 61.4). Установіть курсор на комірці «Греція» і натисніть кнопку Фільтр по виділенній у %f на па інструментів. На екрані з'являться записи про путівки до цієї країни Цей простий фільтр дозволяє відфільтрувати записи лише за однією умовою що накладається на поле. Для виконання фільтрації ще за однією умовою приклад, за полем «Прізд») клацніть по комірці з потрібним значенням (і пустимо, авіа). Далі натисніть кнопку Фільтр по виділенному - і ви одері усього два записи, що містять слово «Греція» у полі «Країна» і слово «авіа» полі «Прізд».

Фільтрація даних стосується лише записів на екран, сама таблиця залишається без змін. Для скасування фільтра й відновлення попереднього вигляду таблиці клацніть по кнопці Удалить фильтр V, розміщеній на панелі інструментів. При установці фільтра, що містить кілька умов, зручно скористатися дещо ширшим інструментом. Відкривши таблицю, у якій збираєтеся виконати фільтрацію, клацніть по кнопці Изменить фильтр '^ на панелі інструментів. На екран виведено лише один рядок таблиці (мал. 63.2). У полі «Країна» з'явиться кутна стрілка розкривного списку. За допомогою цього списку задайте одну умову. Клацніть по іншому полю й аналогічно задайте другу умову (наприклад вид путівки). Задайте всі необхідні умови і натисніть кнопку Применение фильтров. Зверніть увагу, що кнопка Применение фильтров після фільтрації перетворюється на кнопку Удалить фильтр, яка повертає таблицю до початкового

Розширений фільтр.

Для задання складних умов фільтрації використовується розширений фільтр. Припустимо, ви хочете переглянути у таблиці «Путівки» записи про путівки до Болгарії вартістю не більше 1500 грн. Створимо для цього розширений фільтр.

- Відкрийте таблицю «Путівки» бази даних «Фірма «Подорож» і перейдіть до режиму таблиці (клацання по кнопці Вид).
- Оберіть команду меню Записи - Фильтр - Расширенный фильтр.
- У діалозі (мал. 63.3) клацніть по першій комірці рядка Поле в розкривному списку оберіть позицію «Країна». У рядку Условие отбора наберіть «Болгарія».

У сусідній комірці рядка Поле оберіть зі списку позицію «Ціна», а в умові вибору зазначте <«=1500».

Клацніть по кнопці Применение фильтра (або задайте команду Фильтр - Применить фильтр).

На екрані з'являться відфільтровані відомості про путівки до Болгарії в заданому інтервалі вартості

Пошук запису.

Якщо таблиці баз даних великі і важко знайти той чи інший запис, можна звернутися за допомогою до засобу Поиск.

- Працюючи з таблицею (наприклад, «Путівки»), перейдіть у режим таблиці (якщо він не встановлений)

Клацніть мишею в тому полі, значення якого буде використано для пошуку (наприклад, «Код»).

Натисніть на кнопку Найти M на панелі інструментів.

У діалозі (мал. 63.5) введіть значення поля (наприклад, число 5). Запустіть процедуру пошуку, клацнувши по кнопці Найти.

Мал. 63.5. Діалог задання умови пошуку

Якщо необхідно виконати пошук, враховуючи регістр слова або форма полів, установіть перед натисканням кнопки **Найти** відповідні перемикачі діалозі пошуку. У цьому самому діалозі є розкриті списки **Просмотр** і **Совг дение**, що дозволяють задати зону пошуку (все, вверх, вниз) і ознаку (поля целиком, с любой частью поля, с начала поля).

Формування запитів

- 1. створення простих і перехресних запитів;**
- 2. задання умов відбору;**
- 3. створення запиту з обчисленням.**

Доступ до інформації в базі даних забезпечується таким інструментом, як *запити*. Запити дозволяють відібрати дані, що містяться в різних таблицях бази, а також виконати відбір відповідно до заданих умов (наприклад, список товарів, не дорожчих заданої ціни, дані про клієнтів у певному регіоні тощо). Тобто запити нагадують розглянуті раніше фільтри, однак запити - гнучкіший інструмент доступу до інформації. Так, за допомогою запитів можна не лише добути інформацію з БД, а й формувати нові поля, яких немає у первинних таблицях. У запитах можна обробляти початкові дані (знаходження середнього, максимального значення, підсумовування тощо).

Простий запит

Створення запиту в Access (як і створення інших об'єктів) можливо здійснити за допомогою майстра або в режимі конструктора. Під час ознайомлення з технологією запитів ми використовуватимемо майстер простих

Дотримуючись його інструкцій, ви зможете обрати потрібну таблицю і поля даних, переглянути результати відбору на екрані.

Продовжимо розгляд бази «Фірма «Подорож». Наприклад, вас цікавить інформація щодо того, які організації й у якій кількості замовляли путівки. Очевидно, що такий запит має бути зроблений на основі таблиць «Замовлення» і «Клієнти». Вважатимемо, що відповідні таблиці побудовані і що між ними встановлено зв'язок, показаний на мал. 62.4. • Відкрийте вікно бази даних,

перейдіть на вкладку **Запросы** і натисніть кнопку **Создать**. У діалозі **Новый запрос** оберіть опцію **Простой запрос** і наУ першому вікні майстра простих запитів (мал.64.2) зазначте, що вистворюєте запит на основі таблиці «Замовлення». Для цього в списку **Таблицы/запросы** виділіть «Таблица: Замовлення». Потім у списку **Доступные поля** клацніть попозиції «Код клієнта» і натисніть кнопку із символом >. Зазначена вами позиція переміститься до списку **Выбранные**

Оберіть у верхньому списку **Таблицы/запросы** опцію «Таблица: Клієнти» і перемістіть позицію «Організація» із лівого списку до правого.

• Знову установіть у списку опцію «Таблица: Замовлення» і перемістіть з лівого списку до правого поле «Кількість». Потім клацніть по кнопці **Далее**.

- У наступному діалозі залиште установленим перемикач подробиць (вивід кожного поля кожної записи) і клацніть по кнопці Готово.
У вікні Access з'явиться запит (мал. 64.3), що містить дані трьох зазначених полів із двох зв'язаних таблиць

Перехресний запит.

Ефективним засобом аналізу даних є *перехресний запит*. Цей вид запиту і зволяє згрупувати дані рядків або стовпців і вивести підсумкові значення до окремого стовпця.

Розглянемо приклад простого запиту, показаного на мал. 64.3. В ньому клієнти «Київміськбуд-2» і АКБ «Комінвест» представлені кількома рядками. Зручніше було б мати таблицю запиту, в якій дані замовлень кожного клієнта були б підсумовані, а загальна кількість замовлень умістилася б до окремого стовпця. Найпростіше це зробити за допомогою перехресного запиту.

- Відкрийте вікно бази даних, перейдіть на вкладку Запроси і клацніть по кнопці Создать.
- У діалозі Новий запит (мал. 64.1) зазначте опцію Перекрестный запит і натисніть кнопку ОК.

- Відкриється перше діалогове вікно майстра перехресних запитів. Клацніть по перемикачу запити і виділіть у списку існуючих запитів позицію «Замовлення: Запит». Натисніть кнопку Далее.

Наступний діалог призначений для задання заголовків рядків у майбутньому запиті. У списку Доступные поля оберіть позицію «Організація», натисніть кнопку >, і зазначена позиція переміститься до списку Выбранные поля. Клацніть по кнопці Далее.

Третій діалог майстра перехресних запитів служить для задання заголовків стовпців. У списку Доступные поля виділіть позицію «Код клієнта» і натисніть кнопку Далее.

У діалозі (мал. 64.5) потрібно задати ті обчислення, які виконуватимуться в комірках на перехресті рядків і стовпців. Оберіть опцію Зит, яка у нашому прикладі встановить підсумовування за значеннями стовпця «Кількість». Клацніть по кнопці Далее.

- В останньому діалозі майстра введіть ім'я запиту, наприклад «Сума замовлень». Натисніть кнопку Готово й отримайте таблицю запиту. У стовпчику підсумкових значень показано загальну кількість путівок, замовлених кожною організацією. Стовпці праворуч мають заголовки кодів клієнтів і містять значення, на основі яких сформований стовпець підсумків.

Використання виразів у запитах

При формуванні запиту ви можете задати обробку даних, наприклад! знайти суму або середнє значення для будь-якого поля. Для цього використовуються *вирази*, за якими виконуються обчислення, а результати обчислень заносяться в окреме поле. У виразах можна вживати знаки арифметичних операцій +, -, *, /, оператори порівняння =, <, >, <=, >=, а також імена полів,

які взяті у квадратні дужки. Наприклад, вираз $[Ціна]*1,25$ означає^ що вміст поля Цена збільшується у 1,25 разу.

Надалі ми використовуватимемо вирази при заданні умов відбору і під час формування поля, що обчислюється в запиті.

Умови відбору

Запит, сформований згідно з вказівками пункту «Створення запиту», містить всі замовлення путівок. Якщо ж вас цікавлять лише великі замовлення (кількіс путівок перевищує певне число), краще сформулювати запит із заданням умов^ відбору у такий спосіб:

- У вікні бази даних перейдіть на вкладку Запроси і клацніть двічі по пікіграмі «Замовлення: Запрос».
- У відповідь відкриється вікно запиту (мал. 64.3). Перейдіть у режим конструктора запитів, клацнувши по кнопці Вид (мал. 61.2) на панелі інструментів.
 - У діалозі (мал. 64.7) наведено схему даних для розглянутих таблиць, нині це - бланк запиту. Клацніть по комірці, розташованій на перехресті ряд; Условие отбора і стовпця «Кількість». Введіть із клавіатури вираз $\langle > 8 \rangle$ і натисніть Enter.
 - Клацніть по кнопці Вид і ви побачите таблицю із записами, що задовольняють умови відбору. Після введення умови відбору кількість записів у запиті зменшиться до трьох.

Розрахунки в запиті.

Проілюструємо виконання розрахунків на прикладі запиту, сформованого на основі таблиць «Замовлення» і «Путівки». Нас цікавитиме сума кожного замовлення, що обчислюється як добуток ціни путівки та кількості путівок: $[Ціна]*[Кількість]$. Виконується подібний запит таким чином.

- Спочатку сформулюйте запит, показаний на мал. 64.9 (технологію створення запиту ви вивчили в пункті «Створення запиту» цього параграфа). Потрібно використовувати поля таблиць «Замовлення» і «Путівки».

Перейдіть у режим конструктора запитів, клацнувши по кнопці Вид на панелі інструментів.

У вікні (мал. 64.10) клацніть по полю праворуч від поля «Кількість». Введіть вираз $[Ціна]*[Кількість]$ і натисніть клавішу Enter. Перед введеним виразом з'явиться текст Выражение:

Клацніть правою кнопкою миші в зоні поля з виразом і оберіть у контекстному меню команду Свойства. У діалозі Свойства поля задайте формат поля С разделителем (два десяткові знаки після коми) і назву поля «Сума».

Натисніть кнопку Вид і перейдіть у Режим таблиць. Ви отримаєте запит (мал. 64.11), в останньому стовпці якого буде зазначена сума кожного замовлення.

Отже, на основі таблиць бази даних ви отримали запит, у якому було виведено обчислюване поле - сума всіх зроблених замовлень на путівки. Розрахунки виконуються безпосередньо при виведенні запиту. Результати обчи-

слень таблицях не зберігаються. Тому результати запиту завжди представляють поточний вміст бази даних.

Контрольні запитання

1. Для чого призначені запити?
2. Чим запити відрізняються від фільтрів?
3. Опишіть процедуру створення простого запиту в базі даних.
4. Що таке перехресний запит і як його створити?
5. Для чого застосовуються вирази в запитах?
6. Як задати умову відбору в запиті?
7. Як створити поле, що обчислюється в запиті?

Контрольні запитання:

1. Як здійснити сортування записів за зростанням (за спаданням)?
2. У чому полягає перевага фільтрації перед сортуванням?
3. Як задати простий фільтр?
4. Що таке розширений фільтр?
5. Як задається розширений фільтр?
6. Як виконується пошук потрібної інформації в базі даних?