## МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ЧЕРКАСЬКИЙ ДЕРЖАВНИЙ ТЕХНОЛОГІЧНИЙ УНІВЕРСИТЕТ КАФЕДРА ІНФОРМАЦІЙНОЇ БЕЗПЕКИ ТА КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ

## ЛАБОРАТОРНИЙ ПРАКТИКУМ

з дисципліни

## «ОРГАНІЗАЦІЯ БАЗ ДАНИХ»

для студентів напряму 6.050102 – комп'ютерна інженерія усіх форм навчання

Затверджено на засіданні кафедри інформаційної безпеки та комп'ютерної інженерії, протокол №9 від 08.02.2016 р., та Методичною радою ЧДТУ, протокол № \_\_\_\_ від \_\_\_\_2016 р.

Укладачі: Куницька Світлана Юріївна, к.т.н., доцент Бабенко Віра Григорівна, к.т.н., доцент

Рецензент: Рідкокаша А.А., к.т.н., доцент кафедри програмного забезпечення автоматизованих систем Черкаського державного технологічного університету.

Лабораторний практикум з дисципліни «Організація баз даних» для студентів напрямку 6.050102 — комп'ютерна інженерія усіх форм навчання / Укл.: С.Ю. Куницька, В.Г. Бабенко. – Черкаси: ЧДТУ, 2016. – 162 с.

### **3MICT**

ВСТУП5
1. Вимоги до оформлення виконання та захисту лабораторних робіт
ТЕОРЕТИЧНА ЧАСТИНА9
2. Тема №1. "Створення таблиць. Використання форм для введення та
редагування даних"
2.1. Завдання9
2.2. Теоретичні відомості12
2.3. Хід виконання роботи та результати завдань14
3. Тема №2. "Проектування багатотабличних баз даних"
3.1. Завдання
3.2. Теоретичні відомості
3.3. Хід виконання роботи та результати завдань
4. Тема №3. "Підпорядкованні форми. Форматування форм"
4.1. Завдання
4.2. Теоретичні відомості 52
4.3. Хід виконання роботи та результати завдань
5. Тема №4. "Побудова запитів та звітів. Створення кнопок по управлінню та
форм"
5.1. Завдання
5.2. Теоретичні відомості64
5.3. Хід виконання роботи та результати завдань
6. Тема №5. "Мова SQL – створення запитів. Основні оператори мови
SQL"
6.1. Завдання
6.2. Теоретичні відомості
6.3. Хід виконання роботи та результати завдань
7. Тема №6. "Організація зв'язку з базою даних типу Microsoft Access засобами
ADO"105

7.1. Завдання	105
7.2. Теоретичні відомості	106
7.3. Хід виконання роботи та результати завдань	109
ПРАКТИЧНА ЧАСТИНА	115
Лабораторна робота №1	115
Лабораторна робота №2	117
Лабораторна робота №3	118
Лабораторна робота №4	120
Лабораторна робота №5	122
Лабораторна робота №6	124
8. Індивідуальні завдання	125
9. Контрольні питання	153
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	156
ДОДАТКИ	

#### ВСТУП

Лабораторний практикум розроблений для студентів напряму 6.050102 – комп'ютерна інженерія та призначений для формування навиків роботи з СУБД Access, а також направлений на практичне освоєння головних принципів роботи з реляційними базами даних.

Основною метою викладання дисципліни «Організація баз даних» - є засвоєння організації, принципів і методів побудови та застосування баз даних, збереження, обробки, перетворення даних та вміння створювати запити мовою SQL і формувати форми та звіти.

Основні завдання:

• ознайомитися із принципами моделювання предметного середовища для різних моделей даних;

• вивчити організацію реляційних баз даних;

• засвоїти основні команди та інструкції мови SQL;

• навчитися використовувати методи забезпечення, контролю та відновлення цілісності даних;

• вивчити фізичні та логічні методи захисту даних від несанкціонованого втручання.

СУБД Access є професіональною версією офісної системи Microsoft Office, використовується для розробки діалогових офісних інформаційних систем, що використовує реляційну модель даних та графічний інтерфейс Windows. Більшість нескладних інформаційних задач вирішуються без програмування, але для створення більш складних – використовують мову програмування Visual Basic for Application.

База даних (БД) в СУБД Access представлена у вигляді сукупності об'єктів, які зберігаються в одному файлі (mdb). У СУБД Access допускається тільки ті типи об'єктів, які найбільш розповсюджені при роботі з базами даних: таблиці, форми, запити, звіти, макроси, програмні модулі.

Головним об'єктом в БД є *таблиця*, що зберігає данні о предметах реального часу. Інші типи об'єктів – це різноманітні способи представлення інформації з таблиць (форми, звіти, динамічні набори) або дії над таблицями (запити, макроси, модулі).

Запит – це об'єкт, який виводить данні з таблиць згідно інструкції користувачем (вибірка, зміна або групування даних збережених по різним таблицям). Відповідь на запит виглядає також як таблиця та називається динамічним набором записів.

Форма – це об'єкт, необхідний для введення та перегляду записів в потрібному форматі: виділити різним кольором данні, вивести списки значень, вивести інформацію про помилки при введені даних, відображення результатів як на екран так і на друк. При описанні форми використовується методика візуального програмування.

Звіт – це об'єкт, необхідний для друку даних, який дозволяє групувати записи, підрахувати та вивести текучі і загальні значення. Звіт можна оформити використовуючи різноманітні шрифти Windows, а також прикрасити малюнками та діаграмами.

*Макрос* – це набір з однієї або декількох макрокоманд, які виконують Access без участі користувача. Дозволяють автоматизувати постійно виконуючі операції та узгодити роботу різних об'єктів без програмування.

*Модуль* – це набір процедур та функцій згідно мови Visual Basic (Access Basic), які зберігаються як одне ціле. Модулі використовують для створення більш складних інформаційних систем.

В результаті студент повинен уміти:

• розробляти структури та системи управління базами даних, застосовуючи сучасні методи побудови та управління БД;

• виконувати розробку реляційних та розподілених баз даних, застосовуючи сучасні мови та системи управління БД типу Visual FoxPro, Access, SQL;

• забезпечувати оптимальне розміщення даних та їх прискорений пошук в базі даних, застосовуючи методи автоматизованого проектування БД.

Навчання студентів ведеться в формі лекцій, лабораторних, практичних та самостійних занять з використанням сучасних персональних комп'ютерів. Самостійні заняття зводяться до індивідуального вивчення додаткової літератури згідно навчального плану.

## 1. Вимоги до виконання, оформлення та захисту лабораторних робіт

Лабораторні роботи, що представлені в лабораторному практикумі, призначені для формування навиків роботи з СУБД Access, а також направлені на практичне освоєння головних принципів роботи з реляційними базами даних та мовою SQL.

При виконанні кожної лабораторної роботи студент опрацьовує розділ з теоретичними відомостями, аналізує поставлену задачу, створює табличні форми, вносить данні в поля таблиці, вчиться проектувати багато табличні дані, завдяки основним командам та інструкціям мови SQL моделює реляційну базу даних в цілому згідно отриманого індивідуального завдання.

Після виконання кожної лабораторної роботи студент оформлює звіт згідно ДСТУ 3008-95 *(«Документація. Звіти у сфері науки і техніки. Структура і правила оформлення)* на аркушах формату А4 у запропонованій нижче формі і захищає лабораторну роботу у викладача при наявності електронної версії бази даних.

Кожна лабораторна робота повинна мати:

- <u>титульний лист</u> з порядковим номером роботи та відображеною темою цієї роботи;
- <u>зміст</u> (лист 2) з наступними розділами та відповідними сторінками: вступ, теоретичні відомості, хід виконання роботи та результати завдань, висновки;
- <u>вступ</u> (лист 3) включає в себе: назву теми, мету та завдання;
- <u>теоретичні відомості</u> (лист 4) обсягом описання 2-3 сторінки;
- <u>хід виконання роботи та результати завдань</u> (лист N);
- <u>висновки</u> (лист N).

В розділі «*Теоретичні відомості*» студент детально описує процес створення бази даних - розширено описує як створити табличні форми, як відбувається внесення табличних даних в поля таблиці згідно властивостям, як спроектувати багато табличні дані, як побудувати запити та звіти, як створити кнопки по управлінню та форми, як завдяки основним командам та директивам мови SQL змодельована реляційна база даних в повному обсязі згідно отриманого індивідуального завдання.

В розділі <u>«Хід виконання роботи та результати завдань»</u> студент поступово описує кожне із завдань, тобто детальний опис всіх необхідних вимог, які допомагають отримати певний результат завдання згідно отриманого варіанту. В кінці кожного опрацьованого завдання повинен бути результат у вигляді друкованого вікна із заповненою формою табличними даними. Підписувати та нумерувати результат необхідно єдиною нумерацією наскрізь в межах однієї лабораторної роботи.

У *«Висновках»* стисло описується створення бази даних, необхідність побудови запитів та звітів, переваги створення кнопок по управлінню та можливості мови SQL завдяки основним інструкціям і командам SELECT, UPDATE, DELETE отримати певні данні. Обсяг викладення 1 сторінка.

По закінченні курсу лабораторних робіт студент представляє звіти оформлені належним чином по кожній з них, зібравши їх у єдину папку із загальним титульним аркушем *«Журнал звітів з лабораторних робіт по дисципліні «ОРГАНІЗАЦІЯ БАЗ ДАНИХ» на тему: «…»*.

#### ТЕОРЕТИЧНА ЧАСТИНА

# 2. ТЕМА 1: «Створення таблиць. Використання форм для введення і редагування даних»

**Мета:** Вивчити процес створення бази даних та об'єктів типу «*Таблица*» за допомогою «*Конструктора*», вивчити методи визначення властивостей полів таблиці і здійснювати маніпуляції з ними, заповнювати таблиці даними та редагувати. Оволодіти способами задання форм, створювати кнопки переміщення по записах, здійснювати перехід з форми в режим таблиці і роботу зі списками.

#### 2.1. Завдання

#### Завдання 1:

- створити на диску порожню базу даних і назвати її «фамилия группа БАЗА ДАННЫХ»;

- заповнити стовпці таблиці «Имя поля» і «Тип данных» як показано нижче:

Таблиця 1- Структура таблиці

Имя поля	Тип данных
Фамилия	Текстовый (щелкните
	мышкой)
Имя	Текстовый
Отчество	Текстовый
Дата	Дата/время
рождения	
Адрес	Текстовый
Телефон	Текстовый
Группа	Числовой

- назвати таблицю «фамилия\_группа\_база\_таблица\_анкета».

#### Завдання 2:

- визначити властивості полів на вкладці «Общие»:

Ποπο														
поле				~ .										
		Свойства полей												
		Формат	Маска	Подпись	Значение по	Обязательное	Индексированное							
	Размер	поля	ввода		умолчанию	поле	поле							
Фамилия	30			Фамилия		да	Да (совпадения							
							допускаются)							
Имя	12			Имя		да	нет							
Отчество	30			Отчество		нет	нет							
Дата		Краткий	Краткий	Дата		да	нет							
рождения		формат	формат	рождения										
-		даты		-										
Адрес	50			Адрес		да	нет							
Телефон	6					нет	нет							
Группа	Целое	основной		Группа	Номер вашей	да	Да (совпадения							
					группы		допускаются)							

#### Таблиця 2 – Заповнення властивостей полів

#### Завдання 3:

- відкрити таблицю «Вид/Режим таблицы». Заповнити рядки;
- перейменувати таблицю в «Анкета».

#### Завдання 4:

- в режимі «Конструктор» на основі створеної таблиці ввести список

прізвищ студентів групи.

Таблиця 3 – Внесення списку в таблицю

Тип элемента	Поле со списком
управления	
Тип источника	Таблица/запрос
строк	
Источник строк	<ol> <li>Щелкните мышкой,</li> <li>Нажмите кнопку</li> <li>В появившемся окне "построителя запросов":         <ul> <li>добавить таблицу "Анкета",</li> <li>закрыть окно "добавление таблицы"</li> <li>перетащить с помощью мыши поле "фамилия" в "поле" бланка запроса</li> </ul> </li> </ol>
-	• в поле "вывод на экран" должна стоять галочка 4
Ограничиться	Нет
списком	

#### Завдання 5:

- створити форму для заповнення таблиці «Анкета».
- вибрати необхідну «Автоформу».

#### Завдання 6:

- розібратися з кнопками переміщення по записах:

- перехід на один запис вперед;
- перехід на останній запис;
- перехід на один запис назад;
- перехід на чисту сторінку, де можна ввести новий запис;
- перехід на перший запис.

#### Завдання 7:

- створити таблицю «Группы» для введення груп, що навчаються на вашому курсі;

- створити структуру таблиці, тобто заповнити стовпці «Имя поля» і «Тип данных» наступним чином:

Таблиця 4 – Заповнення стовпців таблиці

Имя поля	Тип данных
Группа	Текстовый (щелкните мышкой)

#### Завдання 8:

 створити форму для заповнення таблиці «Группы» та заповнити цю форму існуючими групами певного курсу.

#### Завдання 9:

- зверніть увагу на те, що якщо ви в полі «фамилии» ввести нове прізвище, то воно не з'явиться в списку прізвищ, що розкриваються. Переконайтеся в цьому.

- створити список який автоматично заповнюється прізвищами, таким чином, щоб при введенні нового прізвища це автоматично потрапляло до списку, і згодом це прізвище можна було б вибрати зі списку, а не набирати заново.

#### Завдання 10:

- створити два макроси з ім'ям «Добавление» та «Закрытие»;

- перевірте роботу створених макросів по автоматичному самозапису списку із форми «Анкета»;

- введіть нове прізвище в полі *«фамилия»* заповнивши попередньо обов'язкові поля. Закрити форму **«Анкета»**;

- перевірити список всіх прізвищ, включаючи нову. Якщо виконали, то зберегти форму із змінами.

#### 2.2. Теоретичні відомості

MS Access підтримує реляційну модель даних, тобто база даних складається з множини взаємопов'язаних відношень, де кожному відношенню відповідає окрема таблиця.

Таблиця має стовбці та рядки, де стовбці таблиці відображають загальні властивості або характеристики об'єктів реального світу і які моделюються з використанням певної таблиці, а *рядки* таблиці відповідають різним варіантам реальних об'єктів на називають їх записами. Для кожної окремої таблиці визначається первинний ключ, тобто це може бути стовбець або набір стовбців, що визначають конкретний рядок таблиці. Перш ніж створити таблицю, необхідно відкрити базу даних, в якій MS Access буде зберігати цю таблицю. Потім у вікні бази даних слід вибрати закладку «ТАБЛИЦА», натиснути кнопку «Создать», а потім у вікні діалогу вибрати необхідний режим створення таблиці: *Майстер* або *Конструктор*.

MS Access пропонує велику кількість таблиць – прикладів, що містять дані з різних областей. У списку *«Образцы полей»* з'явиться список полів даних, які можна виділити і помістити в список *«Поля новой таблицы»*. Якщо імена полів недостатньо інформативні, їх можна змінити. Далі в режимі *Мастер* MS Access пропонує встановити первинний ключ.

Для зміни структури таблиці необхідно скористатися режимом *Конструктор*. Поля таблиці розташовані один під одним. Для кожного поля – ім'я, тип та опис. Поле первинного ключа таблиці, як правило, може мати тип

даних «*Счетчик»* (AutoNumber), який гарантує, що під час введення даних MS Access автоматично пронумерує рядки таблиці в зростаючій послідовності.

В режимі *Конструктора* можна змінити наявні поля або додати нові. Якщо таблиця вже містить дані, то при зміні структури таблиці вони губляться тільки у виняткових випадках. MS Access попереджає про це відповідним повідомленням.

В іменах полів заборонено використовувати символи '!', '.', '[', '] '. Максимальна довжина імені 64 символу. Не рекомендується використовувати в іменах полів пустоти, оскільки в подальшому некоректно будуть виконуватися запити. В якості типу даних поля можна використовувати наступні стандартні типи: чисельний, текстовий, лічильник, поле МЕМО, грошовий, логічний, дата/час, поле об'єкта ОLE. Для кожного поля задаються додаткові характеристики. Наприклад, для текстових полів визначається розмір поля, а для чисельного – довжина розрядної сітки і кількість розрядів після коми.

При створенні таблиць за допомогою *Конструктора* MS Access автоматично прописує первинний ключ. Щоб визначити в якості первинного ключа інший набір полів, слід виділити відповідні поля і натиснути піктограму «ОПРЕДЕЛИТЬ КЛЮЧ» на панелі інструментів. Під час введення даних в поля, які визначені в якості первинного ключа, MS Access автоматично стежить за тим, щоб вводилися тільки унікальні значення. МЕМО-поля і поля об'єкта OLE не можуть бути первинними ключами.

Форми є основним засобом створення інтерфейсу користувача, який забезпечує найбільш зручний спосіб представлення, перегляду, редагування даних і управління ходом виконання програми. Переваги форми для введення і редагування даних полягає в простоті і наочності, так як записи таблиці або запиту представлені у формі в зручному вигляді. В режимі **Формы** можна повною мірою скористатися можливостями, що надаються графічним середовищем Windows. Крім того, можна створювати необхідні елементи форм для вирішення певного завдання. У формах користувач самостійно визначає, скільки місця має бути визначено для представлення конкретної інформації. З

їх допомогою значно спрощується внесення змін, додавання та видалення даних з бази.

#### 2.3. Хід виконання роботи та результати завдань

Створена база даних та всі форми представлення структур даних розроблені за допомогою Microsoft Access 2010.

#### Завдання 1.

Перш ніж створити таблицю, необхідно відкрити базу даних, в якій MS Access буде її зберігати. Потім у вікні бази даних вибираємо закладку «ТАБЛИЦА», натиснувши кнопку «Создать», а потім у вікні діалогу вибираємо режим створення таблиці Конструктор і вже створюємо робочу таблицю «Таблиця1», яку перейменовуємо в назву «Артеменко\_КМ\_125\_база\_таблица\_анкета». Вікно бази даних складається з п'яти вкладок, шоста вкладка з'являється при роботі з таблицею, конструктором або іншими елементами (рис. 1).

A   🚽	<b>*) •</b> (* •   <del>•</del>			100				Pa6	ота с таблі	ицами	Артеменко_КМ_125	БАЗА Д	АННЫХ : база д	анных (Access 2007)
Файл	Главная	Создание	Bł	нешние данные	Работа с б	азами	данн	ых	Конструкт	op				
Режим	Ключевое По поле	остроитель Г	ровери	⊒н Вставить ст ⇒ Удалить стр 3 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 5 4 5 4 5 4 5 4 5 5 5 5 5 5 5 5 5 5 5 5 5	гроки роки подстановку	Стра	<b>и</b> ница йств	Индексы	Создать і данн	2 макросы ых т	Переименовать или удалить макрос	Схема данных	Зависимости объектов	
Режимы			Серв	ис		Пока	зать и	ли скрыть	Событи	ія полей	і, записей и таблиц		Связи	
Bce of	ъекты Access		« 📃	Таблица1										
Поиск			ρ	Имя	поля			Тип данн	ных					Описан
Табли	цы	\$		Фамилия			Текс	товый						
Та	блица1			Имя			Текс	товый						
				Отчество			Текс	товый						
				Дата рождени	18		Дата	/время						
				Адрес			Текс	товый						
				Телефон			Текс	товый						
				Группа			Числ	овой						

Рисунок 1 – Створюємо базу даних

Після того, як заповнили стовпці «Имя поля» і «Тип данных» таблиця набуває наступного набуває вигляд наступної структури:

• СГАВИТЬ • Формат по образцу	Фильтр Я⊘Уда.	лить сортировку	<b>ү</b> Фильтр		Обновить Все т	🗙 Удалить 🔹 🧮 Д	ополнительно *	Пан	И 🔓 Выбрать т	ж <u>кч</u>	A * "/ ·
Режимы Буфер обмена 🕞		Сортировка и ф	бильтр			Записи			Найти		Формати
Все объекты Access 💿 « 🔳	Артеменко_КМ	125_база_табли	ца_анкета								×
Поиск Р 🖉	Код 👻	Фамилия 🔹	Имя	٣	Отчество 🔹	Дата рождения	• Адре	c 🔻	Телефон 🔻	Группа	🝷 Щелкни
Таблицы 🎄 👫	(Nº)										
Портеменко_КМ-125_6аза_та											

Рисунок 2 – Наповнення структури даними

#### Завдання 2.

На вкладці «Общие» визначаються властивості полів. Для цього необхідно в режимі *Конструктор* натиснути мишкою на потрібне поле і заповнити всі необхідні властивості цього поля:

Bce объекты Access 🛛 🕞 «		Артеменко_КМ-125_база_таблица_анкета							
Тоиск 🔎		Имя поля		Тип данных					
Тъблици	8	Код		Счетчик					
		Фамилия		Текстовый 🔻					
Артеменко_КМ-125_6аза_та		Има		Текстовый					
		Отчество		Тексторый					
		Лата рожления		Лата/время					
		Annos		Тонсторый					
		Адрес		Текстовыи					
		Телефон		Текстовыи					
		Группа		Числовой					
	Г	Общие Подстановка							
		Размер поля	30						
	4	Формат поля							
		Маска ввода							
		Тодпись	Фамилия						
		спорие на значение							
		Сообщение об ошибке							
		Обязательное поле	Да						
	1	Пустые строки	Да						
		Индексированное поле	Да (Допус	каются совпадения)					
		Сжатие Юникод	Да						
	F	Режим IME	Нет контр	оля					
		Режим предложений IME	Нет						
I		смарт-теги	1						

Рисунок 3 - Встановлення властивостей для поля «Фамилия»

Режимь	4	C	ерви	10		п	оказать или скрыть	Событи	ия полей,		
Bce o	бъекты Access 🛛 🕞	~<		Артеменко_КМ-1	125_6a	за_табл	ица_анкета				
Поиск		2	2	Имя п	юля		Тип данн	ых			
Табли		~	8	Код			Счетчик				
	Annual Khd 125 Care at	^		Фамилия			Текстовый				
	Артеменко_км-125_6аза_та	a		Имя			Текстовый	-			
				Отчество			Текстовый	Текстовый			
				Дата рождения			Дата/время				
				Алрес			Текстовый				
				Телефон			Текстовый				
				Группа			Числовой				
				, pyrnia			menobom				
				Общие Подста	ановка	•					
			P	азмер поля		12					
			Φ	ормат поля							
				аска ввода							
			-	одпись		Имя					
			2	начение по умолча	анию						
			Ċ	ообщение об оши	бке						
			o	бязательное поле		Да					
				устые строки		Да					
			И	ндексированное п	оле	Нет					
			C	Сжатие Юникод Да							
			P	ежим IME		Нет кон	нтроля				
			P	ежим предложени	й ІМЕ	Нет					
			C	март-теги							

Рисунок 4 - Встановлення властивостей для поля «Имя»

Bce объекты Access 🛛 🕞	«	Артеменко	_KM-125_6a	ва_таблиц	а_анкета
Тоиск	ρ	<u> </u>	1мя поля		Тип данных
Тэблицы	- 19	Код			Счетчик
		Фамилия			Текстовый
Артеменко_КМ-125_6аза_та	·   -	Имя			Текстовый
		Отчество			Текстовый
		Лата рожл	ения		Лата/время
		Annec	CIIIIII		Таксторый
		Толофон			Текстовый
		телефон			Текстовыи
		группа			числовои
					· · · · · · · · · · · · · · · · · · ·
		Общие ј	Подстановка	а	
		Размер поля		30	
		Формат поля			
		Маска ввода			
		Подпись		Отчество	
		Значение по у	молчанию		
		Условие на зна	ачение		
		Сообщение об	ошибке		
		Обязательное	поле	Пет	
		Инлексирован	ное поле	Нет	
		Сжатие Юника	од	Дa	
		Режим IME		Нет контр	оля
		Режим предло	жений IME	Нет	
		CHART TALL			

Рисунок 5 - Встановлення властивостей для поля «Отчество»

Режимы	C	ерв	ис	Пок	азать или скрыть	События п
Bce объекты Access	∍ «		Артеменко_КМ-125_6	аза_таблиц	ца_анкета	
Поиск	2	$\square$	Имя поля		Тип даннь	ix
Таблицы	~	8	Код		Счетчик	
Артеменко КМ-125 база	та		Фамилия		Текстовый	
	- Tu		Имя		Текстовый	
			Отчество		Текстовый	
			Дата рождения		Дата/время	-
			Адрес		Текстовый	
			Телефон		Текстовый	
			Группа		Числовой	
		<u> </u>				
			Общие Подстанов	ka		
		d	ормат поля	Краткий с	формат даты	
		N	Ласка ввода	00.00.000	);0;_	
			Іодпись	Дата рож	дения	
		3	начение по умолчанию			
			словие на значение			
			ообщение об ошибке	0.5		
				Да		
			ежим IME	Het KOHT	0.0.9	
		l i	ежим предложений IMI	Нет		
		Ċ	март-теги			
		E	ыравнивание текста	Общее		
		0	Отображать элемент вые	5 Для дат		



Все объекты Access	~~		Артеменко_КМ-125_6а	ва_таблица	а_анкета
Поиск	2		Имя поля		Тип данных
Таблицы	~	8	Код		Счетчик
	~		Фамилия		Текстовый
Артеменко_км-125_6аза_т	a		Имя		Текстовый
			Отчество		Текстовый
			Дата рождения		Дата/время
			Алрес		Текстовый 🔹
			Телефон		Текстовый
			Гоуппа		Числовой
			i pyrma		исловои
			Общие Подстановк	a	
		F	азмер поля	50	
		4	Рормат поля		
			Ласка ввода		
			Тодпись	Адрес	
		LIE	спорие на знашение		
			ообщение об ошибке		
			Обязательное поле	Да	
		ſ	Тустые строки	Да	
		L	Індексированное поле	Нет	
			жатие Юникод	Да	
		F	Режим IME	Нет контр	оля
		F	ежим предложений IME	Нет	
			март-теги		

Рисунок 7 - Встановлення властивостей для поля «Адрес»

Режимы	серв	ис	нока	азать или скрыть со	юыни
Все объекты Access 💿 «		Артеменко_КМ-125_6а	за_таблиц	а_анкета	
Поиск		Имя поля		Тип данных	
Таблицы	8	Код		Счетчик	
	1	Фамилия		Текстовый	
Apremenco_con-125_basa_ta		Имя		Текстовый	
		Отчество		Текстовый	
		Дата рождения		Дата/время	
		Адрес		Текстовый	
		Телефон		Текстовый	-
		Группа		Числовой	
		Общие Подстановк	а		
	F	Размер поля	12		
		рормат поля			
		Ласка ввода			
		Тодпись			
		саовие на значение			
		ообщение об ощибке			
		Обязательное поле	Нет		
	l r	Тустые строки	Дa		
	1	Індексированное поле	Нет		
	(	жатие Юникод	Да		
	F	Режим IME	Нет контр	оля	
	F	ежим предложений IME	Нет		
		март-теги			

Рисунок 8 - Встановлення властивостей для поля «Телефон»

Все объекты Access 🛛 😒 «		Артеменко_	KM-125_6a	за_таблиц	а_анкета
Поиск	$\left  \right  \ge$	И	мя поля		Тип данных
Таблицы	- 8	Код			Счетчик
		Фамилия			Текстовый
Apremenco_km-125_6asa_1a		Имя			Текстовый
		Отчество			Текстовый
		Дата рожде	ния		Дата/время
		Адрес			Текстовый
		Телефон			Текстовый
		Группа			Числовой 👻
		06,000			
		Общие П	одстановка	a  	
		Размер поля		Целое	
		Формат поля Число лесятичн	ых знаков	Авто	
	•	Маска ввода			
	ſ	Подпись		Группа	
	3	Значение по ум	юлчанию	125	
		Условие на знач	чение		
		Сообщение об	ошиьке	<i>Д</i> э	
		Инлексированн	ое поле	Да (Допус)	каются совпадения)
		Смарт-теги		La La crisci	contraget in the

Рисунок 9 - Встановлення властивостей для поля «Группа»

#### Завдання 3.

Оскільки вкладки «*Bud*» в Microsoft Access 2010 не існує потрібно вибрати вкладку «*Главная*», натиснути на кнопку «*Режим*» та обрати «*Режим таблицы*». Заповнити рядки в «*Режим таблицы*» (рис.10). При заповненні колонки «Дата рождения» роздільники прописуються автоматично.

Після цього вибрати вкладку «Файл», натиснути кнопку «Сохранить объект как», на пропозицію системи дати ім'я створеній таблиці ввести ім'я «Анкета» і «Сохранить».

<u>ежимы</u>	руфер оомена		13	( <u> </u>		сортировка и ф	илыр			заниси			паин	и		
Все объекты	Access 🔅	• «		Анкета												
Поиск		2	$\angle$	Код	-	Фамилия 👻	имя -	Отч	ество 🝷	Дата рожде 👻	Адрес 👻	Телефон	-	Группа	-	Щ
Таблицы		\$			1	Бондаренко	Евгений	Мак	симович	17.04.1995	Чехова, 23	3-47-53			125	
🛄 Анкета					2	Возный	Юрий	Cepr	еевич	09.06.1995	Смелянская, 5	3-77-22			125	
					3	Гушлевский	Александр	Оле	гович	11.03.1995	Добролюбова	3-65-11			125	
					4	Коваленко	Ирина	Вале	ериивна	02.12.1995	Орджоникидз	3-12-55			125	
					5	Кунов	Максим	Иван	ювич	09.09.1995	Вербовецкого	3-26-64			125	
					6	Лукашенко	Олег	Влад	цимирови	12.11.1995	Кармелюка, 3	3-27-74			125	
					7	Порублевый	Олексей	Дми	триевич	03.04.1995	Розы Люксем(	3-15-77			125	
					8	Сиваченко	Максим	Петр	ович	29.06.1995	Героев Днепр	3-63-54			125	
					9	Чупрун	Эдуард	Петр	ович	14.04.1995	Волкова, 27	3-22-65			125	
			*		(Nº)										125	

Рисунок 10 – Створення анкети та заповнення рядків таблиці

#### Завдання 4.

На основі створеної таблиці створюємо список прізвищ студентів групи. Для цього в режимі *Конструктор* для поля «фамилия» потрібно відкрити вкладку *«Подстановка»* і встановити необхідні властивості для поля «фамилия». Після цього в полі «фамилия» при натисканні мишею повинна з'явитися кнопка, що забезпечує вибір зі списку.

Все объекты Access 🛛 😒 «		Анкета				
Поиск			Имя поля		Тип данных	
Tafauuu	8	Код			Счетчик	
		Фамилия			Текстовый 🔻	-
Анкета		Има			Текстовый	
		Отчество			Текстовый	
		Дата рож	ления		Лата/время	
		Annec			Токсторый	
		Тарафан			Тенстовый	
		телефон			Текстовый	
		Труппа			Числовои	
		Общие	Подстановка	а		
	l íF	ип элемента	управления	Поле со сг	писком	
	T	ип источний	ка строк	Таблица и	ли запрос	
	L	1сточник стр	юк			
		Трисоединен	ный столбец	1		
		нисло столбы	IOE	1		
		аглавия стол	биов	нет		
		нисло строк	писка	16		
		Цирина спис	ка	Авто		
	•	Ограничитьс	я списком	Нет		
	F	азрешение і	нескольких зі	Нет		
	F	азрешить из	менение спи	Нет		
		рорма измен	ения элемен			
	T	олько значе	ния источни	Нет		

Рисунок 11 – Введення списку прізвищ студентів групи

При заповненні властивості «Источник строк» потрібно натиснути кнопку «…», після чого з'явиться вікно «Построитель запросов». Потім потрібно виконати такі дії (рис. 12):

- додати таблицю «Анкета»;
- закрити вікно «Додавання таблиці»;
- перетягнути за допомогою миші поле «фамилия» в «поле» бланка запиту;
- у полі «вывод на экран» повинна стояти галочка.

Р	зультаты Ги	п за
>>	П Анкета - Анкета : построитель запросов	
бласть навигации	Анкета Код Фамилия Имя Отчество Дата рождения Адрес Телефон	
0	Поле: Имя таблицы: Сортировка: Вывод на экран: Условие отбора: или:	

Рисунок 12 - Заповнення властивостей

Перевірка виконується наступним чином: при натисканні мишею повинна з'явитися кнопка, що забезпечує вибір зі списку (рис. 13):

	Ľ) • (¥ •	Ŧ					F	абота с	таблицами	Артеменко_КМ_125	БАЗА_ДАННЫ	Х1 : база	данных	(Access	2007) - Mi	crosoft Acces	5		- 0
Файл	Главная	Создание	В⊦	іешние даі	нные	Работа с базам	и данных	Поля	Таблица										
Режим	Вставить	Вырезать Копировать Формат по об	разцу	<b>Т</b> Фильтр	АЦ По АЦ По АД Уда	возрастанию убыванию алить сортировку	🎸 Выделение 🎦 Дополните 🍸 Фильтр	е т ельно т	Обновить ВСС *	🚽 Создать ΣИто ┛ Сохранить 🍄 Ор Х Удалить т 📰 До	оги фография полнительно *	<b>А</b> Найти	авадал адада ада ада а ада ада ада а ада а а ада а а а а а а а а а а а а а а а а а а а	іенить рейти * брать *	Calibri (O Ж. К	бласть данны: Ч <mark>А</mark> + ар	* 11 ?	ı • ⊟ ⊟   ≢ ≢ ≜•   ≣ ≣ ≣   ∰•	н - -
Режимы	Буф	ер обмена	5			Сортировка и ф	ильтр			Записи			Найти			Форм	атир	ование текста	Gi -
Bce of	ъекты Ассе	ss 💌 «		Анкета															
Поиск		٩	Z	Код	*	Фамилия 🔻	Имя	- От	чество 🔹	Дата рождения	<b>,</b> Ад	pec		Тел	ефон 🝷	Группа	*	Щелкните для добавле	ения 🔻
Табли	цы	*			1	Бондаренк 🔻	Евгений	Мак	СИМОВИЧ	17.04.1995	Чехова, 23			3-	47-53	125			
A	нкета				2	Бондаренко	Юрий	Сер	геевич	09.06.1995	Смелянска	я, 55		3-	77-22	125			
Форм		^	-		3	Возный	Александр	Оле	ГОВИЧ	11.03.1995	Добролюб	ова, 11		3-	-65-11	125			
		^			4	Гушлевский	Ирина	Вале	ериивна	02.12.1995	Орджоник	идзе, 44		3-	12-55	125			
<u>-</u>	nkcia				5	Коваленко	Максим	Ива	НОВИЧ	09.09.1995	Вербовецк	ого, 4		3-	26-64	125			
					6	Кунов	Олег	Вла,	димирови	12.11.1995	Кармелюка	a, 38		3-	27-74	125			
					7	Лукашенко	Олексей	Дми	триевич	03.04.1995	Розы Люкс	ембург,	48	3.	15-77	125			
					8	Порублевый	Максим	Пет	рович	29.06.1995	Героев Дне	епра, 64		3-	-63-54	125			
				9	Сиваченко	Эдуард	Пет	рович	14.04.1995	Волкова, 2	7		3-	22-65	125				
			*		(Nº)	Чупрун										125			

Рисунок 13 – Перевірка списку студентів групи

#### Завдання 5.

За допомогою майстра форм створюється «ленточная» форма таблиці «Анкета». Для цього потрібно вибрати вкладку «Создание», натиснути кнопку «Мастер форм», як джерела даних вибрати таблицю «Анкета», вибрати всі поля і натиснути кнопку «Далее», надалі вибираємо зовнішній вигляд форми «ленточный», назвати форму «Анкета», натиснути кнопку «Готово». Після цього форма готова до заповнення (рис.14).

A 🚽 🤊 - 🕲 - 🖙				Арт	еменко_КМ_12	25_БАЗА_ДАННЬ	IX1 : база данных (Ас	cess 2007) -	Microso	ft Acces:
Файл Главная Созд	ание Вне	шние данные	Работа с база	ми данных						
Режим	ть <b>Эвать</b> г по образцу	Фильтр Ад↓Г Фильтр Ад↓Г	По возрастанию По убыванию Идалить сортировку	🎸 Выдел 🔚 Допо. у 🍸 Фильт	ление т лнительно т гр	Сбновить все т	оздать ∑ Итоги охранить 🍄 Орфог далить т 📰 Дополн	рафия нительно т	<b>А</b> Найти	а <sup>b</sup> <sub>ac</sub> Зам ⇒ Пер & Выб
Режимы Буфер обмен	a G		Сортировка и	фильтр			Записи			Найти
Bce объекты Access		Анкета 😑	Анкета							
Поиск Таблицы	<b>م</b> *	Анкета	l							
🛄 Анкета		Код	Фамилия	Имя	Отчество	ждения	Адрес	Телефс па		
Формы	*	1	Бондаренко 🗖	- Евгени	Максимови	4	Чехова, 23	3-47-53 ##		
🗄 Анкета		2	Возный 🗖	Юрий	Сергеевич		Смелянская, 55	3-77-22 ##		
		3	Гушлевский	Алекса	Олегович		Добролюбова, 1	3-65-11 ##		
		4	Коваленко 🗖	• Ирина	Валериивна	a ########	Орджоникидзе, 4	3-12-55 ##		
		5	Кунов	<ul> <li>Максил</li> </ul>	Иванович	*****	Вербовецкого, 4	3-26-64 ##		
		6	Лукашенко 🗖	• Олег	Владимиро	вич ########	Кармелюка, 38	3-27-74 ##		
		7	Порублевый 🖣	• Олексе	Дмитриеви	4	Розы Люксембур	3-15-77 ##		
		8	Сиваченко	<ul> <li>Максиі</li> </ul>	Петрович	+++++++++++++++++++++++++++++++++++++++	Героев Днепра, б	3-63-54 ##		
		9	Чупрун	• Эдуард	Петрович	++++++++++++	Волкова, 27	3-22-65 ##		
	*	(Nº)	•	•				##		

Рисунок 14 – Створення форми «ленточная»

#### Завдання 6.

Переміщатися по записам, тобто створювати перехід на один запис вперед, перехід на останній запис, перехід на один запис назад, перехід на чисту сторінку, де можна ввести новий запис, перехід на перший запис можливо за допомогою стрілок, які розташовані на нижній панелі в вікні Microsoft Access (рис.15).



Рисунок 15 – Перехід по записам

#### Завдання 7.

За допомогою режиму *Конструктор* створимо таблицю «Группы». Тобто потрібно вибрати вкладку «*Создание»* і натиснути кнопку «*Конструктор таблиц»*. Після створення структури таблиці, попередньо заповнивши стовпці «Имя поля» і «Тип данных», таблиця набуває наступного вигляду:



Рисунок 16 - Створення таблиці «Группы»

#### Завдання 8.

За допомогою майстра форм створюється «Форма в один столбец», що дозволяє заповнювати таблиці «Группы». Для цього потрібно вибрати вкладку «Создание», натиснути кнопку «Мастер форм», як джерела даних вибрати таблицю «Группы», вибрати всі поля, нажати кнопку «Далее», вибрати

зовнішній вигляд форми «Форма в один столбец», назвати форму «Группы», натиснути кнопку «Готово». Після цього форма готова до заповнення (рис. 17).



Рисунок 17 – Створення форми

Надалі заповнюємо створену форму існуючими групами курсу і отримуємо певний результат по заповненню форми *«Группы»* (рис. 18):



Рисунок 18 – Заповнення форми «Группы»

#### Завдання 9.

Якщо в полі «Фамилия» ввести нове прізвище, то воно не з'явиться в списку прізвищ, що розкривається в таблиці (рис. 19):

A 🛃 🍠 • (	×   <del>-</del>	_	-	-			Pa6	ота с таблицами	Артеменко_КМ	1_125_	БАЗА_ДАННЫХІ : база дан	ных (/	Access 2007)	- Mic	rosoft Acces	s
Файл Гла	вная	Создание	Внешние,	данные	Работа с базам	и данных	По	ля Таблица								
Цасти Части приложения т Шаблоны	Таблица	Конструктор таблиц Таблиць	Списки SharePoint	Маст * запро	ер Конструктор сов запросов Запросы	Форма Кон	стру фор	ктор м форма Формы	Мастер форм Навигация * Другие формы *	Отче	∎ т Конструктор Пустой отчетов отчет Отчеты	Маст Накл	ер отчетов ейки	Мак	🖉 💸 Моду 👷 Моду рос 😭 Visua Макросы и	иль Ильклас al Basic Код
Все объекты	Access		Анкет	ra												
Поиск		Q	🕗 Ко	д -	Фамилия 👻	Имя	-	Отчество 🔹	Дата рожден	ия -	Адрес	-	Телефон	-	Группа	- Щ
Таблицы		â		1	Бондаренко	Евгений		Максимович	17.04.199	5	Чехова, 23		3-47-53		125	
Анкета				2	Возный	Юрий		Сергеевич	09.06.199	5	Смелянская, 55		3-77-22		125	
Грудаы				3	Гушлевский	Александр		Олегович	11.03.199	5	Добролюбова, 11		3-65-11		125	
Car (pyrina)				4	Коваленко	Ирина		Валериивна	02.12.199	5	Орджоникидзе, 44		3-12-55		125	
Формы		~		5	Кунов	Максим		Иванович	09.09.199	5	Вербовецкого, 4		3-26-64		125	
- AHKeld				6	Лукашенко	Олег		Владимирови	12.11.199	5	Кармелюка, 38		3-27-74		125	
🖽 Группы				7	Порублевый	Олексей		Дмитриевич	03.04.199	5	Розы Люксембург, 48		3-15-77		125	
				8	Сиваченко	Максим		Петрович	29.06.199	5	Героев Днепра, 64		3-63-54		125	
				9	Чупрун	Эдуард		Петрович	14.04.199	5	Волкова, 27		3-22-65		125	
				10	Ворсул 💌	Ирина		Олексеевна	08.03.199	5	Чехова, 42		3-33-41		125	
			*	(Nº)	Бондаренко										125	
					Возный											
					Гушлевский											
					Коваленко	_										
					кунов											
					Лукашенко											
					Сирацонико											
					Чупрун											
					110910	_										

Рисунок 19 – Розгорнута таблиця без змін

Для того, щоб новий запис з новим прізвищем з'явився в таблиці необхідно зробити наступне: за допомогою майстра форм створюється «ленточная» форма таблиці «Анкета», тобто потрібно вибрати вкладку «Создание», натиснути кнопку «Мастер форм», як джерела даних вибрати таблицю «Анкета», вибрати всі поля, нажати кнопку «Далее», а потім вибрати зовнішній вигляд форми «ленточный», натиснути на форму «Форма для добавления записей» і кнопку «Готово». Після цього форма готова до заповнення (рис. 20).

Артеменко_КМ_125_БАЗА_ДАННЫХ1 : база данных (Access 2007) - Місто:									
Файл Главная Создание Вне	шние данные Работа с базами данных								
Режимы Буфер обмена Га	Фильтр Фильтр Осотировка и фильтр Сортировка и фильтр	айти айс Замени айти № Выбрат Найти							
Bce объекты Access 💿 «	Анкета В Форма для добавления записей								
Поиск Р Таблицы *	Форма для добавления записей								
Анкета	код Фамилия имя Отчество ждения Адрес Телеф	спа							
Труппы	1 Бондаренко 🗸 Евгени Максимович 🛛 🗰 Чехова, 23 3-47-5	3 ###							
Формы	2 Возный 🔹 Юрий Сергеевич 🛛 🕬 Смелянская, 55 3-77-22	2 ##							
🗐 Группы	3 Гушлевский 🔽 Алекса Олегович ####### Добролюбова, 1: 3-65-1:	1 ##							
🗐 Форма для добавления записей	4 Коваленко 💽 Ирина Валериивна ####### Орджоникидзе, 4 3-12-5	5 ##							
	5 Кунов 💌 Максиі Иванович ####### Вербовецкого, 4 3-26-64	4 ##							
	6 Лукашенко 🔽 Олег Владимирович ####### Кармелюка, 38 3-27-74	4 ##							
	7 Порублевый 💌 Олекса Дмитриевич ####### Розы Люксембур 3-15-7	7 ##							
	8 Сиваченко 💌 Максил Петрович 🛛 🗰 Героев Днепра, 🗄 3-63-54	4 ##							
	9 Чупрун 🔻 Эдуард Петрович ####### Волкова, 27 3-22-65	5 ##							
	10 Ворсул 💽 Ирина Олексеевна ####### Чехова, 42 3-33-4:	1 ##							
	* (Nº)	###							

Рисунок 20 – Форма для додавання нового прізвища

#### Завдання 10.

Створимо макрос «Добавление». Для цього потрібно вибрати вкладку «Создание», натиснути кнопку «Макрос».

Ввести макрокоманду «Открыть форму»:

- Ім'я форми: «Форма для додавання записів»;
- Режим: «Форма»;
- Режим даних: «Добавление»;
- Режим вікна: «Обычное».

Закрити вікно і зберегти макрос з ім'ям «Добавление», як показано на нижче приведеному рис. 21 з результатами створення макросу.

Файл         Главная         Создание         Внешние данные         Работа с базами данных         Конструктор           Выполнить         Преобразовать макросы         Развернуть         Сервис         Развернуть         Свернуть         Развернуть         Свернуть         Развернуть         Свернуть         Развернуть         Свернуть         Развернуть         Свернуть         Развернуть         Свернуть         Показать         Показать<	A 🛃 🖻 • (* •   =	A	Работа с	макросами А	ртеменко_КМ_125_БАЗА	ДАННЫХ1 : база данных (Access 2	007) - Microsoft Access
По шагам       Развернуть       Свернуть       Развернуть       Свернуть       Развернуть       Свернуть       Развернуть       Гоказать         Развернуть       Свернуть       Развернуть       Свернуть       Гоказать       По казать         Все объекты       Ассезя       Свернуть       По казать       По казать       По казать         Все объекты       Ассезя       Свернуть       По казать       По казать       По казать         Поиск       Г       Анкета       С объекты       Обриа для добавления       По казать       По казать         Формы       Х       Каталог       По казать       По казать       По казать       По казать         Поиск       Режим       Форма       Форма       По казать       По казать       По казать         Поиск       Режим       Форма       Форма       По казать       По казать       По казать         Поиск       Режим       Форма       Форма       По казать       По казать       По казать         По казать       По казать       Вернуть       Форма       Форма       По казать       По казать         По казать       По казать       Форма       Форма       По казать       По казать         По казать <td>Файл Главная Создание Вн</td> <td>ешние данные Работа с базами</td> <td>и данных Конст</td> <td>труктор</td> <td></td> <td></td> <td></td>	Файл Главная Создание Вн	ешние данные Работа с базами	и данных Конст	труктор			
Выполнить Свернуть Свернуть Развернуть Свернуть Маталог Показать макрокоманды все все Развернуть или свернуть все все Развернуть или свернуть Все все Развернуть или свернуть Все действия Показать или скрыть Все объекты Ассезя С « Гоиск	<ul> <li>По шагам</li> <li>Преобразовать макросы</li> </ul>	♣ <u> </u>	•] •]	B			
Сервис         Развернуть или свернуть         Показать или скрыть           Все объекты Ассеss                Добавление            Глоиск              //             Добавление               //             Добавление            Таблицы                //             Добавление               //             Добавление                 Добла для добавления               Формы               Форма               Форма                 Добавления записей               Режим форма               Форма               Добавление                 Дориа для добавления записей               Режим форма               Добавление                 Дориа для добавления записей               Режим окна               Обновить парам	Выполнить	Развернуть Свернуть Раз макрокоманды макрокоманды	звернуть Свернуть все все м	Каталог макрокоманд во	Показать е действия		
Все объекты Ассеss	Сервис	Развернуть или свер	НУТЬ	Показать или	скрыть		
Поиск.         Габлицы         С           Таблицы         Анкета         Открыть Форма	Все объекты Access 💿 <	< 🛄 Анкета <u>Д</u> Добавление					×
Таблицы         №           Ш Анкета         Формы         Форма для добавления записей           Формы         №           Формы         №           В Анкета         №           В Анкета         №           В Анкета         №           В Анкета         №           В Орма для добавления записей         Форма           Условие отбора         =           Режим данных         Добавление           Режим окна         Обычное           Обновить парам         Обновить парам	Поиск						~
Ш Анкета         Форма для добавления записей           Формы         ©           Формы         ©           Условие отбора         =           Условие отбора         =           Режим данных         Добавление           Режим окна         Обычное           Обновить парам         Обновить парам	Таблицы 🖈	ОткрытьФорму					×
Режим         Форма           Формы         Форма           Формы         Форма           Э         Анкета         Условие отбора         —           Э         Форма для добавления записей         Добавление         —           Форма для добавления записей         Режим окна         Обычное         Обновить парам	Анкета	Имя формы Форма	для добавления запис	ей			▼
Формы         №           Фармы         №           В Анкета         Условие отбора =           Группы         Режим данных           Добавление         Обычное           Обновить парам         Обновить парам	Пруппы	Режим Форма					•
Формы         ×           З Анкета         Условие отбора           Э Группы         Режим данных           Добавление         Режим окна           Обычное         Обновить парам	<b>A</b>	— Имя фильтра					
Группы     Форма для добавления записей     Фежим окна     Обычное     Обновить парам	Формы 🔅	Условие отбора =					<u></u>
Форма для добавления записей     Режим окна     Обычное     Обновить парам     Обновить парам	🗐 Группы	Режим данных Добавл	тение				
Макарсы	🗐 Форма для добавления записей	Режим окна Обычно	oe				•
Пакросы Адобавить новую макрокоманду	Макросы ☆	Добавить новую макрокол	манду 💌				Обновить Параметры

Рисунок 21 - Створення макросу «Добавление».

Створимо макрос «Закрытие». Для цього потрібно вибрати вкладку «Создание», натиснути кнопку «Макрос».

Ввести макрокоманду «Сохранить запись».

Ввести макрокоманду «Выделить объект»:

- Тип об'єкту: «Форма»;
- Назва об'єкту: «Анкета»;
- У вікні бази даних: «Нет».

Ввести макрокоманду «Обновление»:

• Ім'я елемента: «Фамилия».



Закрити вікно і зберегти макрос з ім'ям «Закрытие» (рис. 22).

Рисунок 22 - Створення макросу «Закрытие»

Для того, щоб працювати в режимі *Конструктор* з самозаписом прізвищ необхідно відкрити форму «Форма для добавления записей», натиснути правою кнопкою миші по полю «Фамилия» та вибрати пункт *«Свойства»*, далі відкрити вкладку *«События»* і тепер можна додати макрос **«Закрытие»** у властивість *«Выход»* після чого необхідно закрити форму (рис. 23).

П		Φο	ома для добавления записей	
		1.1	1 • 1 • 2 • 1 • 3 • 1 • 4 • 1 • 5 • 1 • 6 • 1 • 7 • 1 • 8 • 1 • 9 • 1 • 10 • 1 • 11 • 1 • 12 • 1	13 • 1 • 14 • 1 • 15 • 1 • 16
-1		<b>F</b> 3	головок формы	
ч	· · ·			
		Φ	орма для добавления записей	
	1	H .		
			Код Фамилия Имя Отчество ждения Адрес	Телефена
		<b>#</b> 0	бласть данных	
Ш			Код Фамилия УИМЯ Отчество Дата ро Адрес	Телеф(Гк
	1 ÷ 1			
11		-	римечание формы	
	11	I I	Окно свойств	- ×
Ш	1		Тип выделенного элемента: Форма	
11	121			
11	-			
11	1		Макет Данные События Другие Все	
11	-		Лвойное нажатие кнопки	
11	2		Кнопка вниз	
-11	1 - 1		Кнопка вверх	
11	4		Перемещение указателя	
11			Клавиша вверх	
11	5		Клавиша вниз	
11				
11	6		Открытие	
11	1 ÷ 1		Закрытие	
11	7		Изменение размера	
	1 - 1		Включение	-
11	8			

Рисунок 23 – Результат роботи

Після цього потрібно відкрити в режимі *Конструктор* форму «Анкета», натиснути правою кнопкою миші по полю «Фамилия» та вибрати пункт «Свойства», далі необхідно відкрити вкладку «События» та вказати ім'я макросу «Добавление» в полі введення властивості «Двойное нажатие кнопки», як показано на рис. 24.

· ·	1 · 1 · 1 · 2	: • • • 3 • •	· 4 · 1 · 5	5 • • • 6	· · · 7	· · · 8	· I · 9	· · · 1	0 • • • 1	1 · + • 1	12 • 1 •	13 • • •	14 · I
	Ваголовок	формы											
	Анкета	a											
	Код	Фамилия	a	Имя	Отче	ество		жд	ения/	дрес			Теле
	🖡 Область да	нных											
	Код	Фамили	a 👻	Имя	Отче	ество		Да	ra pc	Адрес			Теле
	Примечані	ие формы											
	Окно сво	йств											- ×
	Тип выде	ленного эл	емента: Г	Толе со	списком	a							
	0.5												
	Фамилия	•		-		-							
	Макет	Данные	События	Други	e B	ce							
	Макет	Данные	События	Други	e B	ce							
	Фамилия Макет Получен Потеря о	Данные ие фокуса фокуса	События	Други	e B	ce							-
	Фамилия Макет Получен Потеря о Двойное	Данные ие фокуса фокуса нажатие и	События	Други	е В	се							
	Фамилия Макет Получен Потеря о Двойное Кнопка в	Данные ие фокуса фокуса нажатие и вниз	События	Други	е Ві Добавле	се							
	Фамилия Макет Получен Двойное Кнопка в	Данные ие фокуса фокуса е нажатие п вниз вверх	События кнопки	Други	е В Цобавле	се							
	Фамилия Макет Получен Потеря о Двойное Кнопка в Кнопка в Перемец	Данные ие фокуса фокуса е нажатие и вниз вверх цение указ	События кнопки ателя	Други	е В Добавле	се							
	Фамилия Макет Получен Потеря с Двойное Кнопка в Кнопка в Перемец Клавиша	Данные ие фокуса фокуса нажатие и вниз зверх цение указ вниз	События кнопки ателя	Други	е Ві	се							
	Фамилия Макет Получен Потеря ( Двойнос Кнопка е Кнопка е Кнопка е Клавиша Клавиша	Данные ие фокуса фокуса е нажатие и вниз вверх вниз вверх вверх	События кнопки ателя	Други	е Ві	се							
	Фамилия Макет Получен Потеря о Двойное Кнопка в Кнопка в Перемеш Клавиша Нажатие Вуход	Данные ие фокуса е нажатие и зниз зверх цение указ вниз вверх вниз вверх склавиши	События кнопки ателя	Други	е Вн	се							
	Макет Получен Потеря о Двойное Кнопка с Перемец Клавиша Нажатие Вход Вчура	Данные ие фокуса фокуса е нажатие и вниз вверх цение указ вниз вверх склавиши	События кнопки ателя	Други	е   Ві	се							

Рисунок 24 – Результат роботи

Після створення макросів «Добавление» і «Закрытие» нове прізвище, яке вводиться в таблицю «Анкета» через певну форму, з'явиться в списку згідно описаного вище процесу введення нового прізвища у раніше створений загальний список прізвищ групи (рис. 25).

A  , ") × (° ×  ∓				Работа с	таблицами Артемен	нко_КМ_125_БАЗА_ДАНН	НЫХ1 : база данных (Access	2007) - Microsoft Acc	ess	- 0 - X
Файл Главная Создание Вне	шние данн	ные Раб	бота с базами дані	ных Поля	Таблица					۵ ()
Режим Вставить	Фильтр	↓По возра ↓По убыва	істанию 🌾 Ві анию 🚰 Д	ыделение * ополнительно *	Создат Создат Обновить	ь ΣИтоги нить ∛у Орфография	аранить аранить аранить аранить аранить аранить аранить аранить аранить аранить аранить аранить аранить	Calibri (Область дани	ны т 11 т :	E E   律 律   M → E E = I = I = I = •
• • • • Формат по образцу Региции	3	6) удалить с Сорт	сортировку у Ф	ильтр	все У Удалит	ь • 📩 дополнительно	<ul> <li>Карать</li> </ul>	- <u>-</u>		×773
Режимы руфер оомена и	1	Copi	провка и фильтр			аписи	паити	Ψ	рматирование те	KCId 19
Все объекты Access 💌 «	AH	кета		14.15	0	0	A	Tanakau	Courses.	X
Поиск		код 🔹	Фамилия •	имя <del>-</del>	Отчество 🔻	дата рождения 🔻	Адрес	• телефон •	труппа 👻	щелкните оля оооавления 🔻
Таблицы 🎗		1	L Бондаренко	Евгений	Максимович	17.04.1995	Чехова, 23	3-47-53	125	
🛄 Анкета		2	2 Возныи	Юрии	Сергеевич	09.06.1995	Смелянская, 55	3-77-22	125	
🛄 Группы		3	3 Гушлевскии	Александр	Олегович	11.03.1995	Добролюбова, 11	3-65-11	125	
Формы		4	Коваленко	Ирина	Валериивна	02.12.1995	Орджоникидзе, 44	3-12-55	125	
🖼 Анкета		5	Кунов	Максим	Иванович	09.09.1995	Вербовецкого, 4	3-26-64	125	
Francisco de la composición de la composicinde la composición de la composición de la composición de l		6	5 Лукашенко	Олег	Владимирович	12.11.1995	Кармелюка, 38	3-27-74	125	
		7	7 Порублевый	Олексей	Дмитриевич	03.04.1995	Розы Люксембург, 48	3-15-77	125	
Форма для добавления записей		8	3 Сиваченко	Максим	Петрович	29.06.1995	Героев Днепра, 64	3-63-54	125	
Макросы 🏦		9	Э Чупрун	Эдуард	Петрович	14.04.1995	Волкова, 27	3-22-65	125	
💈 Добавление		12	Клинтон 🔽	Петро	Владимирович	11.05.1995	Хмельницкого, 73	3-79-31	125	
💈 Закрытие	*	(Nº)	Бондаренко						125	
			Возныи							
			тушлевский							
			Коваленко	-						
			Лукашенко							
			Попублерый							
			Сиваченко							
			Чупрун							
			Клинтон							
				_						

Рисунок 25 – Результат введення нового прізвища

#### 3. ЛАБОРАТОРНА РОБОТА №2

Тема: «Проектування багато табличних баз даних»

**Мета:** Ознайомитись та навчитись створювати і модифікувати структури таблиці «Мастер таблиц», «Мастер форм», закріпити навички роботи з таблицями і формами, вивчити елемент керування типу «поле со списком», закріпити навички роботи з таблицями і формами, вивчити елемент управління типу «флажок», навчитися встановлювати зв'язок між таблицями.

#### 3.1. Завдання

#### Завдання 1:

- створити таблицю під назвою «Преподаватели» за допомогою «Мастер таблиц». Зв'язки не визначати;

- у режимі *Конструктор* визначити тип даних для кожного поля наступним чином:

Имя поля	Тип данных
Фамилия	Текстовый
Имя	Текстовый
Отчество	Текстовый
Должность	Текстовый
Кафедра	Текстовый

Таблиця 1 – Визначення типу даних по полю таблиці

- визначити властивості полів на вкладці «Общие» згідно таблиці 2;

- для поля «Кафедра» змінити елемент керування згідно таблиці 3;

- створити форму для заповнення таблиці «Преподаватели». Заповнити форму. Зверніть увагу на те, що стовпець «Кафедра» заповнюється шляхом підстановки із сформованого Вами списку.

#### Таблиця 2 – Заповнення властивостей полів

Поле										
		Свойства полей								
		Форма	Маска	Подпись	Значение	Обязате	Индексирова			
	Размер	т поля	ввода		по умолч.	льное	нное поле			
						поле				
Фамилия	30			Фамилия		да	Да			
							(совпадения			
							допускаются			
							)			
Имя	12					да	нет			
				Имя						
Отчество	30			Отчество		Дa	нет			
Должность	20					нет	нет			
				Должность						
Кафедра	50			Кафедра		Дa	нет			

Таблиця 3 – Заповнення елементів полів таблиці

Тип элемента	Поле со списком
управления	
Тип источника	Список значений.
строк	
Источник строк	Введите список кафедр, используя в качестве разделителя точку с
	запятой ";"
Ограничиться	Да
списком	

#### Завдання 2:

- створити таблицю «Дисциплины» в режимі *Конструктор* з наступними параметрами згідно таблиці 4:

Таблиця 4 – Наповнення типів даних по кожному полю

Имя поля	Тип данных
Дисциплина	Текстовый

- визначити властивості полів таблиці «Дисциплины» на вкладці «Общие» наступним чином згідно таблиці 5:

Таблиця 5 – Властивості полів

Поле	Свойства полей						
	Размер	Подпись	Обязательное	Индексированное			
			поле	поле			
Дисциплина	20	Дисциплина	Да	Да (совпадения не			
				допускаются)			

- створити форму для заповнення таблиці «Дисциплины». Ввести назви дисциплін у форму, які читаються на вашому курсі.

#### Завдання 3:

- створити таблицю «Экзамен» в режимі Конструктор з параметрами згідно таблиці 6:

Таблиця 6 – Створення нової таблиці

Имя поля	Тип данных
Дисциплина	Текстовый
Группа	Числовой
Студент	Текстовый
Результат контроля	Текстовый

- визначити властивості полів на вкладці «Общие», як описано нижче:

Поле	Свойства полей								
		Формат	Подпись	Обязательное	Индексированное				
	Размер	поля		поле	поле				
Дисциплина			Дисципли	Да	Да (совпадения				
	15		на		допускаются)				
Группа	2		Группа	Дa	Да (совпадения				
	3				допускаются)				
Студент			Студент	Да	Да (совпадения				
	30				допускаются)				
Результат			Результат	Да	Да (совпадения				
контроля	10		контроля		допускаются)				

Таблиця 7 – Заповнення властивостей полів

Увага! В базі даних вже існує таблиця зі списком груп, таблиця зі списком дисциплін та «Анкета» зі списком студентом, тому цими записами можна скористатися для автоматичного заповнення полів «Группа», «Дисциплина» и «Студент» для таблиці «Экзамен». Це можливо виконати за допомогою елемента по управлінню полем.

#### Завдання 4:

- для поля «Группа» змінити елемент керування описано в таблиці 8;

- аналогічно потрібно змінити елементи керування для полів «Дисциплины» і «Студент», використовуючи відповідно таблиці «Дисциплины» та «Анкета»:

Тип элемента	Поле со списком
управления	
Тип источника	Таблица/запрос.
строк	
Источник строк	4. Щелкните мышкой,
	5. Нажмите кнопку
	6. В появившемся окне "построителя запросов":
	• добавить таблицу "Группы",
	• закрыть окно "добавление таблицы"
	• перетацить с помощью мыши поле "Группа" в "поле"
	бланка запроса
	в поле " <b>вывод на экран</b> " должна стоять галочка
Ограничиться	Да
списком	

Таблиця 8 – Створення елементів керування для поля «Группа»

- для поля «Результат контроля» змінити елемент керування за допомогою створення списку оцінок, як показано в таблиці 9:

Таблиця 9 - Створення елементів керування для поля «Результат контроля»

Тип элемента управления	Поле со списком
Тип источника строк	Список значений
Источник строк	Введите список оценок, используя в качестве разделителя точку с запятой "; " (отлично; хорошо; удовлетворительно; неудовлетворительно)
Ограничиться списком	Да

#### Завдання 5:

- створити форму для заповнення таблиці «Экзамен». Використовуючи створену форму «Экзамен» ввести дані з двох дисциплін на Ваш вибір по всім студентам своєї групи в таблицю «Экзамен». Оцінки виставляти довільно (бажано різноманітні).

#### Завдання 6:

- створити таблицю «Зачёт» в режимі *Конструктор* з параметрами, які описані в таблиці 10;

- визначити властивості полів «Дисциплина», «Группа», «Студент», «Результат» на вкладці «Общие»;

Таблиця 10 – Властивості полів

Имя поля	Тип данных
Дисциплина	Текстовый
Группа	Числовой
Студент	Текстовый
Результат	Логический

- для поля «Группа» змінити елемент керування. Аналогічно змінити елемент керування для полів «Дисциплина» і «Студент», використовуючи таблиці «Дисциплины» та «Анкета».

#### Завдання 7:

- створити форму для заповнення таблиці «Зачёт». Користуючись створеною формою «Зачёт», ввести дані по студентам своєї групи з двох дисциплін на свій вибір. Залік відзначати прапорцем;

- потрібно скористатися функцією управління Ііf для того, щоб за допомогою «флажка» в нове створене поле можна було вводити одне з двох значень: «Зачёт», «Незачёт»);

- переглянути форму «Зачёт» в режимі форми. Користуючись новим елементом управління «флажок» проставити результати по заліку спочатку.

#### Завдання 8:

- встановити зв'язок між таблицями «Зачёт» - [«Дисциплины», «Анкета»]. Аналогічно створити зв'язок між полем «Фамилия» таблиці «Анкета» і полем «Студент» таблиці «Зачёт»;

- після встановлення зв'язку зміна полів «Фамилия» і «Дисциплина» відповідно в таблицях «Анкета» і «Дисциплины» автоматично спричинить зміну пов'язаних з цими полями наступних полів «Дисциплина» і «Студент» в таблиці «Зачёт». У свою чергу Ви не зможете ввести в таблицю «Зачёт» нові прізвища, або дисципліни, якщо вони відсутні в таблицях «Анкета» і «Дисциплины». Перевірити це;

- встановити зв'язок між таблицями «Экзамен» - [«Анкета», «Дисциплины»]. Перевірте, як працює зв'язок.

#### 3.2. Теоретичні відомості

MS Access містить в собі великий набір інструментів для автоматичного створення стандартних елементів інтерфейсу. Для створення нової форми спочатку слід відкрити вікно бази даних. Потім необхідно вибрати піктограму «ФОРМА» і натиснути кнопку «Создать». MS Access відкриває діалогове вікно «Создание формы». У полі слід вказати таблицю або запит, дані з яких необхідно представити у формах. Після чого слід натиснути кнопку «Мастера».

MS Access відкриває вікно зі списком доступних майстрів, які необхідні для розробки форм. Виберіть елемент «В один столбец» і натисніть кнопку *OK*. У першому діалоговому вікні майстра необхідно визначити поля даних, які будуть розміщені по формах. Поля можна вибирати окремо і поміщати в список «Порядок полей в форме» натисканням кнопки >. При натискання кнопки « майстер розміщує в цей список всі існуючі поля. Список визначає не тільки які поля існуватимуть у формі, але ї їх чергу по запису.

Продовжіть процес створення форми, натиснувши кнопку *«Далее»*. Якщо більше ніяких установок робити не потрібно, натисніть кнопку «Готово». У цьому випадку для всіх інших установок майстер використовує значення за замовчуванням.

MS Access пропонує користувачеві ще чотири майстри по створенню форм:

- «Мастер табличной формы» - створює форми, дані в яких представлені у вигляді таблиці;

- «Мастер диаграмм» - створює форми з двох- або багатовимірними стовпчастими, круговими, лінійними або штриховими діаграмами. Його використання особливо корисно для графічного представлення числових даних;

- «Мастер составных форм» можна використовувати для створення форм, в яких необхідно відображати інформацію з двох зв'язаних таблиць;

- «Мастер простых форм» - пропонує найшвидший спосіб підготовки форм і може бути завантажений за допомогою кнопки "Простая форма", яка розташована на панелі управління. Він створює стандартну форму за один крок. При запуску «Мастер простых форм» за допомогою піктограми форма буде заснована на таблиці або запиті, обраних в даний момент у вікні бази даних.

Створення елементів форми здійснюється в режимі *Конструктор*. На панелі елементів знаходяться кнопки, що відповідають доступним елементам управління, і які можуть бути присутніми у формі. Це наступні кнопки: "Выключатель", "Группа", "Диаграмма", "Конец страницы", "Кнопка", "Линия", "Несвязанная рамка объекта", "Переключатель", "Подпись", "Подчиненная форма/отчет", "Поле", "Поле со списком", "Прямоугольник", "Связанная рамка объекта", "Список", "Флажок".

Елементи форми можуть бути зв'язаними і незв'язаними. Зв'язані елементи прив'язані до поля вихідної таблиці або запиту. Незв'язані елементи, як правило, відображають результат обчислення.

Відкрийте форму в режимі *Конструктор* і виберіть піктограму "ПОЛЕ" на панелі елементів. Потім вкажіть мишею місце в межах форми, в результаті чого MS Access створить нове поле і підпис. Відкрийте вікно властивостей поля і для властивості *«Данные»* введіть вираз, що задає його значення. Імена полів в таких виразах розташовані у квадратні дужки. У цьому випадку їх легше відрізнити від інших елементів. Крім того, застосування квадратних дужок для імен полів, що містять спеціальні символи, є обов'язковим.

В MS Access немає можливості визначати тип даних, що обраховуються при створенні форми, тому необхідно самостійно прописувати значення властивості «Формат поля».

Таблиця складається з рядків і стовпців і має унікальне ім'я в базі даних. База даних містить безліч таблиць, зв'язок між якими встановлюється за допомогою співпадаючих полів. MS Access підтримує чотири типи відносин

між таблицями:

- <u>Один-до-одного</u>, де кожний запис в одній таблиці відповідає тільки одному запису в іншій таблиці;

- <u>Один-до-багатьох</u>, тобто кожний запис в одній таблиці відповідає багатьом записам в іншій таблиці;

- Багато-до-одного – аналогічно щодо запису один-до-багатьох;

- <u>Багато-до-багатьох</u>, тобто один запис із першої таблиці може бути пов'язаний більш ніж з одним записом із другої таблиці, або один запис із другої таблиці може бути пов'язаний більш ніж з одним записом з першої таблиці.

Тип відносин <u>«Один-до-багатьох»</u> є найбільш загальним.

#### 3.3. Хід виконання роботи та результати завдань

#### Завдання 1.

Оскільки «*Мастер таблиц»* в Microsoft Access 2010 не існує, то таблиця створюється за допомогою режиму *Конструктор* - вибрати вкладку «*Создание»* і натиснути кнопку «*Конструктор таблиц»*. Після створення структури таблиці потрібно її зберегти і назвати «Преподаватели».

	") - (" -   -	лаба 2 :	база д <mark>анных (</mark> Access 2007)	- Microsoft Access	Работа с	таблицами				di terd		
Файл	Главная	Создание	Внешние данные Ра	бота с базами данных	Поля	Таблица						
Режим •	АВ 12 Текст Число	Денежный	щ Дата и время ✓ Логический Другие поля →	Имя и подпись В Значение по умолч Размер поля 255	анию	изменить одстановку	<i>fx</i> Изменить выражение	ар Параметры поля МЕМО т	Тип данных: Формат:	Текстовый ▼ Форматирование ▼	<ul> <li>Обязательное</li> <li>Уникальное</li> <li>Индексировано</li> </ul>	Гроверка Троверка
Режимы		Добавление	и удаление		Ce	ойства			Форм	атирование	Проверка п	оля
! Пр	едупреждение	системы безо	опасности Запуск активно	ого содержимого отключ	ен. Щелкн	ите для пол	учения допо	лнительных све	едений. 🛛 Е	ключить содержимо	)e	
Bce of	ьекты Access		Преподаватели									
Поиск		Q	🕗 Порядковый номе	ер 🔻 Фамилия 👻	Поле	l ⊽ [	1оле2 🔹	Поле3	• Поле4	👻 Щелкните	для добавления	*
Табли	цы	\$	*	(Nº)								
🔲 nj	реподаватели											

Рисунок 1 - Створення таблиці «Преподаватели»

На вкладці *«Общие»* визначаються властивості полів. Для цього необхідно в режимі *Конструктор* натиснути мишкою на потрібне поле і заповнити всі необхідні властивості цього поля, як на рисунках 2,3,4,5,6.

Bce объекты Access	⊗ «		Преподаватели				
Поиск	2		Имя поля		Тип данных		
Тэбанны		8	Порядковый номер	Счетчик			
	~		Фамилия		Текстовый	-	
Преподаватели			Имя		Текстовый		
			Отчество	Текстовый			
			Должность	Текстовый			
			Кафедра		Текстовый		
			···· • • • • • • • • • • • • • • • • •				
		<u> </u>					
		<u> </u>					
			Общие Подстановк	а			
		F	азмер поля	30			
		q	Арска врояа				
		l li	Подпись	Фамилия			
		3	начение по умолчанию				
		У	словие на значение				
			Сообщение об ошибке				
			Обязательное поле	Да			
			Іустые строки	Да			
1			пндексированное поле	да (Допус	каются совпадения)		

Рисунок 2 - Встановлення властивостей для поля «Фамилия»

	0 4						
SCE OOBERTBI ACCESS	· · · ·	преподаватели					
louck		Имя поля	я Тип данных				
Таблицы	~	Порядковый номер	р Счетчик				
Преподаватели		Фамилия	Текстовый				
Преподаватели		Имя	Текстовый 💌				
		Отчество	Текстовый				
		Должность	Текстовый				
		Кафедра	Текстовый				
		Пафедра					
		Общие Полстанов	BK3				
		Pazmen nong	12				
		Формат поля	12				
		Маска ввода					
		Подпись	Имя				
		Значение по умолчаник	0				
		Условие на значение					
		Сообщение об ошибке					
		Обязательное поле	Да				
		Пустые строки	Ца				
		Скатие Юникол	/hei				
		ПСжатие юникод	да				

Рисунок 3 - Встановлення властивостей для поля «Имя»
Все объекты Access 🛛 😒 ≪		Преподава	атели				
Поиск		2 I	Имя поля		Тип данных		
Таблицы	- 8	Порядков	ый номер		Счетчик		
		Фамилия			Текстовый		
Преподаватели		Имя			Текстовый		
		Отчество			Текстовый	-	
		Должност	ь		Текстовый		
		Кафедра			Текстовый		
	ШГ	Общие	Подстановка	3			
		Размер поля		30			
		Формат поля					
		Маска ввода		_			
		Подпись		Отчество			
		Усповие на зн	молчанию				
		Сообщение о	бошибке				
		Обязательное поле		Дa			
		Пустые строки		Да			
		Индексирован	ное поле	Нет			
1		Сжатие Юник	0.7	Ла			



Все объекты Access 💿 <	) B	Преподавате	ели				
Поиск		L L L	мя поля		Тип данных		
Таблицы	8	Порядковый	й номер		Счетчик		
		Фамилия		Текстовый			
Преподаватели		Имя		Текстовый			
		Отчество		Текстовый			
		Должность			Текстовый 💌		
		Кафедра			Текстовый		
		Общие	0.0/13408/	-			
	6	Pazuen nong	одетанювка	20			
		Формат поля		20			
		Маска ввода					
		Подпись		Должност	гь		
		Значение по ум	олчанию				
		Условие на знач	ение				
		Сообщение об ошибке					
		Обязательное поле		Нет			
		Пустые строки		Да			
		Индексированн	ое поле	Нет			
a 1		Сжатие Юникод	1	Да			

Рисунок 5 - Встановлення властивостей для поля «Должность»

Bce объекты Access 💿 «		Преподаватели				
Поиск	2	Имя поля		Тип данных		
Таблицы	8	Порядковый номер		Счетчик		
		Фамилия		Текстовый		
преподаватели		Имя		Текстовый		
		Отчество		Текстовый		
		Должность	Текстовый			
		Кафедра		Текстовый 💌		
			7			
		Общие Подстановк	а			
		Размер поля	50			
	9	Формат поля				
		Маска ввода				
		Подпись	Кафедра			
		Значение по умолчанию				
		условие на значение				
		сообщение об ошибке	-			
		Обязательное поле	да			
		пустые строки	да			
		индексированное поле	нет			

Рисунок 6 - Встановлення властивостей для поля «Кафедра»

Для того, щоб змінити елемент керування для поля «Кафедра» потрібно в режимі *Конструктор* відкрити вкладку *«Подстановка»* і встановити необхідні властивості для поля «Кафедра». При заповненні властивості *«Источник строк»* потрібно ввести список необхідних нам кафедр, які розділяються за допомогою « ; ». Після цього в полі «Кафедра» при натисканні мишею повинна з'явитися кнопка, що забезпечує вибір зі списку, як зображено на рис. 7.

	- v					
Отчество	Текстовыи					
Должность	Текстовый					
Кафедра	Текстовый 💌					
		Свойства поля				
Общие Подстановк	a					
Тип элемента управления	Поле со списком					
Тип источника строк	точника строк Список значений					
Источник строк	ик строк Системного програмирования;Истории Украины;Компьютерных систем;Прикладной математики;Иностранных языков;Физи					
Присоединенный столбе	1					

Рисунок 7 - Зміна елементу керування для поля «Кафедра»

Зробимо перевірку, тобто в полі «Кафедра» при клацанні мишею з'являється кнопка, що забезпечує вибір зі списку, що й зображено на рис. 8.

Все объекты Access 🛛 💿 «		Преподаватели									
Поиск Р	Z	Порядковый номер 🔻	Фамилия	٧	Имя	v	Отчество	Ŧ	Должность 🔻	Кафедра 👻	Щелкнип
Таблицы 🖈	*	(Nº)								•	
Преподаватели										Системного програмирования	
										Истории Украины	
										Компьютерных систем	
										Прикладной математики	
										Иностранных языков	

Рисунок 8 – Результат перевірки

За допомогою «Мастер форм» створюється «ленточная» форма таблиці «Преподаватели». Для цього потрібно вибрати вкладку «Создание», натиснути «Macmep кнопку форм», джерела вибрати таблицю ЯК даних «Преподаватели», вибрати всі поля, нажати кнопку «Далее». Після цього вибрати зовнішній вигляд форми «ленточный» та назвати форму

«Преподаватели», потім натиснути кнопку «Готово». Після цього форма готова до заповнення згідно рис. 9.

		Saripota		+opiner		
Все объекты Access		🔠 Преподаватели 🔳 Препо	даватели			
Поиск	2	Преполавател	И			
Таблицы	*	преподавател	V I			
Преподаватели		ый номер Фамилия	Имя	Отчество	Должность	Кафедра
Формы	*	1 Куницкая	Светла	Юриивна	Преподавате	Системного п 👻
Преподаватели	ĺ	2 Бабенко	Вера	Григорьевна	Преподавате	Системного п 👻
		3 Яшан	Оксана	Олегивна	Преподавате	Истории Укра 👻
		4 Ерофеев	Юрий	Федорович	Преподовате	Компьютерні 👻
		5 Щерба	Валент	Олексеевна	Преподовате	Прикладной / 👻
		6 Палагина	Ольга	Анатолиивна	Преподовате	Прикладной / 👻
		7 Махиня	Наталы	Владимировна	Преподовате	Иностранных 👻
		8 Бондаренко	Максил	Олексеевич	Преподовате	Физики 👻

Рисунок 9 - Створення формі «ленточная» таблиці «Преподаватели»

Стовпець «Кафедра» заповнюється шляхом підстановки із сформованого списку.

Шаблоны	Таблицы	3a	просы	Формы Отчеты			
Все объекты Access		Преподаватели					
Поиск	2	Порядковый ном	иер 🔻 Фамилия	• Имя •	Отчество 🔻	Должность 👻	Кафедра
Таблицы	*		1 Куницкая	Светлана	Юриивна	Преподаватель	Системного програмирования
Преподаватели			2 Бабенко	Вера	Григорьевна	Преподаватель	Системного програмирования
Формы	*		3 Яшан	Оксана	Олегивна	Преподаватель	Истории Украины
Преподаватели	^		4 Ерофеев	Юрий	Федорович	Преподователь	Компьютерных систем
Преподаватели			5 Щерба	Валентина	Олексеевна	Преподователь	Прикладной математики
			6 Палагина	Ольга	Анатолиивна	Преподователь	Прикладной математики
			7 Махиня	Наталья	Владимировн	Преподователь	Иностранных языков
			8 Бондаренко	Максим	Олексеевич	Преподователь	Физики

Рисунок 10 – Результат завдання 2

### Завдання 2.

Для того, щоб створити таблицю «Дисциплины» потрібно вибрати вкладку «*Создание*» і натиснути кнопку *«Конструктор таблиц»*. Після створення структури таблиці потрібно її зберегти у вигляді зображеному на рис. 11.



Рисунок 11 - Створення таблиці «Дисциплины»

На вкладці «Общие» визначаються властивості полів. Для цього необхідно в режимі конструктора клацнути мишкою на потрібне поле і заповнити всі необхідні властивості цього поля згідно рис. 12.

Режимы	Ce	рвис			Показать или скрыть События			
Bce объекты Access	∞ «	П Дисципли	ны 🔳 Ди	сциплины				
Поиск	2		Имя поля		Тип даннь	1×		
Таблицы	*	Дисципли	на		Текстовый	-		
Дисциплины								
Формы	~							
∟⊴ Дисциплины								
		Общие	Подстановк	а				
		Размер поля		80				
		Формат поля						
		Маска ввода		<b>D</b>				
		Значение по	умодчанию	дисципли	па			
		Условие на за	ачение					
		Сообщение	б ошибке					
		Обязательно	е поле	Да				
		Пустые строк	и	Да				
		Индексирова	нное поле	Нет				

Рисунок 12 – Заповнення властивостей полів

Створимо форму таблиці «Дисциплины» «ленточная» за допомогою «Мастера форм». Для цього потрібно вибрати вкладку «Создание», натиснути кнопку «Мастер форм», як джерела даних вибрати таблицю «Дисциплины», вибрати всі поля, нажати кнопку «Далее», вибрати зовнішній вигляд форми «ленточный», назвати форму «Дисциплины», натиснути кнопку «Готово». Після цього форма готова до заповнення, як на рис. 13.

			лу п			
Режимы	Буфер обмена	- Fai	Сортировка и фил	ьтр	Записи	
Все объек	ты Access 💿 «		Дисциплины 🔳 Дисциплины			
Поиск	م		Лисциплины			
Таблицы	*	1	дисциплипр			
🛄 Дисци	иплины		Дисциплина			
Формы	\$					
🔳 Дисци	иплины		•			

Рисунок 13 - Створення форми таблиці «Дисциплины»

Після заповнення форми таблиця «Дисциплины» виглядає наступним чином:

Все объекты Access			Дисциплины 🖽 Дисциплины				
Поиск	2		Дисциплина 👻				
Таблицы	\$		Английский язык				
Дисциплины			Архитектура компьютера				
Формы			Компьютерная логика				
Паралины	~	Организация баз данных					
Дисциплины			Полтология				
			Правознавство				
			Системное програмирование				
			Теория вероятности				
		SIZ.					

Рисунок 14 – Заповнена форма таблиці

# Завдання 3.

Після створення структури таблиці потрібно її зберегти і назвати «Экзамен» (рис. 15), потім визначити властивості полів на вкладці «*Общие»* і заповнити всі необхідні властивості цього поля (рис. 16-18).

i cardinina a a a a a a a a a a a a a a a a a	in contra	1.0		Cohinhone	ω <i>τι</i> ψ	no se per se		1	
Все объекты Access			Экзамен	Дисциплины					
Поиск	2	$\angle$	Дисциплин 👻	Группа	-	Студент	-	Результат к 👻	
Таблицы	*	*							
🛄 Дисциплины									
Экзамен									
Формы	*								
🔳 Дисциплины									

Рисунок 15 – Структура таблиці

Далі встановимо властивості для поля «Дисциплина» (рис. 16), «Группа» (рис. 17), «Студент» (рис. 18) та «Результат контроля» (рис. 19):

Bce объекты Access 🛛 💌 «	Экзамен 🖂 Дисцип	лины		
Поиск Р	Имя поля	Тип данных		
Таблицы	Дисциплина	Текстовый 💌		
	Группа	Числовой		
Дисциплины	Студент	Текстовый		
Экзамен	Результат контроля	Текстовый		
Формы 🛠				
🔳 Дисциплины				
	Общие Подстановка	a		
	Размер поля	15		
	Формат поля			
	Подпись	Дисциплина		
	Значение по умолчанию			
	Условие на значение			
	Сообщение об ошибке	_		
	Обязательное поле	Да Ла		
	Индексированное поле	да Да (Допускаются совпадения)		
	Скатие Юникол	Ла		
Рисунок 16 –	Властивості поля «	<Дисциплина»		

Touci	c	<u> </u>	Имя поля	Тип данных		
Tafu	мпы	~	Дисциплина	Текстовый		
		^	Группа	Числовой 🔻		
	дисциплины		Студент	Текстовый		
	Экзамен		Результат контроля	Текстовый		
Формы		~				
	Дисциплины					
			Общие Полстановка			
			Pasmen nong	Lience		
			Формат поля	Lenoe		
			Число десятичных знаков	Авто		
			Маска ввода			
			Подпись	Группа		
			Значение по умолчанию			
			Условие на значение			
			Сообщение об ошибке			
			Обязательное поле	Да		
			Индексированное поле	Да (Допускаются совпадения)		
			CHAPT TELL	1		

Рисунок 17 - Властивості для поля «Группа»

F CALVINIDI	~	ервис	показате или скрене соовният		
Bce объекты Access	∞ «	🔳 Экзамен 🔳 Дисциг	ллины		
Поиск	2	🚄 Имя поля	Тип данных		
Tafannu		Дисциплина	Текстовый		
таолицы	~	Группа	Числовой		
Дисциплины		Студент	Текстовый 👻		
Экзамен		Результат контроля	Текстовый		
Формы	~				
		Общие Подстановк	a		
		Размер поля	30		
		Формат поля			
		Маска ввода	CTU A A MIT		
		Значение по умодчанию	студент		
		Условие на значение			
		Сообщение об ошибке			
		Обязательное поле	Да		
		Пустые строки	Да		
		Индексированное поле	Да (Допускаются совпадения)		
		Сжатие Юникод	Да		
		Режим ІМЕ	Нет контроля		
		Режим предложений IME	Нет		
		Смарт-теги	I. Contraction of the second se		

Рисунок 18 - Властивості для поля «Студент»

Bce объекты Access		Экзамен 🔳 Дисциг	лины		
Поиск		Имя поля	Тип данных		
Tafauruu		Дисциплина	Текстовый		
		Группа	Числовой		
дисциплины		Студент	Текстовый		
Экзамен		Результат контроля	Текстовый 👻		
Формы	~				
🗐 Дисциплины					
		Общие Подстановк	a		
		Формат поля			
		Маска ввода	Berry at ToT KOUTDO BO		
		Значение по умолчанию	гезультат контроля		
		Условие на значение			
		Сообщение об ошибке			
		Обязательное поле	Да		
		Пустые строки	Да		
1		Индексированное поле	Да (Допускаются совпадения)		

Рисунок 19 - Встановлення властивостей для поля «Результат контроля»

## Завдання 4.

Для того, щоб змінити елемент керування для поля «Группа» потрібно в режимі *Конструктор* відкрити вкладку «Подстановка» і встановити необхідні властивості для поля «Группа» згідно рисунка 20.

Bce объекты Access 🛛 🕞 <	Экзамен	
Поиск Я	🕗 Имя поля	Тип данных
Таблицы	Дисциплина	Текстовый
Анкета	Группа	Текстовый 💌
	Студент	Текстовый
пруппы	Результат контроля	Текстовый
Дисциплины		
Экзамен		
Формы 🛠		
=в Дисциплины		
😑 Экзамен		
	Общие Подстановка	
	Тип элемента управления Поле со	
	Источник строк SELECT	Группы.Группа FROM Группы:
	Присоединенный столбес 1	
	Число столбцов 1	
	Заглавия столбцов Нет	
	Ширина столбцов	
	Число строк списка 16	
	Ширина списка Авто	
	Ограничиться списком Да	

Рисунок 20 – Властивості для поля «Группа»

При заповненні властивості *«Источник строк»* потрібно нажати кнопку «…». Після цього з'явиться вікно *«Построитель запросов»*. Потім потрібно виконати наступні дії (рис. 21):

- додати таблицю «Группы» та закрити вікно «Додавання таблиці»;
- перетягнути за допомогою миші поле «Группы» в поле бланка запиту;
- у полі «Вывод на экран» повинна стояти галочка.

>>	Экзамен Экзамен : построитель запросов
	Группы * Группа
асть навигации	
06/1	Поле: Группа 💌 Имя таблицы: Сортировка: Вывод на экран: 👽 Условие отбора:

Рисунок 21 – Заповнення вікна «Постороитель запросов»

Аналогічно потрібно змінити полів елементи керування для «Дисииплина» i «Студент», відповідно таблиці використовуючи «Дисциплины» «Анкета». Зміна та елемента керування для поля «Дисциплина» має вигляд як на рисунку 22. Для поля «Студент» пройти аналогічні дії:

Bce	объекты Access	R 4	Экзамен					
Dec .	ooberribi Access			14		T		
Поиск						типданных		
Табл	тицы	~	дисципли	ина		Текстовыи	-	ļ
	Анкета		Группа			Текстовый		
			Студент			Текстовый		
	группы		Результа	г контроля		Текстовый		
	Дисциплины							
	Экзамен							
Фор	мы	*						
-8	Дисциплины							
-8	Экзамен							
					_			
			Общие	Подстановка	a			
			Тип элемента	а управления	Поле со с	писком		
			Тип источни	ка строк	Таблица или запрос			
			Источник стр	рок	SELECT Дисциплины. Дисциплина FROM Дисциплины			
			Присоедине	нный столбес	1			
			Заглавия сто	дов	1 Her			
			Ширина столбцов					
			Число строк	Число строк списка				
			Ширина спи	ска	Авто			
			Ограничиты	ся списком	Да			
			Разрешение	нескольких зі	Да			

Рисунок 22 – Зміна елемента керування

Для того, щоб змінити елемент керування для поля «*Результат* контроля» потрібно в режимі Конструктор відкрити вкладку «Подстановка» і встановити необхідні властивості для поля «*Результат контроля*». При заповненні властивості «Источник строк» потрібно ввести список необхідних значень, які розділяються за допомогою «; », після чого при натисканні мишею повинна з'явитися кнопка, що забезпечить вибір зі списку (рис. 23).

Все объекты Access		Экзамен					
Поиск	2		Имя поля		Тип данных		
Таблиции		Дисципли	ина		Текстовый		
Таолицы		Группа			Текстовый		
Анкета		Студент			Текстовый		
Группы		Результат	контроля		Текстовый	-	
💷 Дисциплины		,					
Экзамен							
Формы	*						
🔳 Дисциплины							
П Эктанан							
-a Skamen							
		Общие	Подстановка	а			
		Тип элемента	управления	Поле со с	писком		
		Тип источни	ка строк	Список зн	ачений		
		Источник стр	ок	отлично;х	орошо;удовлетворит	ельно;	неудовлетворительно
		Присоединен	ный столбец	1			
		Число столбы	цов	1			
		Заглавия сто	лбцов	Нет			
		ширина стол	ьцов	16			
		ЧИСТО СТРОК	списка	APTO			
		Ограничить	а списком	Да			
		Description					

Рисунок 23 – Встановлення елементу керування

### Завдання 5.

За допомогою «*Macmep форм*» створюється «ленточная» форма таблиці «Экзамен». Для цього потрібно вибрати вкладку «*Coздание*», натиснути кнопку «*Macmep форм*», як джерела даних вибрати таблицю «Экзамен», вибрати всі поля, нажати кнопку «*Далее*», вибрати зовнішній вигляд форми «ленточный», назвати форму «Экзамен», натиснути кнопку «Готово». Після цього форма готова до заповнення (рис. 24).

		1			
Все объекты Access 💿	~	🌐 Экзамен 💷 Дисципл	ины 😑 Экзамен		
Поиск	P	Экзамен			
Таблицы	2	Shouwen			
🛄 Анкета		Дисциплина	Группа	Студент	Результат контрол:
🏥 Группы			•	-	•
🛄 Дисциплины					
🛄 Экзамен					
Формы	5				
🔳 Дисциплины					
🔳 Экзамен					

Рисунок 24 - «Ленточная» форма таблиці «Экзамен».

Після заповнення форми таблиця «Экзамен» набуває наступного вигляду:

Все объекты Access 💿 «			Экзамен					
Поиск		2	Дисциплина 👻	Группа 👻	Студент 👻	Результат контроля 🚽		
Таблицы	*		Английский язык	KM-125	Бондаренко	отлично		
Анкета			Английский язык	KM-125	Возный	хорошо		
Брудан			Английский язык	KM-125	Гушлевский	удовлетворительно		
			Английский язык	KM-125	Коваленко	неудовлетворительно		
Дисциплины			Английский язык	KM-125	Кунов	отлично		
Экзамен			Английский язык	KM-125	Лукашенко	хорошо		
Формы	*		Английский язык	KM-125	Порублевый	удовлетворительно		
📧 Дисциплины			Английский язык	KM-125	Сиваченко	неудовлетворительно		
😑 Экзамен			Английский язык	KM-125	Чупрун	отлично		
			Системное програмир	KM-125	Бондаренко	хорошо		
			Системное програмир	KM-125	Возный	удовлетворительно		
			Системное програмир	KM-125	Гушлевский	неудовлетворительно		
			Системное програмир	KM-125	Коваленко	отлично		
			Системное програмир	KM-125	Кунов	хорошо		
			Системное програмир	KM-125	Лукашенко	удовлетворительно		
			Системное програмир	KM-125	Порублевый	неудовлетворительно		
			Системное програмир	KM-125	Сиваченко	отлично		
			Системное програмир	KM-125	Чупрун	хорошо		
		*				<b>•</b>		

Рисунок 25 – Заповнена форма

# Завдання 6.

Для того, щоб створити таблицю за допомогою *Конструктор* потрібно вибрати вкладку «*Создание*» і натиснути кнопку «*Конструктор таблиц*». Після створення структури таблиці потрібно її зберегти і назвати «Зачёт» (рис. 26).

—————————————————————————————————————	ча Фильтр	М↓ По убывані Я Удалить сор	<b>ію</b> этировку	<b>Ч Дополн</b> Фильтр	нительно 🔻	Обновит все т	∎С( ЪХУД	
симы Буфер обмена 🗔 Сортировка и фильтр								
екты Access 💿	≪ [ ₽ ∠	Зачет Дисципл	ин 👻 Груг	па 🔻	Студен	т - Ре	зультат 🚽	
ы	* *							
чет								
	Вставить Вставить Буфер обмена жекты Access	Вставить Вставить Формат по образци Буфер обмена жекты Access Собрана	Вставить Формат по образцу Буфер обмена екты Access качет в Ассевить Ассевить Ассевить в Ассевить Ассевить Ассевить в Ассевить Ассевить Ассевить в Ассевить Ассевить Ассевить в Асс	Вставить Фильтр Фильтр Удалить сор Буфер обмена екты Access чет	Вставить Формат по образцу Буфер обмена Сортировка и ф эекты Access С « Бильтр Удалить сортировка и ф Зачет Дисциплин - Группа - *	Вставить Формат по образцу Буфер обмена Га Сортировка и фильтр Эекты Access	Вставить образцу Буфер обмена с Сортировка и фильтр сортировка и фильтр Эекты Access	Вставить Формат по образцу Буфер обмена С Чальтр Сортировка и фильтр Обновит все ч Эекты Access С К С К Дисциплин ч Группа ч Студент ч Результат ч ж чет

Рисунок 26 - Структура таблиці «Зачёт»

На вкладці «Общие» визначаються властивості полів «Дисциплина», «Группа», «Студент» та «Результат» згідно методики описаної вище, а змінити елемент керування для полів «Группа», «Дисциплина» і «Студент» таблиці «Зачёт» можна використовуючи, відповідно, таблиці «Дисциплины» й «Анкета» аналогічно з методикою описаною в завданні 4. Один з результатів показано на рисунку 27:

Bce объекты Access		Зачет	
юиск		Имя поля	Тип данных
<b></b>		Дисциплина	Текстовый
Таблицы	~	Группа	Текстовый
Анкета		Студент	Текстовый 👻
Группы		Результат	Логический
💷 Дисциплины			
Зачет			
Экзамен			
Рормы	*		
В Зачет			
		Общие Подстанов	ка
		Тип	
		Тип исторника строк	Таблица или запрос
		Merouwurk erpor	SELECT Aurers daugage EPOM Aurers
		Присоединенный столбе	1
		Нисто столбнов	1
		Заглавия стоябнов	Нат
		Ширина столбцов	
		Число строк списка	16
		Ширина списка	ABTO
			0.5
		Спраничитася списком	<u></u>

Рисунок 27 - Зміна елемента керування для поля «Студент»

### Завдання 7.

За допомогою «Майстер форм» створюється «ленточная» форма таблиці «Зачёт». Для цього потрібно вибрати вкладку «Создание», натиснути кнопку «Мастер форм», як джерела даних вибрати таблицю «Зачёт», вибрати всі поля, нажати кнопку «Далее», вибрати зовнішній вигляд форми «ленточный», назвати форму «Зачёт», натиснути кнопку «Готово». Після цього форма готова

до заповнення.

Все объекты Access	. ≪	🔠 Зачет 😑 Зачет			
Поиск	٩	Зацет			
Таблицы	\$	Janei			
Анкета		Дисциплина	Группа	Студент	Резу
🛄 Группы			•	•	
🛄 Дисциплины					
🛄 Зачет					
Формы	*				
🔳 Зачет					

Рисунок 28 – Створення форми «Зачёт»

Після заповнення форми таблиця «Зачёт» набуває вигляду, що зображено на рисунку 29:

Bce объекты Access	. ≪		Зачет 😑 Зачет						
Поиск	Q		Дисциплина	-শ	Группа	*	Студент	÷	Результат 👻
Таблицы	*		Организация баз данных		KM-125		Бондаренко		
Анкета			Организация баз данных		KM-125		Возный		
Бруппы			Организация баз данных		KM-125		Гушлевский		
			Организация баз данных		KM-125		Коваленко		
Дисциплины			Организация баз данных		KM-125		Кунов		
Зачет			Организация баз данных		KM-125		Лукашенко		
Формы	*		Организация баз данных		KM-125		Порублевый		
-8 Зачет			Организация баз данных		KM-125		Сиваченко		<b>V</b>
			Организация баз данных		KM-125		Чупрун		
			Политология		KM-125		Бондаренко		
			Политология		KM-125		Возный		
			Политология		KM-125		Гушлевский		<b>V</b>
			Политология		KM-125		Коваленко		
			Политология		KM-125		Кунов		
			Политология		KM-125		Лукашенко		
			Политология		KM-125		Порублевый		<b>V</b>
			Политология		KM-125		Сиваченко		
			Политология		KM-125		Чупрун		
		*		-					

Рисунок 29 – Заповнена форма «Зачёт»

Для того, щоб за допомогою «флажка» в нове створене поле можна було вводити одне з двох значень («Зачёт», «Незачёт») слід виконати наступні дії:

1. Відкрити форму «Зачёт» в режимі Конструктор.

2. Праворуч від поля типу «флажок» додати елемент типу «поле»: клацнути мишкою по значку «аб» на панелі інструментів і праворуч від прапорця ще раз клацнути мишкою, з'явиться поле з довільним написом (поле ...).

3 . Натиснути праву кнопку миші на цьому полі, розкрити пункт «Свойства», ввести у властивість «Всі» наступний вираз: = Iif ( [ Результат ] ; "Зачёт" ; "Незачёт" ).

A   🚽 ") - (" -   - Lab_	4 : база	данных (Access	2007) - Micr	osoft Aco	ess									
Файл Главная Создани	ie B	нешние данные	Работа (	с базами	данных	Конструкт	ор Уг	орядочи	ть Формат					
Режим Темы АШрифты т	4	abl Aa	xxx	Q	[ <sup>XYZ</sup> ]					Вставить юбражение	(∰ Эм ⊒ За - डि. Да	іблема головок іта и время	Добави	ть Стран свой
Режимы Темы					Элементь	управлени:	я				Кол	онтитулы		
Все объекты Access 💿	« 🗄	Зачет 😑	Зачет											
Поиск	<u>م</u>	+ + + 1 + + + 2	· I · 3 · I ·	4 + 1 + 5	6	7 * 1 * 8 *	1 * 9 * 1 *	10 · · · 11	· · · 12 · · · 13 ·	1 + 14 + 1 + 19	5 · i · 16 ·	i · 17 · i · 18	· · · 19 ·	I · 20 · I
Таблицы	*	Заголовок	формы											
Анкета	- - 1	Зачет									Поле9			
Дисциплины		Дисципл	ина		Группа			Студен	нт	P	езу			
Зачет		🗲 Область да	нных	· ·		· · ·		· ·				· · ·		
Формы	*	. Дисципл	ина	· · ·	Группа		-	Студе	нт		=IIf([P	езультат];		
	2			Окна	о свойств выделенног	о элемента:	Поле				I	+ ×		
		• Примечани	ие формы	пол	e				-				<u> </u>	
		·		Ma	кет Данн	ые Событи	ия Други	e Bce	•				I	
	1			Имя			1	толе				*		
	- 2			Дан	ные			=IIf([Резул	ьтат];"Зачет";"Не	зачет")		<b>_</b>		
	- III -			Чис	по десятичн	ых знаков		Авто						
	3			Выв	од на экран			Да Экономи	TAKET					
		•		Наз	мат текста вание табли	цы		зовічный	TEKCI					
	4			Ото	бражать эл	емент выбо	ра даты	Для дат						
				Выс	ота			3см ) 556 см						
	5			Оте	верхнего кр	ая	i	0,106cm						
				От/	евого края			15,397 см				-		
	-													

4. Встановити для властивості «Вывод на экран» значення "Да".

Рисунок 30 – Введення значень «Зачёт» або «Незачёт»

Форма **«Зачёт»** має «всановлений флажок», який буде відповідати значенню «зачёт», а «знятий флажок» – значенню «незачёт» (рис. 31):

Все объекты Access 💿 <	< 📰 Зачет 📑 Зачет	
Поиск	Зачет	Поле9
Таблицы 🛠	Sulei	
🛄 Анкета	Дисциплина Группа Студент	Резу
🛄 Группы		
🛄 Дисциплины	Политология • КМ-125 • Бондаренко	▼ 🛛 Зачет
🛄 Зачет		
Формы		
🗄 Зачет	Организация баз данных 🗸 КМ-125 🔍 Бондаренко	▼ 🗸 Зачет
	Политология 🗸 КМ-125 🗣 Возный	- Незачет
	Организация баз данных — КМ-125	Незачет
	Политология 🗸 КМ-125 🔽 Гушлевский	🛛 🗹 Зачет
		- 7 20007
	Организация баз данных • Пин-125	Javer

Рисунок 31 – Заповнена форма

# Завдання 8.

Для того, щоб встановити зв'язок між таблицями «Зачёт» -[«Дисциплины», «Анкета»], потрібно вибрати вкладку «Работа с базами даннях» і натиснути кнопку «Схема данных». Потім:

• користуючись «правою мишкою» додати в схему всі таблиці «Зачёт», «Анкета», «Дисциплины»;

• закрити вікно «Добавление таблицы». У схемі даних повинні з'явитися три схеми вище зазначених таблиць;

• помістити покажчик миші на полі «Дисциплина» таблиці «Зачёт», натиснути ліву кнопку і, не відпускаючи її, перетягнути значок поля на поле «Дисциплина» таблиці «Дисциплины». З'явиться діалогове вікно «Связи»:

- поставити прапорець «Обеспечение целесности данных»;
- поставити прапорець «Каскадное обновление связанных полей»;
- натиснути кнопку «Объединение»;
- натиснути кнопку «Создать», щоб підтвердити створення зв'язку;

• перейти у вікно «Схема данных». Там повинен з'явитися зв'язок «один-до-багатьох».

Аналогічно створити зв'язок між полем «Фамилия» таблиці «Анкета» і полем «Студент» таблиці «Зачёт».

сервис	Иска				
Все объекты Access 💿 «	🗧 Схема данных				
Поиск					
Таблицы 🛠					
🛄 Анкета					
🛄 Группы					
🛄 Дисциплины	Í	Дисциплины	Зачет		Анкета
🛄 Зачет		Дисциплина	Дисциплина	1	💡 Код 🔺
Формы 🛠			Группа		Фамилия Имя
🕮 Зачет			Результат		Отчество
					Дата рождения
					Адрес
					Телефон

Рисунок 32 - Створення зв'язків

Після встановлення зв'язку зміна полів «Фамилия» і «Дисциплина» відповідно в таблицях «Анкета» і «Дисциплины» автоматично спричинить зміну пов'язаних з цими полями полів «Дисциплина» і «Студент» в таблиці «Зачёт». У свою чергу не можливо ввести в таблицю «Зачёт» нові прізвища, або дисципліни, якщо вони відсутні в таблицях «Анкета» і «Дисциплины» (рис. 33-35).

Для того, щоб встановити зв'язок між таблицями «Экзамен» - [«Анкета», «Дисциплины»] потрібно виконати аналогічні дії, які зазначені вище (рис. 36).

Bce объекты Access	⊗ «	🖽 Зачет 🛄 Дисциплины
Поиск	Q	🚄 Дисциплина 👻
Таблицы	*	Английский язык
🛄 Анкета		<ul> <li>Архитектура компьютера</li> </ul>
Группы		<ul> <li>Компьютерная логика</li> </ul>
		<ul> <li>• Организация баз данных_1</li> </ul>
дисциплины		<ul> <li>Политология</li> </ul>
Зачет		<ul> <li>Правознавство</li> </ul>
Формы	*	<ul> <li>Системное програмирование</li> </ul>
🗐 Зачет		<ul> <li>Теория вероятности</li> </ul>

Рисунок 33 – Внесення змін у таблицю «Дисциплины»

Все объекты Access	• «		Анкета								
Поиск	م	2	Код	🔻 Фамилия 🔻	Имя -	Отчество 🔹	Дата рождения 🔻	Адрес 🗸	Телефон 🔻	Группа	,
Таблицы	\$	÷		1 Бондаренко	Евгений	Максимович	17.04.1995	Чехова, 23	3-47-53	125	
Анкета		÷		2 Возный	Юрий	Сергеевич	09.06.1995	Смелянская, 55	3-77-22	125	
Группы		+		3 Гушлевский	Александр	Олегович	11.03.1995	Добролюбова, 11	3-65-11	125	
		+		4 Ворсул	Ирина	Валериивна	02.12.1995	Орджоникидзе, 44	3-12-55	125	
Дисциплины		+		5 Кунов	Максим	Иванович	09.09.1995	Вербовецкого, 4	3-26-64	125	
🛄 Зачет		+		6 Лукашенко	Олег	Владимирович	12.11.1995	Кармелюка, 38	3-27-74	125	
Формы	\$	÷		7 Порублевый	Олексей	Дмитриевич	03.04.1995	Розы Люксембург, 48	3-15-77	125	
🔠 Зачет		÷		8 Сиваченко	Максим	Петрович	29.06.1995	Героев Днепра, 64	3-63-54	125	
		+		9 Чупрун	Эдуард	Петрович	14.04.1995	Волкова, 27	3-22-65	125	
		+		12 Клинтон	Петро	Владимирович	11.05.1995	Хмельницкого, 73	3-79-31	125	
		*	(	Nº)						125	

Рисунок 34 – Внесення змін у таблицю «Анкета»

Все объекты Access			Зачет			
Поиск	Q	2	Дисциплина 👻	Группа 👻	Студент 👻	Результат
Таблицы	*		Политология	KM-125	Бондаренко	$\checkmark$
Анкета			Организация баз данных_1	KM-125	Бондаренко	
Группы			Политология	KM-125	Возный	$\checkmark$
			Организация баз данных_1	KM-125	Возный	
дисциплины			Политология	KM-125	Ворсул	$\checkmark$
🔜 Зачет			Организация баз данных_1	KM-125	Ворсул	
Формы	*		Политология	KM-125	Гушлевский	$\checkmark$
🗐 Зачет			Организация баз данных_1	KM-125	Гушлевский	
			Политология	KM-125	Кунов	
			Организация баз данных_1	KM-125	Кунов	$\checkmark$
			Политология	KM-125	Лукашенко	$\checkmark$
			Организация баз данных_1	KM-125	Лукашенко	
			Политология	KM-125	Порублевый	$\checkmark$
			Организация баз данных_1	KM-125	Порублевый	$\checkmark$
			Политология	KM-125	Сиваченко	
			Организация баз данных_1	KM-125	Сиваченко	
			Политология	KM-125	Чупрун	<b>V</b>
			Организация баз данных_1	KM-125	Чупрун	$\checkmark$
		*	•			

Рисунок 35 - Автоматичні зміни і у таблиці «Зачёт»



Рисунок 36 – Встановлення зв'язків між таблицями

# 4. ЛАБОРАТОРНА РОБОТА №3

Тема: «Підпорядковані форми. Форматування форм».

Мета: Ознайомитись з процесом створення підлеглих форм з метою перегляду даних, що мають відношення «один-до-багатьох». Навчитися налаштовувати зовнішній вигляд форм.

# 4.1. Завдання

### Завдання 1:

- створити підпорядковану форму за результатами складання іспитів з вибіркою по кожному студенту окремо;

- змініть стиль форми «Анкета» на свій розсуд.

### Завдання 2:

 змініть колір фону і колір шрифту для кожного поля за своїм бажанням;

- на місце поля «Код» поставте малюнок.

#### Завдання 3:

- спробуйте змінити розмір поля для введення прізвища по висоті;

- змініть шрифт (розмір, вид, колір) для поля «Фамилия».

### Завдання 4:

- видалити в підлеглій формі "Экзамен" поле "Студент";

- вийдіть з режиму конструктора і подивіться, як тепер виглядає форма.

# 4.2. Теоретичні відомості

Часто потрібно візуально представити відношення 1:М між елементами даних. Для цього використовується режим підпорядкованої форми. Якщо необхідно подивитись дані про результати екзаменів по кожному студенту окремо або по кожній дисципліні використовують таблицю, де занесені загальні

результати, які отримали студенти по дисциплінам протягом семестру. Для такого перегляду і використовують створені підлеглі форми.

Наприклад, при описанні видавництва, ми хочемо бачити список книг, виданих у даному видавництві. Таблиця з даними про видавництва, пов'язана з таблицею "Книги". Зв'язок здійснюється за допомогою поля "Код видавництва", присутнього в обох таблицях. Виберіть піктограму "НОВА ФОРМА", після чого MS Access відкриє діалогове вікно "Створення форми". Виберіть таблицю "Видавництва" і натисніть кнопку "Майстер". Зі списку майстрів виберіть "Складова форма" і знову натисніть ОК . Даний майстер автоматично доповнює створювану форму так званої підпорядкованою формою. У першому діалоговому вікні майстра запитується ім'я таблиці для підпорядкованої форми. Виберіть зі списку таблицю "Книги" та натисніть кнопку "Далі". Наступне діалогове вікно призначене для вибору полів головної форми. Поля підпорядкованої форми вибираються в наступному діалоговому вікні. Ще одне діалогове вікно дозволяє виконати розміщення полів у формі. В останньому діалоговому вікні залишається задати заголовок форми і закінчити роботу з майстром, натиснувши кнопку "Готово". Потім майстер запропонує ввести ім'я для створеної підпорядкованої форми. Відкрийте головну форму в режимі перегляду. У верхній частині форми розташовані поля таблиці "Видавництва", нижче - таблиця, утворена полями таблиці "Книги". Особливість полягає в тому, що в формі відображаються тільки ті з записів таблиці "Книги", які пов'язані з поточним записом таблиці "Видавництва". При редагуванні даних Access автоматично підтримує цілісність даних. MS Тому запис про видавництво можна видалити лише після того, коли будуть видалені всі записи про відповідні йому книги. При введенні інформації про нові книжки MS пов'язує записи з поточним видавництвом. Access автоматично Тому користувачеві не потрібно самому заповнювати поле "Код видавництва". Це поле можна видалити з підпорядкованої форми.

Складова форма може бути побудована і іншим шляхом. Для цього спочатку розробляється підпорядкована форма у вигляді «ленточной» форми. А

потім розробляється основна форма, а підпорядкована форма вбудовується шляхом простого перетягування готової підпорядкованої форми та розміщення її всередині основної.

Зовнішній вигляд форми можна змінити за такими параметрами:

- стиль;
- шрифт;
- додати надписи;
- видалити поля;
- додати рисунок, діаграму.

Для редагування форми використовується режим *Конструктор*. Форму також можна відкрити відразу в режимі *Конструктор*, для цього у вікні бази даних вкажіть на її ім'я та натисніть кнопку "Конструктор".

При виведенні форми в режимі *Конструктор* автоматично виводиться панель елементів управління (у вигляді панелі, що зміститься). Панель елементів дозволяє додавати елементи управління у форми.

Для вибраних елементів можна змінити шрифт, розмір та інші властивості. Для цього використовуйте списки і кнопки, розташовані на панелі управління. Розташування елемента форми можна змінити, використовуючи техніку буксирування.

Для зміни кольору необхідно відкрити палітру доступних кольорів за допомогою піктограми "ПАЛІТРА". Виберіть один або кілька елементів управління і встановіть необхідний колір.

Для форми MS Access присутні три області: область заголовка (розміщується заголовок і примітки форми), область колонтитулів (відображаються верхній і нижній колонтитули) і область даних (безпосередньо відображаються дані таблиці). Вміст областей заголовка і колонтитулів не змінюється при переміщенні по записах, на відміну від області даних. Різниця між областями заголовка, колонтитулів і даних стає очевидним при роздруківці форми.

Для кожної області форми можна встановити свій колір. Для цього слід спочатку вказати на відповідну область, а потім вибрати необхідний колір на палітрі кольорів.

Для наочності відображення у формі даних таблиць і запитів вміст полів можна позначити підписами. Підпис в цьому випадку буде поясненням до даних, що відображаються в полі. При переміщенні одного з цих елементів переміщається і інший. Для того щоб за допомогою буксирування окремо перемістити підпис або поле, слід зачепитися за великий квадрат, що знаходиться в лівому верхньому кутку відповідного елемента.

Для зміни розміру елемента вікна форми слід перемістити в потрібну позицію один із маркерів виділення.

Колір, розмір і місце розташування є тільки частиною властивостей елементів форми. Повний список всіх властивостей елементів можна отримати, відкривши вікно властивостей шляхом вибору однойменної піктограми. Якщо виділено кілька елементів вікна, можна відразу змінити їх загальні властивості.

# 4.3. Хід виконання роботи та результати завдань

### Завдання 1.

Для того, щоб створити підпорядковану форму за результатами складання іспитів з вибіркою по кожному студенту необхідно відкрити базу даних, яка була створена в результаті виконання лабораторної роботи № 2. Потім потрібно вибрати вкладку «*Создание»* і натиснути кнопку «*Мастер запросов»*. Після цього з'явиться вікно «*Новый запрос»*, потрібно обрати «*Простой запрос»* і натиснути кнопку «*ОК»* (рис. 1).

У вікні «Создание простых запросов» потрібно обрати зі списка «Таблицы и запросы» таблицю «Экзамен» (це буде <u>підпорядкована форма</u>). Перемістити в список «Выбранные поля» поля: *«Дисциплина», «Студент», «Результат контроля»*. Не виходячи з вікна *«Создание простых запросов»,* 

потрібно обрати із списка «Таблицы и запросы» таблицю «Анкета» (це буде <u>головна форма</u>) и помістити в список «Выбранные поля» поле *«Фамилия»*. Потім натиснути кнопку *«Далее»* (рис. 2).

	Простой запрос Перекрестный запрос Повторяющиеся записи
Создание запроса на выборку из	
определенных полеи.	

Рисунок 1 – Початок роботи

	Выберите поля для запроса. Допускается выбор нескольких таблиц или запросов.
<u>Т</u> аблицы и запросы	
таблица: Анкета	
Д <u>о</u> ступные поля:	В <u>ы</u> бранные поля:
Код	Дисциплина
Имя	Студент
Отчество	>> Результат контроля
Дата рождения	Фамилия
Адрес	
Телефон	
Группа	

Рисунок 2 – Створення простих запитів

В полі «Задайте имя запроса» необхідно вказати назву запита, який створюється, в даному випадку це – «Анкета для экзамена». Натиснути кнопку «Готово» (рис. 3).

Создание простых запросов	3
	Задайте имя запроса: Анкета для экзамена
	Указаны все сведения, необходимые для создания запроса с помощью мастера. Дальнейшие действия: © <u>О</u> ткрыть запрос для просмотра данных. © <u>И</u> зменить макет запроса.
	Отмена < Назад Далее > Готово

Рисунок 3 – Введення назви запита

### Створений запит «Анкета для экзамена» виглядає наступним чином:

Анкета для экзамена					
Дисциплина	🔹 Студент	-	Результат контроля	-	Фамилия 👻
Английский язык	Бондаренко		неудовлетворительно		Бондаренко
Системное програмирование	Бондаренко		хорошо		Бондаренко
Английский язык	Возный		хорошо		Возный
Системное програмирование	Возный		удовлетворительно		Возный
Английский язык	Гушлевский		удовлетворительно		Гушлевский
Системное програмирование	Гушлевский		неудовлетворительно		Гушлевский
Английский язык	Коваленко		неудовлетворительно		Коваленко
Системное програмирование	Коваленко		отлично		Коваленко
Английский язык	Кунов		отлично		Кунов
Системное програмирование	Кунов		хорошо		Кунов
Английский язык	Лукашенко		хорошо		Лукашенко
Системное програмирование	Лукашенко		удовлетворительно		Лукашенко
Английский язык	Порублевый		удовлетворительно		Порублевый
Системное програмирование	Порублевый		неудовлетворительно		Порублевый
Английский язык	Сиваченко		неудовлетворительно		Сиваченко
Системное програмирование	Сиваченко		отлично		Сиваченко
Английский язык	Чупрун		отлично		Чупрун
Системное програмирование	Чупрун	-	хорошо		Чупрун

Рисунок 4 – Загальний вигляд створеного запиту

При обрані одного (можна декілька) прізвища студента можна побачити, як змінюється вміст таблиці «Анкета для екзамена». Причому інформація, що з'явиться у таблиці стосується тільки обраного студента. Таким самим чином можна зробити запит на дисципліну або результат контролю (рис. 5).

Все объекты Access	<b>*</b> *		Анкета для экзамена						
Поиск	P	2	Дисциплина	*	Студент	•	Результат контроля	*	Фамилия 👻
Таблицы	\$		Английский язык		Бондаренко	•	неудовлетворительно		Бондаренко
🛄 Анкета			Системное програмирование		Бондаренко		хорошо		Бондаренко
🛄 Группы									
🛄 Дисциплины		$\vdash$							
🛄 Экзамен									
Запросы	\$								
📑 Анкета для экзамена									

Рисунок 5 – Загальний вигляд форми по студенту

Для того, щоб змінити стиль форми «Анкета» потрібно відкрити форму в режимі *Конструктор*, натиснути на панелі інструментів кнопку *«Теми»* та обрати потрібний стиль, як показано на рис. 6.

A   🛃	9-(4-1-	-	-					Инструме	енты конструкто	ра форм	лаба_3 : ба	за данн
Файл	Главная	Создание	Внешние	данные	Работа с	база	ами данных	Конструктор	Упорядочить	Формат		
Режим	Аа Темы Темы А и	Цвета т Шрифты т	ab	Aa 🔤	× 🗋		) [ <sup>xyz</sup> ] •				Бставить вображение т	∰ Эм ⊒ За 53 Да
Режимы					-		Элемен	ты управления				Кол
Все об Поиск Таблии		Aa	Aa _	Aa_		i P	/ппы 🛄 Д • 5 • 1 • 6 •	исциплины	Экзамен 9 · i · 10 · i · 11 ·	ı · 12 · ı · 13 ·	ı · 14 · ı · 15 ·	ı · 16 ·
	Aa_	Aa	Aa	Aa			Имя	Отнество	Аления Ал	pec	Телеф; п	a
Запрос	Aa_	Aa _	Aa _	Aa			• Имя	Отчество	Дата р Ад	pec	Телеф І	
Формь Парала	Aa	Aa	Aa	Aa								
🖼 Д	Aa	Aa	Aa_	Aa_		-						
	Разрен Поиск Сохра	шить обновле тем нить текущую	ние контента тему	с сайта Offi	ce.com							
			5									

Рисунок 6 – Зміна стилю форми «Анкета»

_									
ſ	- В Анк	сета							
	Ан	кет	а						
	_	Код	Фамилия		Имя	Отчество	кдения	Адрес	Телеф та
	▶  [	1	Бондаренко	•	Евгени	Максимович	######	Чехова, 23	3-47-5 ##
Γ		2	Возный	•	Юрий	Сергеевич	######	Смелянская, 55	3-77-2 ##
		3	Гушлевский	•	Алекс	Олегович	######	Добролюбова, 1	3-65-1 ##
Γ		4	Коваленко	•	Ирина	Валериивна	######	Орджоникидзе,	3-12-5 ##
		5	Кунов	•	Макси	Иванович	######	Вербовецкого,	3-26-6 ##
		6	Лукашенко	•	Олег	Владимирович	######	Кармелюка, 38	3-27-7 ##
		7	Порублевый	•	Олекс	Дмитриевич	######	Розы Люксембу	3-15-7 ##
		8	Сиваченко	•	Макси	Петрович	######	Героев Днепра,	3-63-5 ##
		9	Чупрун	•	Эдуар	Петрович	######	Волкова, 27	3-22-6 ##
		12	Клінтон	•	Петро	Владимирович	######	Хмельницкого,	3-79-3 ##
-	*	<b>(</b> №)		•					t#

Рисунок 7 - Форма «Анкета» після зміни стиля

# Завдання 2.

Для того щоб змініти колір фону і колір шрифту поля потрібно відкрити форму в режимі *Конструктор*, натиснути на поле, перейти на вкладку формат та обрати потрібний колір фона та колір шрифту, як показано нижче:



Для того, щоб вставити на місце поля «Код» малюнок необхідно натиснути кнопку «Вставить изображение» на панелі інструментів, натиснути мишкою по тому місцю, де повинен бути малюнок (поле «Код») та обрати будьякий з файлів в папці «Program Files\Microsoft Office\Clipart».

A 🖬 🛃 🔊 ד 🕲 ד 🖛	Инструменты конструктора	форм лаба_3 : база дан								
Файл Главная Создание	Внешние данные Работа с базами данных Конструктор Упорядочить	Формат								
Режим Темы А Шрифты т	abl Aa 🚥 🗀 🧶 🏋 🖶 📑 📫 🔪 븜 1	Г Т Т Т Т Вставить изображение ▼								
Режимы Темы	Элементы управления	Ko								
Все объекты Access 💿 «	Паралини Санана С									
Поиск	• 1 • 1 • 1 • 2 • 1 • 3 • 1 • 4 • 1 • 5 • 1 • 6 • 1 • 7 • 1 • 8 • 1 • 9 • 1 • 10 • 1 • 11 • 1	·12 · I ·13 · I ·14 · I ·15 · I ·16								
Таблицы 🔅	🗲 Заголовок формы									
Анкета	Анкета									
Дисциплины	Код Фамилия Имя Отчество кдения Адре	ес Телеф іа								
Экзамен	Фобласть данных									
Запросы 🖈	- Фамили Имя Отчество Дата р Адр	ес Телеф Г								
📑 Анкета для экзамена										
Формы	- <b>X</b>									
-в Анкета	2									
🖽 Дисциплины										
🔳 Экзамен										
	Рисунок 9 – Встановлення малюнка									

Після форматування форма «Анкета» набула наступного вигляду, що зображено на рис. 10.

A   🛃 🔊 🔹	(~ -   -		1.1				лаба	_3 : база	а данных (Access 2007	- 2010) -	Microsoft A	ccess	
Файл Гл	авная	Создание	Внешние да	ные	Работа с база	ми данных							
Части Части приложения ч Шаблоны	Таблица	Конструктор таблиц Таблицы	Списки SharePoint •	Мастер запросов За	Конструктор з запросов апросы	<b>Ф</b> орма	Конструктор форм	Пустая форма Формы	📉 Мастер форм 🛅 Навигация т 👼 Другие формы т	Отчет	Конструктор отчетов	Пустой отчет Отчеты	💐 Мастер с 📄 Наклейк
Все объекть	Access		🔠 Анкета										
Поиск Таблицы		<b>م</b> *	Анке	ета									
🛄 Анкета			ŀ	Сод Фам	илия	Имя	Отчество		ждения Адрес		Телеф та		
🛄 Группы				<b>1</b> Бон	даренко	• <mark>Евген</mark>	Максимо	вич	###### Чехова,	23	3-47-5. #		
Дисцип	ілины		_~5	į									
Экзаме	н												
Запросы	для экзаме	≈											
Формы		\$	_			_							
🔳 Анкета			EX.	Воз	ный	Юрий	Сергееви	ч	###### Смеляно	жая, 55	5 <mark>3-77-2</mark> #		
🔳 Дисцип	ілины			7									
ї Экзаме	н												
			_\$	Гуш	левский	• <mark>Алекс</mark>	Олегович	I I	####### Добролк	бова, 1	3-65-1 #		
		-	- Ø	Ков	аленко	• Ирина	Валерния	на	####### Орджони	икидзе	, 3-12-5 #		

Рисунок 10 - Створена загальна форма «Анкета»

# Завдання 3.

A   🚽 🔊 - 🤄 -   <del>-</del>	11	Инструменты конструктора форм	лаба_3 : база данных (Access 2007 - 2010)
Файл Главная Создание Вн	Внешние данные Работа с базами данных	Конструктор Упорядочить Формат	
Режим Темы А Шрифты -	ab) Aa 🚥 🗋 🌏 [ <sup>xyz</sup> ] 🖡		Вставить ображение Сарата и время Иславить Сарата и время Иславить Поля
Режимы темы	Элементы	управления	Колонтитулы
Bce объекты Access 💿 « 📑	В Анкета		
Поиск Р	• • • • 1 • 1 • • 2 • • • 3 • • • 4 • • • 5 • • • 6 • • •	7 · · · 8 · · · 9 · · · 10 · · · 11 · · · 12 · · · 13 ·	· · 14 · · · 15 · · · 16 · · · 17 · · · 18 · · · 19 · · ·
Таблицы 🌣	🗲 Заголовок формы		
🛄 Анкета 📋			
. Пруппы 1	Апксіа		
Дисциплины	Код Фамилия Имя О	тчествр ждения Адрес	Телеф на
Экзамен	• Область данных		
Запросы 🖈 🗧	Фамили Имя О	тчество Дата р( <mark>Адрес</mark>	Телеф Г
Анкета для экзамена 1	- <u></u>		
Формы 🌣 .	H H		
🔳 Анкета 2			
. 3			
🖽 Экзамен	🗲 Примечание формы		

Рисунок 11 - Зміна розміру поля «Прізвище» по висоті

Для того, щоб змінити розмір поля для введення прізвища по висоті треба натиснути мишкою по цьому полю і змінити межі поля, що з'явилися. Після цього потрібно змінити шрифт (розмір, вид, колір) для поля «Фамилия» (вкладка «Формат») (рис. 11-12).

	ਅ - (ਦ -   <del>-</del>	-	1.1			лаба_	_3 : база данных (	Access 2007 - 2	010) - Microsoft Ad	cess
Файл	Главная Созда	ние Вн	ешние данны	е Работа с базал	ии данных					
Режим	Вставить Формат	ъ вать по образцу	Фильтр А↓	По возрастанию По убыванию Удалить сортировку	🎸 Выде 🎦 Допо 🍸 Филь	ление т лнительно т гр	Обновить все *	Создать Σ Сохранить 🍄 Далить * 📰	Итоги Орфография Дополнительно т	Найти
Режимы	Буфер обмена	i Gi		Сортировка и ф	бильтр			Записи		
Bce of	ъекты Access	💌 « =	Анкета							
Поиск		2	Δμκρτ	a						
Табли	цы	*	Anter	a						
- AI	нкета		Код	Фамилия	Имя	Отчество	ждения	Адрес	Телеф та	
п Гр	уппы	▶	1	Foudan	Евген	Максимон	8114 #######	Чехова, 23	3-47-5. #	
Д	исциплины			Болоар						
ш э	кзамен			енко						
Запрос	ы	*								
AI	нкета для экзамена									
Формь	d .	*	D-XI	<b>n</b> ~	Юрий	Centeenu	4	Смелянска	ия. 55 <b>3-77-2</b> #	
	нкета		Chi and a state of the state o	возныи						
	исциплины									
<u> = =</u> = >i	сзамен				1					
			Derti		Arous	Orecourse		Ποδροποδι		
			_&}	Гушлев	Alecc	Олегович		дооролюо	JBa, 1 5-05-1 •#	
				ский						
					_					
			No.4		1	-				
			_©;;}	Ковале	ирина	Валеринв	Ha ////////////////////////////////////	Орджоник	идзе, 3-12-5 #	
				нко						

Рисунок 12 - Створена форма «Анкета»

## Завдання 4.

A 🛃 🤊 • 🖓 • 🔤	-			Работа с запроса	ми лаба_3 : база д	данных (Access 2007 - 2
Файл Главная	Создание	Внешние данные Ра	абота с базами данных	Конструктор		
Режим Выполнить	Выборка Создан	ние Добавление Обновле цы	ние Перекрестный Уд	<ul> <li>Объедине</li> <li>Объедине</li> <li>К серверу</li> <li>Управлени</li> </ul>	ние Стобразить таблицу	<ul> <li>Вставить строки</li> <li>Удалить строки</li> <li>Построитель</li> </ul>
	0	[	Sampoca			Haciponia sang
Bce объекты Acces	5 ♥ ≪	Анкета для экзамен	13			
Поиск	<u>م</u>	Aurora		1233 Mark		
Таблицы	*	*		*	1	
🛄 Анкета		😵 Код		Дисциплина		
🛄 Группы		Фамилия		Дисциплина		
🛄 Дисциплины		Отчество		Группа.Valu		
Экзамен		Дата рож	сдения	Студент		
Запросы	\$	Адрес		Студент.Valu		
📑 Анкета для экза	мена					
Формы	\$					
🔠 Анкета						
🔳 Дисциплины						
🔳 Экзамен						
		Поле:	Дисциплина		Результат контроля	Фамилия
		Имя таблицы: Сортировка:	Экзамен		Экзамен	Анкета
		Вывод на экран:				
		Условие отбора:				
1		или:				
1						

Рисунок 13 - Видалення поля "Студент"

Для того, щоб видалити в підпорядкованій формі «Экзамен» поле «Студент» (рис. 13), потрібно відкрити форму в режимі Конструктор, виділити поле «Студент» і «вирізати» це поле (рис. 14).

Все объекты Access	☞ «	Анкета для экзамена		
Поиск	2	Дисциплина 🚽	Результат контроля 👻	Фамилия 👻
Таблицы	\$	Английский язык	неудовлетворительно	Бондаренко
Анкета		Системное програмир	хорошо	Бондаренко
EDVER N		Английский язык	хорошо	Возный
		Системное програмир	удовлетворительно	Возный
Дисциплины		Английский язык	удовлетворительно	Гушлевский
Экзамен		Системное програмир	неудовлетворительно	Гушлевский
Запросы	~	Английский язык	неудовлетворительно	Коваленко
📑 Анкета для экзамена		Системное програмир	отлично	Коваленко
Формы	*	Английский язык	отлично	Кунов
🗐 Анкета		Системное програмир	хорошо	Кунов
		Английский язык	хорошо	Лукашенко
		Системное програмир	удовлетворительно	Лукашенко
-а Экзамен		Английский язык	удовлетворительно	Порублевый
		Системное програмир	неудовлетворительно	Порублевый
		Английский язык	неудовлетворительно	Сиваченко
		Системное програмир	отлично	Сиваченко
		Английский язык	отлично	Чупрун
		Системное програм 💌	хорошо	Чупрун

Рисунок 14 - Вигляд форми після виходу з режиму конструктора

# 5. ЛАБОРАТОРНА РОБОТА №4

**Тема:** «Побудова запитів та звітів. Створення кнопок управління та кнопкових форм».

**Мета:** Вивчити методику створення запитів на вибірку та створення звітів, а також методику створення кнопок управління і кнопкових форм.

# 5.1. Завдання

#### Завдання 1:

- створити запити на вибірку записів, що містять студентів, які отримали оцінки за екзамени «Отлично», та які отримали екзаменаційні оцінки «Неудовлетворительно»;

- створити запити на вибірку записів по результатам контролю по дисципліні, яку викладає певний викладач;

- створити запит по таблиці «Анкета», який формує список студентів, що народились в певний проміжок часу. Запит зберегти під іменем «Запрос по дате рождения»;

- створити форми у режимі перегляду для кожного запиту (автоформа «ленточная»);

-закрити доступ до корегування бази даних у режимі запитів.

#### Завдання 2:

- створити звіт по результатам здачі екзаменів, використовуючи таблицю «Экзамен» та звіт для таблиці «Зачёт»;

- опрацювати область заголовка та область даних. Встановити в поле дати та форматувати кожне поле. Перейти на вкладку «Данные» та обрати формат зображення дати у звіті.

# Завдання 3:

- створити кнопки виходу зі всіх форм, які ви створили:

• Анкета;

- Преподаватели;
- Дисциплины;
- Группы;
- Экзамен;
- Зачёт;
  - створити кнопки виходу для всіх форм і запитів;

- створити кнопки виходу для всіх звітів. Оформлення кнопок можливе в режимі *Конструктор* за допомогою вкладки «Цвет текста».

### Завдання 4:

- створити меню 1-го рівня;

- створити меню 2-го рівня;
- створити меню 3-го рівня;

- створити меню 3-го рівня для виклику запитів та для перегляду звітів згідно структури зображеної на рисунку 1:



Рисунок 1 - Схема 3-х рівневого меню

# 5.2. Теоретичні відомості

Запити забезпечують швидкий і ефективний доступ до даних, що зберігаються в таблиці. При створенні таблиці за допомогою MS Access можна помітити, що програма не дозволяє використовувати поля для обчислювання. Не існує також можливості виконати сортування не за первинним ключем.

Завдяки запитам, можна виконати сортування або обчислити вирази. Наприклад, звести разом дані з пов'язаних таблиць. При виконанні запиту MS Access зчитує дані з таблиць і відображає результат виконання в режимі таблиці. При цьому слід враховувати, що результат виконання запиту не зберігається. Дані завжди зберігаються в таблицях. У запиті MS Access зберігає тільки інструкції про те, як повинні бути організовані дані в результаті виконання запиту.

Для деяких запитів можна внести зміни в результат запиту, що відображається в режимі таблиці. Ці зміни відіб'ються і у вихідних таблицях. На підставі запиту можна розробити форму або звіт.

### Сортування записів таблиці

У таблиці дані завжди відсортовані по первинному ключу. Щоб організувати інший спосіб сортування даних, використовується запит. Для створення такого запиту у вікні бази даних слід вибрати піктограму «ТАБЛИЦА» і вказати на ту таблицю, яку необхідно відсортувати. Вибір піктограми «НОВЫЙ ЗАПРОС», розташованої на панелі інструментів, призводить до відкриття діалогового вікна для створення нового запиту для зазначеної у вікні бази даних таблиці. У результаті вибору піктограми «НОВЫЙ ЗАПРОС» в діалоговому вікні відображається список всіх імен полів вихідної таблиці, на початку якого розташований символ "\*". Перенесіть зірочку в бланк запиту QBE. Вона є символом - замінником всіх полів даних. Крім того, перенесіть в бланк запиту то поле даних, за яким має бути проведено сортування.

Для цього поля зі списку «Сортировка» виберіть спосіб сортування. Виберіть піктограму «ВЫПОЛНИТЬ» і MS Access зобразить на екрані результат запиту, що відображається в режимі таблиці, дані в якому відсортовані потрібним чином.

### Запити за критерієм пошуку

Для вибірки даних з таблиць по певній умові в MS Access використовуються запити з критеріями пошуку. Попередньо записавши умову

відбору, результатом виконання запиту буде набір даних, що задовольняють сформованому критерію.

Для складання запиту слід перейти у вікно бази даних, вибрати піктограму «ЗАПРОС» і натиснути кнопку «Создать», а потім кнопку «Новый запрос». У наступному вікні діалогу слід вибрати одну з таблиць БД і натиснути кнопку «Добавить», після чого MS Access помістить у вікно запиту список полів зазначеної таблиці. Тепер можна вибрати одне або кілька полів і перетягнути їх у бланк запиту QBE.

В рядку «Условие отбора» можна ввести критерії вибору. Для позначення довільної кількості символів використовуйте в якості символу-замінника зірочку (\*), а для позначення одного довільного символу - знак питання (?). Сформуйте критерії запиту. Наприклад , якщо для текстового поля буде заданий тільки критерій « А \*» , то в результаті виконання запиту будуть відображені тільки ті записи, текстове поле яких починається з букви «А» або «а».

Якщо потрібно відшукати комбінацію декількох значень, необхідно ввести їх в рядок «Условие отбора», вживаючи як роздільник логічні оператори AND і OR. Крім звичайних операторів порівняння, MS Access надає три спеціальних оператора, корисних для відбору даних: BETWEEN, IN, LIKE.

### Запити з параметрами

Не завжди можна вирішити на етапі створення запиту, які саме значення повинен відшукувати MS Access, а іноді потрібна гнучка умова відбору, яка може змінюватися в залежності від поточної ситуації. Тому замість умови відбору, що безпосередньо вводиться в бланк QBE, можна включити в запит параметр, і в цьому випадку перед виконанням запиту MS Access кожного разу буде запитувати конкретні умови відбору.

Щоб встановити параметр необхідно ввести в рядок «Условие отбора» ім'я або фразу, укладену в квадратні дужки ([]). Те, що записано всередині квадратних дужок, MS Access розглядає як ім'я параметра. Це ім'я виводиться в

діалоговому вікні при виконанні запиту, тому необхідно в якості ім'я параметра використовувати змістовну фразу. В одному запиті можна задати декілька параметрів, при цьому ім'я кожного параметра має бути унікальним та інформативним.

За замовчуванням MS Access перетворює введені в запит параметри в текстовий тип даних. Однак, так як текстові поля не можна порівнювати з полями дат, тому існує можливість для параметрів тип даних встановлювати точно. Для цього виберіть команду *Запрос/Параметры* з меню, в результаті чого MS Access відкриє вікно діалогу «Параметры запроса». Введіть параметр без квадратних дужок у діалогове вікно і виберіть необхідний тип даних.

Тепер для виконання запиту спочатку необхідно ввести значення параметра, а результат запиту буде містити тільки ті поля, які задовольняють заданому значенню параметра.

### Обчислюване поле в запитах

Існує можливість обчислення над будь-якими полями таблиці і можливість зробити обчислюване значення новим полем в наборі записів. Для цього використовується безліч вбудованих в MS Access функцій. Можна створити поле в запиті з використанням арифметичних операцій над полями таблиць, а можна також створити нове поле як результат конкатенації текстових полів або символьних констант.

Створення обчислюваного поля здійснюється шляхом простого введення виразу для обчислення в порожньому стовпці бланка запиту QBE. Необхідно всього лише скласти відповідний запит з використанням імен полів:

### Вираз 1: [Поле1] «оператор» [Поле2]

У квадратні дужки записують тільки імена полів. MS Access автоматично використовує «Вираз 1», як ім'я обчисленого виразу. Це ім'я можна змінити на більш змістовне.

При побудові складних виразів MS Access пропонує використовувати утиліту, під назвою «Построитель выражений». Для цього необхідно клацнути

по пустому полю в бланку QBE, а потім - по кнопці «Построить» на панелі інструментів.

### Підсумкові запити

Підсумкові запити значно відрізняються від звичайних. В них можна використовувати два типи полів. Існують поля, за якими здійснюється угруповання даних, і поля, для яких проводяться обчислення. Додатково можна включити ще поля для визначення умов запиту.

Для того щоб сформувати підсумковий запит, перебуваючи в режимі *Конструктор запроса*, слід вибрати піктограму «ГРУППОВЫЕ ОПЕРАЦИИ» або скористатися командою *Вид/Групповые операции*. В результаті чого в бланку запиту з'явиться рядок "Групповая операция". Потім для відповідного поля зі списку необхідно вибрати функцію "Группировка".

### Багатотабличні запити

Розглянувши можливості запитів, заснованих на одній таблиці, легко організувати перегляд об'єднаних даних з декількох пов'язаних таблиць. Вибір даних з кількох таблиць особливо корисно використовувати при створенні форм і звітів.

По-перше відкрийте вікно *Конструктор запроса* та додайте таблиці, в яких міститься необхідна інформація. Якщо раніше за допомогою команди *Связи* було встановлено зв'язок між таблицями, то MS Access визначить це автоматично. Цей зв'язок буде відображено у вигляді лінії, проведеної між полями таблиць. Якщо зв'язок між таблицями відсутній, ви повинні його встановити.

Далі необхідно визначити параметри зв'язку. Вкажіть на сполучну лінію і виберіть команду *Bud/Параметры объединения*. В результаті розгорнеться діалогове вікно, в якому пропонуються три опції для визначення параметрів зв'язку. Після цього перенесіть в бланк запиту поля з головної і підлеглої таблиці. Як правило, існує більше записів, що містять відомості про об'єкти,

ніж записів про їх категорії. Тому MS Access відображає категорію стільки разів, скільки це необхідно. Отже, якщо однієй категорії відповідає десять різних об'єктів, категорія буде повторена десять разів.

<u>Примітка</u>: Поля головної таблиці при відношенні «один- до-багатьох» не можна змінити в результаті запиту, що зображено в режимі таблиці. Крім того, ці поля можуть містити повторювані значення.

Будь-які запити вибору можуть бути перетворені в запити створення таблиці. У цьому випадку створюється таблиця, яка містить результат запиту, отриманий в даний момент.

### Запити на видалення

Всі вище описані запити відносять до запитів вибору, які дозволяють тільки вибирати та проглядати дані з таблиць. Наступні запити є діючими, тобто завдяки яким маємо можливість модифікувати збережену інформацію.

На основі запиту на видалення відбувається видалення відповідних даних з таблиці. Перед тим, як сконструювати цей запит необхідно чітко розуміти які запити надалі будемо видаляти. Надійним методом є попереднє створення запиту-вибору.

В режимі «Конструктора запроса» необхідно обрати таблицю, в якій пізніше буде проводитись дія на видалення. Далі необхідно визначити критерії та потім продивитись результат виконання запиту, що відтворюється в режимі таблиці. Швидке переключення між режимами відбувається за допомогою іменних піктограм, що розташовані на панелі інструментів.

Тільки у випадку, якщо в результаті запиту існують записи на видалення є можливість перетворення запиту-вибору в запит-видалення. Для виконання цієї дії необхідно в рядку меню вибрати команду *Запрос/Удаление*. Після цього MS Access вимикає в бланку запиту рядки "Сортировка" та "Вывод на экран". Дія цих рядків вже непотрібна. Крім того, для всіх існуючих колонок бланка запиту в новому рядку "Удаление" відтворюється текст "Условие". Критерії при цьому незмінні.

Якщо виконати запит через піктограму "ВЫПОЛНИТЬ", MS Access не відобразить результат виконання запиту на екрані. Замість нього з'явиться повідомлення про кількість записів на видалення по первинній таблиці. Є можливість переривання процесу видалення при натисненні кнопки "Отмена". А при натисненні OK всі записи безповоротно видаляються за обраними критеріями.

#### Запити-оновлення

Запит – оновлення дозволяє проводити оновлення деякої збереженої інформації, своєчасність якої вже не має сенсу на цей момент. Тому спочатку необхідно створити новий запит-вибору та ввести критерії відбору щодо змінених записів. Якщо результат запиту відповідає конкретним вимогам, тоді з'являється нагода перетворення цього запиту в запит-зміну. Для цього необхідно вибрати команду *Запрос/Обновление*. При цьому будуть вимкнені рядки "Вывод на экран" та "Сортировка" в бланку запиту, тому що вони не потрібні на певний момент. Замість них підключається новий рядок "Обновление", де потрібно ввести нові вирази для змінених записів. Причому тип даних вираз обов'язково повинен співпадати з типом даних поля таблиці.

# 5.3. Хід виконання роботи та результати завдань

#### Завдання 1.

Почнемо зі створення запиту під назвою Запрос по отличникам. Для цього вибору записів, які містять студентів, що склали всі екзамени на «відмінно» потрібно клацнути на вкладці «Создать» головного меню, а потім натиснути піктограму, Конструктор запросов, після чого Ассезя запропонує вибрати необхідну таблицю по якій буде створено запит. Вибрали в цьому вікні таблицю «Экзамен» і натиснули кнопку «Додати», після чого закрили діалогове вікно. Далі послідовно з таблиці вибрали поля «Студент»,

«Дисциплина», «Результат» і натиснули двічі по обраним полям, або можна перемістити кожне поле в бланк запиту. Після даних операцій в бланку запиту з'являться три поля, тепер натискаємо на кнопку «Построить» в головному меню, або за допомогою правої кнопки мишки натискаємо на полі «Условие отбора». Далі обираємо необхідний пункт в розгорнутому вікні, в якому необхідно ввести команду *Like «Отлично»*, як показано на рисунку 2.

Like "Отлично"       ОК         Отмена       Назад         + - // * & => < <> And Or Not Like ()       Вставить Справка         Вапрос1       Вставить Справка         Эапрос1       В таблицы         Эапросы       В потов         Forms       Reports         Эмроть       Ос пораторы         Операторы       Общие выражения	Построитель выражений	? 🛛
,	Like "Отлично" + - / * & = > < <> And Or Not Like () Вставить Запрос1 Таблицы Запросы Рогтав Рогтав Рогтав Операторы Общие выражения	ОК Отмена <u>Н</u> азад Справка

Рисунок 2 – Будівник виразів

Надалі з'явиться запит, який необхідно зберегти під назвою «Запрос по отличникам», як показано нижче:

	іне 🔳 Запрос	по отличникам 📑 А	нкета для экзамена	📑 Запрос по отлич	никам				
Экзамен Экламен Экламен Фисциплина Дисциплина Группа.Valu Студент.Valu									
Поле: Имя таблицы: Сортировка:	Студент Экзамен	Дисциплина.Value Экзамен	Результат контроля Экзамен						
Вывод на экран:									

Рисунок 3 – Запит по відмінниках

Наступний запит буде мати назву «Запрос по двоечникам». Він створюється аналогічно попередньому запиту, але як умову відбору він повинен використовувати Like "Неудовлетворительно":

🗐 Запрос по дисципли	ине 🔚 Запрос	пос	отличникам	Запрос по двоечникам	
Экзамен * — Дисципли Дисци — Группа Группа — Студент Студен	ина плина a.Valu				
Поле: Имя таблицы: Сортировка: Вывод на экран: Условие отбора:	Студент Экзамен ✓	•	Дисциплина.Value Экзамен	Результат контроля Экзамен И Like "неудовлетвори-	
или:					

Рисунок 4 – Запит з умовою

Тепер створимо запит на вибір всіх записів по результату контролю за дисципліною, яку викладає певний викладач. Щоб виконати завдання необхідно додати в таблицю поле «Преподаватель». Для того, щоб створити запит потрібно натиснути на вкладці головного меню «Coздать», на іконку Конструктор запросов, після чого Access запропонує вибрати необхідну таблицю по якій буде створений запит. Виберемо в цьому вікні таблицю «Экзамен» і натиснемо кнопку «Додати», після цього діалогове вікно потрібно закрити і перед нами з'явився бланк запитів та вікно таблиці. Далі необхідно перемістити поля «Студент», «Дисциплина», «Результат», «Преподаватель» в бланк запиту і в пункті умови відбору вказати умову - Like «Имя преподавателя», як показано на рисунку 5. Після виконання описаних дій закриваємо діалогове вікно і зберігаємо запит під назвою «Запрос по дисциплине».
	Экзамен				
	*				
	дисципли	пна			
	Дисци	плина			
	🖃 Группа				
	Труппа	a.Valu			
	🖂 Студент				
	Студен	rr.Valu			
•					
	Поле:	Студент	Дисциплина	Преподаватель	Результат контроля
	Имя таблицы:	Экзамен	Экзамен	Экзамен	Экзамен
	Сортировка:				
	Вывод на экран:	<b>V</b>	<b>V</b>	<b>V</b>	<b>V</b>
	Условие отбора:			Like "Махиня"	
	или:				

Рисунок 5 – Запит по дисципліні

Тепер потрібно створити наступний запит під назвою «Запрос по дате рождения». Знову відкриваємо Конструктор запросов. В діалоговому вікні вибираємо таблицю «Анкета» та поля: «Имя», «Отчество», «Фамилия», «Дата рождения». Для того, щоб вибрати всіх студентів з певною датою народження, потрібно використати умову відбору Between, тобто відкрити пункт «Построить» і обрати оператор порівняння Between, в якому необхідно вказати дати між якими знаходиться бажана дата народження (приклад на рисунку 6). Далі зберігаємо запит під назвою «Запрос по дате рождения».

Построитель выражений			<u> 8 23</u>			
Between 05.09. 1994 And 05.09. 1996 + - / * & = > < <> And Or Not Like () Вст <u>а</u> вить						
Запрос1     Таблицы     Запросы     Запросы     Forms     Reports     Функции     Константы     Операторы     Общие выражения	<Все> Арифметические Сравнения Логические	< <= <> = > >= Between				

Рисунок 6 – Будівник виразів

Тепер використовуючи навички, які були отриманні на попередніх лабораторних роботах потрібно створити форми для кожного запита, використовуючи *Macmep форм*, та обираємо тип форми «ленточную».

Закрити доступ до корегування бази даних у режимі запитів можна наступним чином: відкрити форму в режимі *Конструктор* і помітити поле, після цього викликати властивості і на вкладці «Данные» встановити властивість «Блокировка – да», як показано на рисунку 7:

Результат контроля					
Макет Данные События Дру	гие Все				
Данные	Результат контроля				
Источник строк	Відмінно;Добре;Задовільно;Незадо				
Тип источника строк	Список значений				
Присоединенный столбец	1				
Ограничиться списком	Да				
Разрешить изменение списка знач	є Нет				
Форма изменения элементов списк					
Наследовать список значений	Да				
Только значения источника строк	Нет				
Маска ввода					
Значение по умолчанию					
Условие на значение					
Сообщение об ошибке					
Доступ	Да				
Блокировка	Да 👻				
Автоподстановка	Да				
Смарт-теги					

Рисунок 7 – Блокування корегувань

### Завдання 2.

Створимо звіт по результатам екзаменів, використовуючи таблицю «Экзамен». Потрібно відкрити вкладку головного меню «*Coзdamь*» і вибрати пункт «*Maйcmep отчётов*», після чого з'явиться діалогове вікно, в якому можна вибрати джерело, тобто таблицю «Экзамен». Тепер перенесемо у «*Bыбранные поля»* наступні поля: «*Дисциплина*», «*Группа*», «*Cmydeнm*», «*Peзультат контроля*», як показано на рисунку 8 і натиснемо кнопку «Далее».

Создание отчетов	a despense Spenser has		
	Выберите поля для отчета. Допускается выбор нескольких таблиц или запросов.		
<u>Т</u> аблицы и запросы Таблица: Экзамен			
<u>До</u> ступные поля:	В <u>ы</u> бранные поля:		
Дисциплина Группа Студент Преподаватель	Дисциплина. Value       Группа. Value       Студент. Value       Результат контроля		
	Отмена < Назад Далее > Готово		

Рисунок 8 – Створення звітів за вибором полів

Тепер потрібно визначити, чи потрібно згруповувати дані по де-якому з полів. Поля, за якими буде здійснюватися угруповування, містяться у верхній частині правого списку в окремій рамці і виділяються на екрані синім кольором. Нам потрібно згрупувати данні по дисципліні і натиснути «Далее».

Потім задати режим в зростаючому порядку по полю «*Студент»* і натиснути «Далее». Тепер нам потрібно вибрати вид макета, в нашому випадку вибираємо «Ступенчастая» і натискаємо «Далее». Вибираємо стиль на свій розсуд і задаємо назву звіту «Отчёт по экзаменам» натиснувши в кінці кнопку «Готово». Тепер ми можемо подивитись результат натиснувши «*Файл\Предварительный просмотр»* згідно рисунка 9.

Экзамен				За	крить отчет		
Дисциплина.Value	Студент	Дисциплина	Группа	Группа	Студент	Результат контроля	Преподава
Англиискии язык	Бондаренко	Английский я:	KM-125	KM-125	Бондаренко	неудовлетворительно	Махиня
	Возный	Английский я:	KM-125	KM-125	Возный	хорошо	Махиня
	Гушлевский	Английский я:	KM-125	KM-125	Гушлевский	удовлетворительно	Махиня
	Коваленко	Английский я:	KM-125	KM-125	Коваленко	неудовлетворительно	Махиня
	Кунов	Английский я:	KM-125	KM-125	Кунов	отлично	Махиня
	Лукашенко	Английский я:	KM-125	KM-125	Лукашенко	хорошо	Махиня
	Порублевый	Английский я:	KM-125	KM-125	Порублевый	удовлетворительно	Махиня
	Сиваченко	Английский я:	KM-125	KM-125	Сиваченко	неудовлетворительно	Махиня
	Чупрун	Английский я:	KM-125	KM-125	Чупрун	отлично	Махиня

Системное програ

Рисунок 9 - Приклад звіту

Для того щоб редагувати дані необхідно відкрити звіт в режимі *Конструктор* і вибрати поле, яке необхідно змінити. Наприклад, змінимо фон заголовка на синій, для цього потрібно виділити дане поле і відкривши його властивості зазначити «*Колір фону*» - синій. Зробити такі самі дії можна з полями даних, тобто змінити колір шрифту та його розмір (рисунок 10).

Экзамен				Закр	ить отчет		
Дисциплина.Value	Студент	Дисципл	Группа	Группа	Студент	Результат контроля	Преподава
Английский язык							
	Бондаренко	Английский язык	KM-125	KM-125	Бондаренко	неудовлетворительно	Махиня
	Возный	Английский язык	KM-125	KM-125	Возный	хорошо	Махиня
	Гушлевский	Английский язык	KM-125	KM-125	Гушлевский	удовлетворительно	Махиня
	Коваленко	Английский язык	KM-125	KM-125	Коваленко	неудовлетворительно	Махиня
	Кунов	Английский язык	KM-125	KM-125	Кунов	отлично	Махиня
	Лукашенко	Английский язык	KM-125	KM-125	Лукашенко	хорошо	Махиня
	Порублевый	Английский язык	KM-125	KM-125	Порублевый	удовлетворительно	Махиня
	Сиваченко	Английский язык	KM-125	KM-125	Сиваченко	неудовлетворительно	Махиня
	Чупрун	Английский язык	KM-125	KM-125	Чупрун	отлично	Махиня
Системное програ	a						

Рисунок 10 – Зміна стилю звіту

Тепер встановимо поле дати друку звіту (рисунок 11). Для цього потрібно відкрити звіт в режимі *Конструктор* і вибрати на панелі інструментів кнопку «Поле», після цього розмістити дане поле в області верхнього колонтитула, де повинно розміщатися поле. В результаті з'явиться об'єкт, який складається з

Экзамен		09.05.20	15			Закрить отчет	
Дисциплина.Value Английский язык	Студент	Дисципл 	Группа	Группа	Студент	Результат контроля	Преподава
	Бондаренко	Английский язык	KM-125	KM-125	Бондаренко	неудовлетворительно	Махиня
	Возный	Английский язык	KM-125	KM-125	Возный	хорошо	Махиня
	Гушлевский	Английский язык	KM-125	KM-125	Гушлевский	удовлетворительно	Махиня
	Коваленко	Английский язык	KM-125	KM-125	Коваленко	неудовлетворительно	Махиня
	Кунов	Английский язык	KM-125	KM-125	Кунов	отлично	Махиня

Рисунок 11 – Встановлення дати функцією Now

### Завдання 3.

Тепер створимо кнопки виходу для всіх форм які ми створили «Анкета», «Дисциплины», «Преподаватели», «Экзамен», «Группы». Для цього нам потрібно відкрити форму в режимі *Конструктор* та вибрати режим Використання майстра на панелі інструментів, після цього натиснути «Кнопка» з панелі інструментів та натиснути лівою кнопкою миші в будь-якому місці форми, де ви б хотіли створити кнопку, після цього запуститься вікно майстра створення форми. В першому вікні діалогу розміщенні два списки «Категории» і «Действия», виберіть в списку «Категории» пункт «*Работа с формой»*, а в списку «Действия» пункт «*Закрытие формы»* і натисніть «Далее». Тепер у вікні створення кнопок введіть текст «Выход», натисніть «Далее» і задайте ім'я кнопки «**Выход с формы**». Після цього ви отримаєте кнопку на формі, яка при натисканні закриватиме форму (рис. 12).

E	🗐 Запрос по дисциплине 📳 Экзамен 📑 Анкега							
	A	нкета			Вых	код с формы		
		Код Фамилия	Имя Отчест	во жден	ия Адрес	Телефс па		
		Бондаре	ЧКО - Евген	Максимович	17.04.1995	Чехова, 23	3-47-53	125
	[	2 Возный	🗸 Юрий	Сергеевич	09.06.1995	Смелянская, 55	3-77-22	125
		<sup>3</sup> Гушлевс	<i>кий</i> - <sup>Алекс</sup>	Олегович	11.03.1995	Добролюбова, 1	3-65-11	125

Рисунок 12 – Форма з кнопкою виходу

Тепер виконуючи ті самі дії створіть кнопки для закриття форми в усіх інших формах.

Экзамен		09.05.20	15			Закрить отчет	
Дисциплина.Value Английский язык	Студент	Дисципл ЧЧС	Группа	Группа	Студент	Результат контроля	Преподава
	Бондаренко	Английский язык	KM-125	KM-125	Бондаренко	неудовлетворительно	Махиня
	Возный	Английский язык	KM-125	KM-125	Возный	хорошо	Махиня
	Гушлевский	Английский язык	KM-125	KM-125	Гушлевский	удовлетворительно	Махиня
	Коваленко	Английский язык	KM-125	KM-125	Коваленко	неудовлетворительно	Махиня
	Кунов	Английский язык	KM-125	KM-125	Кунов	отлично	Махиня

Рисунок 13 - Звіт з кнопкою виходу

Далі створимо кнопки для виходу зі звітів, для цього потрібно відкрити звіт в режимі *Конструктор* і виконати ті самі дії які були виконанні при створенні кнопки в формах, також потрібно змінити формат кнопок з допомогою вікна властивостей. Після даних дій ви отримаєте звіт з кнопкою виходу (рис. 13).

## Завдання 4.

Тепер нам потрібно створити кнопкові форми. Почнемо з створення кнопкових форм першого рівня. Для початку нам потрібно закрити всі форми, звіти та таблиці. Після даних операцій виберіть пункт меню «Работа с базами данных/Диспетчер кнопочных форм», перед вами з'явиться діалогове вікно в якому запитуватиметься «Создать кнопочную форму?». На дане питання потрібно натиснути «Да», перед вами з'явиться диспетчер кнопкових форм, після цього вам потрібно натиснути кнопку змінити ввести назву кнопкової форми «Студенти\_ЧДТУ» і натиснути кнопку «Закрити».

Диспетчер кнопочных форм	
<u>С</u> траницы кнопочной формы:	<u>З</u> акрыть
Студенти ЧДТУ (По умолчанию) Виход	Создать
Запросы Отчеты Формы	<u>И</u> зменить
	<u>У</u> далить
	По умолчанию

Рисунок 14 – Диспетчер кнопкових форм

Після цього потрібно натиснути кнопку «Создать» та у вікні «Создать» ввести текст «Формы». Після цього потрібно натиснути кнопку «Создать» і також у вікні «Создать» ввести текст «Запросы». Аналогічно створити пункти для «Отчёты» та «Выход», результат зображено на рисунку 14.

Тепер перейдемо до створення меню 2 рівня, для цього нам прийдеться розкривати кожний пункт меню першого рівня.

Для цього нам потрібно виділити курсором головну кнопкову форму «Студент» і натиснути кнопку «Изменить». В меню яке з'явилось натисніть кнопку «Создать» у вікні, яке показано на рисунку 15, потрібно ввести дані. В поле «Текст» ввести значення «Формы», в полі «Команда»: «Переход к кнопочной форме» і вибрати зі списку «Кнопкова форма» значення «Формы».

<sup>х от</sup> Изменение стра Изменение элемента	ницы кнопочной формы	
<u>Т</u> екст: Ком <u>а</u> нда: <u>К</u> нопочная форма:	Формы Перейти к кнопочной форме Формы	ОК ОТмена
		<u>В</u> верх В <u>н</u> из

Рисунок 15 – Створення елементів кнопкової форми

Аналогічно потрібно створити елементи кнопкової форми для форм «Запросы», «Отчеты». Створюючи елемент кнопкової форми «Выход» потрібно вибрати команду «Выход из приложения».

Тепер створимо меню 3-го рівня.

Для цього встановіть курсор на пункт зі списку форми і натисніть кнопку «Изменить», ви перейдете в вікно «Изменение страниц кнопочной формы». В даному вікні натисніть кнопку створити, у вікні яке з'явилось введіть такі дані: текст «Анкета», команда «Открыть форму для изменения», кнопкова форма – вибрати з списку «Анкета». Використовуючи аналогічні дії створіть елементи кнопкової форми для форм «Преподаватели», «Дисциплины», «Группы», ви повинні отримати результат який представлений на рисунку 16.

Изменение страницы кнопочной формы					
Название кнопочной формы:	<u>З</u> акрыть				
Формы					
<u>Э</u> лементы данной кнопочной формы:	Создать				
Анкета	Изменить				
і руппи Дисциплины	<u>У</u> далить				
	<u>В</u> верх				
	Вниз				

Рисунок 16 – Меню третього рівня для форм

Тільки що ви створили меню введення і корегування інформації в базу даних. Тепер побудуємо меню 3-го рівня для виклику запитів.

Для цього встановіть курсор на пункт зі списку «Запросы» і натисніть кнопку «Изменить», тоді ви перейдете у вікно «Изменение страниц кнопочной формы». В даному вікні натисніть кнопку «Создать» та введіть такі дані: текст «Запрос по двоечникам», команда «Открыть форму для изменения», кнопкова форма – вибрати з списку «Запрос по двоечникам» як показано на рисунку 17.

Изменение элемента кнопочной формы							
<u>т</u> екст:	Запрос по двоечникам	ОК					
Ком <u>а</u> нда:	Открыть форму для добавления	• Отмена					
<u>Ф</u> орма:	Запрос по двоечникам	<b></b>					

Рисунок 17 – Додавання елементів кнопкової форми

Аналогічно створіть елементи кнопкової форми для інших запитів: «Запрос по дате рождения», «Запрос по отличникам», «Запрос по дисциплине», після цих операцій ви повинні отримати результат представлений на рисунку 18.

Изменение страницы кнопочной формы	
Название <u>к</u> нопочной формы:	<u>З</u> акрыть
Запросы	
<u>Э</u> лементы данной кнопочной формы:	Создать
Запрос по двоечникам	
Запрос по отличникам	Изменить
Запрос по дате рождения	
Запрос по дисциплине	<u>У</u> далить
	Вверх
	Вниз

Рисунок 18 - Меню третього рівня для запитів

Аналогічно до попередніх завдань виконуємо створення кнопкової форми для звітів. Додаючи певний звіт, як команду потрібно вибирати «Открыть отчёт» і ввести його назву.

Тепер ви повинні закрити диспетчер кнопкових форм і перевірити чи правильно працює меню. В результаті виконання 4 завдання, ви повинні отримати кнопкову форму представлену на рис. 19. Кнопкова форма повинна містити при переході по пунктам всі звіти, форми та запити, при натисненні на пункт «Выход» база даних разом з додатком повинна бути закрита.



Рисунок 19 – Кнопкова форма

# 6. ЛАБОРАТОРНА РОБОТА №5

Тема: «Мова SQL – створення запитів. Основні оператори мови SQL».

**Мета:** за допомогою операторів мови SQL зробити вибірку даних з таблиць: додавання, видалення та редагування інформації.

# 6.1. Завдання

Не користуючись режимами *Конструктор* та *Мастер запросов* розробити перелічені нижче типи запитів до бази даних за допомогою команд мови SQL.

Завдання 1. Розробити простий вибір даних (select, \*, from, where, order by, обчислювальні поля, distinct):

- створити простий запит на вибірку з однієї таблиці. Включити кілька полів таблиці;

- включити в запит всі поля за допомогою знака "\*";

- створити запит на вибірку даних з двох зв'язаних таблиць;

вибрати кілька полів, за якими сортується висновок.;

- визначити умови відбору («І» та «АБО»). Створити не менше 2-х запитів;

визначити умови відбору за допомогою параметра запиту;

- створити обчислювальні поля;

- створити відсортований по обчислювальному полю запит з декількох таблиць, в якому визначені умови «І» та «АБО»;

- використати Distinct.

Завдання 2. Застосувати зовнішнє об'єднання таблиць:

- створити запит на зовнішнє об'єднання таблиць;

- продемонструвати використання пропозицій: Is null, Is not null;

- використати пропозицію Like;

- використати оператор UNION.

Завдання 3. За допомогою групуючих запитів з умовами (group by, having, min(), max(), sum(), count(), ...) зробити вибір даних:

- створити підсумковий запит, який містить кілька підсумкових цифр;

- створити простий групуючий запит;

- створити групуючий запит з угрупованням по декількох полях;

- створити групуючий запит, в якому визначаються умови, причому спочатку виконуються обчислення, а потім відбувається відбір;

- створити групуючий запит, в якому визначаються умови, причому спочатку відбувається відбір, а потім виконуються обчислення.

Завдання 4. Показати вибір даних за допомогою підзапитів:

- створити запит з вибором за допомогою In;

- використати All, Any, Exists.

## 6.2. Теоретичні відомості

Для отримання даних з бази даних використовується мова SQL (Structured Query Language - мова структурованих запитів). SQL - це мова програмування, яка дуже нагадує англійську, але призначена для програм управління базами даних. Кожен запит на виконання насправді грунтується на мові SQL. Розуміння принципів роботи SQL допомагає створювати покращені запити і спрощує виправлення запитів, які повертають неправильні результати.

*SQL* - це мова програмування, призначена для роботи з наборами фактів і відносинами між ними. У програмах керування базами даних, таких як Microsoft Office Access, мова SQL використовується для роботи з даними та є міжнародним стандартом, визнаним комітетами зі стандартизації ICO та ANSI. При використанні SQL необхідно застосовувати правильний синтаксис, тобто

певний набір правил, які дозволяють правильно об'єднувати різні елементи мови.

Наприклад, інструкція SQL, яка відображає список прізвищ контактів з іменем Магу має наступний вигляд:

SELECT Last\_Name FROM Contacts WHERE First\_Name = 'Mary';

Інструкція SQL складається з декількох частин, які називаються реченнями. Кожне речення в інструкції SQL має своє призначення. У наведеній нижче таблиці вказані речення SQL, використовуються найчастіше:

Таблиця	1	-	Опис	інстр	укцій	SQL
,				1	2 1	· ·

Речення SQL	Призначення						
SELECT	Визначає поля, які містять необхідні надалі данні						
FROM	Визначає таблиці, які містять ті поля, що вказані в SELECT						
WHERE	Визначає умову відбору полів, яким повинні відповідати всі						
	записи, що будуть знаходитись в результатах						
ORDER BY	Визначає порядок сортування результатів						
GROUP BY	Визначає поля для яких не обраховується об'єднане значення						
HAVING	Визначає умову, яка застосовується до полів в яких не						
	обраховується об'єднане значення						

Таблиця 2 - Терміни SQL

Термін SQL	Визначення					
Ідентифікатор	Ім'я, яке використовують для ідентифікації об'єкта бази даних					
Оператор	Ключове слово, яке визначає певну дію або змінює його (AS)					
Константа	Значення, яке не змінюється та приймає значення певної					
	цифри або NULL					
Вираз	Поєднування ідентифікаторів, операторів, констант та					
	функцій, що призначені для обрахування одного значення					
	(>=)					

Кожне речення SQL складається з термінів, які відповідають певним частинам мови.

Команда **SELECT** – це основний оператор мови SQL, який виконує відбір необхідної інформації з таблиць.

Структура оператора:

SELECT що\_відбирається [AS псевдонім] (через кому) FROM таблиці БД [WHERE умова\_відбору\_з\_таблиць, умова\_з'єднання\_таблиць] [GROUP BY поля\_об'єднання (через кому)] [HAVING умова\_об'єднання] [ORDER BY поля\_для\_сортування (через кому)];

Пояснення кожної частини цієї команди:

- **SELECT**- ключове слово, яке повідомляє БД, що ця команда-запит;

- що\_відбирається [AS псевдонім] - список полів (стовпчиків таблиці), які будуть відібрані. Будь-які поля не перераховані тут не будуть включені у результат команди. Це не значить, що вони будуть стерті з таблиць, тому що запит не впливає на інформацію в таблицях; він тільки показує дані на екрані. Якщо потрібно вказати всі поля всіх таблиць використовується символ «\*». Псевдонім дозволяє задати нові імена, які будуть заголовками стовпчиків замість назв полів.

- **FROM** - вказує ті таблиці, з яких відбирається інформація.

- ; - використовується у всіх інтерактивних командах SQL щоб повідомити, що команда готова до виконання.

Наприклад:

SELECT \* FROM STUDENT; SELECT fio AS [Прізвище] FROM STUDENT;

- WHERE - дозволяє вибрати тільки ті рядки з таблиць, для яких виконується умова.

Наприклад: 1) відібрати шифри груп, які відносяться до 2 курсу:

SELECT shifr\_gr AS [uudp rpynu] FROM GRUPA WHERE kurs=2;

Лапки не використовуються тут тому, що курс - це не числове поле, а в текстових використовуються лапки.

2) Відібрати інформацію про групу СП-126:

SELECT \* FROM GRUPA WHERE shifr\_gr='СП-126';

Можливі варіанти умов пошуку

а) використання знаків порівняння =, >,<,>=,<=,<>

Наприклад: необхідно побачити всіх студентів з оцінкою вище 3:

SELECT\* FROM OZENKA WHERE oz>3;

б) використання логічних операцій: AND, OR, NOT

AND (i) - оцінює, чи вірні всі умови;

**ОR** (або) - оцінює, чи вірна одна з умов;

**NOT** (ні) - заміняє значення умови на протилежне.

Наприклад: відібрати студентів, які мають номер не більше 10 та оцінку не менше 4:

SELECT\* FROM OZENKA WHERE n\_st<=10 AND oz>=4;

в) Використання спеціальних операторів: IN, BETWEEN, LIKE.

Оператор **IN** визначає набір значень, з яких вибирається будь-яке. Наприклад: вибрати номери студентів, які отримували або 2, або 3, або 4:

> SELECT\* FROM OZENKA WHERE oz IN (2,3,4);

Оператор **BETWEEN** визначає діапазон значень від початкового до кінцевого. Наприклад: вибрати номери студентів, які отримували оцінки в першій половині 2015 року:

SELECT n\_st FROM OZENKA WHERE data BETWEEN #01-01-15# AND #06-30-15#;

Оператор **LIKE** застосовується тільки для текстових полів і шукає значення поля за зразком. В зразках можна використовувати такі символи:

? - заміщає любий один символ;

\* - заміщає любу послідовність символів

Наприклад: знайдемо всіх студентів, чиї прізвища починаються з літери Л:

SELECT\* FROM STUDENT\ WHERE fio Like "Л\*";

- GROUP BY- вказує поля, за яким однакові значення будуть об'єднані в один запис.

Наприклад: підрахувати середню оцінку для кожного студента:

SELECT AVG(oz), n\_st FROM OZENKA GROUP BY n\_st;

- Команду **ORDER BY** використовують, щоб упорядкувати виведення даних. Записи можна вивести в порядку зростання (ASC) або в порядку зменшення (DESC) для кожного стовпчика.

Наприклад: порядкувати прізвища студентів в алфавітному порядку від А до Я:

# SELECT fio FROM STUDENT ORDER BY fio ASC;

Після **SELECT** можна використовувати *функції*:

COUNT (\*) – кількість рядків,

SUM (поле) - арифметична сума значень даного поля,

AVG (поле) – середнє арифметичне значень даного поля,

MIN (поле) - найменше із значень даного поля,

МАХ (поле) - найбільше із значень даного поля.

Наприклад: підрахувати найбільшу оцінку серед всіх студентів в січні 2015 року:

SELECT MAX(oz) AS [найбільша оцінка] FROM OZENKA WHERE data BETWEEN #01-01-15# AND #01-31-15#;

# 6.3. Хід виконання роботи та результати завдань

### Завдання 1.

Використовуючи базу даних, яку ми розробили в попередніх лабораторних роботах виконаємо простий вибір даних за допомогою запитів SQL, таких як select, \*, from, where, order by, обчислювальні поля, distinct.

Для того щоб створити запит з допомогою мови SQL в Access, потрібно створити звичайний запит з допомогою конструктора запитів і переключити режим в режим SQL, натиснувши кнопку SQL в нижньому правому куті, або натиснувши на пункт SQL в головному меню у вас повинно з'явитись вікно, як показано на рисунку 1.



Рисунок 1 – Вікно SQL запиту

Тепер необхідно створити створення перший запит на вибірку полів з таблиці «Анкета». Для цього виберемо поля «Фамилия», «Имя», «Отчество», і в блоці SELECT через кому перелічимо дані поля. Якщо поля мають прочерк, як в нашому випадку то краще взяти дані поля в квадратні дужки. Після перелічування полів в блоці FROM написати таблицю з якої вибиратимуться данні (рис. 2) і зберегти запис під назвою «Студенты»:



Рисунок 2 – Запит на вибірку полів з таблиці «Анкета»

В результаті ви отримаєте таблицю, яка відображатиме вибранні поля (рис. 3).

3anpoc1						
2	Фамилия 👻	Имя 👻	Отчество 👻			
	Бондаренк 🔫	Евгений	Максимович			
	Возный	Юрий	Сергеевич			
	Гушлевский	Александр	Олегович			
	Коваленко	Ирина	Валериивна			
	Кунов	Максим	Иванович			
	Лукашенко	Олег	Владимирович			
	Порублевый	Олексей	Дмитриевич			
	Сиваченко	Максим	Петрович			
	Чупрун	Эдуард	Петрович			
	Клинтон	Петро	Владимирович			
*						

Рисунок 3 – Результати запиту

Для того щоб вивести всі записи потрібно використати знак «\*» як значення SELECT і вказати в FROM з якої таблиці взяти дані значення.

Тепер створимо запит з допомогою мови SQL для того щоб вивести всіх студентів, які здали всі предмети на відмінно. Для виконання цього завдання використаємо оператор SELECT, де потрібно вказати поля «Студент» та «Дисциплина» з таблиці «Экзамен». В операторі WHERE потрібно вказати, що поле «Результат контроля» з даної таблиці повинно бути рівне «Отлично», для цього потрібно використати оператор «=» або ключове слово «Like» та збережемо запит як «Запрос по отличниках», як показано на рис. 4.



Рисунок 4 – Запит на вибірку відмінників

anpoc1							
2	Экзамен.Студен 👻	Результат контроля 🛛 👻	Экзамен.Дисциплин 👻				
	Кунов 🔫	отлично	Английский язык				
	Чупрун	отлично	Английский язык				
	Коваленко	отлично	Системное програмир				
	Сиваченко	отлично	Системное програмир				

В результаті ви отримаєте таблицю, яка відображатиме вибранні поля (рис. 5).

Рисунок 5 - Результат запиту по відмінниках

Тепер створимо запити які будуть використовувати такі операції, як «AND» та «OR», для цього грунтуючись на попередньому запиту виберемо студентів які здали на задовільно по всих предметах крім англійської мови. Для цього потрібно взяти запит який ми створили на попередньому кроці та використати оператор OR в області WHERE де використати таку умову, що «Результат контроля» рівний «Удовлитворительно», для цього потрібно використати оператор «=» або ключове слово «Like», як на рисунку 6. В результаті ви отримаєте таблицю, яка відображатиме вибранні поля (рис. 7).

<mark>म Запрос1</mark> 💷 Группы (Ш. Дисциплины (Ш. Преподаватели) (Ш. Экзамен) (Ш. Анкета) (Ш. Здано Екзам SELECT Экзамен.Студент.Value, Экзамен.[Результат контроля], Экзамен.Дисциплина.Value

SEECT Экзамен.студент.value, Экзамен.[resynbla1 контроля], Экзамен.дисциплина.value FROM Экзамен

WHERE (((Экзамен.[Результат контроля]) Like "отлично") OR ((Экзамен.[Результат контроля]) Like "удовлетворительно"))

# Рисунок 6 – Запит на вибірку студентів з оцінкою «Отлично» та «Удовлетворительно»

<mark>🗊 Запрос1</mark> 🌐 Группы 🖽 Дисциплины 🌐 Преподаватели 🖽 Экзамен 🖽 Анкета 🖽								
🔀 Экзамен.Студен 👻	Результат контроля	Ŧ	Экзамен.Дисциплина.Value	-				
Гушлевский 🔽	удовлетворительно		Английский язык					
Кунов	отлично		Английский язык					
Порублевый	удовлетворительно		Английский язык					
Чупрун	отлично		Английский язык					
Возный	удовлетворительно		Системное програмирование					
Коваленко	отлично		Системное програмирование					
Лукашенко	удовлетворительно		Системное програмирование					
Сиваченко	отлично		Системное програмирование					

Рисунок 7 – Результат запиту на вибірку студентів

Тепер здійснимо вибірку даних з двох зв'язних таблиць. Наприклад виберемо з таблиці «Анкета» певні дані про студента та з таблиці «Экзамены» виберемо дисципліни та результати здачі студентом даної дисципліни. Для цього використаємо операцію INNER JOIN – внутрішнє Внутрішнє об'єднання означає, що таблиць об'єднання. 3 пов'язаних вибираються тільки ті записи, значення зв'язаних полів яких збігаються, інші ігноруються, дану операцію потрібно виконати між таблицями «Экзамен» та «Анкета» для об'єднання полів «Студент» та «Фамилия», операція ОN показує між якими полями встановлений зв'язок. В SELECT ми вибираємо поля «Фамилия», «Имя», «Отчество», «Дата рождения» з таблиці «Анкета» та поля «Дисциплина» та «Результат контроля» з таблиці «Экзамен», як показано на рис 8. В результаті ви отримаєте таблицю, яка відображатиме вибранні поля (рис. 9).

Запросі Экзамен Запит по всім результатам
SELECT Анкета.Фамилия, Анкета.Имя, Анкета.Отчество, Анкета.[Дата рождения], Экзамен.Дисциплина.Value, Экзамен.[Результат контроля]
FROM Экзамен INNER JOIN Анкета ON Экзамен.Студент.Value = Анкета.Фамилия;

	🗊 Запросі 🌐 Экзамен 📳 Запит по всім результатам										
2	Фамилия 👻	имя 👻	Отчество 👻	Дата рождения 🕞	Экзамен.Дисциплин 👻	Результат контроля					
	Бондаренк 🔻	Евгений	Максимович	17.04.1995	Английский язык	неудовлетворительно					
	Бондаренко	Евгений	Максимович	17.04.1995	Системное програмир	хорошо					
	Возный	Юрий	Сергеевич	09.06.1995	Английский язык	хорошо					
	Возный	Юрий	Сергеевич	09.06.1995	Системное програмир	удовлетворительно					
	Гушлевский	Александр	Олегович	11.03.1995	Английский язык	удовлетворительно					
	Гушлевский	Александр	Олегович	11.03.1995	Системное програмир	неудовлетворительно					
	Коваленко	Ирина	Валериивна	02.12.1995	Английский язык	неудовлетворительно					
	Коваленко	Ирина	Валериивна	02.12.1995	Системное програмир	отлично					
	Кунов	Максим	Иванович	09.09.1995	Английский язык	отлично					
	Кунов	Максим	Иванович	09.09.1995	Системное програмир	хорошо					
	Лукашенко	Олег	Владимирович	12.11.1995	Английский язык	хорошо					
	Лукашенко	Олег	Владимирович	12.11.1995	Системное програмир	удовлетворительно					
	Порублевый	Олексей	Дмитриевич	03.04.1995	Английский язык	удовлетворительно					
	Порублевый	Олексей	Дмитриевич	03.04.1995	Системное програмир	неудовлетворительно					
	Сиваченко	Максим	Петрович	29.06.1995	Английский язык	неудовлетворительно					
	Сиваченко	Максим	Петрович	29.06.1995	Системное програмир	отлично					
	Чупрун	Эдуард	Петрович	14.04.1995	Английский язык	отлично					
	Чупрун	Эдуард	Петрович	14.04.1995	Системное програмир	хорошо					

Рисунок 8 – Вибірка з двох зв'язних таблиць

Рисунок 9 – Результат вибірки з двох зв'язних таблиць

Тепер створимо запит який використовуватиме умови відбору з допомогою параметра відбору. Наприклад створимо запит який вибирає викладачів які належать тій чи іншій кафедрі. Для початку узагальнимо поняття параметра та умови. **Параметр** - це частина відомостей, що надаються запитом при його виконанні. Параметри можна використовувати окремо або в складі довгих виразів для формування умови запиту. **Умови** - це "фільтри", що додаються в запит для вказівки елементів, які він повинен повернути. Тепер створимо запит, для цього використаємо операцію **SELECT** де в якості аргументу використаємо «\*», у секції **FROM** вкажемо з якої таблиці вибирати, в даному випадку це таблиця «**Преподаватели**» в секції **WHERE** вкажемо умову для поля «Кафедра» вона повинна бути рівна параметру, який вказується в квадратних дужках, як показано на рис. 10.



Рисунок 10 – Запит по кафедрі

Збережіть цей запит під назвою «Запрос по кафедре», коли ви натиснете на режим перегляду перед вами з'явиться вікно де ви зможете ввести назву кафедри, після цього ви отримаєте результат, який представлений на рис. 11.

💼 Запросі 🚛 Экзамен 🎼 Запит по всім результатам 🎽								
2	🕗 Фамилия 👻 Имя 👻 Отчество 👻							
	Бондаренко	Максим		Олексеевич				
*								

Рисунок 11 – Запит по кафедрі, результат виконання

Тепер створимо запит який використовує операцію **Distinct**. Предикат Distinct - виключає записи, що містять повторювані дані в обраних полях. Для включення в результати запиту значення кожного з полів, перерахованих в інструкції **SELECT**, повинні бути унікальними. Щоб побачити, як працює

даний предикат давайте додамо декілька записів в таблицю «Анкета», які мають прізвище, яке вже наявне в таблиці, приклад показаний на рисунку 12.

13 Порублевый	Максим	Владимирович	12.11.1990	Хрещатик 95	3-15-65	125
14 Бондаренко	Олег	Петрович	12.12.1212	Роумсона 33	3-15-84	125

Рисунок 12 – Додані два записи в таблицю

Створимо запит, який використовуватиме предикат **Distinct**, для цього в блоці **SELECT** поставимо **Distinct** перед полем «Фамилия» з таблиці викладачі, як показано на рисунку 13. Результат виконання запиту зображений на рисунку 14.

P	Запрос1	] Экзамен	Запі
SELE FROI	CT DISTINCT Ан И Анкета;	кета.Фамилия	

Рисунок 13 – Створення запиту на виведення прізвищ студентів без повторень



Рисунок 14 – Виведення прізвищ без повторень

### Завдання 2.

Тепер попрацюємо з зовнішніми об'єднаннями таблиць. Створимо запити. При зовнішньому об'єднанні, на відміну від внутрішнього, в результат вибірки потрапляють не тільки всі зв'язані рядки обох таблиць, а й рядки однієї з таблиць, для яких немає пов'язаних в іншій таблиці. Для зовнішнього об'єднання таблиць використовуються операції LEFT JOIN, RIGHT JOIN і FULL JOIN.

Операція **LEFT JOIN** використовується для створення ліве зовнішнє з'єднання. За допомогою лівого зовнішнього об'єднання вибираються всі записи першої (лівої) таблиці, навіть якщо вони не відповідають записам у другій (правої) таблиці.

Операція **RIGHT JOIN** використовується для створення праве зовнішнє з'єднання. За допомогою правого зовнішнього об'єднання вибираються всі записи другий (правої) таблиці, навіть якщо вони не відповідають записам у першій (лівої) таблиці.

Так, наприклад, створимо запит, який вибирає всіх викладачів з таблиці «Преподаватели», які провели екзамен та які не провели його ще. Для цього використаємо LEFT JOIN. Для того щоб побудувати даний запит потрібно використати функцію SELECT, в якій взяти поля «Фамилия», «Имя», «Отчество» з таблиці «Преподаватели» та поле «Дисциплина» з таблиці «Экзамен». В секції FROM використаємо операцію LEFT JOIN де першою буде таблиця «Преподаватели», шоб вибору стояти задати умови використовується оператор **ON**, як показано на рисунках 15-16.

 Image: Second constraints
 I

		🗊 -Запрос1 🔠 Экзамен 📑 Запит по всім результатам 📑 Запит по кафедрі_S									
	2	Фамилия 👻	имя 👻	Отчество 👻	Экзамен.Дисциплин 👻						
		Бабенко	Вера	Григорьевна							
		Бондаренко	Максим	Олексеевич							
		Ерофеев	Юрий	Федорович							
		Куницкая	Светлана	Юриивна	Системное програмир						
		Махиня	Наталья	Владимировн	Английский язык						
		Палагина	Ольга	Анатолиивна							
		Щерба	Валентина	Олексеевна							
		Яшан	Оксана	Олегивна							
1											

Рисунок 15 – Вибір викладачів, які провели та не провели екзамен

Рисунок 16 – Виконання запиту по вибору викладачів

Аналогічно до операції LEFT JOIN використовується операція RIGHT JOIN, тільки вона вибирає поля з правої таблиці. Тепер створимо запити, які використовуватимуть такі операції IS NULL та IS NOT NULL.

Для демонстрації операції **IS NULL** давайте створимо таблицю «Здано Экзаменов» і додамо туди поля, такі як «Студент» та «Всего здано» - де буде вказуватись кількість зданих екзаменів, та заповнити її довільним чином, приклад на рис. 17.

📑 3anpoc1 🔠	🗊 Запрос1 🔠 Экзамен 🎦 Запит по всім результа 📳 Запит по кафедрі_SQL 🗐									
🕗 Номер студ 👻	Студент 👻	Всього здан 👻	Щелкните для добавления 🕞							
1	Бондаренко	1								
2	Возний									
3	Гушлевский	3								
4	Коваленко									
5	Кунов	2								
6	Лукашенко									
7	Порубльовий	2								
<i>.</i> /	Сиваченко 💌	2								
9	Чупрун	1								
* (Nº)										

Рисунок 17 – Таблиця «Здано Экзаменов»

Тепер, коли таблиця заповнена створимо запит (рис. 18.), який буде вибирати записи, поле «Всего здано» яке рівне NULL. Для створення запиту використаемо функцію SELECT, додамо поля з таблиці «Здано Экзаменов». В області WHERE вкажемо, що поле «Всего здано» не повинно бути NULL. Результат роботи даного запиту показаний на рисунку 19.



Рисунок 19 – Результат запиту 96

Тепер використовуючи дану таблицю, змінимо запит, як показано на рисунку 20, тобто для того щоб він вибирав всі значення де поле «Всего здано» рівне NULL. Результат роботи запиту, який використовує операцію IS NULL, показаний на рисунку 21.



Рисунок 20 - Приклад запиту IS NULL

🔠 Здано Екзаменів 🔂 🗐 Запрос1						
2	Студент 👻	Всього здан 👻				
	Возний 💌					
	Коваленко					
	Лукашенко					
	Чупрун					
*						

Рисунок 21 – Результат запиту

Тепер створимо запит який використовує вираз Like для пошуку записів.

Оператор Like можна використовувати для пошуку значень полів, відповідних вказаним шаблоном. Як шаблон, можна вказати повне значення (наприклад, Like "Smith") або використовувати символи узагальнення, щоб отримати діапазон значень

Створимо запит, який буде вибирати з таблиці «Преподаватели» всіх викладачів які мають ім'я по-батькові та яке починається на «Ол». Для цього використовуючи оператор SELECT виберемо таблиці поля 3 області оператора «Преподаватели», В WHERE зазначимо, ЩО поле «Отчество» повинно починатись з «Ол», як показано на рисунку 22.

🗇 Запрос1 🗐 Экзамен 🗇 Запит по всім результатам 🏚 Запит по кафедрі SQL 🥅 Анкета SELECT DISTINCT Преподаватели.Фамилия, Преподаватели.Имя, Преподаватели.Отчество, Преподаватели.Должность, Преподаватели.Кафедра FROM Преподаватели WHERE (Преподаватели.Отчество Like "Ол\*");

Рисунок 22 – Використання оператора Like

В результаті виконання даного запиту ми отримаємо список викладачів які мають імя по-батькові, яке починається на «Ол», як показано на рисунку 23.

	🗗 Запрос1	🔲 Экзамен [	Запит по всім р	езультатам 📑 Запі	ит по кафедрі_SQL 🔲 Анкета 📑 З	апи
2	Фамилия 👻	Имя 👻	Отчество 👻	Должность 👻	Кафедра 🗸	r -
	Бондаренко	Максим	Олексеевич	Преподователь	Физики	
	Щерба	Валентина	Олексеевна	Преподователь	Прикладной математики	
	Яшан	Оксана	Олегивна	Преподаватель	Истории Украины	

Рисунок 23 – Результат виконання запиту

Далі створимо запит, який буде використовувати оператор UNION. За допомогою однієї операції UNION можна виконати злиття двох або більше запитів, таблиць та інструкцій SELECT в будь-якій послідовності.

Створимо запит, який буде містити в собі декілька запитів, які будуть отримувати всі записи з таблиці «Дисциплины» та «Группы». Для цього нам потрібно створити запит з допомогою SELECT який буде діставати всі записи з таблиці «Дисциплины» та з допомогою SELECT створити запит який буде діставати всі записи з таблиці «Группы» і об'єднаємо ці запити операцією UNION, як показано на рисунках 24-25.



Рисунок 24 – Використання UNION



Рисунок 25 – Результат виконання запиту

## Завдання 3.

Виконаємо завдання 3, яке полягає в створенні результуючого запиту з декількома результуючими цифрами. Для виконання цього та наступних пунктів додамо в таблицю «Экзамен» колонку, яка матиме дані про те, яку оцінку по 100 бальній шкалі отримав студент як показано на рисунку 26.

	Экзамен 📑 Запит по всім результа	атам 📑 Запит	по кафедрі_SQL 🔳	Анкета 📑 Запрос по дисциплин	е 🗊 Запрос по отличникам	Запит по д
2	Дисциплина 🗸	Группа 👻	Студент 👻	Результат контроля 🛛 👻	Преподаватель 👻	Оценка 👻
	Английский язык	KM-125	Бондаренко	неудовлетворительно	Махиня	65
	Английский язык	KM-125	Возный	хорошо	Махиня	57
	Английский язык 💌	KM-125	Гушлевский	удовлетворительно	Махиня	49
	Английский язык	KM-125	Коваленко	неудовлетворительно	Махиня	85
	Английский язык	KM-125	Кунов	отлично	Махиня	80
	Английский язык	KM-125	Лукашенко	хорошо	Махиня	76
	Английский язык	KM-125	Порублевый	удовлетворительно	Махиня	72
	Английский язык	KM-125	Сиваченко	неудовлетворительно	Махиня	74
	Английский язык	KM-125	Чупрун	отлично	Махиня	55
	Системное програмирование	KM-125	Бондаренко	хорошо	Куницкая	90
	Системное програмирование	KM-125	Возный	удовлетворительно	Куницкая	74
	Системное програмирование	KM-125	Гушлевский	неудовлетворительно	Куницкая	54
	Системное програмирование	KM-125	Коваленко	отлично	Куницкая	55
	Системное програмирование	KM-125	Кунов	хорошо	Куницкая	80
	Системное програмирование	KM-125	Лукашенко	удовлетворительно	Куницкая	80
	Системное програмирование	KM-125	Порублевый	неудовлетворительно	Куницкая	85
	Системное програмирование	KM-125	Сиваченко	отлично	Куницкая	93
	Системное програмирование	KM-125	Чупрун	хорошо	Куницкая	74

Рисунок 26 - Таблиця «Экзамен»

Щоб створити результуючий запит потрібно використати операцію SELECT, де додамо декілька полів з таблиці «Экзамен». Поле «Дисциплина» та результуючі поля називатимуться «Средняя оценка по дисциплине» і «Максимальная оценка по дисциплине». Для того, щоб створити поля дисциплине» «Средняя оценка ПО та «Максимальная оценка ПО дисциплине» потрібно згрупувати дані по полю «Дисциплина» - це можна зробити з допомогою оператора GROUP BY, для того щоб вирахувати середню оцінку по дисципліні потрібно використати функцію Avg для того щоб дізнатись максимальне значення по групі потрібно використати відповідну функцію під назвою Мах. Результуючий запит показаний на рисунках 27-28.



Рисунок 27 – Результуючий запит

		Use UNION <b>3anpoc1</b>			
	2	Экзамен.Дисциплина.Value	*	"Средняя оценка по дисципли 👻	"Максималі 👻
		Английский язык	•	68,111111111111	85
		Системное програмирование		76,111111111111	93
1					

Рисунок 28 – Результат роботи результуючого запиту

Тепер виконаємо наступне завдання, яке передбачає створення простого групового запиту. Давайте створимо запит який буде виводити кількість викладачів, які є на певній кафедрі. Для цього потрібно використати оператор **SELECT**, де вкажемо поля з таблиці «**Преподаватели**» та використовуючи функцію групування **COUNT** підрахуємо кількість викладачів. Щоб виконати груповий запит потрібно використати оператор **GROUP BY** по полю «**Кафедра**», як показано на рис. 29. Результат виконання групового запиту представлений на рис. 30.

Use UNION 3anpoc1								
SELECT Преподаватели.Кафедра, Count(Преподаватели.Фамилия) AS [Количество преподавателей]								
FROM Преподаватели								
GROUP ВУ Преподаватели.Кафедра:								

	Use UNION 📑 3anpoc1			
2	Кафедра	Ŧ	Количество 👻	
	Иностранных языков	•	1	
	Истории Украины		1	
	Компьютерных систем		1	
	Прикладной математики		2	
	Системного програмирования		2	
	Физики		1	

Рисунок 29 – Груповий запит

Рисунок 30 – Результат групового запиту

Тепер виконаємо завдання, яке полягає в тому щоб створити груповий запит з групуванням по декількох полях. Давайте створимо запит, який буде групувати дані по дисципліні та по оцінках. Для цього використаємо оператор SELECT, де виберемо поля з таблиці «Анкета» та «Экзамен», використовуючи LEFT JOIN об'єднаємо дані таблиці, де прізвище рівне полю студент з таблиці «Экзамен», після цього об'єднаємо запит по полю «Дисциплина» та «Оценка» з таблиці «Экзамен», як показано на рисунку 31. Результат даного запиту представлений на рисунку 32.

🗊 Use UNION 📳 Запросі 🚰 Запит\_по\_двох\_полях

SELECT Экзамен.Дисциплина.Value AS Выражение1, Экзамен.Оценка, Count(Экзамен.Студент.Value) AS ["количество студентов"] FROM Экзамен LEFT JOIN Анкета ON Экзамен.Студент.Value = Анкета.Фамилия

GROUP BY Экзамен.Оценка, Экзамен.Дисциплина.Value;

## Рисунок 31 – Груповий запит по декількох полях

	Use UNION 📑 Запрос1 📑 Запит	_по_двох_полях	
$\square$	Выражение1 👻	Оценка 👻	"количество студентов" 👻
	Английский язык	49	1
	Системное програмирование	54	1
	Английский язык	55	1
	Системное програмирование	55	1
	Английский язык	57	1
	Английский язык	65	1
	Английский язык	72	1
	Английский язык	74	1
	Системное програмирование	74	2
	Английский язык	76	1
	Английский язык	80	1
	Системное програмирование	80	2
	Английский язык	85	1
	Системное програмирование	85	1
	Системное програмирование	90	1
	Системное програмирование	93	1

Рисунок 32 – Результат запиту

Тепер виконаємо завдання, яке полягає в тому, що потрібно створити запит в якому визначаються умови, причому спочатку виконуються обчислення, а потім відбувається відбір. Давайте використаємо попередній запит, але для того щоб виконати відбір використаємо операцію **HAVING** і встановимо що кількість студентів більше 1 (рис. 33).

Use UNION - Запит з постумовою - Запит_по_двох_полях
SELECT Экзамен.Дисциплина.Value AS Выражение1, Экзамен.Оценка, Count(Экзамен.Студент.Value) AS ["количество студентов"] FROM Экзамен LEFT JOIN Анкета ON Экзамен.Студент.Value = Анкета.Фамилия GROUP BY Экзамен.Оценка, Экзамен.Дисциплина.Value HAVING count(Экзамен.Студент.Value)>1; 

L.	Use UNION 📑 Запит з постумов	ою	Запит_по_двох_полях			
Z	Выражение1	*	Оценка	Ŧ	"количестві 👻	
	Системное програмирование	•		74	2	
	Системное програмирование			80	2	

Результат роботи запиту представлений на рисунку 34.

Рисунок 34 – Результат виконання запиту

Виконаємо наступне завдання, яке полягає в тому, що потрібно створити запит в якому визначаються умови, причому спочатку відбувається відбір, а потім виконуються обчислення. Створимо запит, який буде згрупований по дисципліні, який буде обчислювати кількість студентів які здали екзамен і які мають оцінку не більше 70 балів. Для цього використаємо оператор **SELECT**, де виберемо поля з таблиці «Экзамен» і перед тим як згрупувати запит задамо посту мови, що оцінка не повинна бути більше 70. Даний запит показаний на рисунку 35. Результат запиту зображений на рисунку 36.



Рисунок 35 – Запит з передумовою

	🗐 Use UNION 📑 Запит з постумовою 📑 Запит_по_двох_полях							
2	Экзамен.Дисциплина.Value	*	Количество 👻					
	Английский язык	•	4					
	Системное програмирование		2					

Рисунок 36 – Результат запиту

## Завдання 4.

Далі виконаємо завдання 4, яке полягає в створенні запиту, що використовує оператор In. Оператор In повертає всі записи які відповідають певному набору значень. Створимо наприклад запит, який буде вибирати викладачів з таблиці «Преподаватели», які працюють на кафедрі «Прикладной математики» та «Физики», приклад запиту показаний на рисунку 37. Результат роботи запиту представлений на рисунку 38.



Рисунок 37 – Використання IN

•	Запит з по	стумовою 📑 З	апит_по_двох_по	лях 🖽 Дисциплины 🖽	Преподават
$\mathbb{Z}$	Фамилия 👻	Имя 👻	Отчество 👻	Кафедра	*
	Щерба	Валентина	Олексеевна	Прикладной математики	
	Палагина	Ольга	Анатолиивна	Прикладной математики	
	Бондаренко	Максим	Олексеевич	Физики	
*					

Рисунок 38 – Результат виконання запиту

Виконаємо наступне завдання, яке полягає в використанні операторів All, Any, Exists. Створимо запит який використовує оператор All. Оператор All використовується для задіяні одного порівняння для декількох значень, які вибираються з інших таблиць. Наприклад, виберемо студентів з таблиці «Анкета», які не отримали оцінку відмінно по екзаменах, за допомогою предиката All, створимо запит як показаний на рисунку 39.

▲ 3апит_по_двох_полях ДШ Дисциплины Ш Преподаватели Запрос1 ДР Апу,АШ SELECT * FROM Анкета WHERE Анкета.Фамилия <> All (SELECT экзамен.Студент.Value FROM Экзамен WHERE Экзамен.[Результат контроля] = "Отлично[];									
🗊 Use UNION 📳 Запит з постумовою 📳 Запит_по_двох_полях 🔠 Дисциплины 🏢 Преподаватели 📳 Запросі 📳 Апу,АІІ									
$\square$	Код 👻	Фамилия 👻	имя 👻	Отчество 🔹	Дата рождения 🕞	Адрес 👻	Телефон 🝷	Группа 👻	
	1	Бондаренко	Евгений	Максимович	17.04.1995	Чехова, 23	3-47-53	125	
	2	Возный	Юрий	Сергеевич	09.06.1995	Смелянская, 55	3-77-22	125	
	3	Гушлевский	Александр	Олегович	11.03.1995	Добролюбова, 11	3-65-11	125	
	6	Лукашенко	Олег	Владимирович	12.11.1995	Кармелюка, 38	3-27-74	125	
	7	Порублевый	Олексей	Дмитриевич	03.04.1995	Розы Люксембург, 48	3-15-77	125	
	12	Клинтон	Петро	Владимирович	11.05.1995	Хмельницкого, 73	3-79-31	125	
	13	Порублевый	Максим	Владимирович	12.11.1990	Хрещатик 95	3-15-65	125	
	14	Бондаренко	Олег	Петрович	12.12.1212	Роумсона 33	3-15-84	125	
*	(Nº)							125	

Рисунок 39 – Використання предиката All

Тепер створимо запит, який використовує предикат Any. Даний предикат використовується для того щоб примінити одне й те саме порівняння для всіх значень, які обєднанні операцією OR. Давайте створимо запит, який буде демонструвати роботу предиката **Any**, наприклад давайте виберемо всіх студентів оцінки яких є менше 65, або більше 90. Використовуючи оператор **SELECT** створимо запит, який вибиратиме записи відповідно до умов, приклад запиту та результати його роботи представлено на рисунку 40.

📑 Use UNION 📑 Запит з постумовою 📑 Запит_по_двох_полях 🖽 Дисцип										
SELECT Анкета.Фамилия, Анкета.Имя, Анкета.[Отчество], Анкета.[Дата рождения] FROM Анкета WHERE Анкета.Фамилия = ANY (SELECT Экзамен.Студент.Value FROM Экзамен WHERE (Экзамен.Оценка >90 OR Экзамен.Оценка <65 ));										
📰 Use UNION 🗊 Запит з постумовою 📳 Запит_по_двох_полях 🖽 Дисци										
$\square$	Фамилия 🔻	Имя 👻	Отчество 👻	Дата рождения 🕞						
	Возный 💌	Юрий	Сергеевич	09.06.1995						
	Гушлевский	Александр	Олегович	11.03.1995						
	Коваленко	Ирина	Валериивна	02.12.1995						
	Сиваченко	Максим	Петрович	29.06.1995						
	Чупрун	Эдуард	Петрович	14.04.1995						
*										

Рисунок 40 – Використання Апу

Тепер застосуємо предикат Exists, який використовується в комбінації з під запитом, що здійснює пошук по таблиці і повертає кілька результатів. Після цього предикат здійснює пошук по таблиці і шукає чи існують записи, які повернув під запит. Даний предикат може використовуватись з різними операторами, такими як SELECT, INSERT, UPDATE, або DELETE. Для створення даного запиту давайте використаємо оператор SELECT, який виводить інформацію про екзамен того студента, який ще не здав ні одного екзамену. Для цього запиту використаємо оператор NOT EXISTS, як показано на рисунку 41. Результат запиту показано на рисунку 42.



Рисунок 41 – Запит який використовує EXISTS

	Use UNION		Запит з посту	мо	вою 📑 За	пит	_по_двох_полях	] Дисциплины 🔳 Пре	еподаватели 📑 Запросі (	J.	Any,All	ny_запит	Б	cist
$\square$	Код	•	Фамилия	-	Имя	Ŧ	Отчество 👻	Дата рождения 🕞	Адрес 🚽	•	Телефон 👻	Группа	*	
		12	Клинтон		Петро		Владимирович	11.05.1995	Хмельницкого, 73	:	3-79-31		125	
*	(	(Nº)											125	
Рисунок 42 – Результат запиту														

# 7. ЛАБОРАТОРНА РОБОТА №6

Тема роботи: «Організація зв'язку з базою даних типу Microsoft Access засобами ADO»

**Мета роботи:** Вивчити механізм доступу до таблиць Microsoft Access за технологією ADO. Створити додаток для внесення даних.

# 7.1. Завдання

Завдання 1: Створити папку для бази даних.

Завдання 2: Скопіювати базу даних з попередньої роботи в створену папку.

Завдання 3: Відкрити проект програми в середовищі Delphi, організувати відображення значень таблиці.

- добавити модуль даних: меню File/New/DataModule;

- добавити в модуль об'єкти: ADOConnection1, ADOTable1, DataSource1;

- налаштувати об'єкт ADOConnection1 на на зв'язок з БД Access:

ADOConnection1. ConnectionString

Завдання 4: Програму зберегти в папці з БД.

Завдання 5: Налаштувати властивості об'єктів для зв'язку з ADOConnection1:

- ADOTable1.Connection:= ADOConnection1;

- DataSource1.DateSet:= ADOTable1;

- ADOTable1.TableName:= Анкета;

- ADOTable1.Active:=True.

Завдання 6: Підключити модуль до форми 2 меню File/Use Unit...

Завдання 7: На форму винести об'єкт DBGrid1, налаштувати його властивості:

- DBGrid1.DataSourse:= DataModule2.DataSource1.

Завдання 8: Створити інтерфейс користувача для роботи з БД на власний розсуд.

# 7.2. Теоретичні відомості

Для роботи з базами даних минаючи BDE в Microsoft була розроблена технологія ActiveX Data Objects (ADO). ADO - це користувальницький інтерфейс до будь-яких типів даних, включаючи реляційні і не реляційні бази даних, електронну пошту, системні, текстові та графічні файли. Зв'язок з даними здійснюється за допомогою так званої технології OLE DB.

Використання ADO забезпечує більш ефективну роботу з даними. Для peaniзaції цієї можливості повинна бути встановлена система ADO 2.1 або більш старша версія. Крім того повинна бути встановлена клієнтська система доступу до даних, наприклад, Microsoft SQL Server, а в ODBC повинен мати драйвер OLE DB для того типу баз даних, з яким ви працюєте.

Треба сказати, що можливості ADO в Delphi поки в деяких відносинах нижче, ніж можливості BDE.

У Delphi для використання технології ADO розроблено ряд компонентів, які знаходяться на вкладці ADO.

Розглянемо особливості компонентів наборів даних ADO на прикладі ADOTable. Цей компонент може використовуватися в додатках замість компонента Table, що виконує аналогічні функції. Він вступає в контакт з вказаною таблицею бази даних. База даних задається властивостями Connections-string aбо Connection. Для управління таблицею в додаток вводиться, крім компонента ADOTable, звичайний компонент джерела даних DataSource, у властивості DataSet якого задається ім'я компонента ADOTable. Далі до цього джерела даних DataSource підключаються будь компоненти відображення даних.

Ім'я таблиці задається властивістю TableName.

З'єднання з базою даних здійснюється так само, як і в компонентах BDE, методом Open або установкою в true властивості Active. Але при цьому, якщо зв'язок з базою даних здійснюється через компонент ADOConnection, треба враховувати взаємозв'язок властивості Active компонента ADOTable і властивості Connected компонента ADOConnection.

Основні способи роботи з ADOTable не відрізняються від способів для Table. Точно так само подвійне клацання на компоненті викликає Редактор Полів, в якому можна задати властивості окремих полів, нічим не відрізняються від полів компонентів BDE. Але в компонентах ADO неможливо працювати зі словниками. Але не всі драйвери ADO можуть працювати з будь-якими типами полів. Наприклад, драйвер Paradox ADO не працює з полями зображень. З редактора полів так само, як в компонентах BDE, можна перетягувати поля мишею на форму. При цьому на формі автоматично будуть створюватися відповідні компоненти відображення даних.

Зв'язок один з одним компонентів ADOTable, які працюють з різними таблицями, одна з яких головна, а інша - допоміжна, здійснюється так само, як в компонентах Table за допомогою властивостей MasterSource i MasterFields.

Упорядкування відображаються записів провадиться установкою властивості IndexFieldNames.

Фільтрація відображуваних даних може здійснюватися так само, як в компонентах BDE, за допомогою властивості Filter, в якому записуються умови відбору. Відмінність від компонентів BDE полягає в тому, що в компонентах ADO в рядку Filter імена полів обов'язково повинні відділятися пробілами від операцій відношення. Також пробілами повинні оточувати логічні операції and і ог. Властивість Filter працює, якщо властивість Filtered = true.

Можна використовувати для фільтрації обробник подій OnFilterRecord. Ці події наступають кожного разу при зміні активної записи, якщо властивість Filtered = true. В обробнику цих подій ви можете аналізувати будь-які поля запису і повертати значення параметра Accept = true, якщо поля записи задовольняють сформульованим вами умов, і Accept = false в іншому випадку.

При використанні обробника подій OnFilterRecord треба стежити за тим, щоб рядок властивості Filter не перечила умовам, сформульованим у обробнику OnFilterRecord. Не можна також змінювати індексацію (властивість IndexFieldNames), поки Filtered = true, так як це може привести до зациклення програми. Перед зміною індексації треба встановити Filtered в false, потім змінити IndexFieldNames, а потім встановити Filtered в true.

Додаткові можливості фільтрації забезпечує властивість тільки часу виконання FilterGroup.

Методи, використовувані при програмуванні роботи з базою даних, в ADOTable в основному ті ж, що в Table. Навігація по таблиці здійснюється методами First, Next, Last і Prior. При редагуванні даних використовуються також методи, розглянуті раніше для Table: Insert, Edit, Post та інші. З методів пошуку в ADO реалізовані тільки методи Locate і Lookup.

Розглянемо компонент ADOQuery, який є аналогом компонента Query, використовуваного при роботі з BDE. Цей компонент використовується для виконання довільних запитів SQL. Його основна властивість SQL, що містить запит і методи виконання цього запиту, нічим не відрізняються від компонента Query. А з'єднання з базою даних, властивості і методи фільтрації та пошуку аналогічні розглянутим вище для компонента ADOTable.

Відмінність від компонента Query полягає в методиці роботи з параметрами при динамічних запитах. Якщо в запиті SQL вказані параметри, то в компоненті Query об'єкти, відповідні цим параметрам, розташовані у властивості Params типу TParams. Це масив параметрів, причому доступ до значень окремих параметрів під час виконання може здійснюватися або за індексом, або по імені за допомогою методу ParamByName.

Компонент TADOCommand інкапсулює командний об'єкт ADO. Він призначений для виконання будь-яких команд, але найбільш часто використовується для команд, які не повертають дані, наприклад, для створення наборів Інші та реорганізації даних. команди зазвичай виконуються компонентами TADODataSet, TADOQuery, TADOStoredProc, які більш зручні.
# 7.3. Хід виконання роботи

# Завдання 1.

Створимо папку для даної практичної роботи, де будуть зберігатись всі файли.

🤁 Л.Р.№З	27.05.2015 14:48	Папка с файлами
🔁 Л.Р.№4	11.06.2015 13:38	Папка с файлами
🤤 Л.Р.№5	05.06.2015 12:39	Папка с файлами
🔚 Лаби	02.06.2015 10:14	Папка с файлами
🔚 Лекції	12.06.2015 11:14	Папка с файлами
📴 П.р.№1	14.06.2015 20:35	Папка с файлами
<mark></mark> Π.p.№2	09.06.2015 21:09	Папка с файлами
📴 П.р.№3	12.06.2015 13:01	Папка с файлами
<mark></mark>	14.06.2015 21:47	Папка с файлами
📴 Перевірено	12.06.2015 14:07	Папка с файлами

Рисунок 1 – Створення папки

# Завдання 2.

Копіюємо в цю папку базу даних, яка використовувала в попередніх лабораторних роботах.

_								
Локальный диск (D:)	<ul> <li>2014-2015 II семестр</li> </ul>	▶ ОБД ▶ П.р.№4						
в библиотеку ▼ Общий доступ ▼ Новая папка								
Имя	Дата изменения	Тип	Размер					
🕙 Мороз_СП-136_БАЗА ДАННЫХ - копия	14.06.2015 21:40	Microsoft Access Database	388 KE					
🚇 Мороз_СП-136_БАЗА ДАННЫХ	14.06.2015 21:39	Microsoft Access База данных	948 KE					
🎒 Мороз_СП-136_БАЗА ДАННЫХ	14.06.2015 21:39	Microsoft Access Record-Locking Inf	1 KB					
🐴 Мороз_СП-136_БАЗА ДАННЫХ	14.06.2015 21:40	Microsoft Access Database	388 KE					
∰ Π.P.№4	14.06.2015 21:50	Документ Microsoft Word	299 KG					

Рисунок 2 – Копіювання бази даних в папку

# Завдання 3.

Відкриємо новий проект та налаштуємо відображеня нашої бази даних.

	þ De	lphi 7 - Project1			to AT \$5 Pasts spectrum have been been as
	File	Edit Search View Proje	ct F	Run Component	Database Tools Window Help
		New 🕨	P	Application	tional   Win32   Sustem   Data Access   Data Controls   dbExpress   DataSnap   BDE   AL
Ì	3	Open	F	CLX Application	
Ľ	- 🗁	Open Project Ctrl+F11		Data Module	
C	)ł	Reopen	<b>—</b>	Form	Абзац із
1		Save Ctrl+S		Frame	
	1	Save As	j,	Unit	
	1	Save Project As	<b>*</b>	Other	
	9	Save All Shift+Ctrl+S	÷.		
	B.,	Close		]	
	<b>1</b>	Close All			
		Use Unit Alt+F11			
	4	Print			
	i.	Exit			
			1		

Рисунок 3 – Добавлення нового модуля даних

Добавимо модуль даних, використовуючи меню «File» та вибравши в ньому «/New/DataModule».

Добавимо в щойно створений модуль наступні об'єкти:

ADOConnection1, ADOTable1, DataSource1.



Рисунок 4 – Добавлення об'єктів в модуль

Налаштуємо об'єкт ADOConnection1 на зв'язок з нашою базою даних.

🅻 Delphi 7 - Project1	Manager Theorem Andrews Manager and Andrews
File         Edit         Search         View         Proj <sup>1</sup>	ect Run Component Database Tools Window Help   (None> 💌 😤 🦛 Standard   Additional   Win32  Sustem Data Access   dbExaress   DataSnao   BDE   ADD   InterBase   WebServices   InternetExaress   InternetExaress
	Шрифт 🕼 Абзац 🕼 Стили
Object TreeView	
¥axa + +	
itaModule2	Vajables/Constants unit Unit2;
2 S (ADDTable1)	
DataSource1	interface
MSDASQL (ADOConnection1)	
	uses DataModule2 ADOConnectionString
	SysUtils, Classes; Datamedia(2), Botamedia Connections and
	Source of Connection
	type C Use Data Link File
< · · · · · · · · · · · · · · · · · · ·	TDataModule2 = class(TDataModule
	ADOConnection: TADOConnection
Object Inspector	Abolabiel: Tabolabie; © Use Connection String
ADDConnection1 TADDConnect	Provider=Microsoft Jet OLEDB.4.0;Data Source=Mopos CTI-136 EA Build
	/ Anizate declarations )
Properties Events	
	(Public declarations)
CommandTime 30	end;
Connected False	
ConnectionStrip	var
Connection 1 Im 15	DataModule2: TDataModule2;
Cursort ocation of IseClient	
DefaultDatabas	implementation
IsolationLevel ilCursorStability	
KeepConnectic True	1. 1 Modified Incent Code/Diagram/
LoginPrompt True	
Mode cmUnknown	
Name ADOConnection1	

Рисунок 5 – Налаштування зв'язку з базою даних

# Завдання 4.

Збережемо програму в папці з базою даних.



Рисунок 6 – Збереження програми в папці з базою даних

# Завлання 5.

Налаштуємо властивості об'єктів для зв'язку з ADOConnection1 наступним чином:



# Рисунок 7 – Надання властивостей об'єкту ADOTable1



Рисунок 8 – Надання властивостей об'єкту DataSource1



Рисунок 9 – Надання властивостей об'єкту ADOTable1

# Завдання 6.

Підключимо модуль Unit1 до форми.



Рисунок 10 – Підключений модуль Unit1

# Завдання 7.

Винесемо на форму об'єкт DBGrid1 та налаштуємо його властивості наступним чином:



Рисунок 11 – Налаштування об'єкту DBGrid1

# Завдання 8.

Створимо інтерфейс користувача на власний розсуд.

Код	Прізвище	Ім'я	По-батькові	Дата народження	Адреса	
Þ	1 Варварецький	Аңдрій	Володимирович	11.07.1995	Черкаси	_
	2 Водолад	Ілля	Богданович	15.03.1995	Черкаси	
	3 Жашкевич	Олег	Вячеславович	21.01.1995	Черкаси	
	4 Коваленко	Вадим	Юрійович	16.10.1995	Черкаси	
	5 Mopos	Богдан	Олександрович	04.07.1992	Кобринова Гребля	
	6 Назаренко	Семен	Миколайович	06.08.1996	Черкаси	
	7 Слабенко	Олександр	Сергійович	07.04.1996	Черкаси	
	8 Таченко	Дмитро	Олександрович	23.01.1995	Черкаси	
	9 Траченко	Володимир	Миколайович	24.07.1996	Черкаси	
	10 Ходаковський	Олексій	Васильович	08.07.1996	Черкаси	
	11 Хоменко	Віталій	Олександрович	06.02.1996	Черкаси	
•						
					<ul> <li>Виконати</li> </ul>	

Рисунок 12 – Створення інтерфейсу користувача

# ПРАКТИЧНА ЧАСТИНА

# Лабораторна робота №1

**Тема роботи:** «Створення таблиць. Використання форм для введення і редагування даних».

Мета роботи: Вивчення процесу створення бази даних та об'єктів типу "Таблиця" за допомогою режиму *Конструктор*, методів задання властивостей полів таблиці і маніпуляцій з ними, заповнення таблиці даними і її редагування. Освоїти способи задання форм, кнопки переміщення по записах, перехід з форми в режим таблиці і роботу зі списками.

#### Завдання роботи

 Створити нову базу даних і назвати її «БАЗА ДАННЫХ», де БАЗА ДАННЫХ – це назва бази згідно індивідуального завдання (наприклад, СТРАХОВА КОМПАНІЯ).

2. Створити структури таблиць, для цього треба заповнити стовпці «Имя поля» і «Тип данных».

3. Визначити властивості полів на вкладці «Общие».

Відкрити таблиці «Вид/Режим таблицы». Заповнити рядки.
 Створити список записів.

5. Створити форми для заповнення кожної з таблиць згідно варіанту.

6. Розібратися з кнопками переміщення по записах:

- Перехід на один запис вперед;
- Перехід на останній запис;
- Перехід на один запис назад;
- Перехід на чисту сторінку, де можна ввести новий запис;
- Перехід на перший запис.

7. Створити самозаповнювальний список записів, щоб при введенні нового запису він автоматично потрапляв до списку, і згодом його можна було б вибрати зі списку, а не набирати заново. Для цього необхідно створити два макроса з імя'м «Добавление» та «Закрытие». Перевірте роботу створених макросів по створенню самозаповнювального списку.

# Лабораторна робота №2

**Тема роботи:** «Проектування багато табличних баз даних. Структуровані типи даних».

Мета роботи: Вивчити роботу режимів *Майстер таблиць*, *Майстер форм* і модифікацію структури таблиці. Закріпити навички роботи з таблицями і формами. Освоїти елементи керування типів полів «поле со списком» та «флажок», навчитися встановлювати зв'язок між таблицями. Навчитися представляти структуровані данні у різних їх формах.

#### Завдання роботи

1. Обрати з будь-якої таблиці поле для подальшої роботи, тобто для одного будь-якого поля певної таблиці змінити елемент керування.

2. Створити передбачувану умову для певного поля, де можливо реалізувати налаштування відмітки за допомогою поля типу «флажок». Переглянути форму таблиці в режимі *Форми*. Користуючись новим елементом керування «флажок» проставити результати.

3. Встановити зв'язок між всіма таблицями. Перевірити правильність роботи зв'язку у формі цих таблиць.

4. Представити табличну форму структури даних.

5. Побудувати графову форму структури даних, яка дозволяє моделювати сутності записами із відповідним схематичним позначенням сутностей та зв'язків між ними.

6. Зобразити графічну форму структури даних у вигляді діаграми, що дозволяє деталізувати елементи даних згідно схематичного позначення сутностей, атрибутів та відношень між різними сутностями з різними типами зв'язків.

# Лабораторна робота №3

**Тема:** «Підлеглі форми. Форматування форм. Представлення даних за допомогою моделі «Сутність-зв'язок».

Мета: Освоїти механізм створення підлеглих форм з метою перегляду даних, що мають відношення «один-до-багатьох». Навчитися налагоджувати зовнішній вигляд форм. Навчитися відображати елементи моделі, кардинальність та степені зв'язку за допомогою ER-моделі.

### Завдання роботи

1. Створити підлеглу форму за результатами роботи елемента керування «флажок» з вибіркою по кожному запису окремо.

2. Змінити стиль форм всіх таблиць.

3. Змінити колір фону і колір шрифту для кожного поля за своїм розсудом. Вставити на місце поля «Код» малюнок.

4. Спробуйте змінити розмір поля для одного поля по висоті та змініть шрифт (розмір, вид, колір) для іншого поля.

5. Видаліть в підлеглій формі однієї з таблиць одне будь-яке поле. Вийдіть з режиму *Конструктор* і подивіться, як тепер виглядає форма.

6. На основі заповненої табличної форми структури даних згідно лабораторної роботи №2 завдання 4 створити ER-модель:

- визначити всі набори сутностей та їх атрибути використовуючи загальні позначення для зображення моделі «Сутність-зв'язок» у вигляді, наприклад: СПІВРОБІТНИК (Табельний\_номер, Ім'я, Вік);

- визначити ключ сутності - це один чи більше атрибутів, які унікально визначають дану сутність.

7. Графічно зобразити зв'язок (relationship), тобто асоціацію, яка існує між кількома сутностями:

- зобразити степені для бінарних зв'язків;

- визначити клас приналежності сутностей та кардинальність зв'язку;

- визначити та графічно зобразити відношення залежної і незалежної сутності і позначити зв'язок між ними.

# Лабораторна робота №4

**Тема роботи:** «Створення реляційної моделі даних. Побудова запитів і звітів. Створення кнопок управління та кнопкової форми».

Мета роботи: Навчитися відтворювати основні компоненти реляційного відношення в моделі даних. Освоїти механізм створення запитів на вибірку, механізм створення звітів, механізми створення кнопок управління і кнопкових форм.

#### Завдання роботи

1. На основі заповненої табличної форми структури даних згідно лабораторної роботи №2 завдання №4 представити відношення у вигляді таблиці, наприклад:



Рисунок 1 - Основні компоненти реляційного відношення

2. Описати заголовок відношення та всі існуючі кортежі по цьому відношенні у вигляді:

СПІВРОБІТНИКИ (Табельний\_номер, Ім'я, Посада, Зарплата, Премія):

(2931, Іванов, Інженер, 112, 40)

(2935, Петров, вед. інженер, 144, 50)

(2936, Сідоров, бухгалтер, 92, 35).

3. Показати базу даних, відносно до реляційної моделі у вигляді:



Рисунок 2 - База даних про співробітників та підрозділи підприємства

4. Створити запити, відповідно до індивідуального завдання. Створити форми для кожного запиту. Закрити доступ до корегування бази даних у режимі запитів.

- 5. Створити звіти, відповідно до індивідуального завдання.
- 6. Створити кнопку для виходу з кожної форми бази даних.
- 7. Створити кнопку виходу для всіх звітів бази даних.
- 8. Створити кнопкову форму та меню 1-3 рівнів.
- 9. Перевірити правильність функціонування кнопкової форми.

# Лабораторна робота №5

Тема роботи: «Мова SQL - засіб створення запитів. Команди SQL».

**Мета роботи:** Вибірка даних з таблиць. Додавання, видалення, редагування інформації. Набуття практичних навичок використання мови SQL.

### Завдання роботи

#### Завдання 1:

1.1 Створити простий запит на вибірку з однієї таблиці. Включити кілька полів таблиці.

1.2 Включити в запит всі поля за допомогою знака "\*".

1.3 Створити запит на вибірку даних з двох зв'язаних таблиць.

1.4 Вибрати кілька полів, за якими сортується висновок.

1.5 Визначити умови відбору («І» та «АБО»). Створити не менше 2-х запитів.

1.6 Визначити умови відбору за допомогою параметра запиту.

1.7 Використати пропозицію Distinct.

#### Завдання 2:

2.1. Створити запит на зовнішнє об'єднання таблиць.
2.2. Продемонструвати використання пропозицій Is null, Is not null.
2.3. Використовувати пропозицію Like.

2.4. Використовувати оператор UNION.

## Завдання 3:

3.1. Створити результуючий запит, який містить декілька результуючих полів.

3.2. Створити простий групуючий запит.

3.3. Створити групуючий запит з групуванням по декількох полях.

3.4. Створити групуючий запит, в якому визначаються умови, причому спочатку виконуються обчислення, а потім відбувається відбір.

3.5. Створити групуючий запит, в якому визначаються умови, причому спочатку відбувається відбір, а потім виконуються обчислення.

# Завдання 4:

4.1. Створити запит з вибором за допомогою Іп.

4.2. Використовувати пропозиції All, Any, Exists.

# Лабораторна робота №6

Тема роботи: «Організація зв'язку з базою даних типу Microsoft Access засобами ADO»

**Мета роботи:** Вивчити механізм доступу до таблиць Microsoft Access за технологією ADO. Створити додаток для внесення даних.

# Завдання роботи

- 1. Створити папку для бази даних.
- 2. Скопіювати базу даних з попередньої роботи в створену папку.

3. Відкрити проект програми в середовищі Delphy, організувати відображення значень таблиці:

- добавити модуль даних: меню File/New/DataModule;
- добавити в модуль об'єкти: ADOConnection1, ADOTable1, DataSource1;
- налаштувати об'єкт ADOConnection1 на на зв'язок з БД Access:

ADOConnection1. ConnectionString.

- 4. Програму зберегти в папці з БД.
- 5. Налаштувати властивості об'єктів для зв'язку з ADOConnection1:
  - ADOTable1.Connection:= ADOConnection1;
  - DataSource1.DateSet:= ADOTable1;
  - ADOTable1.TableName:= Анкета;
  - ADOTable1.Active:=True.
- 6. Підключити модуль до форми 2 меню File/Use Unit...

7. На форму винести об'єкт DBGrid1, налаштувати його властивості: DBGrid1.DataSourse:= DataModule2.DataSource1.

8. Створити інтерфейс користувача для роботи з БД на власний розсуд.

# 8. Індивідуальні завдання

### **BAPIAHT 1.**

Фірма-постачальник добрив господарствам агросектору уклали угоди з певними замовниками. Фірма надає такі пільги: замовлення на 50 тон та вище – 90% вартості, на 100 тон та вище-85%, на 200тон та вище-83%. Для реєстрації замовлень, збереження інформації про замовників, про асортимент добрив та обробки цієї інформації необхідно розробити базу даних, котра складається з трьох таблиць:

1. "Замовлення":

Поля: код замовлення, дата заповнення, код замовника, назва добрива, код добрива, категорія пільг, площа для обробки, дата постачання.

2. "Добрива":

Поля: назва добрива, код добрива, виробник, норма використання на один гектар, вартість однієї тони, термін зберігання.

3. "Господарства":

Поля: код замовника, назва господарства, область, район, індекс, телефон, голова господарства.

#### <u>Запити:</u>

- на певну дату постачання (яка може бути введена як параметр) всі замовлення;

- збільшити вартість добрива на10% для певного виробника;

- для кожного замовника розрахувати необхідну кількість добрива, вартість цієї маси, суму до сплати (вартість замовлення-пільги+20%ПДВ);

- суму замовлень за кожен місяць поточного року;

- які добрива не користуються попитом(замовлення на них відсутні на протязі попереднього та поточного років).

## Задача для створення форм та обробки даних

Розробити систему, що дозволяє реєструвати замовлення з підрахуванням необхідної кількості добрива, вартості цієї маси, суми до сплати (вартість замовлення-пільги+20%ПДВ). Необхідно передбачити можливість вибору добрива, що мають однакові назви, але відрізняються по інших атрибутах.

Також необхідно передбачити можливість оновлення таблиці Добрива, коли термін зберігання певного виду добрива закінчився (видалення цих записів).

<u>Звіти</u>:

- кількість поставлених добрив по місяцях для кожної області з підрахунком відсоткового входження для кожної області до загальної суми поставлених добрив;

- суму замовлень за кожен квартал поточного року по кожному господарству та загальну суму.

#### **BAPIAHT 2.**

Для ведення підрахунків сплат абонентів телефонної мережі розробити базу даних "Сплати за телефон", що містить <u>три таблиці</u> :

1. "Абоненти" - містить інформацію про абонентів телефонної мережі.

Поля: прізвище, ім'я та по батькові, № телефону, код тарифу, кількість несплачених хвилин за міські розмови, кількість несплачених хвилин за міжміські розмови, кількість несплачених хвилин за міжнародні розмови.

2. "Тарифи" - відображає пільги, які надаються при сплаті абонентам деяких категорій (ветерани ВВВ, інваліди, чорнобильці та т. і.).

Поля: код тарифу, назва пільгової категорії, розмір сплати.

3. "Платежі" – це журнал сплат абонентів, який повинен оновлюватись щомісячно. Вважаємо, що кількість хвилин розмов додається до значень "кількість несплачених хвилин" першого числа наступного місяця.

Поля: № телефону, дата платежу, внесена сума за міські розмови, внесена сума за міжміські розмови, внесена сума за міжнародні розмови, розрахунковий рахунок.

Плата за міжміські та міжнародні розмови повинна вноситися до 15 числа наступного місяцю. Пеня, в розмірі 1% від вартості розмови, стягується, якщо сплачено не вчасно.

Запити:

- для кожного абоненту на певну дату загальну внесену суму;

- для кожного абоненту всі дати платежів, суми необхідні до сплати, фактичні платежі, їх різниця та кількість несплачених хвилин по кожному виду розмов;

для певної пільгової категорії зменшити тариф на 10%;

- для кожного абоненту суму всіх платежів по кожному виду розмов та їх загальну суму за поточний рік;

- які абоненти зовсім не сплачували за телефон?

Задача для створення форм та обробки даних

Розробити систему, що дає можливість реєструвати платежі, підраховувати суми сплат за кожен вид розмов, підраховувати загальну суму платежу.

Необхідно надати можливість підраховувати заборгованість загальну та по кожному виду сплат, також необхідно передбачити можливість оновлення таблиці "Абоненти" наприкінці кожного місяцю, вносячи дані у поля "кількість несплачених хвилин" та наприкінці року "Платежі", видаливши всі записи за попередній рік з паралельним внесенням кількості несплачених хвилин до таблиці "Абоненти".

<u>Звіти:</u>

- для кожного абоненту суму всіх платежів по кожному виду розмов за кожен місяць та їх загальну суму за поточний рік;

- кількість хвилин розмов кожного виду за кожен місяць, за рік, загальна сума та відсоткове входження кожного виду розмов до загальної суми.

### **BAPIAHT 3.**

Туристична фірма розміщує гостей міста у готелях. Фірма надає пільги при розміщені для гостей з дітьми. При поселені їх у дво- або трьохмісних номерах сплачується 80% від вартості проживання.

Для отримання повної інформації про вільні місця у готелях міста та обробки наданої інформації розробити базу даних "Розміщення гостей", що містить <u>три таблиці:</u> "Готелі міста", "Замовлення", "Гості міста":

1. "Готелі міста" - включає інформацію про готелі міста.

Поля: назва, код готелю, рівень сервісу (вибір зі списку від 1 до 5 зірок), адреса, район міста, кількість номерів люкс, кількість номерів одномісних, кількість номерів двомісних, кількість номерів трьохмісних, вартість проживання у номерах люкс, вартість проживання у номерах одномісних, вартість проживання у номерах двомісних, вартість проживання у номерах трьохмісних.

2. "Замовлення" - заповнюється на кожний замовлений номер.

Поля: код замовлення, код готелю, код гостя, дата заселення, кількість діб, тип номеру.

 "Гості міста" - заповнюється при реєстрації гостей у туристичній фірмі.

Поля: код, прізвище, ім'я, країна проживання, з дітьми або ні.

Запити:

- для кожного району міста готелі, у яких певний рівень сервісу (рівень сервісу використовувати як параметр);

- для кожного гостя загальну вартість проживання (вартість за одну добу\*термін проживання\*пільги+20%ПДВ);

- для певного готелю збільшити вартість проживання на 5%;

 на певну дату кількість вільних місць одного типу по всіх готелях міста;

- які готелі не користуються попитом(не мають жодного замовлення за квартал).

## Задача для створення форм та обробки даних

Розробити систему реєстрації гостей. Необхідно надати можливість вибору готелю(по району міста, по тарифам, по рівню сервісу) на певну дату. При реєстрації необхідно обчислювати дату від'їзду, вартість проживання(вартість за одну добу\*термін проживання\*пільги+20%ПДВ)

Звіти:

- загальну кількість замовлень за поточний рік по районах міста та по рівню сервісу;

- для кожного готелю отриману суму за проживання по місяцях.

#### **BAPIAHT 4.**

Для обслуговування потреб будівельного майданчику розробити базу даних, що містить <u>три таблиці</u>: "Матеріали", "Постачальники", "Замовлення".

1. Таблиця "Матеріали" - містить дані про будівельні матеріали, що може бути поставлена на майданчик. Необхідно передбачити, що матеріали можуть мати однакові назви, але відрізняться по інших атрибутах.

Поля: назва матеріалу, код матеріалу, виробник, вартість одиниці продукції, мінімальна партія, термін зберігання, код постачальника.

2. Таблиця "Замовлення" - містить дані про будівельні матеріали, що необхідно поставити на будівельний майданчик.

Поля: код замовлення, дата заповнення, назва матеріалу, код матеріалу, кількість, одиниця виміру, дата постачання, на яку потрібно отримати матеріали.

3. Таблиця "Постачальники" містить дані про постачальників будівельних матеріалів, з якими укладені угоди.

Поля: код постачальника, назва постачальника, адреса, телефон, прізвище начальника.

<u>Запити:</u>

- на певну дату постачання (яка може бути введена як параметр) всі замовлення;

- для певного виробника збільшити вартість продукції на 15%;

- для кожного замовлення розрахувати вартість та суму до сплати (вартість замовлення +20%ПДВ);

- суму замовлень для кожного постачальника на певну дату;

- які матеріали зовсім не використовувались під час будівництва (замовлення на них відсутні).

#### Задача для створення форм та обробки даних

Розробити систему, що дозволяє реєструвати замовлення з підрахуванням вартості замовлення, суми до сплати (вартість замовлення+20%ПДВ). Необхідно передбачити можливість вибору матеріалу з тих, що мають однакові назви, але відрізняються по інших атрибутах.

Також необхідно передбачити можливість оновлення таблиці "Матеріали", коли термін зберігання певної продукції закінчився(видалення цих записів).

<u>Звіти:</u>

- для кожного з матеріалів кількість постачань, загальна вартість з підрахунком відсоткового входження кожного з матеріалів до загальної суми постачань;

- суму замовлень для кожного постачальника на певну дату.

## **BAPIAHT 5.**

Для ведення підрахунків сплат за використану електроенергію споживачами розробити базу даних "Сплати за електроенергію", що містить <u>три таблиці</u>: "Споживачі", "Тарифи", "Платежі".

1. "Споживачі" - містить інформацію про споживачів електроенергії.

Поля: прізвище, ім'я та по батькові, код споживача, код тарифу, показання лічильника на кінець попереднього року, заборгованість за попередній рік.

2. "Тарифи" - відображає пільги, які надаються при сплаті споживачами деяких категорій (ветерани ВВВ, інваліди, чорнобильці та т. і.).

Поля: код тарифу, назва пільгової категорії, розмір сплати.

3. "Платежі" – це журнал сплат споживачів, який повинен оновлюватись щорічно.

Поля: код споживача, дата платежу, внесена сума, розрахунковий рахунок, попередній показник лічильника, теперішній показник лічильника.

Вважаємо, що кількість кВт/год вноситься на початку наступного року.

Запити:

- для кожного споживача всі сплати за поточний рік та кількість кВт/год, що сплачується;

- для певної пільгової категорії зменшити тариф на 10%;

- для кожного споживача всі дати платежів, суми необхідні до сплати, фактичні платежі, їх різниця;

- для кожного споживача підрахувати заборгованість по сплатах за рік;

- які споживачі зовсім не сплачували за електроенергію.

## Задача для створення форм та обробки даних

Розробити систему, що дає можливість реєструвати платежі, підраховувати суми сплат, їх підсумок за поточний рік та заборгованість на дату перевірки.

Необхідно надати можливість підраховувати заборгованість за попередній рік.

Також необхідно передбачити можливість оновлення таблиці "Споживачі" та "Платежі" наприкінці року. Для таблиці "Споживачі" підсумувати заборгованість кожного споживача (з урахуванням і тих, що зовсім не сплачували за спожиту електроенергію, вважаючи, що показання лічильника на кінець року відомі). До таблиці "Платежі " занести записи по кожному споживачу за датою 1 січня, нульовою сумою внеску та показанням лічильника на кінець попереднього року.

<u>Звіти:</u>

для кожного споживача суму всіх платежів ;

- для кожного споживача заборгованість та загальну суму боргу усіх споживачів по місяцях та їх відсоткову величину до загальної суми.

## **BAPIAHT 6.**

Міський молокозавод виконує замовлення магазинів на постачання своєї продукції. Для реєстрації замовлень, збереження інформації про замовників, про асортимент продукції, що виробляється, та організації поставок необхідно розробити базу даних, котра складається з <u>трьох таблиць:</u>

1. "Замовлення магазинів".

Поля: код замовлення, дата заповнення, код магазина-замовника, назва продукції, код продукції, об'єм замовленої партії, об'єм поставленої партії, дата постачання.

2. "Продукція".

Поля: назва продукції, жирність, код продукції, вартість одиниці продукції, мінімальна партія, термін зберігання.

3. "Магазини".

Поля: код магазина-замовника, назва магазину, адреса, район міста, телефон, директор магазину.

Запити:

- на певну дату постачання (яка може бути введена як параметр) всі замовлення для кожного магазину;

- для продукції певної назви збільшити вартість на10%;

- на кожну дату для кожного замовника замовлення, вартість замовленої партії продукції, суму до сплати (вартість замовлення +20%ПДВ);

- суму замовлень за останній квартал поточного року по кожному магазину;

- яка продукція не користуються попитом(замовлення на неї відсутні на протязі попереднього та поточного місяцю).

## Задача для створення форм та обробки даних

Розробити систему, що дозволяє реєструвати замовлення з підрахуванням вартості замовленої партії продукції, суми до сплати (вартість замовлення +20%ПДВ), об'єму продукції, що необхідно допоставити, та її вартості.

Необхідно передбачити можливість вибору продукції, що має однакові назви, але відрізняється по інших атрибутах.

Також необхідно передбачити можливість оновлення таблиці «Продукція», коли термін зберігання певного виду продукції закінчився(видалення цих записів).

<u>Звіти:</u>

- кількість поставленої продукції по місяцях для кожного району міста та загальну суму;

- об'єм продукції, що була недопоставлена за кожен місяць поточного року з підрахунком відсоткового входження кожного виду продукції до загальної суми.

#### **BAPIAHT 7.**

Автомобільний завод випускає різні моделі автомобілів. Для забезпечення потреб складального цеху автозаводу розробити базу даних, що містить <u>три таблиці</u>: "Вузли", "Постачальники", "Замовлення".

1. Таблиця "Вузли" містить дані про вузли автомобілю, що поставляються до складального цеху. Необхідно передбачити, що вузли можуть мати однакові назви, але відрізняться по інших атрибутах.

Поля: назва вузла, код вузла, виробник, вартість одиниці продукції, мінімальна партія, код постачальника(вибір зі списку).

2. Таблиця "Замовлення" містить дані про вузли автомобілю, що необхідно поставити до цеху.

Поля: код замовлення, дата заповнення, назва вузла, код вузла, кількість, дата постачання, на яку потрібно отримати замовлення.

3. Таблиця "Постачальники" містить дані про постачальників автомобільних вузлів, з якими укладені угоди.

Поля: код постачальника, назва постачальника, адреса, телефон, прізвище начальника.

### Запити:

- на певну дату постачання (яка може бути введена як параметр) всі замовлення;

- для певного виробника збільшити тариф на 10%;

- для кожного замовлення розрахувати вартість та суму до сплати (вартість замовлення +20%ПДВ);

- суму замовлень для кожного постачальника на певну дату;

- які вузли зовсім не використовувались за попередній квартал.

### Задача для створення форм та обробки даних

Розробити систему, що дозволяє реєструвати замовлення з підрахуванням вартості замовлення, суми до сплати (вартість замовлення+20%ПДВ). Необхідно передбачити можливість вибору вузла відповідно до виробника.

<u>Звіти:</u>

- для кожного з вузлів та для кожного постачальника кількість постачань, загальна вартість;

- суму замовлень для кожного постачальника на певну дату.

#### **BAPIAHT 8.**

Для автоматизації роботи у касах аеропорту розробити базу даних "Продаж авіаквитків", що містить <u>три таблиці:</u> "Авіалайнери", "Рейси", "Продаж". При сплаті за квитки надаються такі пільги: ветеранам війн-20% від вартості квитка; дітям-50%;працівникам авіасервісу – безкоштовно (100%).

1. Таблиця "Авіалайнери " містить інформацію про літаки.

Поля: бортовий номер; тип літаку, дата останнього техогляду, кількість місць бізнес-класу, вартість квитків у цьому салоні, кількість місць салону

першого класу, вартість квитків у цьому салоні, кількість місць другого класу, вартість квитків у цьому салоні.

2. Таблиця "Рейси " відображає інформацію про рейси аеропорту.

Поля: № рейсу, бортовий номер, дні вильоту (кожен день, по парним чи непарним числам), пункт вильоту, пункт призначення, пункти поміж ними.

3. Таблиця "Продаж " – це журнал продажу квитків у касі.

Поля: № рейсу, дата продажу, тип салону (вибір зі списку), кількість квитків, категорія пільг, дата вильоту (необхідно перевіряти по полю " Дні вильоту" з попередньої таблиці).

## Запити:

- усі рейси до певного пункту(вибір пункту можна вводити, як параметр);

- для певного типу літака збільшити тариф на 10%;

- для кожного зареєстрованого продажу авіаквитків № рейсу, дата продажу, тип салону, вартість квитків у цьому салоні, кількість квитків, категорія пільг, дата вильоту підрахувати суму до сплати (вартість квитка\* кількість квитків\*пільги +20%ПДВ);

- для одного з рейсів за певною датою підрахувати кількість проданих квитків та їх загальну вартість для кожного типу салону;

- який рейс не користується попитом(продажу квитків на нього не відбувалось за три доби до вильоту).

## Задача для створення форм та обробки даних

Розробити систему, що дозволяє реєструвати у касах аеропорту продаж авіаквитків з підрахуванням суми до сплати (вартість квитка\* кількість квитків\*пільги +20%ПДВ), загальної вартості квитків. Необхідно перевіряти по полю " Дні вильоту" з таблиці "Рейси" поле "Дата вильоту". Також необхідно передбачити можливість отримання довідки про кожен рейс з підрахуванням кількості місць, що залишилися, по кожному виду з салонів. Довідка про рейс: № рейсу, бортовий номер, дні вильоту(кожен день, по парним чи непарним числам), пункт вильоту, пункт призначення, пункти поміж ними, кількість

місць бізнес-класу, вартість квитків у цьому салоні, кількість місць салону першого класу, вартість квитків у цьому салоні, кількість місць другого класу, вартість квитків у цьому салоні.

# <u>Звіти:</u>

- загальна вартість проданих квитків за кожен місяць з початку поточного року;

- для кожного рейсу загальну кількість проданих квитків з підрахунком відсоткового входження до загальної кількості проданих квитків по всіх рейсах.

### **BAPIAHT 9.**

Розробити базу даних "Комерційна лікарня", що складається <u>з трьох</u> <u>таблиць</u>: "Відділення", "Лікарі", "Пацієнти".

1. Таблиця "Відділення".

Поля: назва відділення, код відділення, прізвище завідуючого, телефон, вартість лікування за добу у відділенні, кількість місць у відділенні.

2. Таблиця "Лікарі".

Поля: код лікаря, прізвище, ім'я та по батькові, дата народження, категорія, стать, код відділення, де він працює, домашній телефон.

3. Таблиця "Пацієнти".

Поля: прізвище, ім'я та по батькові, дата народження, стать, категорія пацієнта, дата надходження у лікарню, термін лікування, код лікаря, що спостерігає.

#### Запити:

- для кожного лікаря список пацієнтів, що перебували під його доглядом на певний період(вводити як параметр, наприклад, місяць та таке інше);

- для певного відділення збільшити вартість лікування за добу на 5%;
- для кожного пацієнта вартість лікування та сума до сплати;

обчислити суму премії для кожного лікаря по-місячно;

- який лікар не займався лікувальною практикою на протязі поточного року.

#### Задача для створення форм та обробки даних

Розробити систему для використовування у комерційних лікарнях для реєстрації пацієнтів, що надходять до лікарні, та їх розміщення по відділенням, збереження інформації про лікарів цієї лікарні. Необхідно надати можливість обчислювати вартість лікування, суму до сплати та премію лікаря. Розрахунки робити за такими правилами. Усі пацієнти поділяються на категорії:

- перша - пільгова, пацієнти обслуговуються безкоштовно( ветерани війн, діти до 7 років, чорнобильці);

- друга – 50% сплати(пенсіонери, діти до 16 років, інваліди);

- третя – повна сплата.

Лікарі отримують такі премії: 10% від суми до 300грн., 5% - до 1000грн., 2,5% - вище 1000грн.

<u>Звіти:</u>

- для кожного відділення загальну суму, що сплачена пацієнтами, помісячно за той рік з підрахунком відсоткового входження до загальної суми;

- загальна сума премій для кожного лікаря за поточний рік.

# **BAPIAHT 10.**

Видавництво книг виконує замовлення магазинів на поставку книг для продажу. Від кількості замовленої партії книг залежить величина пільг, що надає видавництво: 1000 примірників та більше-90% вартості, 2000 примірників та більше-87%, більше 3000-85%. Для реєстрації замовлень, збереження інформації про замовників, про книжкові видання та організації поставок необхідно розробити базу даних, котра складається з <u>трьох таблиць</u>:

1. "Замовлення магазинів".

Поля: № замовлення, дата заповнення, код магазина-замовника, код книги, замовлена кількість примірників, відправлена кількість, дата постачання.

2. "Книжкові видання".

Поля: код книги, автор, назва книги, тираж, об'єм сторінок, роздрібна ціна.

3. "Магазини".

Поля: код замовника, назва магазину, адреса, район міста, телефон, директор магазину.

Запити:

- на певну дату постачання (яка може бути введена як параметр) всі замовлення для кожного магазину;

- зменшити роздрібну ціну книг певного автора на 5%;

 на кожну дату для кожного замовника замовлення, вартість замовленої партії продукції, суму до сплати (вартість замовлення \* пільги+20%ПДВ);

- суму замовлень за останній квартал поточного року по кожному магазину;

- яка продукція не користуються попитом(замовлення на неї відсутні на протязі попереднього та поточного місяцю).

#### Задача для створення форм та обробки даних

Розробити систему, що дозволяє реєструвати замовлення з підрахуванням вартості замовленої партії, суми до сплати (вартість замовлення +20%ПДВ), об'єму книжкової партії, що необхідно доставити, її вартість. Передбачити довідкову форму по певному виданню.

<u>Звіти:</u>

- кількість поставлених книг по місяцях для кожного району міста та загальну суму;

- об'єм книжкової партії, що була недопоставлена за кожен місяць поточного року.

## **BAPIAHT 11.**

Розробити довідкову систему по асортименту продукції, що виробляється на промисловому підприємстві різними цехами та перевозиться до складу. При

передачі партії продукції у цех вона реєструється. Певну продукцію (наприклад, різного гатунку) можуть виробляти різні цехи цього підприємства. Для цього створіть базу даних, що містить <u>три таблиці:</u> "Продукція", "Цех", "Партія продукції".

1. Таблиця "Продукція" містить дані про види продукції, що виробляються підприємством. Необхідно передбачити, що продукція може бути різного ґатунку.

Поля: назва продукції, вартість одиниці продукції вищого гатунку, вартість одиниці продукції першого гатунку, вартість одиниці продукції другого гатунку, мінімальна партія, термін зберігання.

2. Таблиця "Цех" містить дані про цехи підприємства.

Поля: № цеха, назва, прізвище начальника цеха, телефон.

3. Таблиця "Партія продукції" містить дані про партію продукції, що розміщується на складі.

Поля: код партії, назва продукції, № цеха, показник якості, об'єм партії продукції, дата реєстрації.

Запити:

- довідку про певний вид продукції (у яких цехах виробляється, якого гатунку, найдавніша та остання партія, яка зберігається на складі);

- для певної продукції збільшити вартість одиниці продукції вищого гатунку на 5%;

- для кожної партії продукції підрахувати її загальну вартість, дату, до якої можливе зберігання;

- для кожного цеху за кожен місяць загальну вартість виробленої продукції, що зберігається на складі;

яка продукція зовсім відсутня на складі.

## Задача для створення форм та обробки даних

Розробити систему, що дозволяє реєструвати партію продукції, що розміщується на складі, підраховувати її загальну вартість, дату, до якої можливе зберігання. Надати можливість отримувати довідку про певний вид

продукції (у яких цехах виробляється, якого гатунку, найдавніша та остання партія, яка зберігається на складі).

Також необхідно передбачити можливість оновлення таблиці "Партія продукції", коли термін зберігання певної продукції закінчився (видалення цих записів).

<u>Звіти:</u>

- для кожного цеху за кожен місяць загальну вартість виробленої продукції, що зберігається на складі;

- кількість продукції кожного виду (по ґатунках), що зберігається на складі.

### **BAPIAHT 12.**

Розробити базу даних "Комерційна хірургічна лікарня", що складається з <u>трьох таблиць</u>: "Тарифи", "Хірурги", "Пацієнти".

1. Таблиця "Тарифи".

Поля: категорія операції, назва, вартість, приблизний термін після операційної реабілітації, вартість однієї доби у цей період.

2. Таблиця "Хірурги".

Поля: код хірурга, прізвище, ім'я та по батькові, дата народження, категорія, стать, домашній телефон.

3. Таблиця "Пацієнти".

Поля: прізвище, ім'я та по батькові, дата народження, стать, категорія пацієнта, дата операції, код хірурга, категорія операції, фактичний термін післяопераційної реабілітації.

Запити:

- для кожного лікаря список операцій на певну дату (яку вводити як параметр);

- для певної категорії операцій зменшити тариф на 10%;

- для кожного пацієнта вартість операції, загальна сума лікування та сума до сплати;

- обчислити суму премії для кожного хірурга помісячно;
- який хірург не займався лікувальною практикою на протязі місяцю.

## Задача для створення форм та обробки даних

Розробити систему для використовування у комерційних хірургічних лікарнях для реєстрації пацієнтів, збереження інформації про хірургів цієї лікарні та тарифів. Необхідно надати можливість обчислювати попередню вартість лікування, вартість лікування на дату виписки, суму до сплати та премію хірурга. Розрахунки робити за такими правилами. Усі пацієнти поділяються на категорії:

- перша - пільгова, пацієнти обслуговуються безкоштовно (ветерани війн, діти до 7 років, чорнобильці);

- друга – 50% сплати (пенсіонери, діти до 16 років, інваліди );

- третя – повна сплата.

Хірурги отримують такі премії за проведені операції: 15% від суми до 300грн., 10% - до 1000грн., 5% - вище 1000грн.

Звіти:

- для кожного хірурга загальну суму, що сплачена пацієнтами, за той рік з підрахунком відсоткового входження до загальної суми;

- загальна сума премій для кожного хірурга помісячно за поточний рік.

#### **BAPIAHT 13.**

Розробити базу даних "Комерційна ветеринарна лікарня", що складається з <u>трьох таблиць</u>: "Тарифи", "Лікарі", "Журнал відвідувань".

1. Таблиця "Тарифи".

Поля: код захворювання, назва захворювання, ступень важкості(наприклад, початкова, середньої важкості, тяжка форма, дуже тяжка ), вартість лікування.

2. Таблиця "Лікарі".

Поля: код лікаря, прізвище, ім'я та по батькові, категорія, стать, домашній телефон.

3. Таблиця "Журнал відвідувань". Певна тварина може мати декілька діагнозів, по яким може лікуватися. У цьому випадку для неї створюється необхідна кількість окремих записів за однією датою звернення.

Поля: прізвище господаря, тип тварини, порода, прізвисько, вік тварини, стать тварини, дата відвідування, код лікаря, код захворювання.

Запити:

- для певного господаря (прізвище вводити як параметр) знайти дату першого відвідування на рік;

- для певного захворювання для всіх його ступенів підвищити вартість лікування на 5%;

- для кожного відвідування загальна сума лікування за однією датою звернення та сума до сплати;

- кількість відвідувань одного господаря за місяць;

- який лікар не займався лікувальною практикою на протязі місяця.

### Задача для створення форм та обробки даних

Розробити систему для використовування у комерційних ветеринарних лікарнях для реєстрації тварин, збереження інформації про лікарів цієї лікарні та тарифів. Необхідно надати можливість обчислювати вартість відвідування, суму до сплати та відрахування преміальних лікарю. Розрахунки робити за такими правилами. При першому відвідуванні на рік господаря тварин він сплачує внесок у розмірі 5гривень. Під час третього відвідування за місяць одного господаря (можливо, навіть, з різними тваринами) сума до сплати зменшується на 10%.

Лікарі отримують такі премії: 20% від суми до 300грн., 15% - до 1000грн., 12% - вище 1000грн.

<u>Звіти:</u>

 для кожного лікаря загальну суму, що сплачена пацієнтами, помісячно за попередній рік з підрахунком відсоткового входження до загальної суми;

загальна сума премій для кожного лікаря за поточний рік.

#### **BAPIAHT 14.**

Для ведення підрахунків сплат за використаний споживачами газ розробити базу даних "Сплати за газ", що містить <u>три таблиці</u>: "Споживачі", "Тарифи", "Платежі":

1. Таблиця " Споживачі " містить інформацію про споживачів газу. Таблиця "Тарифи" відображає пільги, які надаються при сплаті споживачами деяких категорій (ветерани BBB, інваліди, чорнобильці та т.і.).

Поля: прізвище, ім'я та по батькові, код споживача, код тарифу, показання лічильника на кінець попереднього року, заборгованість за попередній рік.

2. Таблиця "Платежі" – це журнал сплат споживачів, який повинен оновлюватись щорічно.

Поля: код споживача, дата платежу, внесена сума, розрахунковий рахунок, попередній показник лічильника, теперішній показник лічильника.

3. Таблиця «Тарифи».

Поля: код тарифу, назва пільгової категорії, розмір сплати.

Вважаємо, що кількість газу вноситься на початку наступного року.

Запити:

- для кожного споживача всі сплати за поточний рік та кількість газу, що сплачена;

- для певної пільгової категорії операцій зменшити тариф на 10%;

- для кожного споживача всі дати платежів, суми необхідні до сплати, фактичні платежі, їх різниця;

- загальна сума сплат помісячно за рік;

- які споживачі зовсім не сплачували за газ.

## Задача для створення форм та обробки даних

можливість Розробити систему, шо лає реєструвати платежі. підраховувати суми сплат, їх підсумок за поточний рік та заборгованість на дату перевірки. Необхідно надати можливість підраховувати заборгованість за попередній рік. Також необхідно передбачити можливість оновлення таблиці "Споживачі" та "Платежі" наприкінці року. Для таблиці "Споживачі" підсумувати заборгованість кожного споживача (з урахуванням і тих, що зовсім не сплачували за спожитий газ, вважаючи, що показання лічильника на кінець року відомі). До таблиці "Платежі " занести записи по кожному споживачу за датою 1 січня, нульовою сумою внеску та показанням лічильника на кінець попереднього року.

<u>Звіти:</u>

- всі платежі помісячно;

- для кожного споживача заборгованість, відсоткову величину до загальної суми та загальну суму боргу усіх споживачів.

### **BAPIAHT 15.**

Фірма-постачальник медикаментів аптекам міста уклала угоди з певними замовниками. Фірма надає такі пільги: замовлення на суму 1000 грн та вище – 99% вартості, на 10<sup>4</sup> та вище-95%, на 10<sup>5</sup> та вище-90%. Для реєстрації замовлень, збереження інформації про замовників, про асортимент медикаментів та обробки цієї інформації необхідно розробити базу даних, котра складається з трьох таблиць:

1. Таблиця "Замовлення".

Поля: код замовлення, дата заповнення, код замовника, назва медикаментів, код медикаментів, об'єм замовленої партії, об'єм поставленої партії, категорія пільг, дата постачання.

2. Таблиця "Медикаменти".

Поля: назва медикаментів, код медикаментів, виробник, вартість одиниці продукції, мінімальна партія, термін зберігання.
3. Таблиця "Аптеки".

Поля: код замовника, номер аптеки, адреса, район міста, телефон.

Запити:

- на певну дату постачання (яка може бути введена як параметр) всі замовлення для кожної аптеки;

для певного виробника збільшити вартість одиниці продукції на 10%;

- на кожну дату для кожної аптеки замовлення, вартість замовленої партії продукції, суму до сплати (вартість замовлення -пільги+20%ПДВ);

 суму замовлень за останній квартал поточного року по кожній аптеці;

- яка продукція не користуються попитом (замовлення на неї відсутні протягом попереднього та поточного кварталу).

#### Задача для створення форм та обробки даних

Розробити систему, що дозволяє реєструвати замовлення з підрахуванням вартості замовленої партії продукції, суми до сплати (вартість замовлення - пільги +20%ПДВ), об'єму продукції, що необхідно допоставити, її вартості;

Необхідно передбачити можливість вибору продукції, що має однакові назви, але відрізняється по інших атрибутах.

Також необхідно передбачити можливість оновлення таблиці Медикаменти, коли термін зберігання певного виду продукції закінчився(видалення цих записів).

Звіти:

- суму, на яку була поставлена продукція по місяцях для кожної аптеки міста та загальну суму;

 об'єм продукції, що була недопоставлена за кожен місяць поточного року з підрахунком відсоткового входження кожного виду продукції до загальної суми.

#### **BAPIAHT 16.**

Для аналізу та покращання використання сільськогосподарських земель області розробити базу даних, що містить <u>три таблиці</u>: "Господарства", "Культури", "Поставлено по держзамовленню".

1. Таблиця "Господарства".

Поля: код господарства, назва господарства, район, телефон, голова господарства, посівна площа, загальна площа.

2. Таблиця "Культури".

Поля: код культури, назва, середня врожайність, ціна за продукцію першого гатунку, ціна за продукцію другого гатунку, ціна за продукцію вищого гатунку.

3. Таблиця "Поставлено по держзамовленню".

Поля: код господарства, код культури, маса замовлення, маса, що поставлена, врожайність по господарству, ґатунок продукції (обирати зі списку), дата постачання.

#### Запити:

- для кожної культури усі господарства, що її вирощують, врожайність по господарству та ґатунок поставленої продукції;

- підвищити вартість певної продукції по всіх гатунках на 10%;

- для аналізу роботи господарства для кожного з них усі культури, що вони постачають, різницю між врожайністю середньою та по господарству, вартість поставленої продукції;

- для кожної культури загальну кількість постачань по кожному гатунку на кінець року та їх вартість;

- яку культуру зовсім не поставили по держзамовленню.

#### Задача для створення форм та обробки даних

Розробити систему, що надає можливість реєструвати постачання, обчислювати різницю між врожайністю середньою та по господарству, вартість поставленої продукції, сума, що сплачується державою з урахуванням наданих пільг(якщо врожайність по господарству вище на 15% від середньої, продукція

вищого гатунку та держзамовлення виконано повністю, то держава доплачує 5% від вартості поставленої продукції ). Також необхідно передбачити можливість отримання довідки про культуру з підрахуванням загальної кількості постачань по кожному гатунку та їх вартість на дату запиту.

<u>Звіти:</u>

- для кожної культури загальну кількість постачань по кожному гатунку на кінець року та їх вартість;

- для кожного району області по кожній культурі загальний об'єм постачань, їх вартість та кількість недопоставленої продукції.

#### **BAPIAHT 17.**

Розробити базу даних "Комерційна стоматологічна лікарня", що складається з <u>трьох таблиць</u>: "Тарифи", "Лікарі", "Пацієнти".

1. Таблиця "Тарифи".

Поля: код захворювання або виду роботи, назва, ступень важкості (наприклад, початкова, середньої важкості, тяжка форма, дуже тяжка), вартість.

2. Таблиця "Лікарі".

Поля: код лікаря, прізвище, ім'я та по батькові, дата народження, категорія, стать, домашній телефон.

3. Таблиця "Пацієнти". Певний пацієнт може мати декілька діагнозів, по яким може лікуватися. У цьому випадку для нього створюється необхідна кількість окремих записів за однією датою звернення.

Поля: прізвище, ім'я та по батькові, дата народження, стать, категорія пацієнта, дат звернення, код лікаря, код захворювання або виду роботи.

Запити:

- для кожного лікаря список пацієнтів, що зверталися до нього на певний період (вводити як параметр, наприклад, місяць та таке інше);

- для певного захворювання для всіх його ступенів підвищити вартість лікування на 5%;

- для кожного пацієнта загальна сума лікування за однією датою звернення та сума до сплати;

- обчислити суму премії для кожного лікаря по-місячно;

- який лікар не займався лікувальною практикою на протязі місяця.

#### Задача для створення форм та обробки даних

Розробити систему для використовування у комерційних стоматологічних лікарнях для реєстрації пацієнтів, збереження інформації про лікарів цієї лікарні та тарифів. Необхідно надати можливість обчислювати вартість лікування, суму до сплати та премію лікаря. Розрахунки робити за такими правилами. Усі пацієнти поділяються на категорії:

- перша - пільгова, пацієнти обслуговуються безкоштовно (ветерани війн, діти до 7 років, чорнобильці);

- друга – 50% сплати (пенсіонери, діти до 16 років, інваліди);

- третя – повна сплата.

Лікарі отримують такі премії: 10% від суми до 300грн., 5% - до 1000грн., 2,5% - вище 1000грн..

Звіти:

- для кожного лікаря загальну суму, що сплачена пацієнтами, помісячно за той рік з підрахунком відсоткового входження до загальної суми;

- загальна сума премій для кожного лікаря за поточний рік.

#### **BAPIAHT 18.**

Для автоматизації роботи у касах залізниці розробити базу даних "Продаж квитків", що містить <u>чотири таблиці</u>: "Поїзда", "Рейси", "Продаж", "Тарифи". При сплаті за квитки надаються такі пільги: ветеранам війн-20% від вартості квитка; дітям-50%;працівникам авіасервісу – безкоштовно (100%).

1. Таблиця "Поїзда" містить інформацію про поїзда.

Поля: номер поїзда, дні відправлення (кожен день, по парним чи непарним числам), пункт відправлення, пункт призначення, пункт поміж ними 1, пункт поміж ними 2.

2. Таблиця "Рейси" відображає інформацію про рейси вокзалу. Рейс одностайно визначається номером поїзду та датою відправлення.

Поля: номер поїзду, дата відправлення, кількість плацкартних вагонів, кількість купейних вагонів, кількість СВ.

3. Таблиця "Продаж" – це журнал продажу квитків у касі.

Поля: номер поїзду, дата відправлення, дата продажу, тип вагону (вибір зі списку), кількість квитків, категорія пільг.

 Таблиця "Тарифи" містить інформацію про вартість квитків для певного поїзда.

Поля: номер поїзду, вартість квитків у плацкартному вагоні, вартість квитків для купейного вагона, вартість квитків для СВ.

Запити:

- усі рейси до певного пункту (вибір пункту можна вводити, як параметр);

- для певного номеру поїзда підвищити вартість квитків по СВ на 20%;

- для кожного зареєстрованого продажу квитків номер поїзду, дата відправлення, дата продажу, тип вагону (вибір зі списку), вартість квитків у цьому вагоні, кількість квитків, категорія пільг. Підрахувати суму до сплати (вартість квитка\* кількість квитків\*пільги +20%ПДВ), загальну вартість квитків;

- який рейс не користується попитом (продажу квитків на нього не відбувалось на протязі трьох діб).

#### Задача для створення форм та обробки даних

Розробити систему, що дозволяє реєструвати у касах залізниці продаж квитків з підрахуванням суми до сплати (вартість квитка\* кількість квитків\*пільги +20%ПДВ), загальної вартості квитків. Також необхідно передбачити можливість отримання довідки про кожен рейс з підрахуванням кількості місць, що залишилися, по кожному виду з вагонів. Довідка про рейс: номер поїзду, дата відправлення, кількість плацкартних вагонів, вартість

квитків у цьому вагоні, кількість купейних вагонів, вартість квитків у цьому вагоні, кількість СВ, вартість квитків у цьому вагоні.

#### <u>Звіти:</u>

- загальна вартість проданих квитків за кожен місяць з початку поточного року;

- для кожного рейсу загальну кількість проданих квитків з підрахунком відсоткового входження до загальної кількості проданих квитків по всіх рейсах.

#### **BAPIAHT 19.**

Фірма-постачальник автомобілів приймає замовлення на поставку автомобілів різних виробників та різних моделей автомобілів однієї марки. Фірма надає такі пільги: замовлення на 5 автомобілів –90 % вартості, для постійних клієнтів (які звертались більш ніж два рази на рік). Для реєстрації замовлень, збереження інформації про замовників, про асортимент автомобілів та обробки цієї інформації необхідно розробити базу даних, котра складається з трьох таблиць:

1. Таблиця "Замовлення".

Поля: код замовлення, дата заповнення, код клієнту, код моделі, кількість, дата постачання.

2. Таблиця "Автомобілі".

Поля: код моделі, назва марки, назва моделі, фірма-виробник, вартість моделі, об'єм двигуна, потужність двигуна, кількість пального на 100 км, тип двигуна (дизель чи карбюратор виводити списком).

3. Таблиця "Клієнти".

Поля: код замовника, назва чи ім'я, адреса, телефон, розрахунковий рахунок.

#### Запити:

- перевірити, чи певний клієнт може користуватися пільгами, якщо так, то якими(наприклад по категоріям);

- зменшити вартість автомобілів певного виробника;

- для кожного замовлення вартість замовленої партії продукції, суму до сплати (вартість замовлення -пільги+20%ПДВ);

 суму замовлень за останній квартал поточного року по кожній моделі;

- яка модель не користуються попитом (замовлення на неї відсутні протягом попереднього та поточного кварталу).

#### Задача для створення форм та обробки даних

Розробити систему, що дозволяє реєструвати замовлення з підрахуванням вартості замовлених автомобілів, суми до сплати (вартість замовлення - пільги +20%ПДВ);

<u>Звіти:</u>

- кількість замовлень для кожної моделі на кожний квартал та загальна сплачена сума;

- для кожного клієнта замовлення за попередній та поточний роки.

#### **BAPIAHT 20.**

Фірма надає транспортні послуги. Фірма має можливість використовувати різні види транспорту, інформація про які зберігається окремо. При реєстрації замовлення фіксуються дані про замовників та вимоги до перевезення. При повторних зверненнях інформація про замовника не повторюється, але може оновлюватися, якщо були зміни. На перевезення існують такі пільги:100-999км –знижка на 5%, 1000-4999 – 10%, 5000 та вище – 15%. Для фіксації, зберігання та обробки цієї інформації розробити базу даних, що містить <u>три таблиці:</u> "Замовлення на перевезення", "Транспорт та його вартість", "Замовники".

1. Таблиця "Замовлення на перевезення".

Поля: № перевезення, дата перевезення, відстань, маса, накладні витрати, код транспортного засобу, код замовника.

2. Таблиця "Транспорт та його вартість".

Поля: назва, код транспортного засобу, вартість т/км, об'єм вантажу, маса вантажу.

3. Таблиця "Замовники".

Поля: код замовника, назва чи ім'я замовника, адреса, телефон, розрахунковий рахунок.

Запити:

- види транспорту для певної ваги чи певного об'єму (вносити, як параметр);

- для певного замовника збільшити відстань на 20км;

- замовлення з підрахуванням вартості перевезення та суми до сплати (відстань\* вартість т/км\* маса+ накладні витрати - пільги);

- загальна вартість перевезень за останній місяць;

- який вид транспорту зовсім не замовлявся.

#### Задача для створення форм та обробки даних

Розробити систему, що дозволяє реєструвати замовника (з попередньою перевіркою на наявність такої інформації). Система дозволяє вносити замовлення. Необхідно передбачити можливість отримання довідки про різні види транспорту для певної ваги чи певного об'єму. Необхідно підраховувати вартість перевезення та суму до сплати(відстань\* вартість т/км\* маса+ накладні витрати - пільги).

<u>Звіти:</u>

 загальну вартість перевезень з підрахунком відсоткового входження до загальної вартості для кожного виду транспорту за кожен місяць поточного року;

- загальна вартість перевезень на відстань менш ніж 100км, 100-999км, 1000-4999км, більше 5000км за поточний рік.

## 9. Контрольні питання

- 1. Що таке СУБД і які основні функції вона вирішує?
- 2. На які три групи поділяють типи даних?
- 3. Класифікація типів даних.
- 4. Що таке домен?
- 5. Як виглядає схема структури даних, її різновиди?

6. Що таке таблична форма структури даних?

- 7. Які існують методи доступу до даних?
- 8. Що таке ER-модель?
- 9. Поняття сутності, атрибут, набір сутностей, ключ, зв'язок, степінь зв'язку.

10. Як називають модель «Сутність-зв'язок», що представлена у графічному вигляді?

11. Назвіть основні компоненти реляційного відношення?

12. Поняття кортежу та первинного ключа в реляційній моделі даних.

13. Що таке схема відношень та схема бази даних?

14. Які існують елементи графічного зображення ER-діаграми згідно нотації Чена? Поясніть призначення кожного з елементів.

15. Які існують елементи графічного зображення ER-діаграми згідно нотації Мартіна? Поясніть призначення кожного з елементів.

16. Поняття 1NF, 2NF, 3NF. Загальні властивості форм.

17. Як створити таблицю в режимі «Конструктор таблиц»:

- як задати ім'я поля?

- як задати тип даних?

- як визначити конкретні значення типу даних (формат даних)?
- як відкрити таблицю, заповнити рядки та перейменувати таблицю?

- як використати тип «Мастер подстановок»?

- як задати ключове поле та складений ключ?

18. Як створити таблицю в режимі «Мастер таблиц»?

19. Як встановити типи зв'язків у базі даних: «один до одного», «один до багатьох», «багато до багатьох». «багато до одного»?

20. Як створити форму та автоформу для заповнення таблиці?

21. Як відбувається перехід по записам?

22. Що називають степенем зв'язку, класом приналежності та кардинальністю. Як ці поняття зображуються на діаграмі «Сутність-Зв'язок»?

23. Для чого необхідна форма для додавання нового запису і як її створити?

24. Як створити макрос «Добавление» та макрос «Закрытие»?

25. Реляційні бази даних, табличне представлення. Групи даних, що повторюються, ключі. Переваги і недоліки реляційних баз даних.

26. Як змінити елементи керування для полів у відповідних таблицях?

27. Для чого необхідна функція управління Iif?

28. Як створити підлеглу форму за певними результатами?

29. Які дії необхідно зробити щоб:

- змінити стиль форми;

- змінити колір фону і колір шрифту для кожного поля;

- вставити на місце певного поля малюнок;

- змінити шрифт (розмір, вид, колір) для обраного поля.

30. Що таке запит і за допомогою чого він будується?

31. Методика створення запитів:

- на вибірку певних записів;

- запит по таблиці, який буде формувати список за певними критеріями.

32. Як створити форму запитів?

- 33. Закрити доступ до коректування бази даних у режимі запитів. Дії.
- 34. Створення звітів по обраним таблицям.
- 35. Вставити поле дати та форматувати кожне поле.

36. Які дії необхідно провести, щоб:

- створити кнопки виходу з всіх форм;

- створити кнопки виходу зі всіх звітів.

37. Що таке мова SQL і для реалізації чого вона необхідна?

38. Структура SQL-оператору створення таблиці мережевої БД.

39. Команда Select. Синтаксис написання команди.

40. Використання в операторі Select рядка WHERE.

41. Використання в операторі Select рядка ORDER BY.

42. Використання в операторі Select рядка GROUP BY.

43. Арифметичні функції оператору Select.

44. Команда Alter Table.

45. Команда Create View.

46. Сортування результатів запиту.

47. Команда Insert.

48. Команда Update.

49. Команда Delete.

50. Логічні оператори.

51. Введення обмежень на групові записи за допомогою рядка HAVING оператору Select.

52. Як за допомогою мови SQL створити простий запит на вибірку з однієї таблиці. Включити кілька полів таблиці.

53. Як працює знак «\*»?

54. Як створити запит на вибірку даних з двох зв'язаних таблиць операторами мови SQL?

55. За допомогою яких інструкцій можливо визначити умови відбору використовуючи параметр запиту?

56. Як створити відсортований по обчислювальному полю запит з декількох таблиць, в якому визначені умови «І» та «АБО»?

57. Як створити запит на зовнішнє об'єднання таблиць?

58. Які дії виконують: пропозиція Distinct, пропозиції Is null, Is not null, пропозиція Like, оператор UNION?

59. Як створити простий групуючий запит?

60. Як створюється запит за допомогою: In, All, Any, Exists.

### СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- 1. Карпова Т. С. Базы данных. Модели, разработка реализация. СПб.: Питер, 2001. 304 с.
- Єрьоміна Н. В. Проектування баз даних. Навчальний посібник. К.: КНЕУ, 1998. – 208 с.
- Дейт К. Дж. Введение в системы баз данных.: Пер с англ. 7-е изд. К.: Диалектика, 2001. – 1084 с.
- 4. Мартин Дж. Организация баз данных и знаний. М. Мир, 1982 г, 756 с.
- 5. М. Мейер. Теория реляционных баз данных : Пер. с англ. М: Мир, 1987.
  608 с.
- Т. Тиори, Дж. Фрай. Проектирование структур баз данных: В 2-х кн. Пер. с англ.- М.: Мир, 1985.
- Голицина О.Л., Максимов Н.В., Попов И.И. Базы данных: учебное пособие М.: Форум: Инфра-м, 2004. – 352 с.
- Д. Ульман. Основы систем баз даннях / Пер. с англ. М.Р. Когаловского и В. В. Когутовского. М.: Финансы и статистика, 1983. - 334 с.
- Проектирование баз данных. СУБД Microsoft Access: учебное пособие для вузов / Н.Н.Гринченко, Е.В. Бойко, Н.П. Макаров и др.- М.: Горячая Линия-Телеком, 2004. – 240 с.
- 10. Диго С. М. Проектирование баз данных: Учебник М.: Финансы и статистика, 1988. 216 с.
- 11. Галузинський Г. П., Гордієнко І. В. Сучасні технологічні засоби обробки інформації: Навчальний посібник. К.: КНЕУ, 1998.- 224 с.
- 12. Корнеев В. В, Гареев А. Ф. и др. Базы данных. Интеллектуальная обработка информации. М.: Нолидж, 2001. 496 с.
- Коннолли Т., Бегг К., Страчан А. Базы данных. Проектирование, реализация и сопровождение. Теория и практика. 2-е изд. М.: Вильямс, 2000. – 1120 с.

- 14. Искусственный интеллект. В 3-х кн. Кн.2. Модели и методы: Справочник / Под ред. Д.А. Поспелова М.: Радио и связь, 1990. 304 с.
- 15. Шафрин Ю. Информатика. Информационные технологии М.: ООО Лаборатория базовых знаний, 1998. 700 с.
- 16. Кириллов В.В. Структурированный язык запросов (SQL). СПб.: ИТМО, 1994. 80 с.
- 17. С.Д. Кузнецов. Введение в стандарты языка баз данных SQL. Центр Информационных Технологий, 1998.

# додатки

Додаток А

## Форми представлення структур даних

1. Таблична форма представлення даних

## Таблиця 1 – Загальний вигляд табличної форми структури даних

Тип поля 1	Тип поля 2	 Тип поля n	
Значення рядка 1.1	Значення рядка 2.1	 Значення рядка n.1	
Значення рядка 1.n	Значення рядка 2.n	 Значення рядка n.n	

## Таблиця 2 - Приклад табличної форми

Поле	Свойства полей							
	Размер	Формат поля	Маска ввода	Подпись	Значение по умолчанию	Обязательное поле	Индексирован ное поле	
Прізвище	30			Фамилия		да	Да (совпадения допускаются)	
Ім'я	12			Имя		да	нет	
По-батькові	30			Отчество		нет	нет	
Дата народження		Краткий формат даты	Краткий формат	Дата рождения		да	нет	
Адреса	50			Адрес		да	нет	
Телефон	6					нет	нет	
Група	Цел ое	основной		Группа	Номер вашей группы	да	Да (совпадения допускаются)	



Рисунок 1 – Таблична форма визначення властивостей полів

2. Графова форма для структурних таблиць «Викладачі» та «Залік»



3. Графічна форма для структурних таблиць «Іспит» та «Предмети»



Рисунок 3 – Графічна форма

## Додаток Б

Реляційна	модель	даних
,	, ,	, ,

	Рядок	Рядок	Рядок	Рядок	Рядок	Рядок	Типи даних
	Прізвище	Ім'я	По-батькові	Предмет	Посада	Кафедра	Домени
•	Прізвище	Ім'я	По-батькові	Предмет	Посада	Кафедра	Атрибути
	Куницька	Світлана	Юріївна	ОБД	доцент	ІБтаКІ	
	Нестеренко	Алла	Миколаївна	Вища математика	доцент	ПМ	Кортежі
,	Півень	Олег	Борисович	Архітектура ком'ютерів	доцент	ІБтаКІ	J

Відношення

Рисунок 1 - Представлення реляційного відношення таблиці «Викладачі»



Рисунок 2 - Представлення реляційного відношення таблиці «Предмет»



## Діаграма ER-моделі