

"Прикладне програмування в телекомунікаційних системах."

**Методичні вказівки
до виконання лабораторних робіт**

**для студентів напрямів підготовки
6.050803 "Акустотехніка" та 6.050903 "Телекомунікації" усіх форм
навчання**

Затверджено методичною радою НТУУ „КПІ”

ЗМІСТ

Вступ	
Витяг з РСО щодо оцінювання лабораторних робіт.....	
Лабораторна робота №1	
Лабораторна робота №2	
Лабораторна робота №3	
Лабораторна робота №4	
Лабораторна робота №5	
Лабораторна робота №6	
Лабораторна робота №7	
Лабораторна робота №8	
Лабораторна робота №9	
Перелік посилань	

Вступ

Методичні вказівки до виконання лабораторних занять з дисципліни "Прикладне програмування" містять стислі теоретичні відомості, опис структури вікна та системи меню Microsoft Access, особливості та рекомендації щодо виконання лабораторних робіт та обробки результатів.

Методичні вказівки складені відповідно до програми курсу та призначені для виконання лабораторних робіт студентами усіх форм навчання за напрямом 6.050.903 "Телекомунікації" які вивчають курс "Прикладне програмування" протягом семестру.

Тематика лабораторних занять погоджена з лекційним курсом і сприяє більш поглибленому вивченню та закріпленню теоретичних знань студентів з дисципліни, а також дозволяє оволодіти практичними навиками редагування та створення баз даних.

Вивчення теоретичних відомостей, що представлені в цих вказівках, а також виконання лабораторних робіт в програмному пакеті Microsoft Access допоможе студентам ґрунтовніше підготуватись до модульного та семестрового контролю, а також краще засвоїти практичні та теоретичні навички отримані на лекціях і лабораторних заняттях .

Метою виконання лабораторних робіт є:

вивчення методів створення та редагування реляційних баз даних ;
закріплення теоретичних знань, що безпосередньо стосуються особливостей використання програмного пакеті Microsoft Access ;
вдосконалення навиків практичної роботи з SQL-редактором в програмному пакеті Microsoft Access .

Основними завданнями лабораторних робіт є:

вивчення принципів побудови реляційних баз даних за допомогою конструктора та майстра;
вивчення особливостей створення запитів та встановлення зв'язків між базами даних;
- засвоєння базових команд, що використовуються в SQL-редакторі для створення та редагування СУБД;

Лабораторна робота №1

Проектування БД на основі таблиць

Мета роботи - отримання навичок роботи зі створення структури таблиць, модифікації структури таблиць, заповнення таблиць. Створення ключових полів, індексованих полів, установка зв'язків між таблицями.

Теоретичні відомості

Таблиця містить дані по певній темі, наприклад, відомості про співробітників або товари. Кожен запис у таблиці містить дані про один елемент, наприклад про конкретний співробітника. Запис складається з полів і включає такі відомості, як ім'я, адреса та номер телефону. Крім того, запис зазвичай називається рядком, а поле - стовпцем.

Код контакта	Имя	Фамилия	Адрес	Город	Индекс	Рабочий телефон
1	Самаойлов	Иван	Бакінських комісарів 4	Кишенів	3223	5649345
2	Мироненко	Петро	Полькова 24 кв 5	Київ	02334	5402341
3	Анкудінова	Ірина	Урицького 10 кв 17	Київ	02311	2110843

Запись: 3 из 3

Рисунок 1.1 - приклад таблиці.

На Рис 1.1 показано типову таблицю. 1 - Запис або рядок, 2 - Поле або стовпець. База даних може включати безліч таблиць, в яких зберігаються дані з різних тем. Кожна таблиця може складатися з безлічі полів різного типу, включаючи текст, числа, дати і малюнки.

На Рис 1.2 представлено таблицю у режимі конструктора. В цьому режимі можна детально встановити всі атрибути полів. Ключове поле (ключ) - спеціальне поле, що визначає унікальність кожного запису в цій таблиці; ключовим може бути будь-яке поле або група полів, аби вони мали унікальний набір значень для кожного запису в таблиці: в більшості випадків унікальність забезпечується введенням до

запису спеціального поля - порядкового номера або коду запису (тип - лічильник); при видаленні або додаванні в таблицю записів унікальність значень цих полів відстежується автоматично; ключові поля мають і інше використання, наприклад - для сортування даних по порядку і для зв'язку таблиць один з одним.

На те що поле являється ключовим вказую піктограма у вигляді ключа перед іменем поля.

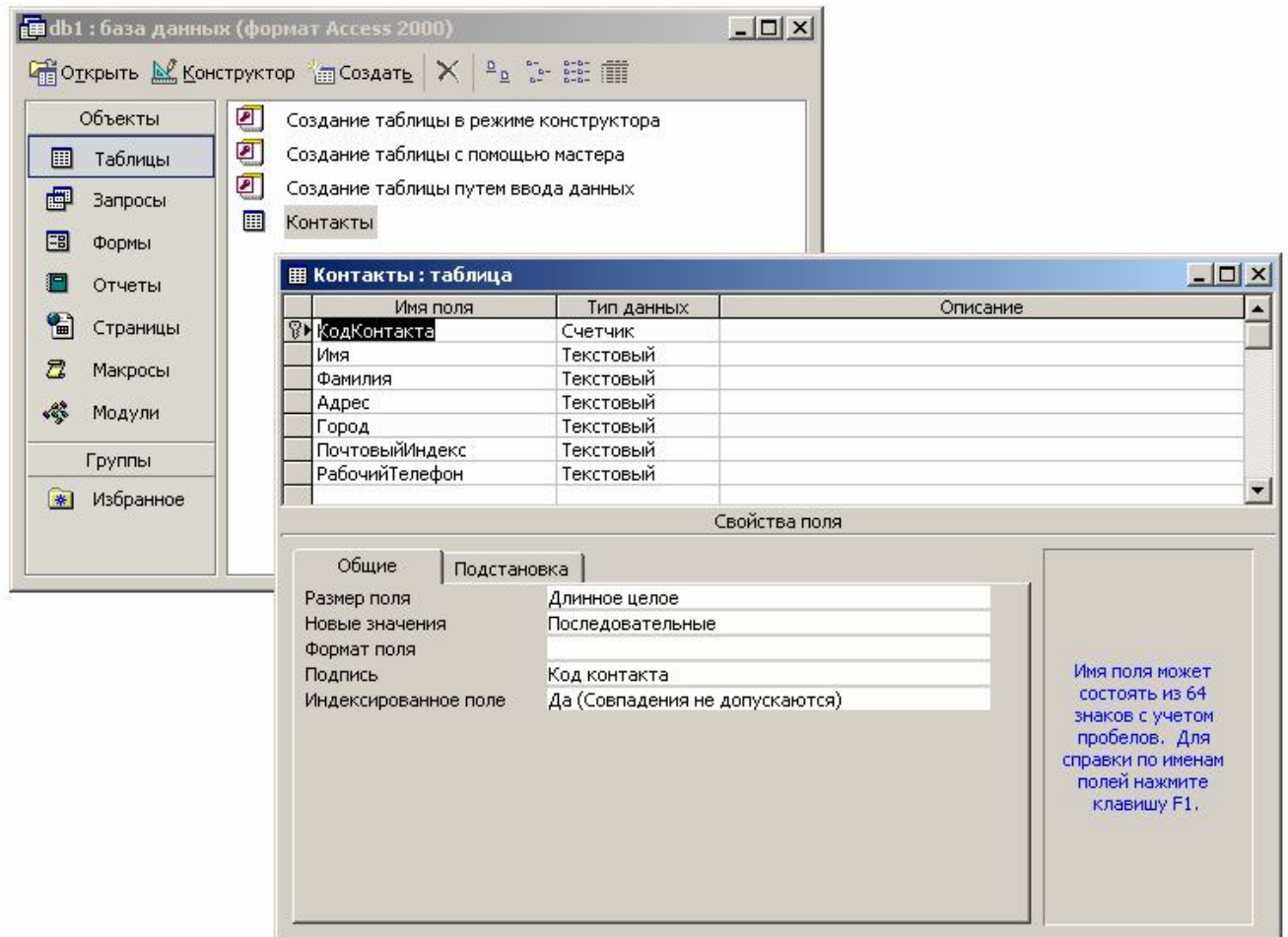


Рисунок 1.2 – вікно редактора баз даних.

В Microsoft Access возможностей работы с индексами набагато менше, ніж в інших програмах цього типу. Автоматично створюються індекси для полів, на які призначені обмеження первинного або вторинного ключа. Для решти стовпців індекси можна створити явно в режимі конструктора для таблиці: властивості поля, рядок "Індексоване поле" (Рис 1.2). Для цього поля передбачено три варіанти: Ні (за замовчуванням), Так (допускаються збіги) і Ні (Збіги не допускаються).

Зв'язок між таблицями - це відношення між співпадаючими значеннями в ключових полях. У більшості випадків пов'язують первинний ключ головної таблиці із

зовнішнім ключем підпорядкованої таблиці (часто мають те ж ім'я), наприклад, перший ключ "Код контрагента" таблиці "Контрагенти" і зовнішній ключ "Код контрагента" таблиці "ЖГО". Існують чотири типи відносин

1. Один - до одного. Запис таблиці А може мати не більше однієї зв'язаного запису в таблиці В і навпаки. Ключове поле в таких таблицях повинно бути унікальним.
 2. Багато - до одного. Одному запису в таблиці А може відповідати один запис у таблиці В, а одному запису в таблиці В - кілька записів у таблиці А. У таблиці В ключове поле повинно бути унікальним.
 3. Один - до багатьох. Кожному запису в таблиці А можуть відповідати кілька записів у таблиці В, а запис у таблиці В не може мати більше однієї відповідного запису в таблиці А. У таблиці А ключове поле повинно бути унікальним.
 4. Багато хто - до багатьох. Така таблиця не нормалізована. Однією запису таблиці А може відповідати кілька записів таблиці В і навпаки. Унікальних ключів немає. Всі ключі зовнішні. У цьому випадку таблицю треба нормалізувати.
- Пов'язані поля не обов'язково повинні мати однакові імена (наприклад поле "Номер*" рахунку таблиці "План рахунків" і поле "Дебет" таблиці "ЖГО"),

Робоче завдання

1. Запустити програму Microsoft Access.
2. Створити БД.
3. Створити 4 таблиці, а саме: дві у режимі таблиці, дві у режимі майстра.
4. Задати для всіх таблиць конкретні типи полів.
5. Задати для всіх таблиць конкретні ключові поля.
6. Створити зв'язки між таблицями у режимі конструктора та у режимі майстра.

Таблиця 1.1 – Завдання по варіантах.

№ варіанту	Завдання
1	Створити базу даних на тему “Деканат”. Створити такі таблиці. “Список викладачів”, “Список дисциплін”, “Список студентів”, “Оцінки”. Побудувати всі необхідні в’язки.
2	Створити базу даних на тему “Довідник рослин”. Створити такі таблиці. “Назви рослин”, “Назви місяців”, “Кліматичні зони”, “Вегетативний тип”. Побудувати всі необхідні зв’язки.
3	Створити базу даних на тему “Реєстратура комерційної лікарні”. Створити такі таблиці. “Лікарі”, “Пацієнти”, “Вартість послуг”, “Кваліфікація лікаря”. Побудувати всі необхідні зв’язки.
4	Створити базу даних на тему “Магазин запчастин”. Створити такі таблиці. “Назва запчастини”, “Постачальники”, “Марка автомобіля”, “Вартість послуг”. Побудувати всі необхідні зв’язки.
5	Створити базу даних на тему “Студенти”. Створити такі таблиці. “Картки студентів”, “Предмети”, “Оцінки”, “Стипендія”. Побудувати всі необхідні зв’язки.
6	Створити базу даних на тему “Юридичний відділ”. Створити такі таблиці. “Картки юристів”, “Справи”, “Витрати”, “Норми часу”. Побудувати всі необхідні зв’язки.
7	Створити базу даних на тему “Автомайстерня”. Створити такі таблиці. “Назва операції”, “Список робітників”, “Кваліфікація робітника”, “Вартість послуг”. Побудувати всі необхідні зв’язки.
8	Створити базу даних на тему “Бухгалтерія ресторану”. Створити такі таблиці. “Назва рецепту”, “Вид страви”, “Інгредієнти”, “Вартість роботи”. Побудувати всі необхідні в’язки.
9	Створити базу даних на тему “Магазин квітів”. Створити такі таблиці. “Назви рослин”, “Постачальники”, “Вартість послуг”, “Вегетативний тип”. Побудувати всі необхідні в’язки.

10	Створити базу даних на тему “Магазин іграшок”. Створити такі таблиці. “ Назва іграшки”, “Постачальники ”, “Тип іграшки”, “ Вартість ”. Побудувати всі необхідні в’язки.
----	---

Контрольні питання

1. Які типи зв'язків розрізняють в програмі Microsoft Access?
2. Дайте визначення поняттям "Ключове поле" , "Індексоване поле" .
3. Які типи полів дозволяється пов'язувати між собою?
4. При якому типу зв'язку ключові поля повинні бути зовнішніми?
5. Опишіть послідовність створення зв'язку типу "Один - до багатьох" привівши довільний приклад.

Лабораторна робота №2

Форми як основний засіб введення інформації до БД

Мета роботи - отримання навичок роботи зі створення форм.

Теоретичні відомості

Форма - це об'єкт бази даних, який можна використовувати для введення, зміни або відображення даних з таблиці або запиту. Форми застосовуються для управління доступом до даних, наприклад для визначення того, які поля або рядки даних мають відображатися. Ефективна форма прискорює роботу з базою даних, оскільки користувачам не потрібно шукати потрібні відомості. Приваблива форма робить роботу з базою даних не тільки ефективною, але і більш приємною. Крім того, форми можуть запобігати введення невірних даних.

На рис 2.1 та рис 2.2 представлено типові форми.

The screenshot shows a form window titled 'Контакты'. It contains several input fields for contact data. The fields are organized as follows:

Код контакта	Имя	Фамилия
	Самаойлов	Иван

Below this is a large text area for 'Адрес' containing 'Бакінських комісарів 4'.

Город	Индекс	Рабочий телефон
Кишенів	3223	5649345

At the bottom, there is a status bar with 'Запись: 1 из 3' and navigation buttons.

Рисунок 2.1 – форма вирівняного типу.

The screenshot shows a form window titled 'Контакты1'. It displays a table with contact information. The table has the following structure:

	Код контакта	Имя	Фамилия	Адрес	Город	Индекс	Рабочий телефон
▶	1	Самаойлов	Иван	Бакінських комісарів 4	Кишенів	3223	5649345
	2	Мироненко	Петро	Полькова 24 кв 5	Київ	02334	5402341
	3	Анкудінова	Ірина	Урицького 10 кв 17	Київ	02311	2110843
*	(Счетчик)						

At the bottom, there is a status bar with 'Запись: 1 из 3' and navigation buttons.

Рисунок 2.2 – форма табличного типу.

Форми створюють після створення таблиць. Фактично форма це інструмент доступу до даних які зберігаються у таблиці. Форми можна створювати у режимі майстра та у режимі конструктора. Як видно на Рис 2.3, режим створення можна вибрати з основного вікна бази даних.

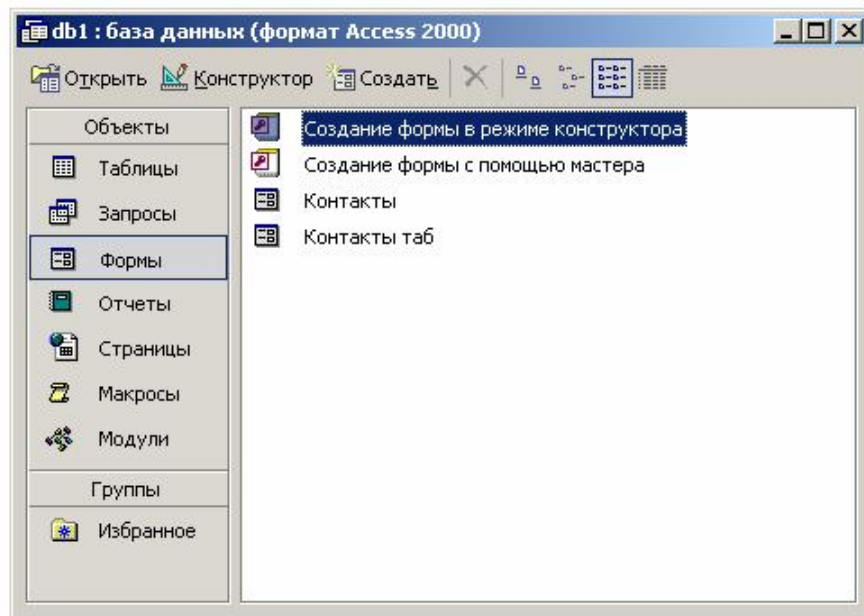


Рисунок 2.3 – основне вікно БД.

За допомогою майстра створювати форми простіше, але цей режим не дає повного контролю. У режимі конструктора можна налаштувати будь-які параметри форми, але це потребує багато роботи. Найкращий спосіб створення форм - це створення форм за допомогою майстра, а потім „тонке” налаштування за допомогою режиму конструктора.

Робоче завдання

1. Запустити програму Microsoft Access.
2. Взяти БД з лабораторної роботи № 1.
3. Створити форми для кожної таблиці.
4. Проаналізувати який тип форми підходить більше.

Таблиця 2.1 – Завдання по варіантах.

№ варіанту	Завдання (Типи форм для створення)
1	В один стовпчик, стрічкова.
2	В один стовпчик, таблична.
3	В один стовпчик, вирівняна.
4	Стрічкова, таблична.
5	Стрічкова, вирівняна.
6	Таблична, вирівняна

7	В один стовпчик, таблична.
8	Стрічкова, таблична.
9	Стрічкова, вирівняна.
10	В один стовпчик, вирівняна.

Контрольні питання

1. Що таке форми?
2. Для чого використовують форми?
3. Що дозволяє робити режим майстра?
4. Що дозволяє робити режим конструктора?
5. Чим відрізняються різні типи форм?

Лабораторна робота №3

Запити – основний інструмент роботи з БД.

Мета роботи - ознайомлення з різними типами запитів, отримання навичок зі створення запитів.

Теоретичні відомості

Коли потрібно переглянути, додати, змінити або видалити дані з бази даних, зручно використовувати запити.

За допомогою запитів можна отримати відповіді на дуже специфічні питання про дані, відповісти на які, просто подивившись на дані в таблиці, було б непросто. Запити можна використовувати для фільтрації даних, виконання розрахунків на основі даних і відображення зведених даних. Крім того, запити дозволяють автоматизувати виконання багатьох завдань управління даними і переглядати зміни до даних перед їх використанням.

Запит є звернення до даних для отримання інформації та виконання дій з даними. Запит можна використовувати для отримання відповіді на просте запитання, виконання розрахунків, об'єднання даних з різних таблиць або навіть додати, змінити або видалення даних у таблиці. Запити, які використовуються для отримання даних з таблиці або виконання розрахунків, називаються запитами на вибірку. Запити, які використовуються для додавання, зміни або видалення даних, називаються запитами на зміну.

Запити можна також використовувати для включення даних у форму або звіт. У добре структурованій базі даних відомості, які потрібно представити з використанням форми або звіту, часто зберігаються в різних таблицях. За допомогою запиту можна зібрати необхідні дані перед проектуванням форми або звіту.

Для того щоб створити запит треба вибрати відповідний пункт у основному меню БД. Запити можна створювати у режимі майстра або конструктора. За допомогою майстра створювати запити простіше, але цей режим не дає повного контролю. У режимі конструктора можна налаштувати будь-які параметри запита, але це потребує багато роботи. Найкращий спосіб створення форм - це створення

форм за допомогою майстра, а потім „тонке” налаштування за допомогою режиму конструктора. На Рис 3.1 показано створення запиту у режимі майстра. Видно що майстер дозволяє вибрати будь-яку кількість полів з таблиці.

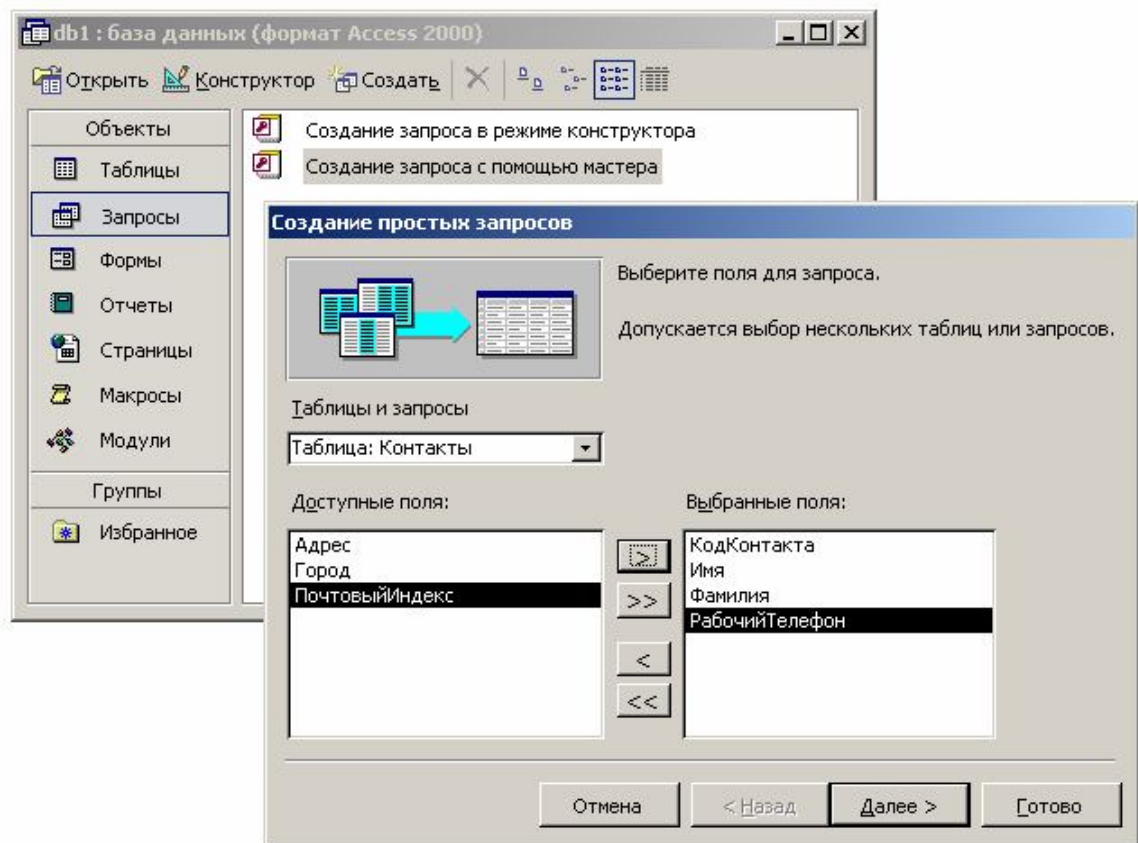


Рисунок 3.1 – режим майстра.

Припустимо що необхідно вибрати тільки імена, прізвища та робочі телефони. Слід вказати майстру на потрібні поля та натиснути кнопку „готово”.

На Рис 3.2 представлено вибірку тільки цих трьох полів.

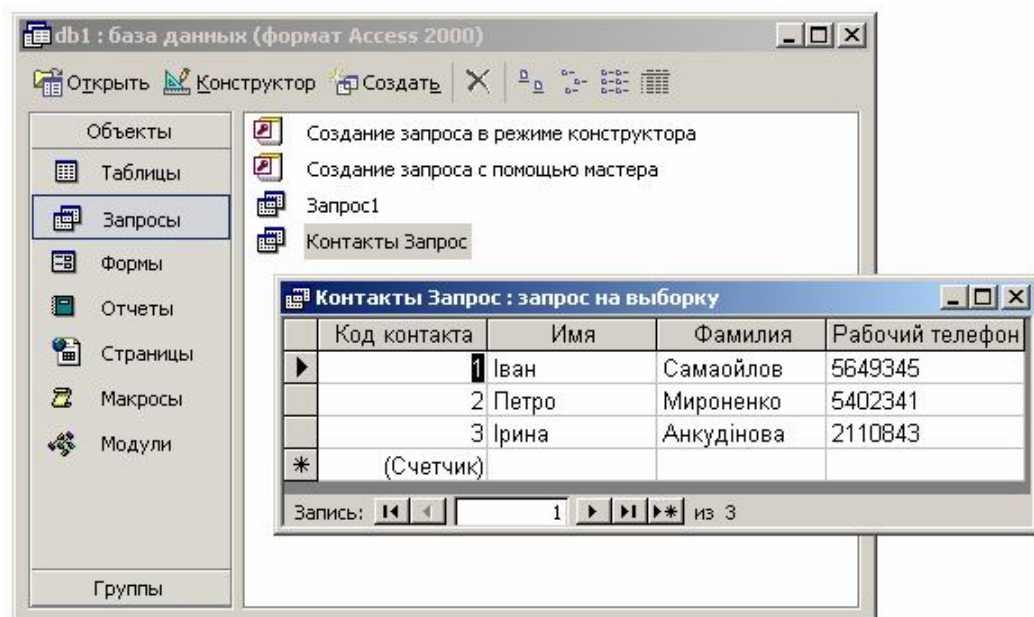


Рисунок 3.2 – вибірка з таблиці.

Режим конструктора надає більші можливості. В цьому режимі можна створювати запити з умовами. На Рис 3.3 представлено такий запит, у результаті побачимо тільки імена та прізвища людей номер телефону яких починається з цифри 5.

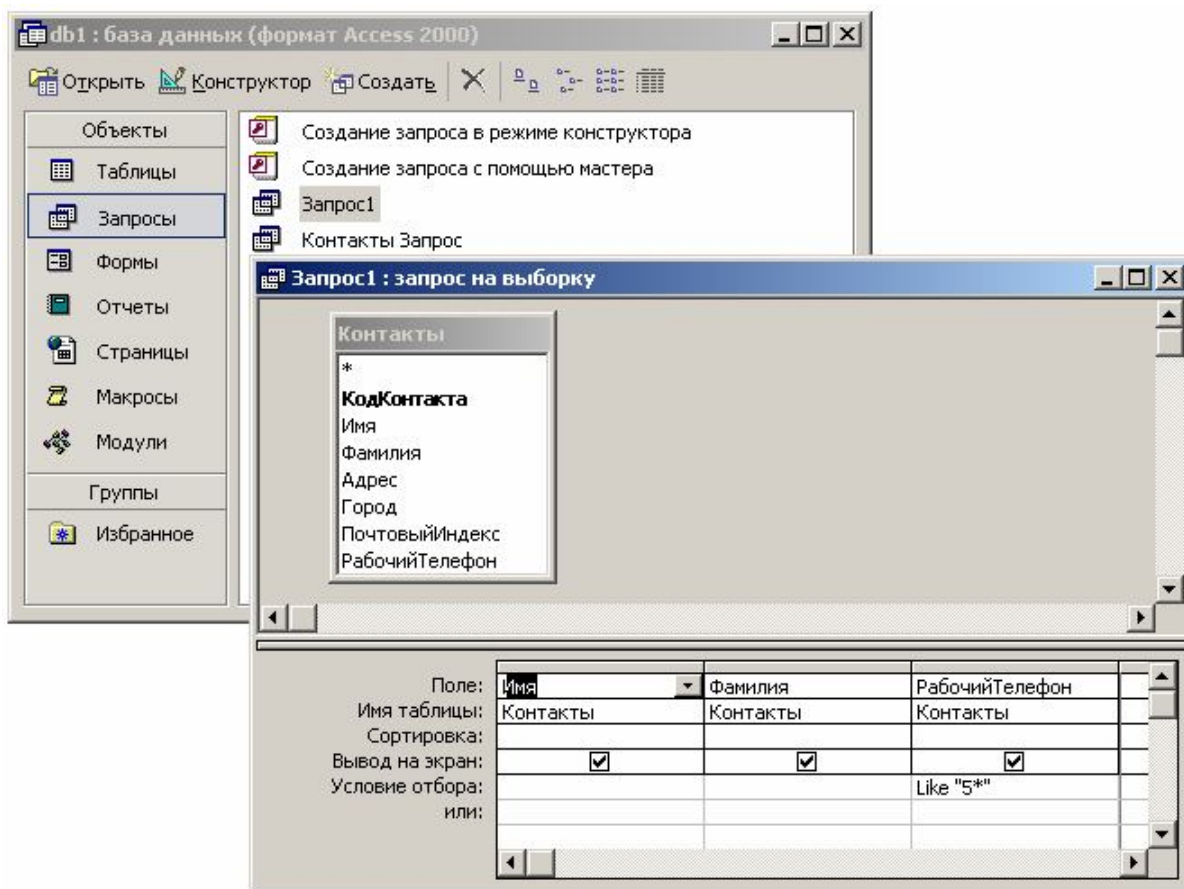


Рисунок 3.3 – запит з умовою у режимі конструктора.

Результат запиту представлено на Рис 3.4.

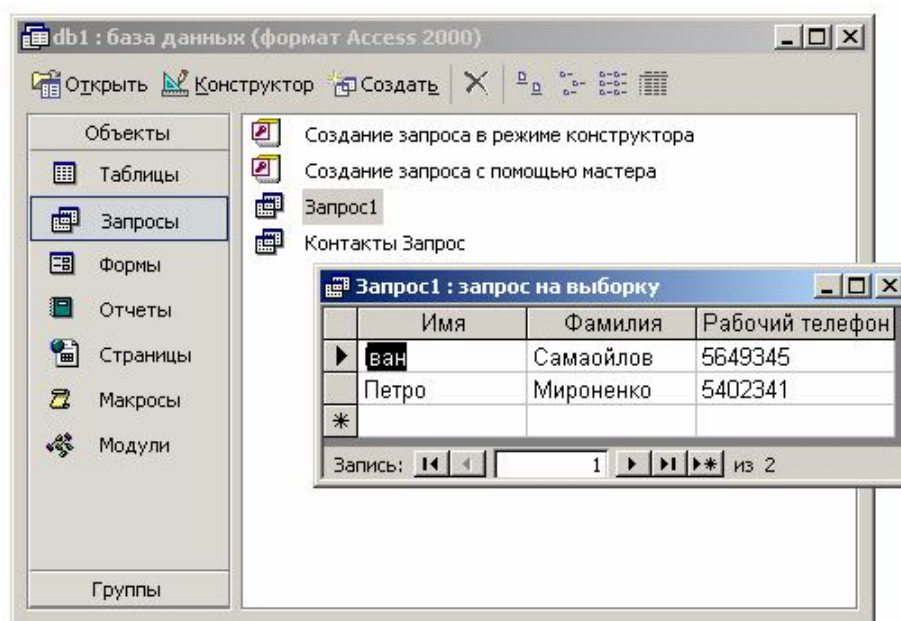


Рисунок 3.4 – результат запиту з умовою.

Робоче завдання

1. Запустити програму Microsoft Access.
2. Взяти БД з лабораторної роботи № 1.
3. Створити запити для кожної таблиці, простий у режимі майстра, та з умовою у режимі конструктора.
4. Проаналізувати який тип форми підходить більше.

Таблиця 3.1 – Завдання по варіантах.

№ варіанту	Завдання	
	простий	З умовою
1	Взяти таблицю „Список викладачів,,, вивести прізвище, ім’я та телефон викладачів.	Взяти таблицю „Список викладачів,,, вивести прізвище, ім’я та телефон тих викладачів прізвище яких починається з літери Д
2	Взяти таблицю „Назви рослин,,, вивести тільки назви рослин українською мовою.	Взяти таблицю „Назви рослин,,, вивести назви тих рослин які починається з літери М
3	Взяти таблицю „Лікарі,,, вивести прізвище, ім’я та телефон.	Взяти таблицю „Лікарі,,, вивести прізвище, ім’я та телефон тих лікарів прізвище яких починається з літери А
4	Взяти таблицю „Назва запчастини,,, вивести тільки назви (без номера за каталогом)	Взяти таблицю „Назва запчастини,,, вивести назви тих запчастин які починається з літери Н.
5	Взяти таблицю „Картки студентів,,, вивести прізвище, ім’я та телефон.	Взяти таблицю „Картки студентів,,, вивести прізвище, ім’я та телефон тих студентів прізвище яких починається з літери Б.

6	Взяти таблицю „Картки юристів,,, вивести прізвище, ім'я та телефон.	Взяти таблицю „Картки юристів,,, вивести прізвище, ім'я та телефон тих юристів прізвище яких починається з літери Г.
7	Взяти таблицю „Список робітників,,, вивести прізвище, ім'я та телефон.	Взяти таблицю „Список робітників,,, вивести прізвище, ім'я та телефон тих ремонтників прізвище яких починається з літери К.
8	Взяти таблицю „Назва рецепту,,, вивести тільки назви рецептів (без опису приготування).	Взяти таблицю „Назва рецепту,,, вивести тільки ті рецепти назва яких починається з літери Н.
9	Взяти таблицю „Назви рослин,,, вивести тільки назви рослин українською мовою.	Взяти таблицю „Назви рослин,,, вивести назви тих рослин які починається з літери О.
10	Взяти таблицю „Назва іграшки,,, вивести тільки назви іграшок (без опису).	Взяти таблицю „Назва іграшки,,, вивести тільки ті іграшки назва яких починається з літери П.

Контрольні питання

1. Що таке запити?
2. Для чого використовують запити?
3. Що дозволяє робити режим майстра?
4. Що дозволяє робити режим конструктора?
5. Як задати умову для відбирання записів за певними критеріями?
6. Які умови можна задавати у запитах?

Лабораторна робота №4

Звіти – засіб „візуалізації” інформації.

Мета роботи - ознайомлення з різними типами звітів, отримання навичок зі створення звітів.

Теоретичні відомості

Програма Access допомагає створювати безліч різних звітів будь-якого ступеня складності. Перш ніж приступати до роботи, слід вибрати джерело, з якого будуть вилучатись записи звіту. Звіт може бути як простий список, так і детальну зведення даних про продажі, згрупованих по районах. Однак у кожному випадку необхідно спочатку визначити, в яких полях містяться дані, які повинні увійти до звіту, і в яких таблицях або запитах розташовані ці поля.

Після того як обраний джерело записів для створення звіту, найпростіше скористатися майстром звітів. Майстер звітів - це засіб Access, що допомагає створити звіт на підставі відповідей, отриманих на задані користувачеві питання.

Поряд з даними, видобуваються з таблиць і запитів, в звіті міститься інформація про макеті звіту: підписах, заголовках, малюнках і т. п. Таблиці та запити, що містять базові дані, називаються джерелом записів звіту. Якщо всі поля, які потрібно включити до звіту, знаходяться в одній таблиці, ця таблиця і буде джерелом записів. Якщо поля знаходяться в декількох таблицях, як джерело записів доведеться використовувати один або декілька запитів.

Звіти також можна створювати у двох режимах, майстра та конструктора. Режим конструктора дозволяє зробити тонке налаштування але потребує багато роботи. Звіти частіше всього створюють у режимі майстра.

На Рис 4.1...Рис 4.6 показано поетапне створення звіту.

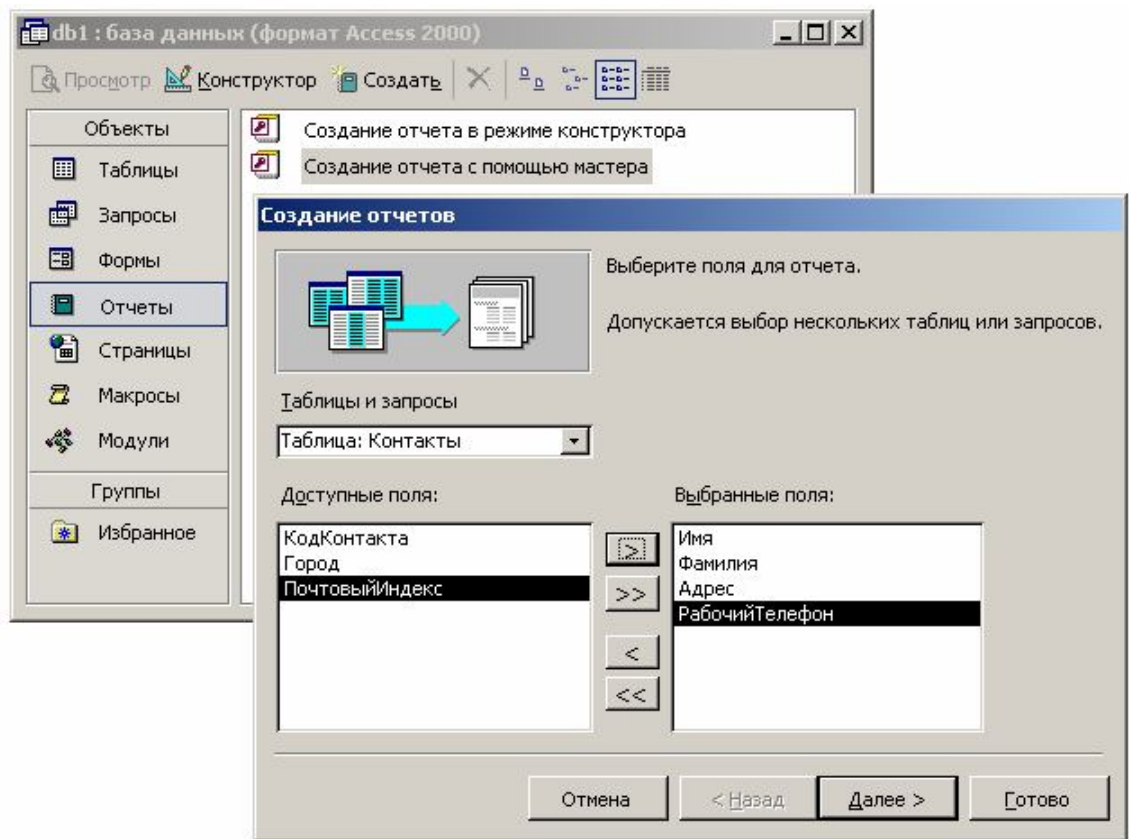


Рисунок 4.1 – створення звіту, перше вікно.

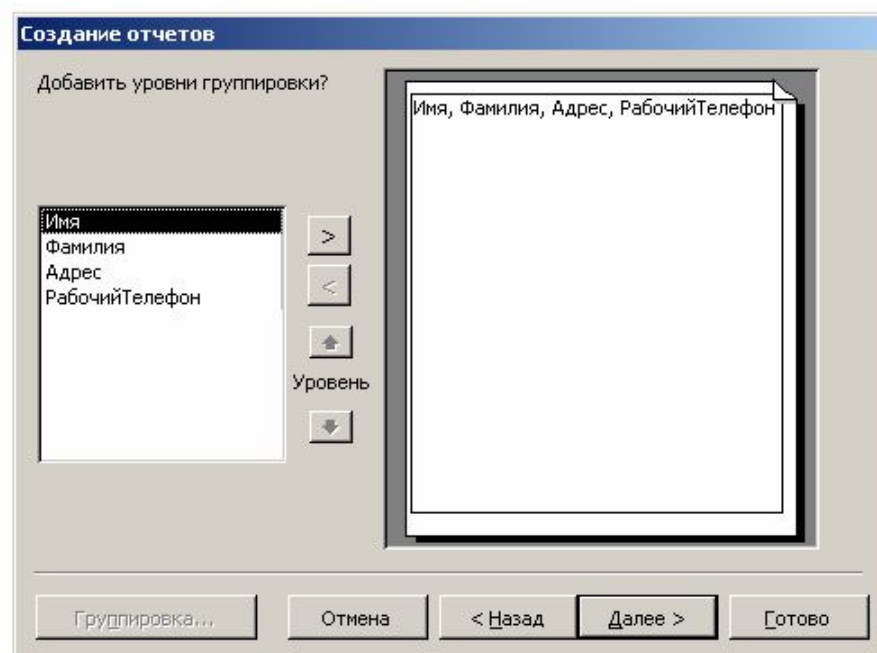


Рисунок 4.2 – створення звіту, рівні групування.

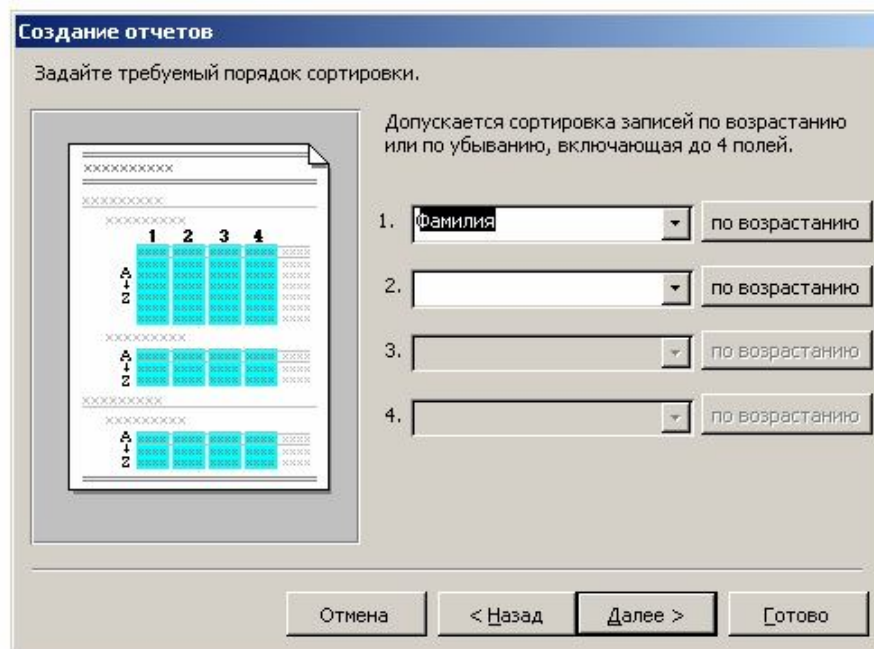


Рисунок 4.3 – створення звіту, сортування.

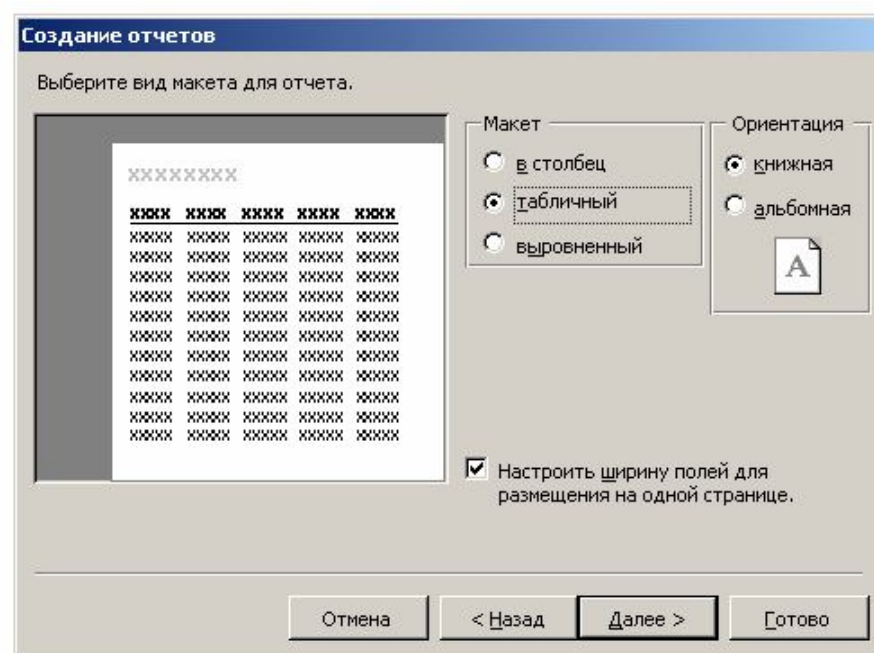


Рисунок 4.4 – створення звіту, вигляд звіту.

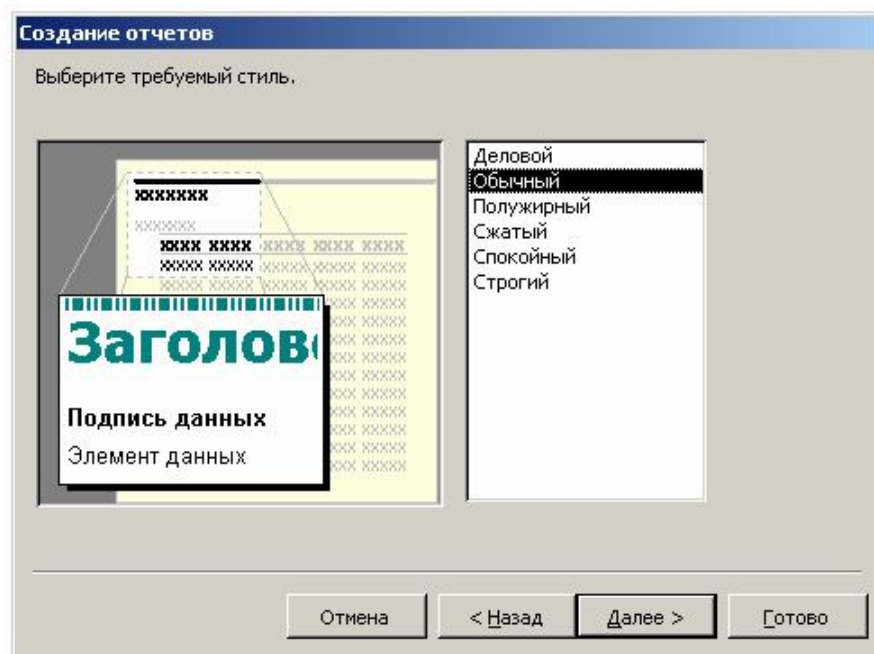


Рисунок 4.5 – створення звіту, стиль оформлення.

Контакты

Фамилия	Имя	Город	Рабочий телефон
Анкудинова	Ирина	Киев	2110843
Мироненко	Петро	Київ	5402341
Самаойлов	Иван	Кишинев	5649345

Рисунок 4.6 – створення звіту, вигляд звіту.

Видно що процес створення звітів, навіть у режимі майстра, має багато налаштувань. У вікні Рис 4.1 вибираються ті поля які потрібно надрукувати. У вікні Рис 4.2 задається групування для складних багато табличних звітів. У вікні Рис 4.3 вибирають по якому полю слід сортувати результати для звіту. У вікні Рис 4.4 вибирають режим групування інформації на папері, частіше всього використовують табличний режим. У вікні Рис 4.5 вибирають стиль оформлення звіту, як видно вибрано звичайний стиль. Результуючий звіт представлено на Рис 4.6.

Робоче завдання

1. Запустити програму Microsoft Access.
2. Взяти БД з лабораторної роботи № 1.
3. Створити звіти у режимі конструктора.

4. Проаналізувати який для яких даних який тип звіту підходить більше.

Таблиця 4.1 – Завдання по варіантах.

№ варіанту	Завдання	
	На основі таблиці	На основі запита з умовою
1	Взяти таблицю „Список викладачів,,, вивести прізвище, ім'я та телефон викладачів. Тип звіту „табличний”. Стиль оформлення „звичайний”.	Взяти таблицю „Список викладачів,,, вивести прізвище, ім'я та телефон тих викладачів прізвище яких починається з літери Д. Створити звіт. Тип звіту „у стовпець”. Стиль оформлення „діловий”.
2	Взяти таблицю „Назви рослин,,, вивести тільки назви рослин українською мовою. Тип звіту „табличний”. Стиль оформлення „діловий”.	Взяти таблицю „Назви рослин,,, вивести назви тих рослин які починається з літери М. Тип звіту „табличний”. Стиль оформлення „звичайний”.
3	Взяти таблицю „Лікарі,,, вивести прізвище, ім'я та телефон. Тип звіту „табличний”. Стиль оформлення „напівжирний”.	Взяти таблицю „Лікарі,,, вивести прізвище, ім'я та телефон тих лікарів прізвище яких починається з літери А. Тип звіту „у стовпець”. Стиль оформлення „діловий”.
4	Взяти таблицю „Назва запчастини,,, вивести тільки назви (без номера за каталогом). Тип звіту „табличний”. Стиль оформлення „стиснутий”.	Взяти таблицю „Назва запчастини,,, вивести назви тих запчастин які починається з літери Н. Тип звіту „табличний”. Стиль оформлення „діловий”.
5	Взяти таблицю „Картки студентів,,, вивести прізвище, ім'я та телефон. Тип звіту „табличний”. Стиль оформлення „спокійний”.	Взяти таблицю „Картки студентів,,, вивести прізвище, ім'я та телефон тих студентів прізвище яких починається з літери Б. Тип звіту „у стовпець”. Стиль оформлення „напівжирний”.

6	Взяти таблицю „Картки юристів,,, вивести прізвище, ім'я та телефон. Тип звіту „табличний”. Стель оформлення „суворий”.	Взяти таблицю „Картки юристів,,, вивести прізвище, ім'я та телефон тих юристів прізвище яких починається з літери Г. Тип звіту „устовпець”. Стель оформлення „напівжирний”.
7	Взяти таблицю „Список робітників,,, вивести прізвище, ім'я та телефон. Тип звіту „табличний”. Стель оформлення „звичайний”.	Взяти таблицю „Список робітників,,, вивести прізвище, ім'я та телефон тих ремонтників прізвище яких починається з літери К. Тип звіту „табличний”. Стель оформлення „суворий”.
8	Взяти таблицю „Назва рецепту,,, вивести тільки назви рецептів (без опису приготування). Тип звіту „табличний”. Стель оформлення „діловий”.	Взяти таблицю „Назва рецепту,,, вивести тільки ті рецепти назва яких починається з літери Н. Тип звіту „устовпець”. Стель оформлення „напівжирний”.
9	Взяти таблицю „Назви рослин,,, вивести тільки назви рослин українською мовою. Тип звіту „табличний”. Стель оформлення „напівжирний”.	Взяти таблицю „Назви рослин,,, вивести назви тих рослин які починається з літери О. Тип звіту „табличний”. Стель оформлення „спокійний”.
10	Взяти таблицю „Назва іграшки,,, вивести тільки назви іграшок (без опису). Тип звіту „табличний”. Стель оформлення „спокійний”.	Взяти таблицю „Назва іграшки,,, вивести тільки ті іграшки назва яких починається з літери П. Тип звіту „устовпець”. Стель оформлення „звичайний”.

Контрольні питання

1. Що таке звіт?

2. Для чого використовують звіти?

3. На що впливає порядок сортування?
4. На що впливає вибір типу макету?
5. На що впливає вибір стилю оформлення звіту?

Лабораторна робота №5

Мова запитів SQL оператор CREATE.

Мета роботи - ознайомлення з мовою запитів SQL, ознайомлення з оператором CREATE. Отримання навичок зі створення запитів мовою SQL.

Теоретичні відомості

SQL (англ. Structured Query Language - мова структурованих запитів) - універсальна мова, що застосовується для створення, модифікації й керування даними в реляційних базах даних. Ця мова був розроблений IBM (експериментальна СУБД "System R") в 1974-1979 роках. Перший стандарт мови SQL був прийнятий Американським національним інститутом стандартизації (ANSI) в 1987 (так званий SQL level 1) і кілька уточнень у 1989 році (SQL level 2), 1992 і 1999 році. Після цього, індустрія почала широко використовувати SQL як мову реляційних баз даних, і на сьогоднішній день практично кожна база даних заснована на SQL.

Архітектура даних, до якої звертається SQL називається реляційною. У реляційних базах даних всі дані представлені у вигляді простих таблиць, розбитих на рядки і стовпці, на перетині яких розташовані дані. Запити до таких таблиць повертають таблиці, які самі можуть ставати предметом подальших запитів. Кожна база даних може включати декілька таблиць, які, як правило, пов'язані один з одним, звідки і пішла назва реляційні.

Оператор CREATE DATABASE створює базу даних з вказаним іменем. Якщо база даних вже існує, то виникає помилка виконання команди.

Загальний синтаксис CREATE DATABASE db_name. db_name – ім'я бази даних, бажано вказувати англійською мовою.

Наприклад команда

```
CREATE DATABASE FflowersFshop
```

створить базу даних з іменем FflowersFshop (магазин квітів).

Оператор CREATE TABLE створює таблицю з заданим ім'ям у поточній базі даних. Якщо таблиця вже існує, то виникає помилка виконання команди.

Загальний синтаксис цієї команди CREATE TABLE tbl_name (перелік полів), де tbl_name – ім'я таблиці яку будемо створювати а у дужках треба вказати перелік імен полів для БД.

Наприклад команда

CREATE TABLE test (Name text, Age int) створить таблицю з іменем test у якої буде 2 поля name (ім'я) та Age(вік)

Робоче завдання

1. Запустити програму Microsoft Access.
2. Створити БД.
3. Створити таблицю у режимі SQL.

У новій (чистій) базі даних створити таблицю аналогічну першій таблиці з лабораторної роботи №1.

Таблиця 5.1 – Завдання по варіантах.

№ варіанту	Завдання
1	Створити базу даних на тему “Деканат”. У режимі SQL створити таблицю “Список викладачів”
2	Створити базу даних на тему “Довідник рослин”. У режимі SQL створити таблицю “Рослин”.
3	Створити базу даних на тему “Реєстратура комерційної лікарні”. У режимі SQL створити таблицю “Лікарі”.
4	Створити базу даних на тему “Магазин запчастин”. У режимі SQL створити таблицю “Назва запчастини”.
5	Створити базу даних на тему “Студенти”. У режимі SQL створити таблицю “Картки студентів”.
6	Створити базу даних на тему “Юридичний відділ”. У режимі SQL створити таблицю “Картки юристів”.
7	Створити базу даних на тему “Автомайстерня”. У режимі SQL створити таблицю “ Операції”.

8	Створити базу даних на тему “Бухгалтерія ресторану”. У режимі SQL створити таблицю “Рецепти”.
9	Створити базу даних на тему “Магазин квітів”. У режимі SQL створити таблицю “ Рослини ”.
10	Створити базу даних на тему “Магазин іграшок”. У режимі SQL створити таблицю “ Іграшки”.

Контрольні питання

1. Що таке мова запитів SQL?
2. Для чого застосовується мова SQL?
3. Що робить оператор CREATE?
4. Які види оператора CREATE Ви знаєте?
5. Які типи полів бувають у базах даних?
6. Що станеться якщо при виконанні оператора CREATE таблиця з введеним ім'ям буде вже існувати?

Лабораторна робота №6

Мова запитів SQL. Оператори ALTER, DELETE, DROP, INSERT.

Мета роботи - ознайомлення з мовою запитів SQL, ознайомлення з операторами ALTER, DELETE, DROP, INSERT CREATE. Вдосконалення навичок зі створення запитів мовою SQL.

Теоретичні відомості

Для того щоб повноцінно працювати за БД треба вміти не тільки створювати бази даних і таблиці, а й вміти вносити в них дані, видаляти дані, видаляти таблиці. Познайомимось з цими операторами.

Оператор ALTER TABLE забезпечує можливість змінювати структуру існуючої таблиці. Наприклад, можна додавати або видаляти стовпці, створювати або знищувати індекси або перейменовувати стовпці або саму таблицю.

Загальний синтаксис цієї команди ALTER TABLE tbl_name alter_spec [, alter_spec ...] команда складна має багато різних варіантів використання, див. додаткову літературу.

Наприклад

```
ALTER TABLE t1 RENAME t2;
```

Перейменувати таблицю t1 у таблицю t2

```
ALTER TABLE t2 ADD d INT;
```

Додати до таблиці t2 нове поле з іменем d та типом ціле.

Оператор DROP DATABASE видаляє усі таблиці у зазначеній базі даних і саму базу. Будьте ДУЖЕ уважні при роботі з цією командою!

Загальний синтаксис цієї команди DROP DATABASE db_name, де db_name – ім'я бази даних яку треба знищити.

Оператор DROP TABLE видаляє одну або декілька таблиць. Всі табличні дані і визначення видаляються, так що будьте уважні при роботі з цією командою!

Загальний синтаксис цієї команди DROP TABLE tbl_name [, tbl_name,...], де tbl_name – одна або декілька таблиць для знищення.

Оператор INSERT вставляє нові рядки в існуючу таблицю. Форма даної команди INSERT ... VALUES вставляє рядки відповідно зазначеним в команді значеннями.

Загальний синтаксис цієї команди INSERT INTO tbl_name (col_name,...) VALUES (...),(...),... де tbl_name – імя таблиці в яку вставляти значення, col_name – імена полів куди вставляти , у дужках вказують що вставляти.

Наприклад

```
INSERT INTO ForSale (col1,col2) VALUES(15, 2);
```

Вставити в таблицю ForSale, в поле з іменем col1 число 15, а у поле з іменем col2 – число 2.

Оператор DELETE видаляє з таблиці рядки, що задовольняють заданим в умовам.

Якщо оператор DELETE запускається без визначення WHERE, то видаляються всі рядки.

Загальний синтаксис цієї команди DELETE FROM table_name [WHERE where_definition], де де tbl_name – імя таблиці, where_definition - умова

Наприклад

```
DELETE FROM t3 WHERE col1 >0;
```

Видалить всі рядки з таблиці t3, які більші нуля.

```
DELETE FROM t3 WHERE col2 LIKE "a*";
```

Знищить всі записи які починаються з літри a

```
DELETE FROM t3 WHERE col2 LIKE "*s";
```

Знищить всі записи які закінчуються на літру s.

Робоче завдання

1. Запустити програму Microsoft Access.
2. Створити БД.
3. Створити таблицю у режимі SQL.
4. Перейменуйте таблицю за допомогою оператора ALTER.

5. Заповнити таблицю даними за допомогою оператора INSERT.

6. Видалить деякі рядки за певної умови, умову взяти з таблиці 6.1

У новій (чистій) базі даних створити таблицю аналогічну першій таблиці з лабораторної роботи №1. Не використовуйте оператори DROP, але про них слід знати.

Таблиця 6.1 – Завдання по варіантах.

№ варіанту		
	Завдання для таблиці	Умова видалення
1	Створити базу даних на тему “Деканат”. У режимі SQL створити таблицю “Список викладачів”	Прізвище починається на літру П
2	Створити базу даних на тему “Довідник рослин”. У режимі SQL створити таблицю “Рослин”.	Назва починається на літри З
3	Створити базу даних на тему “Реєстратура комерційної лікарні”. У режимі SQL створити таблицю “Лікарі”.	Ім’я починається на літру А
4	Створити базу даних на тему “Магазин запчастин”. У режимі SQL створити таблицю “Назва запчастини”.	Назва закінчується на літру Б
5	Створити базу даних на тему “Студенти”. У режимі SQL створити таблицю “Картки студентів”.	Ім’я закінчується на літру В
6	Створити базу даних на тему “Юридичний відділ”. У режимі SQL створити таблицю “Картки юристів”.	Ім’я починається на літру С
7	Створити базу даних на тему “Автомайстерня”. У режимі SQL створити таблицю “Операції”.	Назва закінчується на літру У
8	Створити базу даних на тему “Бухгалтерія ресторану”. У режимі SQL створити таблицю “Рецепти”.	Назва починається на літри С

9	Створити базу даних на тему “Магазин квітів”. У режимі SQL створити таблицю “ Рослини ”.	Назва закінчується на літри Ь
10	Створити базу даних на тему “Магазин іграшок”. У режимі SQL створити таблицю “ Іграшки”.	Назва починається на літри В

Контрольні питання

1. Що Ви знаєте про оператор ALTER?
2. Що Ви знаєте про оператор DELETE?
3. Що Ви знаєте про оператор DROP?
4. Що Ви знаєте про оператор INSERT?
5. Чому треба бути обережним з оператором DELETE,?
6. Чому треба бути обережним з оператором DROP?

Лабораторна робота №7

Мова запитів SQL. Оператори SELECT проста форма.

Мета роботи - ознайомлення з мовою запитів SQL, ознайомлення з оператором SELECT.

Теоретичні відомості

Під час роботи за БД велику частину часу займають різного типу вибірки. Основним оператором для здійснення певної вибірки є оператор SELECT

SELECT застосовується для вилучення рядків, вибраних з однієї або декількох таблиць. Вираз `select_expression` задає стовпці, в яких необхідно проводити вибірку.

Загальний синтаксис цієї команди `SELECT fields, .... FROM mytable;` де `fields` – список полів які треба вибрати, `mytable` – ім'я таблиці з якої проводити вибірку.

Наприклад

```
SELECT last_name, first_name FROM mytable;
```

Вибрати поля `last_name` та `first_name` з таблиці `mytable`

```
SELECT * FROM mytable;
```

Вибрати всі поля з таблиці `mytable`

Робоче завдання

1. Запустити програму Microsoft Access.
2. Відкрити БД яка була створена у роботі №1.
3. Зробити вибірку, завдання взяти з таблиці 7.1

Таблиця 7.1 – Завдання по варіантах.

№ варіанту	Завдання
1	Взяти таблицю „Список викладачів,, вивести прізвище, ім'я та телефон викладачів.
2	Взяти таблицю „Назви рослин ,, вивести тільки назви рослин українською мовою. Другим запитом вивести всі поля.
3	Взяти таблицю „Лікарі,, вивести прізвище, ім'я та телефон.
4	Взяти таблицю „Назва запчастини,, вивести тільки назви (без номера за каталогом) Другим запитом вивести всі поля.
5	Взяти таблицю „Картки студентів,, вивести прізвище, ім'я та телефон.
6	Взяти таблицю „Картки юристів,, вивести прізвище, ім'я та телефон.
7	Взяти таблицю „Список робітників,, вивести прізвище, ім'я та телефон.
8	Взяти таблицю „Назва рецепту,, вивести тільки назви рецептів (без опису приготування). Другим запитом вивести всі поля.
9	Взяти таблицю „Назви рослин,, вивести тільки назви рослин українською мовою. Другим запитом вивести всі поля.
10	Взяти таблицю „Назва іграшки,, вивести тільки назви іграшок (без опису). Другим запитом вивести всі поля.

Контрольні питання

1. Що Ви знаєте про оператор SELECT?
2. Як оператором SELECT вивести всі поля?
3. Як оператором SELECT вивести тільки поле з іменем col1?
4. Напишіть оператор SELECT який дозволить вивести поля col1 та col2?

Лабораторна робота №8

Мова запитів SQL. Оператори SELECT повна форма, оператори WHERE, ORDER.

Мета роботи – продовження ознайомлення з оператором SELECT та ознайомлення з операторами WHERE та ORDER.

Теоретичні відомості

Через те що оператор SELECT є одним з найчастіше використовуваних операторів мови SQL він має багато форм. Це один з найпотужніших операторів.

Познайомимось з деякими можливостями оператора SELECT.

Якщо до оператора SELECT вкінці додати оператор ORDER то результат оператора SELECT можна відсортувати за певним критерієм.

Наприклад

```
SELECT last_name, first_name FROM mytable ORDER by last_name;
```

Виведе список відсортований за полем last_name.

```
SELECT last_name, first_name FROM mytable ORDER by last_name DESC;
```

Виведе список відсортований за полем last_name але не за зростанням, а за зменшенням.

Як правило, під час реальної роботи не буває простих вибірок, в реальних задачах доводиться вибирати значення за певними критеріями.

Для того щоб здійснити вибірку за певними критеріями до оператора SELECT додають оператор WHERE. Цей оператор дозволяє вибирати дані з БД використовуючи певні умови.

Наприклад

```
SELECT * FROM Products WHERE CategoryID = 4
```

З таблиці Products вибрати всі записи у полі CategoryID яких стоїть значення 4

Загальний синтаксис цієї команди WHERE expression1 [{AND | OR} expression2 [...]], де expression1, expression2 – умовні вирази, AND, OR оператори логічного поєднання декількох виразів.

Умови можуть бути такими

Таблиця 8.1 – умовні вирази оператора WHERE

Оператор	Умова
<	менше
>	більше
=	дорівнює
<>	не дорівнює
<=	менше або дорівнює
>=	Більше або дорівнює

Наприклад

```
SELECT * FROM Products WHERE CategoryID = 2 AND SupplierID > 10
```

Виведе всі поля з таблиці Products, номер категорії яких дорівнює 2 і одночасно рядок SupplierID більший числа 10

Для фільтрації обробки текстових полів використовують оператор LIKE, за ним вказують маску-фільтр. У масці символ '%' (відсоток) або '*' (зірочка) замінює будь-яку послідовність символів, а символ '_' (підкреслення) -один довільний символ.

Наприклад

```
SELECT CompanyName, ContactName FROM Customers WHERE  
CompanyName LIKE 'M%'
```

Виведе усі компанії назва яких починається на літеру M

За допомогою оператора SELEC та оператора WHERE можна виводити складні запити з декількох таблиць.

Наприклад

```
SELECT Name, Score FROM Students, Scores WHERE Name LIKE 'K%'  
AND StudentID = ID AND Algebra < 3
```

Виведе список студентів прізвище яких починається на літеру К і оцінки за алгебру менші за 3 бали.

Робоче завдання

1. Запустити програму Microsoft Access.
2. Відкрити БД яки була створена у роботі №1.
3. Зробити вибірку, завдання взяти з таблиці 8.1

Таблиця 8.1 – Завдання по варіантах.

№ варіанту	Завдання			
	Таблиці для БД	Сортування	Умова 1	Умова 2
1	БД “Деканат”. Взяти такі таблиці. “Список студентів”, “Оцінки”.	Від більшого до меншого	Прізвище починається на літеру П	Алгебра більше 5 балів
2	БД “Довідник рослин”. Взяти такі таблиці. “Назви рослин”, “Назви місяців”	Від меншого до більшого	Назва починається на літри З	Цвіте у липні
3	БД “Реєстратура комерційної лікарні”. Взяти такі таблиці. “Лікарі”, “Вартість послуг”, “	Від більшого до меншого	Ім’я починається на літеру А	Вартість послуг менша за 100
4	БД “Магазин запчастин”. Взяти такі таблиці. “ Назва запчастини”, “Вартість послуг ”.	Від меншого до більшого	Назва закінчується на літеру Б	Вартість послуг > 200
5	БД “Студенти”. Взяти такі таблиці. “Картки студентів”, “Стипендія”.	Від більшого до меншого	Ім’я закінчується на літеру В	Стипендія більша за 300

6	БД “Юридичний відділ”. Взяти такі таблиці. “Картки юристів”, “Витрати”,	Від меншого до більшого	Ім’я починається на літру С	Витрати більше за 5000
7	БД “Автомайстерня”. Взяти такі таблиці. “ Назва операції”, “Список робітників”, “ Вартість послуг ”.	Від більшого до меншого	Назва закінчується на літру У	Вартість > 50
8	БД “Бухгалтерія ресторану”. Взяти такі таблиці. “Назва рецепту”, “Вартість роботи”.	Від меншого до більшого	Назва починається на літри С	Вартість < 150
9	БД “Магазин квітів”. Взяти такі таблиці. “ Назви рослин ”, “ Вартість послуг ”	Від більшого до меншого	Назва закінчується на літри Ь	Вартість <= 20
10	БД “Магазин іграшок”. Взяти такі таблиці. “ Назва іграшки”, “Вартість”.	Від меншого до більшого	Назва починається на літри В	Вартість >= 15

Контрольні питання

1. Які оператори використовуються разом з оператором SELECT?
2. Що робить оператор WHERE?
3. Що робить оператор ORDER?
4. Що робить оператор DESC?
5. Як використовують оператор LIKE?
6. Які існують умовні оператори?
7. Як об’єднати декілька умов у одному запиті?
8. Чи можна будувати запити одразу до кількох таблиць?
9. Як відсортувати таблицю у порядку від більшого до меншого?

Лабораторна робота №9

Мова запитів SQL. Оператори SELECT, GROUP, HAVING.

Агрегатні функції. SELECT з підзапитом.

Мета роботи – продовження ознайомлення з оператором SELECT та ознайомлення з операторами GROUP та HAVING та агрегатними функціями.

Теоретичні відомості

Агрегатні функції – це спеціалізовані функції які можна використовувати у SQL запитах.

Агрегатна функція проводить обчислення над одиничним результатом від безлічі записів.

Наприклад

```
SELECT max(price) FROM Toys;
```

Видасть вартість найдорожчої іграшки у таблиці магазину іграшок.

У таблиці 8.1 представлено список агрегатних функцій які частіше всього використовують.

Таблиця 9.1 – агрегатні функції

Функція	Призначення
AVG	Обчислює середнє
COUNT	Обчислює кількість непустих значень у цій колонці
MAX	Обчислює найбільше значення у колонці
MIN	Обчислює найменше значення у колонці
SUM	Обчислює суму значень у колонці

Для обчислення сумарних значень на основі даних однієї або декількох таблиць можна використовувати пропозицію GROUP BY, що має такий синтаксис:

GROUP BY {column1} [, ...] де column1 – список полів

Наприклад, наступний запит пов'язує дві таблиці, сортує їх по полю CustomerID, для кожного значення CustomerID створює один рядок в результуючому наборі даних і обчислює кількість значень поля OrderID для кожного значення CustomerID:

```
SELECT Customers.CustomerID,  
COUNT (Orders.OrderID)  
FROM Customers INNER JOIN Orders  
ON Customers.CustomerID = Orders.CustomerID  
GROUP BY Customers.CustomerID
```

Оператор HAVING

Визначає умова пошуку для групи або статистичного вираження. Оператор HAVING можна використовувати тільки в інструкції SELECT. Оператор HAVING зазвичай використовується в реченні GROUP BY. Коли GROUP BY не використовується, оператор HAVING працює так само, як і оператор WHERE..

Загальний синтаксис HAVING умова_пошуку

Приклад

```
SELECT SalesOrderID, SUM(LineTotal) AS SubTotal  
FROM Sales.SalesOrderDetail  
GROUP BY SalesOrderID  
HAVING SUM(LineTotal) > 100000.00  
ORDER BY SalesOrderID ;
```

У цьому прикладі з таблиці SalesOrderDetail вибирається сума всіх полів SalesOrderID, значення яких перевищує \$ 100000.00.

Робоче завдання

1. Запустити програму Microsoft Access.
2. Відкрити БД яки була створена у роботі №1.
3. Зробити вибірку, завдання взяти з таблиці 9.1

Таблиця 9.2 – Завдання по варіантах.

№ варіанту	Завдання	
	Таблиці для БД	Завдання для вибірки
1	БД “Деканат”. Взяти такі таблиці. “Список студентів”, “Оцінки”.	Вивести список студентів і порахувати їх рейтинг (суму всіх оцінок по всіх предметах)
2	БД “Довідник рослин”. Взяти такі таблиці. “Назви рослин”, “Вартість ”	Вивести список квітів і порахувати їх вартість по типу рослини
3	БД “Реєстратура комерційної лікарні”. Взяти такі таблиці. “Лікарі”, “Вартість послуг”, “	Вивести список лікарів і порахувати вартість їх послуг по видах
4	БД “Магазин запчастин”. Взяти такі таблиці. “ Назва запчастини”, “Вартість послуг ”.	Вивести список запчастин і порахувати їх вартість по марках авто
5	БД “Студенти”. Взяти такі таблиці. “Картки студентів”, “Стипендія”.	Вивести список студентів і порахувати їх рейтинг (суму всіх оцінок по всіх предметах)
6	БД “Юридичний відділ”. Взяти такі таблиці. “Картки юристів”, “Витрати”,	Вивести список юристів і порахувати вартість їх послуг по видах діяльності
7	БД “Автомайстерня”. Взяти такі таблиці. “ Назва операції”, “Список робітників”, “ Вартість послуг ”.	Вивести список майстрів і порахувати вартість їх послуг по видах діяльності

8	БД “Бухгалтерія ресторану”. Взяти такі таблиці. “Назва рецепту”, “Вартість роботи”.	Вивести список рецептів і порахувати середню вартість страви по категоріях
9	БД “Магазин квітів”. Взяти такі таблиці. “ Назви рослин ”, “ Вартість послуг ”	Вивести список квітів і порахувати середню вартість складання букету з них
10	БД “Магазин іграшок”. Взяти такі таблиці. “Назва іграшки”, “Вартість”.	Вивести список іграшок і порахувати середню вартість іграшки у категорії

Контрольні питання

1. Для чого використовують агрегатні функції?
2. Як порахувати середнє значення у стовпчику?
3. Як знайти максимальне значення у стовпчику?
4. Що робить функція COUNT?
5. Що робить оператор GROUP?
6. Що робить оператор HAVING?