

Курсова робота

На тему: Автоматизація лікарняного процесу

Зміст розрахунково-пояснювальної записки:

Вступ.....	3
Розділ 1. Постановка задачі та аналіз предметної області.....	5
Аналіз предметної області.....	5
Постановка задачі.....	6
Розділ 2. Теоретичні відомості.....	7
Вибір засобів / методологій проектування. Вибір СУБД.....	7
Розділ 3. Побудова алгоритму рішення задачі.....	9
Побудова інфологічної (концептуальної) моделі наочної області.....	9
Інфологічна модель бази даних.....	10
Проектування логічної структури бази даних.....	13
Розділ 4. Структура програмного комплексу, тексти вихідних програм (модулів, процедур).....	15
Проектування фізичної структури бази даних.....	15
Проектування структури бази даних.....	15
Створення структури бази даних.....	21
Заповнення таблиць початковими даними.....	23
Створення форм для введення даних.....	26
Створення запитів.....	33
Створення для всіх таблиць і запитів звіти.....	35
Розділ 5. Описання роботи створеного програмного продукту.....	37
Результати роботи програмного продукту.....	38
Висновки.....	39
Список використаної літератури:.....	40

Вступ

Не викликає сумнівів, що останніми роками потреба в базах даних різної архітектури в нашій країні значно зросла. База даних (БД) є організованою структурою, використовуваною для зберігання даних, тобто будь-яких відомостей про явища, процеси, діях і так далі

Якщо ще 20 років тому в більшості організацій використовувалися картотеки, то сьогодні «шафи із скриньками» можна зустріти все рідше і рідше – комп'ютерні бази даних, значно компактніші (раніше в наукових бібліотеках каталоги займали не одну кімнату, тепер же поміщаються на крихітному компакт-диску), зручніші в роботі і дозволяючи досягти високих швидкостей обробки інформації, нестримно завойовують популярність.

Залежно від роду діяльності компанії формуються її вимоги до використовуваної бази даних. Так, операторам стільникового зв'язку потрібні складні білінгові системи – мільйони абонентів, десятки тарифних планів, мільярди дзвінків, кожен з яких треба правильно протарифицировать. Зрозуміло, вони не жаліють грошей на оплату праці програміста, який здатний написати таку систему. У більшості ж випадків записів в базі даних порівняно небагато, алгоритми обробки теж стандартні (пошук і вибірка). Прикладом може служити база даних, що розробляється в даному завданні, присвячена обліку розрахунків з клієнтами в банці.

Представте роботу радіостанції: слухач, що додзвонився, просить поставити йому пісню. Звичайно, пояснити, чого він хоче, у формі «ім'я композитора – ім'я виконавця – назва пісні – рік випуску» він не може. Зазвичай в пам'яті відкладаються лише уривчасті відомості. Співробітники студії теж не в змозі запам'ятати розміщення тисяч записів – а знайти треба швидко, не можна ж сказати слухачеві: «Почекайте годинки три, ми доки пошукаємо». Тут і приходиться база даних, що на допомогу розробляється, дозволяє знайти запис по жанру (відповідно до побажання «Хочу яку-небудь

сумну і мелодійну пісеньку»), по мові («Лише щоб по-російськи співали! А то незрозуміло нічого»), по артистові («Мені дуже подобається, як співає Лариса Доліна. Поставте яку-небудь її пісеньку») і тому подібне

Для створення нескладних баз даних ідеально личить СУБД Access, що входить в той, що є практично на будь-якому комп'ютері пакет програм MS Office. Одне з основних достоїнств даної системи – вона дає можливість створити базу навіть непрофесіоналам, звичайним користувачам персональних комп'ютерів.

Перш ніж приступити як розробці бази даних, розглянемо основні етапи цього процесу.

Проектування БД - це впорядкований формалізований процес створення системи взаємозв'язаних описів, тобто таких моделей наочної області, які зв'язують (фіксують) ті, що зберігаються в базі дані з об'єктами наочної області, що описуються цими даними. Прикладне призначення таких описів полягає в тому, аби користувач практично не має уявлення про організацію даних в БД, обертаючи запит до бази, мав би практичну можливість отримати адекватну інформацію про стан об'єкту наочної області.

Проектування БД складається з трьох частин: концептуальне, логічне і фізичне проектування. Основною метою моєї роботи є розгляд автоматизованої інформаційної системи, про яку без прикладів не розповіси. Тому кожен крок проектування БД супроводитиметься заданою темою роботи - «Лікарня».

Розділ 1. Постановка задачі та аналіз предметної області

Аналіз предметної області

У даній курсовій роботі зачіпається лише та діяльність лікарня, яка охоплює відомості про підрозділи лікарні, данні про лікарів, їх спеціалізацію, типи лікування, список діагнозів, дані про пацієнтів лікарні, а також записи на прийом та записи прийому пацієнтів лікарями. Всі ці дані будуть занесені в базу даних лікарні.

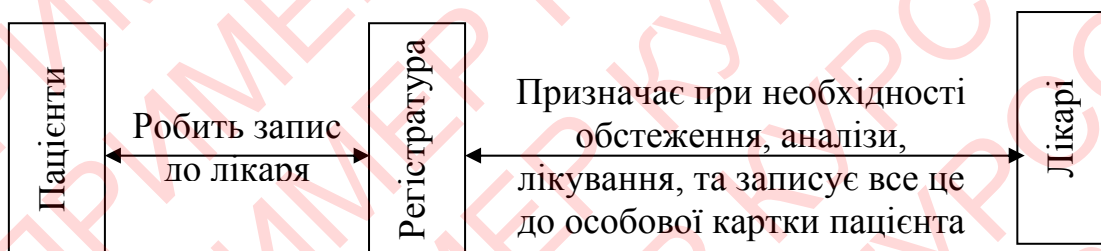
Пацієнт звертаючись до лікарні записується на прийом, на конкретну дату та час, до певного лікаря, якого вибирає за спеціалізацією та приналежністю до певного підрозділу лікарні.

По цим записам на прийом кожен лікар може бачити скільки пацієнтів прийде до нього певного дня.

Коли до лікаря приходять на прийом пацієнти, кожен лікар заводить запис прийому пацієнтів в особовій картці пацієнта. В цьому запису лікар вказую ПІБ пацієнта, своє ПІБ, дату і час прийому, з якими скаргами на своє здоров'є прийшов пацієнт, результати перинного огляду пацієнта, ставить первинний діагноз, а також визначає необхідність проведення певних обстежень та аналізів, та назначає певне лікування пацієнтові, по конкретному типу лікування, та вказує пацієнтові коли прийти наступного разу.

Завдяки записам в особових картках пацієнтів лікар може відстежувати хід лікування та виздоровлення пацієнтів, тобто знати історію хвороби пацієнта. А також бачити якими ще хворобами болів пацієнт, звертаючись до інших лікарів цієї ж лікарні.

В даний час лікарня працює за наступною структурною схемою:



Малюнок 1 - Структурна схема роботи лікарні.

Постановка задачі

Тому в даній курсовій роботі було поставлено наступне завдання: спроектувати і створити базу даних, яка б вела облік підрозділів та спеціалізації лікарні, відомості про пацієнтів, дані про лікарів, стандартизовані типи лікування та загальноприйняті діагнози захворювань, а також журнали запису на прийом та особові картки пацієнтів.

Розділ 2. Теоретичні відомості

Вибір засобів / методологій проектування. Вибір СУБД

На сьогоднішній день існує багато всіляких систем управління базами даних. Це такі СУБД як Paradox, FoxPro, Clipper, Access і ін. Для роботи з більшістю з них потрібні досить глибокі знання даної СУБД і досвід програмування.

Успіх Microsoft Access полягає в прекрасній реалізації продукту, розрахованого як на початківця, так і кваліфікованого користувача. Microsoft Access – це найпопулярніша сьогодні настільна система управління базами даних.

У Microsoft Access присутня мова програмування Visual Basic, який дозволяє створювати масиви, своїх типів даних, контролювати роботу застосувань. MS Access має один з самих кращих наборів візуальних засобів розробки і представлення інформації серед аналогічних програмних продуктів.

Одна з основних переваг MS Access – інтеграції з популярним офісним пакетом Microsoft Office.

Вся робота з базою даних здійснюється через вікно контейнера бази даних. Звідси здійснюється доступ до всіх об'єктів: таблицям, запитам, формам, звітам, макросам, модулям.

Вбудована мова запитів SQL дозволяє максимально гнучко працювати з даними і значно прискорює доступ до зовнішніх даних.

Access сприймає велику кількість форматів даних, включаючи файлові структури інших СУБД. Тому застосування в Access може імпортувати з текстових файлів або електронних таблиць і експорт в них: надавати прямий

доступ і оновлювати файли Paradox, FoxPro і інших БД. Можна також імпортувати дані з цих файлів в таблиці Access.

Перевагою Access є наявність засобів проектування застосування БД без знання мови програмування. Робота в Access починається з визначення реляційних таблиць і полів, призначених для зберігання даних. Відразу після цього за допомогою форм, звітів, макросів і VBA можна визначати дії над цими даними. Форми і звіти використовуються для виводу на екран і додаткових обчислень при роботі з таблицями. В разі розробки складнішого застосування можна використовувати мову Visual Basic.

Архітектура Access називає об'єктами все, що може мати ім'я. У БД Access основними об'єктами є таблиці, запити, форми, звіти, макроси і модулі. Термін БД зазвичай відноситься лише до файлів, в яких зберігаються дані. У Access БД включає всі об'єкти, пов'язані з даними, що зберігаються, у тому числі і ті, які визначаються для автоматизації роботи.

У таблицях зберігаються дані. Використовуючи форми, можна виводити дані на екран або змінювати їх. Форми і звіти отримують дані як безпосередньо з таблиць, так і через запити. Для виконання обчислень запити можуть використовувати вбудовані функції або функції, створені за допомогою Visual Basic для застосувань.

Розділ 3. Побудова алгоритму рішення задачі

Побудова інфологічної (концептуальної) моделі наочної області

Концептуальне проектування починається з аналізу ПО, включає аналіз концептуальних вимог і інформаційних потреб, виявлення інформаційних об'єктів (ІО) і зв'язків між ними, побудову концептуальної моделі (схеми) даних, названою ER-діаграма.

Одна з особливостей ER-діаграми полягає в тому, що результати аналізу наочної області відбиті в досить простому, наочному, але в той же час формалізованому і досить інформативному вигляді. У ER-діаграмі поєднуються функціональний і інформативний підходи, що дозволяє представляти як сукупність виконуваних функцій, так і стосунки між елементами системи, що задаються структурами даних. Графічна форма дозволяє відображувати типологію і властивості сутностей і зв'язків, а формалізми, покладені в основу діаграми, дозволяють використовувати на наступному кроці проектування логічної БД строгий апарат нормалізації.

Основними конструктивними елементами ER-діаграм є сутства, зв'язки між ними і їх атрибути (властивості).

Сутність - будь-який конкретний (реальний) або абстрактний об'єкт в тому, що розглядається ПО.

Зв'язок - спостережуваний взаємозв'язок (асоціація) між сутностями.

У ER-діаграмах сутності позначаються прямокутниками, асоціації (характери об'єднання сутностей) - ромбами або шестикутниками, а зв'язки між атрибутами - ребрами, над якими проставляються типи зв'язків.

Для проектування структури БД необхідна вихідна інформація про наочної області бажано у формалізованому вигляді. Наочною областю (ПО)

називається частина реальної системи, що представляє інтерес для даного дослідження.

При проектуванні автоматизованих інформаційних систем ПО представляється у вигляді моделей даних декількох рівнів. Число використовуваних рівнів залежить від складності системи, але в будь-якому разі включає логічний і фізичний рівні. ПО може відноситися до будь-якого типу організації (інститут, завод, мале підприємство і тому подібне).

Необхідно розрізняти повну ПО підприємства в цілому і організаційну одиницю цієї ПО. Організаційна одиниця, у свою чергу, може представляти свою ПО (відділи).

Інформація, необхідна для опису ПО, залежить від реальної моделі і може включати відомості про персонал, заробітну плату, товари, тобто відомості про людей, місця, предмети, події і поняття.

Опис ПО, без орієнтації на використовувані програмні і технічні засоби, називається інфологічною моделлю (ІЛМ).

Інфологічна модель бази даних

Компоненти інфологічної моделі (ІЛМ) включають ряд описів об'єктів ПО і зв'язки між ними. Опис ПО завжди представлено в якійсь знаковій системі. Тому окрім стосунків, властивих ПО, виникають ще і стосунки, обумовлені особливостями відображення ПО в мовному середовищі.

В даний час не існує єдиного стандарту або загальноприйнятого способу побудови ІЛМ. Для опису ІЛМ використовуються як мови аналітичного (описового) типу, так і графічні засоби.

В процесі аналізу в ПО виділяють класи об'єктів. Класом об'єктів називають сукупність об'єктів, що володіють однаковим набором властивостей. Наприклад, якщо в якості ПО розглянути вуз, то в ній можна виділити наступні класи об'єктів: що вчаться, викладачі, аудиторії і так далі

Об'єкти можуть бути реальними або абстрактними, як, наприклад, предмети, які вивчають студенти.

При відзеркаленні в ІС кожен об'єкт представляється своїм ідентифікатором, який відрізняє один об'єкт класу від іншого, а кожен клас об'єктів представляється ім'ям цього класу. Так, для об'єктів класу «предмети», що «Вивчаються, ідентифікатором кожного об'єкту буде «Назва предмету». Ідентифікатор має бути унікальним. Кожен об'єкт володіє певним набором властивостей. Для об'єктів одного класу набір цих властивостей однаковий, а їх значення можуть розрізнятися.

Кожному класу об'єктів в ІЛМ привласнюється унікальне ім'я.

При описі ПО треба відобразити зв'язку між об'єктом і властивостями, що характеризують його. Це зображується у вигляді лінії, що сполучає позначення об'єкту і його властивостей.

Зв'язок між об'єктом і його властивістю може бути різним. Об'єкт може володіти лише одним значенням якоїсь властивості. Наприклад, кожна людина може мати лише одну дату народження. Назвемо такі властивості одиничними. Для інших властивостей можливе існування одночасно декількох значень в одного об'єкту. Хай, наприклад, при описі «Співробітника» фіксується як його властивість «Іноземна мова», якою він володіє. Оскільки співробітник може знати декілька іноземних мов, то таку властивість називатимемо множинною. При зображенні зв'язку між об'єктом і його властивостями для одиничних властивостей використовуватимемо одинарну стрілку, а для множинних властивостей – подвійну. Крім того, деякі властивості є постійними, їх значення не може змінитися з часом. Назвемо такі властивості статичними, а ті властивості, значення яких може змінюватися з часом, називатимемо динамічними.

Окрім зв'язку між об'єктом і його властивостями, в ІЛМ фіксуються зв'язки між об'єктами різних класів. Розрізняють зв'язки типу «один до одного», «один до багатьом», «багато до одного» і «багато до багатьом». Інколи ці типи зв'язків називаються мірою зв'язку. Окрім міри зв'язку в ІЛМ

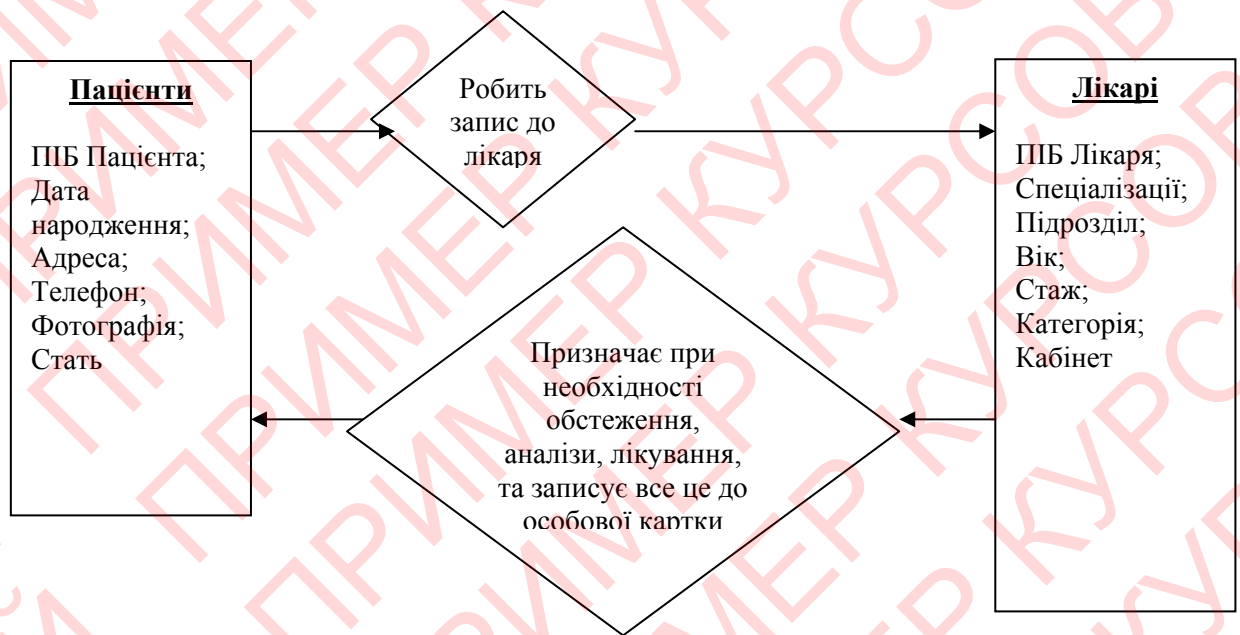
для характеристики зв'язку між різними сутностями треба вказувати так званий клас приналежності, який показує, чи може бути відсутнім зв'язок об'єкту даного класу з яким-небудь об'єктом іншого класу. Клас приналежності ества має бути або обов'язковим, або необов'язковим.

З врахуванням вище перелічених особливостей побудови ІЛМ були розроблені 8 об'єктів даних:

1. Об'єкт **Підрозділи** містить: дані про підрозділи лікарні;
2. Об'єкт **Спеціалізація** містить: дані про спеціалізації лікарні;
3. Об'єкт **Лікарі** включає: дані про лікарів, що працюють в лікарні;
4. Об'єкт **Діагнози** містить: містить дані про загальноприйняті діагнози хвороб;
5. Об'єкт **Тип лікування** містить: дані про стандартні типи лікування;
6. Об'єкт **Пацієнти** містить: дані про пацієнтів, що звертаються до лікарні;
7. Об'єкт **Запис_на_прийом** містить: дані про записи пацієнтів до певних лікарів;
8. Об'єкт **Записи_прийому_пацієнтів** містить особові картки пацієнтів з даними про лікування, та історію хвороби.

На підставі структурної схеми діяльності бібліотеки і описаних вище властивостей об'єктів, будується ІЛМ проектованої БД.

ER-діаграма «Лікарні» змальована на малюнку 2.



Малюнок 2 - ER-діаграма «Лікарні».

Проектування логічної структури бази даних

Завданням наступної стадії проектування системи бази даних є вибір відповідної СУБД і відображення в її середовище (структуру даних) специфікацій інфологічної моделі наочної області. Іншими словами, модель наочної області системи, що розробляється, має бути представлена в термінах моделі даних концептуального рівня вибраної конкретної СУБД. Цю стадію називають логічним або даталогічним проектуванням бази даних, а її результатом є концептуальна схема бази даних, що включає визначення всіх інформаційних елементів і зв'язків, у тому числі завдання типів, характеристик і імен.

Завдання цього етапу - перетворення ER-діаграми в реляційну схему.

Реляційна база даних - база даних, в якій вся інформація, доступна користувачеві, організована у вигляді таблиць, що зазвичай мають унікальні імена, складаються з рядків і стовпців, на пересіченні яких містяться значення даних, а операції над даними зводяться до операцій над цими таблицями.

Основними поняттями, що визначають реляційну модель, є:

Домен - сукупність значень, з яких беруться значення відповідних атрибутів певного відношення (з точки зору програмування - це тип даних, визначуваний системою або користувачем).

Первинний ключ - це стовпець або деяка підмножина стовпців, яка унікальна, тобто єдиним чином визначають рядки.

Зовнішній ключ - це стовпець або підмножина однієї таблиці, який може служити як первинний ключ для іншої таблиці. Зовнішній ключ таблиці є засланням на первинний ключ іншої таблиці.

Всі створювані таблиці повинні відповідати наступним вимогам:

- дані у вічках мають бути структурно неділимими
- дані в одному стовпці мають бути одного типа
- кожен стовпець має бути унікальним
- стовпці розміщуються в довільному порядку
- рядки розміщуються в довільному порядку
- стовпці мають унікальні найменування

Реляційний підхід, в основі якого лежить принцип розділення даних і зв'язків, забезпечує, з одного боку, незалежність даних, а з іншої - простіші способи зберігання і оновлення.

Кожне єство ER-діаграми перетворюється на таблицю, кожна властивість стає атрибутом - стовпцем відповідної таблиці.

Далі необхідно перетворити зв'язки в зовнішні ключі, використовуючи різні зв'язки: «багато до багатьом», «один до багатьом» і так далі. Таблиці можуть знаходитися також в різних формах: першою, другою і третьою.

Розділ 4. Структура програмного комплексу, тексти вихідних програм (модулів, процедур)

Проектування фізичної структури бази даних

Стадія фізичного проектування БД в загальному випадку включає:

- вибір способу організації БД;
- розробку специфікації внутрішньої схеми засобами моделі даних її внутрішнього рівня;
- опис відображення концептуальної схеми у внутрішню.

Етапи роботи:

Проектування структури бази даних.

Запустивши програму MS Access була створена нова база даних «Лікарня.mdb», що описує пацієнтів, лікарів, талони до лікарів, записи до особових карток пацієнтів, відомості про підрозділи, спеціалізації лікарні, типи лікування та діагнози.

База даних складатиметься з восьми таблиць: Підрозділи, Спеціалізація, Лікарі, Тип_лікування, Діагнози, Пацієнти, Запис_на_прийом, Записи_прийому_пацієнтів.

Таблиці слід створювати в режимі конструктора таким чином:

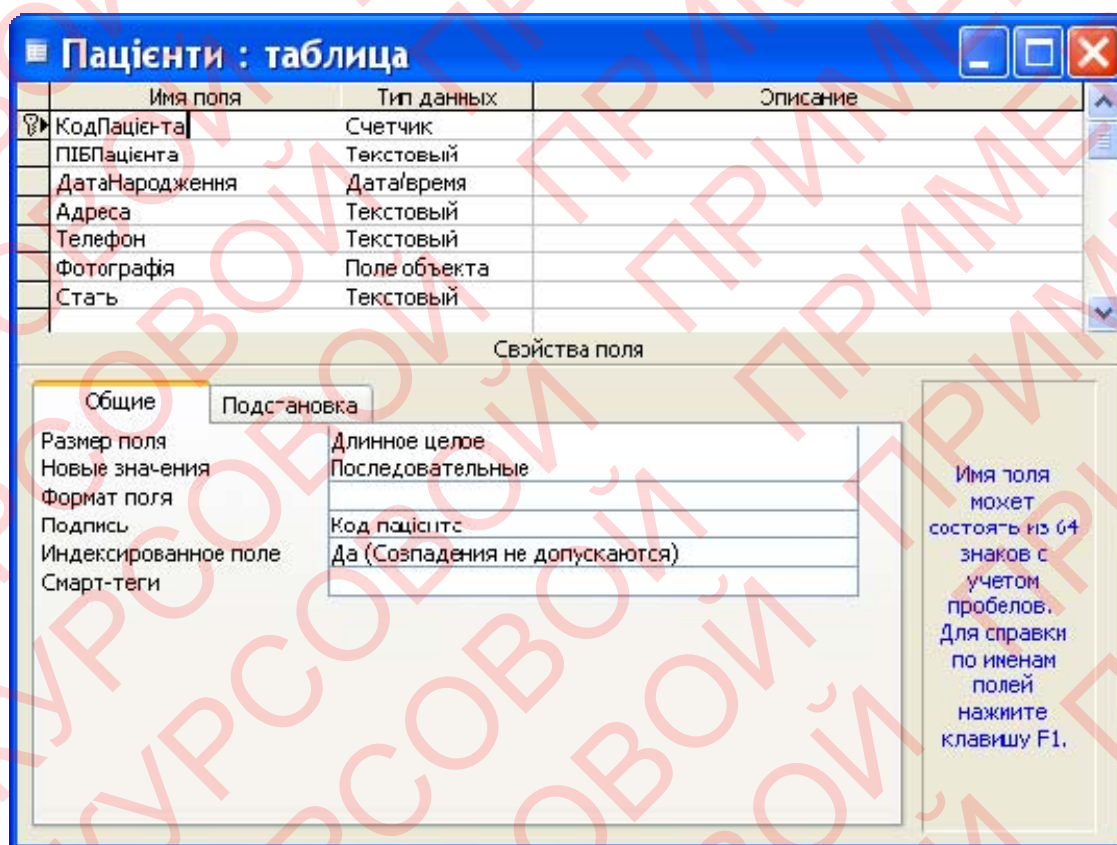
1. Вибрати закладку Таблиця, натискувати Створити
2. У діалоговому вікні, що з'явилося, натискувати Створення таблиці в режимі конструктора

Далі були створені в режимі конструктора наступні таблиці:

- «Пацієнти» з такими полями:

№ п.п	Ім'я	Тип	Розмір	Підпис	Обов'яз-кове	Індексоване
1	КодПацієнта	Счетчик	Длинное целое	Код пацієнта	да	Да (Совпадения не доп-ся)
2	ПІБПацієнта	Текстовый	250	ПІБ пацієнта	да	Да (Совпадения доп-ся)
3	ДатаНародження	Дата/время	Длинный формат даты	Дата народження	да	Да (Совпадения доп-ся)
4	Адреса	Текстовый	250	Адреса	да	Нет
5	Телефон	Текстовый	15	Телефон	да	Нет
6	Фотографія	Поле объекта OLE		Фотографія	нет	нет
7	Стать	Текстовый	50	Стать	да	Да (Совпадения доп-ся)

Ключовое поле - КодПацієнта.



Малюнок 3 – Структура таблиці «Пацієнти»

Існують наступні типи даних:

текстовий - тип даних, використовуваний для зберігання простого неформатованого тексту, число символів в якому не повинне перевищувати 255;

поле МЕМО - спеціальний тип даних, вживаний для зберігання великих об'ємів тексту (до 65 535 символів);

числовий - тип даних для зберігання чисел;

дата/час - тип даних для зберігання значень дати і часу;

грошовий - тип даних для зберігання грошових значень (довжина поля 8 байт);

лічильник - спеціальний тип даних, використовуваний для автоматичної нумерації записів;

логічний - для зберігання логічних даних, які можуть мати одне з двох можливих значень (Так чи ні);

поле об'єкту OLE - спеціальний тип даних, призначений для зберігання об'єктів OLE (електронних таблиць Microsoft Excel, документів Microsoft Word, звукозаписів і ін.);

гіперпосилання - спеціальне поле для зберігання адрес URL Web-об'єктів;

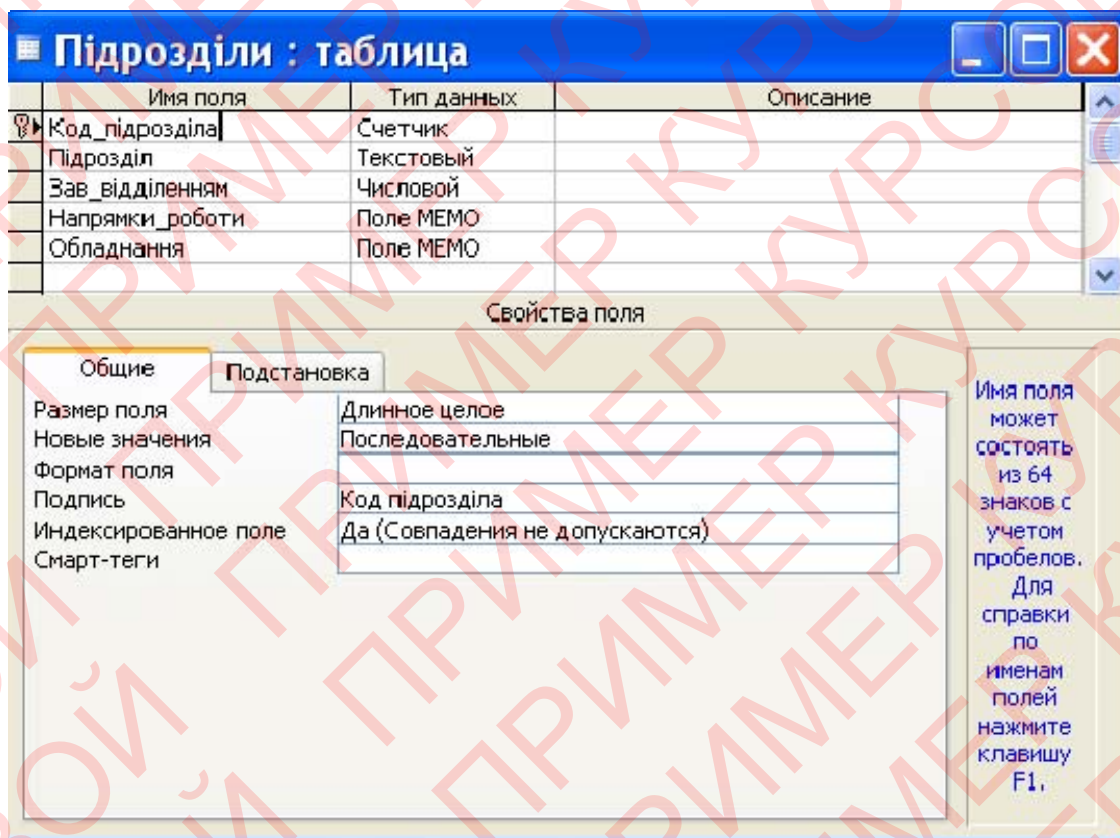
майстер підстановок - тип даних, запускаючи майстер підстановок, що дозволяє вибирати дані із списку, що розкривається, а не вводити їх в полі уручну.

Ключове поле - це поле з унікальними записами. Таблиці зв'язуються по ключах; ключ може складатися з одного або декількох полів.

По наведеному вище прикладу створюються структури останніх сьоми таблиць.

Це побудова порожніх таблиць БД.

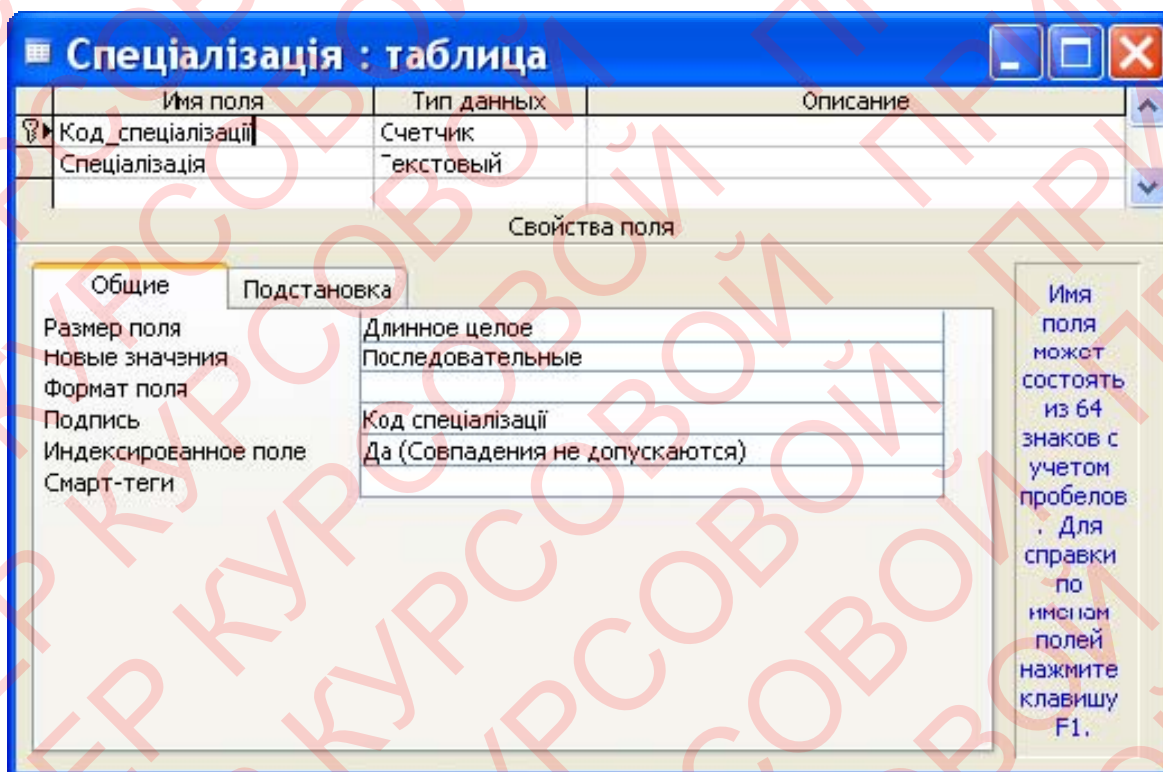
- «Підрозділи» з такими полями:



Малюнок 4 – Структура таблиці «Підрозділи»

Ключевое поле - Код_підрозділа.

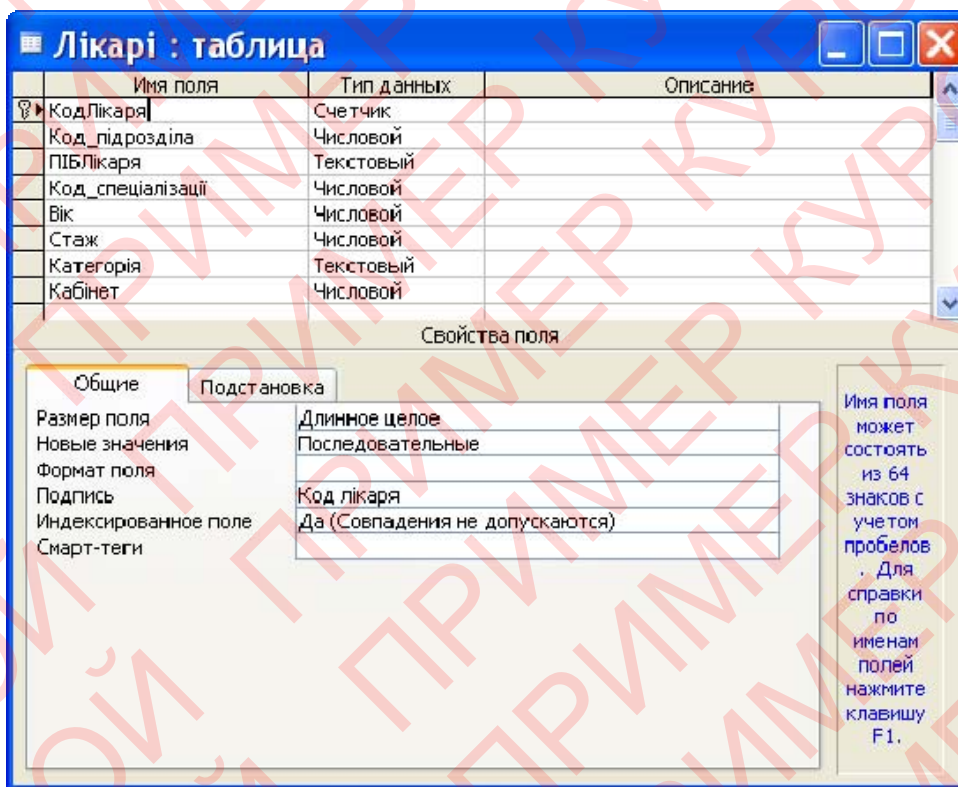
- «Спеціалізація» з такими полями:



Малюнок 5 – Структура таблиці «Спеціалізація»

Ключове поле - Код_спеціалізації.

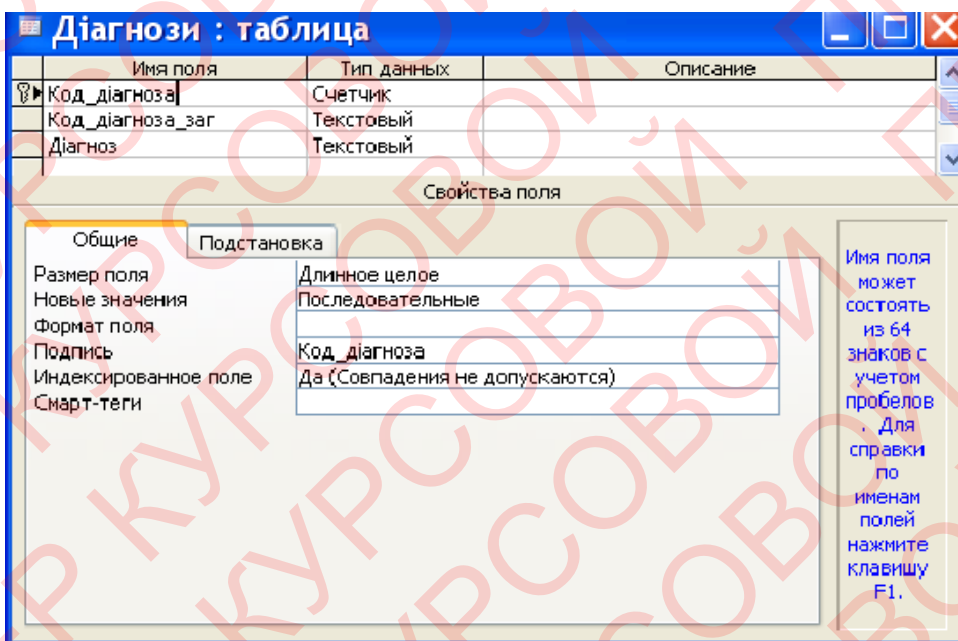
- «Лікарі» з такими полями:



Малюнок 6 – Структура таблиці «Лікарі»

Ключове поле - КодЛікаря. Поля Код_підрозділа, Код_спеціалізації – поля зовнішні ключі, для зв'язку з іншими таблицями.

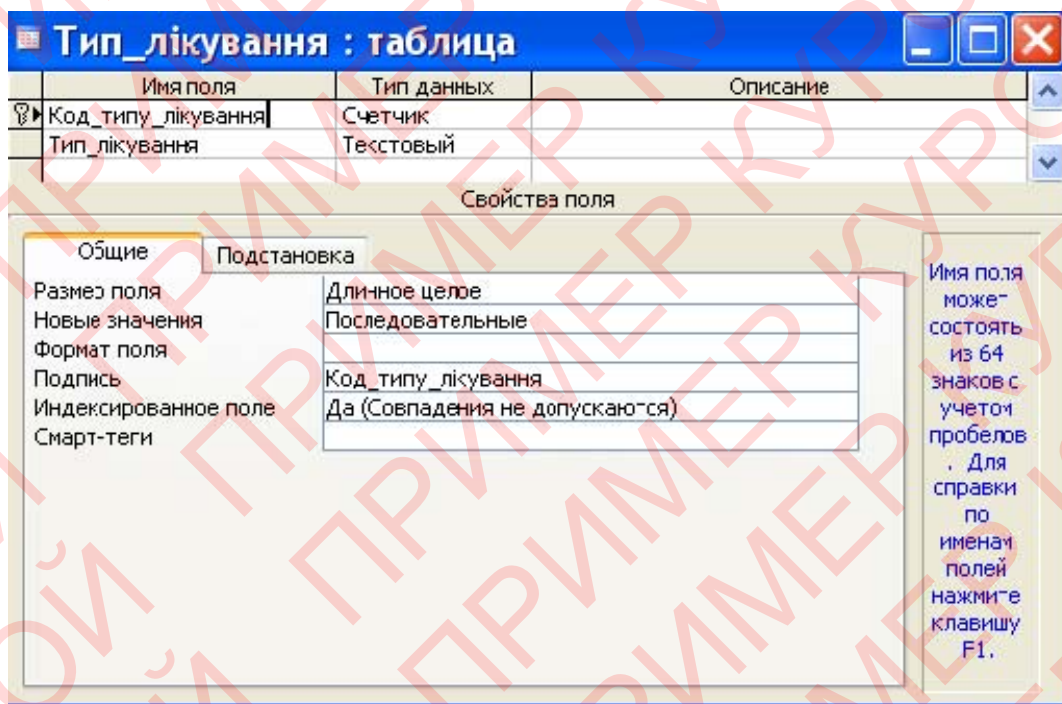
- «Діагнози» з такими полями:



Малюнок 7 – Структура таблиці «Діагнози»

Ключовое поле - Код_диагноза.

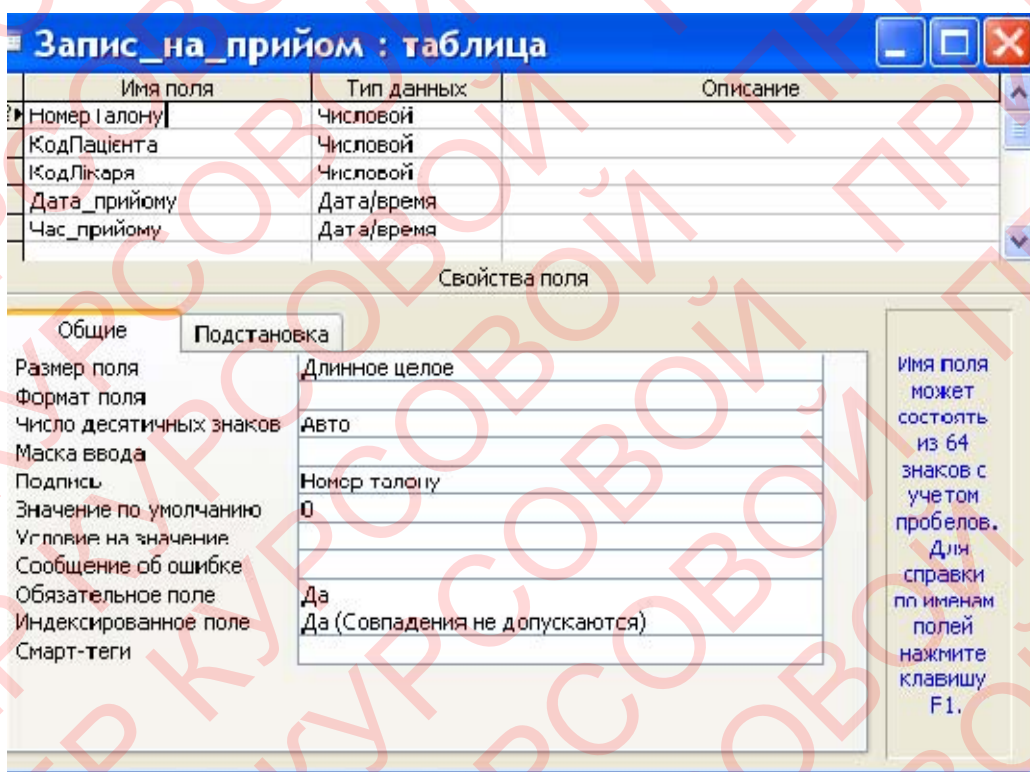
- «Тип_лечения» з такими полями:



Малюнок 8 – Структура таблиці «Тип_лечения»

Ключовое поле - Код_типу_лечения.

- «Запис_на_прием» з такими полями:



Малюнок 9 – Структура таблиці «Запис_на_прием»

Ключове поле - Код_запису_пацієнта. Поля КодПацієнта, КодЛікаря – поля зовнішні ключі, для зв'язку з іншими таблицями.

- «Записи_прийому_пацієнтів» з такими полями:

Имя поля	Тип данных	Описание
Код_запису_пацієнта	Счетчик	
КодПацієнта	Числовой	
КодЛікаря	Числовой	
Дата_прийому	Дата/время	
Час_прийому	Дата/время	
Скарги_пацієнта	Поле MEMO	
Огляд	Поле MEMO	
Діагноз	Числовой	
Необхідність_обстеження	Логический	
Направлення_на_обстежен	Поле MEMO	
Направлення_на_аналізи	Поле MEMO	
Результати_обстеження	Поле MEMO	
Результати_аналізів	Поле MEMO	
Лікування	Поле MEMO	
Код_типу_лікування	Числовой	
Дата_наступного_прийому	Дата/время	

Свойства поля	
Общие	Подстановка
Размер поля	Длинное целое
Новые значения	Последовательные
Формат поля	
Подпись	Код_запису_пацієнта
Индексированное поле	Да (Совпадения не допускаются)
Смарт-теги	

Имя поля может состоять из 64 знаков с учетом пробелов. Для справки по именам полей нажмите клавишу F1.

Малюнок 10 – Структура таблиці «Записи_прийому_пацієнтів»

Ключове поле - НомерТалону. Поля КодПацієнта, КодЛікаря, Діагноз, Код_типу_лікування – поля зовнішні ключі, для зв'язку з іншими таблицями.

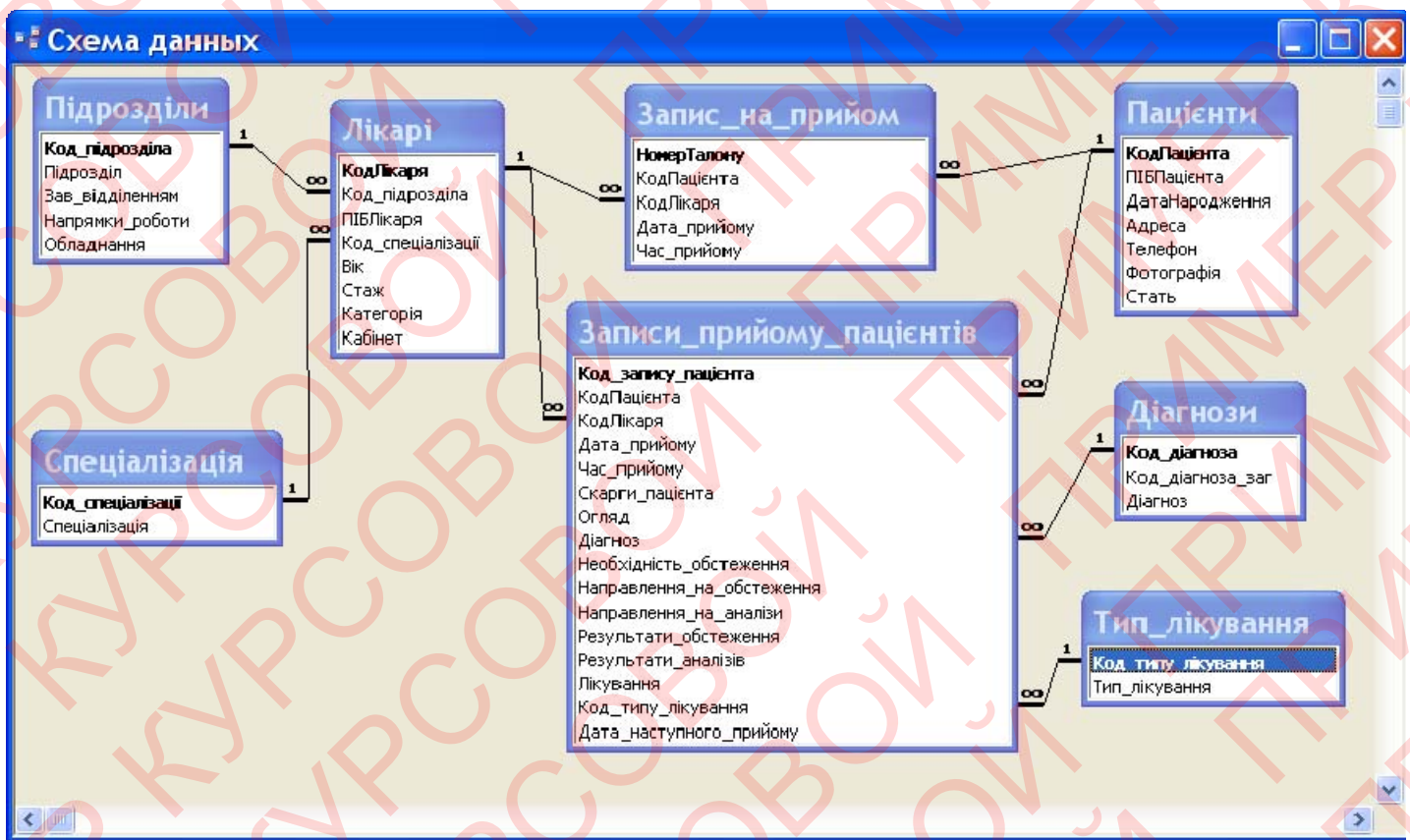
Створення структури бази даних

Для створення структури бази даних необхідно:

- Клацнути по піктограмі - Схема даних. З'явиться вікно Додавання таблиці.
- Виділити таблиці Підрозділи, Спеціалізація, Лікарі, Тип_лікування,

Діагнози, Пацієнти, Запис_на_прийом, Записи_прийому_пацієнтів і клацнути по кнопці Додати. У вікні Схема даних з'явиться умовний вид цих таблиць.

- Поставити мишку на ім'я поля КодЛікаря в таблиці Лікаря і, не відпускаючи кнопку мишки, перетягнути її на полі КодЛікаря в таблиці Запис_на_прийом. Відпустити мишку, і в діалоговому вікні Зв'язку, що з'явилося, включити значок Забезпечення цілісності даних, потім Каскадне оновлення зв'язаних полів і Каскадне видалення зв'язаних полів. Клацнути по кнопці Створити, з'явиться зв'язок один-до-багатьох. По такому ж алгоритму зв'язати решту таблиць так, щоб вийшла схема БД, показана на малюнку 11.



Малюнок 11 - Схема бази даних «Лікарня»

Заповнення таблиць початковими даними

Далі необхідно заповнити всі таблиці початковими даними:

- Заповнити таблицю Підрозділи даними, як показано на малюнку 12.

Підрозділи : таблиця					
Код підрозділа	Підрозділ	Зав відділення	Напрямки роботи	Обладнання	
1	Анестезіологія	9	удосконалення анестезіологічного забезпечення оперативних вт	наркозно-дихальна апаратура. Моніторна станція на 12 пацієнтів	
2	Ортопедія	3	У відділенні лікуються хворі з пошкодженими опорно – рухови	у відділенні триває процес оснащення апаратури і обладнання	
3	Гінекологія	6	Відділення надає допомогу хворим з наступними захворювання	лапароскоп, гістероскоп, апарат УЗД, пертубатор, камери зонал	
4	Ендокринологія	1	Відділення надає допомогу хворим з наступними захворювання	аналізи крові на соматотропний гормон, на гормони щитовидної	
5	Кардіологія	10	Відділення надає допомогу хворим з наступними захворюванн	монітори - 2 шт., дефібрилятори - 2 шт., ЕКГ - 3 шт., велоергоме	
6	Неврологія	7	Відділення надає допомогу хворим з наступними захворювання	окремий фізіотерапевтичний кабінет.	
7	Офтальмологія	4	Відділення надає допомогу хворим з наступними захворюванн	операційні мікроскопи, апарати УЗД, ультразвуковий апарат для	
8	Хірургія	8	Відділення надає допомогу хворим з наступними захворюванн	лапароскоп діагностичний і операційний, операційні столи вироб	
*	(Счетчик)	0			

Запись: 1 из 8

Малюнок 12 - Заповнення таблиці «Підрозділи»

- Заповнити таблицю Спеціалізація даними, як показано на малюнку 13.

Спеціалізація : таблиця		
Код спеціаліза	Спеціалізація	
1	хірург	
2	ЛОР	
3	терапевт	
4	окуліст	
5	проктолог	
6	геніколог	
7	акушер	
8	стоматолог	
9	глав. Врач	
10	анестезіологія	
11	ендокринолог	
12	кардіолог	
*	(Счетчик)	

Запись: 1 из 12

Малюнок 13 - Заповнення таблиці «Спеціалізація»

- Заповнити таблицю Лікарі даними, як показано на малюнку 14.

Лікарі : таблиця

	Код лікаря	Код підрозділа	ПІБ лікаря	Код спеціаліза	Вік	Стаж	Категорія	Кабінет
+	1	1	Дорошенко Максим Леонідович	10	35	10	вища	305
+	2	3	Яненко Максим Петрович	6	27	3	вища	402
+	3	4	Шевченко Борис Іванович	11	45	15	вища	602
+	4	5	Масенко Віталій Вадимович	12	28	4	вища	615
+	5	2	Сороштан Володимир Іванович	1	41	10	середня	102
+	6	3	Бойко Марія Петрівна	6	32	4		316
+	7	7	Зайцева Олена Іванівна	4	36	8	середня	404
+	8	8	Перебийнос Віктор Олександрович	1	38	9	середня	103
+	9	5	Іщенко Вадим Миколаєвич	12	51	25	середня	605
+	10	2	Харітоненко Петро Петрович	1	47	20		104
*	(Счетчик)			0	0	0		0

Запись: 1 из 10

Малюнок 14 - Заповнення таблиці «Лікарі»

- Заповнити таблицю Діагнози даними, як показано на малюнку 15.

Діагнози : таблиця

	Код діагноза	Код діагноза загальний	Діагноз
+	1	E10	Інсулінзалежний цукровий діабет
+	2	E22	Гіперфункція гіпофіза
+	3	G20	Хвороба Паркінсона
+	4	G81	Геміплегія
+	5	I70	Атеросклероз
+	6	I73.0	Синдром Рейно
+	7	I80	Флебіт та тромбофлебіт
+	8	I89.1	Лімфангіт
+	9	M10	Подагра
+	10	M12.1	Хвороба Кашина-Бека
+	11	M91	Юнацький остехондроз стегна та таза
*	(Счетчик)		

Запись: 1 из 11

Малюнок 15 - Заповнення таблиці «Діагнози»

- Заповнити таблицю Тип лікування даними, як показано на малюнку 16.

Тип лікування : таблиця

	Код типу лікування	Тип лікування
+	1	клінічне
+	2	амбулаторне
+	3	операція
*	(Счетчик)	

Запись: 1 из 3

Малюнок 16 - Заповнення таблиці «Тип лікування»

- Заповнити таблицю Пацієнти даними, як показано на малюнку 17.

Пацієнти : таблиця

	Код пацієнта	ПІБ пацієнта	Дата народження	Адреса	Телефон	Фотографія	Стать
+	1	Петров Іван Федорович	12 февраля 1972 г.	вул. Леніна 5	25-69-78	Рисунок	чоловіч
+	2	Кузнецова Марія Петрівна	26 марта 1983 г.	вул. Праці 12	24-65-78	Рисунок	жіноча
+	3	Панасенко Дмитро Васильович	8 июля 1968 г.	вул. Супруна 85	65-89-47	Рисунок	чоловіч
+	4	Ганжа Олена Миколаївна	13 декабря 1986 г.	вул. Могильна 15	64-26-66	Рисунок	жіноча
+	5	Денисенко Віктор Іванович	1 января 1981 г.	вул. Кобзаря 66	66-66-66	Рисунок	чоловіч
+	6	Іванов Роман Леонідович	27 мая 1991 г.	вул. Горького 24	27-68-95	Рисунок	чоловіч
+	7	Сідоров Данило Максимович	29 декабря 1987 г.	вул. Ольшанського 18	28-69-43		чоловіч
+	8	Козуниця Вадим Іванович	15 июля 1975 г.	вул. Кірова 2	22-66-55		чоловіч
+	9	Гарячая Людмила Петрівна	15 февраля 1964 г.	вул. Соборна 12	65-24-89		жіноча
+	10	Самодай Микола Петрович	2 апреля 1942 г.	вул. Кірова 15	65-12-98		чоловіч
*	(Счетчик)						

Запись: 1 из 10

Малюнок 17 - Заповнення таблиці «Пацієнти»

- Заповнити таблицю Запис_на_прийом даними, як показано на малюнку 18.

Запис_на_прийом : таблиця

	Номер талону	Пацієнт	Лікарь	Дата прийому	Час прийому
▶	1	4	1	5 мая 2005 г.	12:00
	2	3	2	1 января 2005 г.	12:45
	4	5	5	15 марта 2005 г.	13:10
	5	6	5	26 августа 2005 г.	11:15
	6	1	9	20 мая 2005 г.	10:50
	7	9	5	15 июля 2004 г.	9:20
	8	4	1	4 марта 2005 г.	10:30
	9	10	8	5 декабря 2004 г.	15:10
	10	7	10	2 мая 2004 г.	8:40
*	0	0	0		

Запись: 1 из 9

Малюнок 18 - Заповнення таблиці «Запис_на_прийом»

- Заповнити таблицю Записи_прийому_пацієнтів даними, як показано на малюнку 19.

Записи_прийому_пацієнтів : таблиця

	Код запису_пд	КодПацієнта	Код лікаря	Дата прийому	Час прийому	Скарги_пацієнт	Огляд	Діагноз	Необхідність о	Направлення_1	Направлення_2	Результати_об	Результати_ан	Лікуванн
▶	5	4		6 2 марта 2009 г.	10:20	біль	все в нормі	4	<input checked="" type="checkbox"/>	пройти спеціал	здати кров			таблетки
*	(Счетчик)													

Запись: 1 из 1

Малюнок 19 - Заповнення таблиці «Записи_прийому_пацієнтів»

Створення форм для введення даних

Для перегляду, введення або зміни даних прямо в таблиці застосовуються форми. Форма дозволяє відібрати дані з однієї або декількох таблиць і вивести їх на екран, використовуючи стандартний або створений користувачем макет. Форми Access дозволяють створювати призначений для користувача інтерфейс для таблиць бази даних. Хоча для виконання тих же самих функцій можна використовувати режим таблиці, форми надають переваги для представлення даних у впорядкованому і привабливому вигляді. Форми Access є багатофункціональними; вони дозволяють виконувати завдання, які не можна виконати в режимі таблиці. Форми дозволяють виробляти перевірку коректності даних, що містяться в таблиці. Access дозволяє створювати форми, що включають інші форми (форма усередині форми називається підлеглою). Форми дозволяють обчислювати значення і виводити на екран результат.

Для того, щоб створити форму, необхідно:

- У вікні бази даних вибрати вкладку Форми, там Створення форми за допомогою майстра
- Вибрати поля, що цікавлять, з таблиці Запис_на_прийом і перенести їх з розряду Доступних в розряд Вибраних полів
- Вибрати зовнішній вигляд таблиці
- Вибрати стиль таблиці (у мене - стандартний)
- Задати ім'я форми і Готово

Таким чином, одержали форму Запис_на_прийом.

Номер талону	Пацієнт	Лікар	Дата прийому	Час прийому
1	Ганжа Олена Миколаївна	Дорошенко Максим Леонідович	5 мая 2005 г.	12:00
2	Панасенко Дмитро Васильович	Яненко Максим Петрович	1 января 2005 г.	12:45
4	Денисенко Віктор Іванович	Сероштан Владимир Іванович	15 марта 2005 г.	1:10
5	Іванов Роман Леонідович	Сероштан Владимир Іванович	26 августа 2005 г.	11:15
6	Петров Іван Федорович	Іщенко Вадим Миколаєвич	20 мая 2005 г.	13:50
7	Гаряча Людмила Петрівна	Сероштан Владимир Іванович	15 июля 2004 г.	9:20
8	Ганжа Олена Миколаївна	Дорошенко Максим Леонідович	4 марта 2005 г.	13:30
9	Самодай Микола Петрович	Перебійнос Віктор Олександрович	5 декабря 2004 г.	3:10
10	Сідоров Данило Максимович	Харітоненко Петро Петрович	2 мая 2004 г.	8:40

Малюнок 20 - Форма Запис_на_прийом

У даній формі використовується два списки.

Перший список з даними з таблиці «Пацієнти», які вибираються по SQL-запиту:

```
SELECT Пацієнти.КодПацієнта, Пацієнти.ПІБПацієнта
FROM Пацієнти ORDER BY [КодПацієнта], [ПІБПацієнта];
```

Другий список з даними з таблиці «Лікарі», які вибираються по SQL-запиту:

```
SELECT Лікарі.КодЛікаря, Лікарі.ПІБЛікаря, Лікарі.Спеціалізація
FROM Лікарі ORDER BY [КодЛікаря], [ПІБЛікаря];
```

Аналогічним образом одержали форму Записи_прийому_пацієнтів.

У даній формі використовується чотири списки.

Перший список з даними з таблиці «Пацієнти», які вибираються по SQL-запиту:

```
SELECT Пацієнти.КодПацієнта, Пацієнти.ПІБПацієнта
FROM Пацієнти ORDER BY [ПІБПацієнта];
```

Другий список з даними з таблиці «Лікарі», які вибираються по SQL-запиту:

```
SELECT Лікарі.КодЛікаря, Лікарі.ПІБЛікаря
FROM Лікарі ORDER BY [ПІБЛікаря];
```

Третій список з даними з таблиці «Діагнози», які вибираються по SQL-запиту:

```
SELECT Діагнози.Код_діагноза, Діагнози.Діагноз  
FROM Діагнози ORDER BY [Діагноз];
```

Четвертий список з даними з таблиці «Тип_лікування», які вибираються по SQL-запиту:

```
SELECT Тип_лікування.Код_типу_лікування,  
Тип_лікування.Тип_лікування  
FROM Тип_лікування ORDER BY [Тип_лікування];
```

The screenshot shows a software window titled "Записи_прийому_пацієнтів" (Patient Appointment Records). The form contains the following fields and sections:

- Код запису пацієнта**: [Empty]
- Дата прийому**: 12 марта 2009 г.
- Час прийому**: 10:20
- ПІБ Пацієнта**: Ганжа Олена Миколаївна
- ПІБ Лікаря**: Бойко Марія Петрівна
- Скарги пацієнта**: біль
- Огляд**: все в нормі
- Діагноз**: Геміплегія
- Необхідність обстеження**:
- Направлення на обстеження**: пройти спеціальне обстеження
- Направлення на аналізи**: здали кров
- Результати обстеження**: [Empty]
- Результати аналізів**: [Empty]
- Лікування**: таблетки
- Тип лікування**: амбулаторне
- Дата наступного прийому**: 16 марта 2009 г.

At the bottom, there is a status bar: "Запись: 1 из 1" with navigation icons.

Малюнок 21 - Форма Записи_прийому_пацієнтів

Форма Підрозділи має наступний вигляд:

Підрозділи

Код підрозділа

Підрозділ
Анестезіологія

Зав. відділенням: Іщенко Вадим Миколаєвич

Напрямки роботи
удосконалення анестезіологічного забезпечення оперативних втручань, оптимізація лікувально-діагностичного процесу при проведенні інтенсивної терапії та реанімаційних заходів у відділенні.

Обладнання
наркозно-дихальна апаратура. Моніторна станція на 12 пацієнтів сірми "Сіменс", Апаратура для екстракорпоральної детоксикації, центрифуги для плазмаферезу, інфузомати.

Запись: 1 из 8

Малюнок 22 – Форма Підрозділи

Форма Спеціалізація має наступний вигляд:

Спеціалізація

Код спеціалізації	Спеціалізація
1	хірург
2	ЛОР
3	терапевт
4	окуліст
5	проктолог
6	геніколог
7	акушер
8	стоматолог
9	глав. Врчч
10	анестезіологія
11	ендокринолог
12	кардіолог
*	[Счетчик]

Запись: 1 из 12

Малюнок 23 – Форма Спеціалізація

Форма Лікарі має наступний вигляд:

Код лікаря	Підрозділ	
	Анестезіологія	
ПІБ лікаря		
Дорошенко Максим Леонідович		
Вік	Стаж	Категорія
35	10	вища
Спеціалізація		
анестезіологія		
Кабинет		
305		

Запись: 1 из 10

Малюнок 24 – Форма Лікарі

У даній формі використовується два списки.

Перший список з даними з таблиці «Підрозділи», які вибираються по SQL-запиту:

```
SELECT Підрозділи.Код_підрозділа, Підрозділи.Підрозділ  
FROM Підрозділи ORDER BY [Підрозділ];
```

Другий список з даними з таблиці «Спеціалізація», які вибираються по SQL-запиту:

```
SELECT Спеціалізація.Код_спеціалізації, Спеціалізація.Спеціалізація  
FROM Спеціалізація ORDER BY [Спеціалізація];
```

Форма Пацієнти має наступний вигляд:

Пацієнти

Код пацієнта

ПІБ пацієнта
Денисенко Віктор Іванович

Дата народження
1 января 1981 г.

Адреса
вул. Кобзаря 66

Телефон
66-66-66

Стать
Чоловіча

Фотографія

Запись: 5 из 10

Малюнок 25 – Форма Пацієнти

Форма Діагнози має наступний вигляд:

Діагнози

Код_діагноза	Код_діагноза_загальний
E10	

Діагноз
Інсулінзалежний цукровий діабет

Запись: 1 из 11

Малюнок 26 – Форма Діагнози

Форма Тип_лікування має наступний вигляд:

Тип_лікування

Код_типу_лікування
1

Тип лікування
клі пічче

Запись: 1 из 3

Малюнок 27 – Форма Тип_лікування

Потім створили головну форму бази даних «Лікарня», яка має три вкладки:

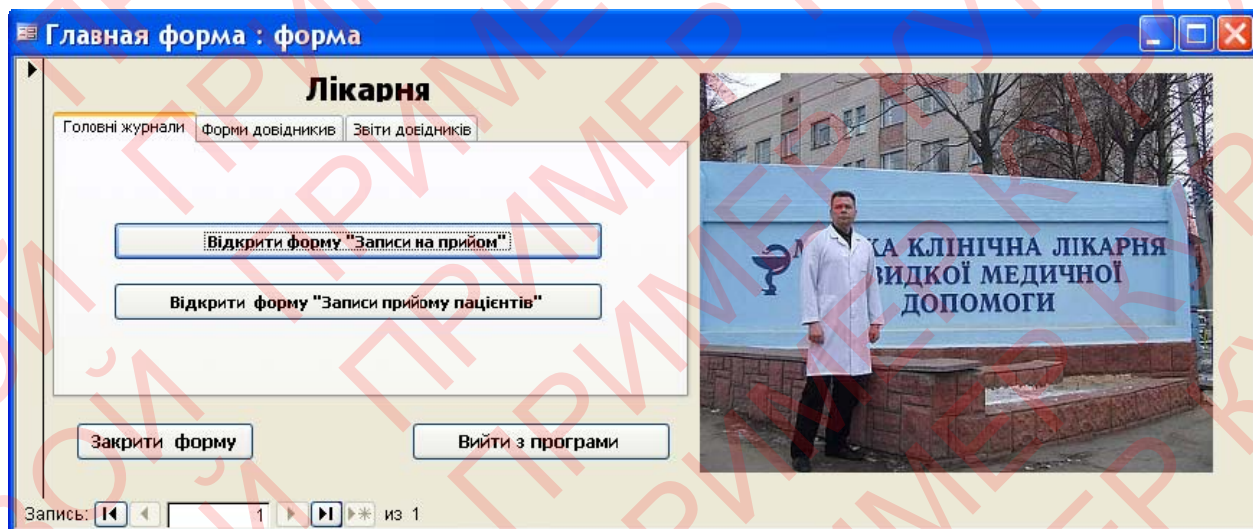
на першій - кнопка по відкриттю форми «Записи на прийом» і кнопки по відкриттю форми «Записи прийому пацієнтів»;

на другій - кнопки по перегляду форм довідників;

на третій - кнопки по перегляду звітів довідників.

А також кнопки закриття головної форми і виходу з програми.

Ця головна форма буде відкриватися відразу після запуску бази даних «Полуклініка» - для цього в головному меню вибрали «Сервис \ Параметры запуска», далі у відкритому вікні в полі «Вывод формы» вказали «Главная форма». Ця форма має такий вигляд:



Малюнок 28 Головна форма

Створення запитів

Для послідуочого створення звітів були створені допоміжні запити.

- Запрос1-підрозділи – вибирає усі дані про підрозділи лікарні, створений на вкладці запити в режимі SQL і має такий код:

```
SELECT Підрозділи.Код_підрозділа, Підрозділи.Підрозділ, Лікарі.ПІБЛікаря,  
Підрозділи.Напрямки_роботи, Підрозділи.Обладнання  
FROM Підрозділи INNER JOIN Лікарі ON Підрозділи.Код_підрозділа =  
Лікарі.Код_підрозділа;
```

- Запрос2-лікарі – вибирає усі дані про лікарів, створений на вкладці запити в режимі SQL і має такий код:

```
SELECT Лікарі.КодЛікаря, Підрозділи.Підрозділ, Лікарі.ПІБЛікаря,  
Спеціалізація.Спеціалізація, Лікарі.Вік, Лікарі.Стаж, Лікарі.Категорія,  
Лікарі.Кабінет  
FROM Спеціалізація INNER JOIN (Підрозділи INNER JOIN Лікарі ON  
Підрозділи.Код_підрозділа = Лікарі.Код_підрозділа) ON  
Спеціалізація.Код_спеціалізації = Лікарі.Код_спеціалізації;
```

- Запрос3-записи_на_прийом – вибирає усі дані про підрозділи лікарні, створений на вкладці запити в режимі SQL і має такий код:

```
SELECT Запис_на_прийом.НомерТалону, Пацієнти.ПІБПацієнта,  
Лікарі.ПІБЛікаря, Спеціалізація.Спеціалізація, Підрозділи.Підрозділ,  
Запис_на_прийом.Дата_прийому, Запис_на_прийом.Час_прийому  
FROM Спеціалізація INNER JOIN (Підрозділи INNER JOIN (Пацієнти INNER  
JOIN (Лікарі INNER JOIN Запис_на_прийом ON Лікарі.КодЛікаря =  
Запис_на_прийом.КодЛікаря) ON Пацієнти.КодПацієнта =  
Запис_на_прийом.КодПацієнта) ON Підрозділи.Код_підрозділа =  
Лікарі.Код_підрозділа) ON Спеціалізація.Код_спеціалізації =  
Лікарі.Код_спеціалізації;
```

- Запрос4-Записи_прийому_пацієнтів – вибирає усі дані про підрозділи лікарні, створений на вкладці запити в режимі SQL і має такий код:

SELECT Записи_прийому_пацієнтів.Код_запису_пацієнта,
Пацієнти.ПІБПацієнта, Лікарі.ПІБЛікаря, Підрозділи.Підрозділ,
Спеціалізація.Спеціалізація, Записи_прийому_пацієнтів.Дата_прийому,
Записи_прийому_пацієнтів.Час_прийому,
Записи_прийому_пацієнтів.Скарги_пацієнта,
Записи_прийому_пацієнтів.Огляд, Діагнози.Діагноз,
Записи_прийому_пацієнтів.Необхідність_обстеження,
Записи_прийому_пацієнтів.Направлення_на_обстеження,
Записи_прийому_пацієнтів.Направлення_на_аналізи,
Записи_прийому_пацієнтів.Результати_обстеження,
Записи_прийому_пацієнтів.Результати_аналізів,
Записи_прийому_пацієнтів.Лікування, Тип_лікування.Тип_лікування,
Записи_прийому_пацієнтів.Дата_наступного_прийому
FROM Тип_лікування INNER JOIN (Спеціалізація INNER JOIN (Підрозділи
INNER JOIN (Пацієнти INNER JOIN (Лікарі INNER JOIN (Діагнози INNER
JOIN Записи_прийому_пацієнтів ON Діагнози.Код_діагноза =
Записи_прийому_пацієнтів.Діагноз) ON Лікарі.КодЛікаря =
Записи_прийому_пацієнтів.КодЛікаря) ON Пацієнти.КодПацієнта =
Записи_прийому_пацієнтів.КодПацієнта) ON Підрозділи.Код_підрозділа =
Лікарі.Код_підрозділа) ON Спеціалізація.Код_спеціалізації =
Лікарі.Код_спеціалізації) ON Тип_лікування.Код_типу_лікування =
Записи_прийому_пацієнтів.Код_типу_лікування;

Створення для всіх таблиць і запитів звіти

Кінцевим продуктом більшості додатків баз даних є звіт. У Access звіт є спеціальним типом безперервних форм, призначених для друку. Для створення звіту, який можна роздрукувати і розподілити між споживачами, Access комбінує дані в таблицях, запитах і навіть формах. Роздрукована версія форми може служити звітом.

Створювані Access звіти діляться на такі основні типи:

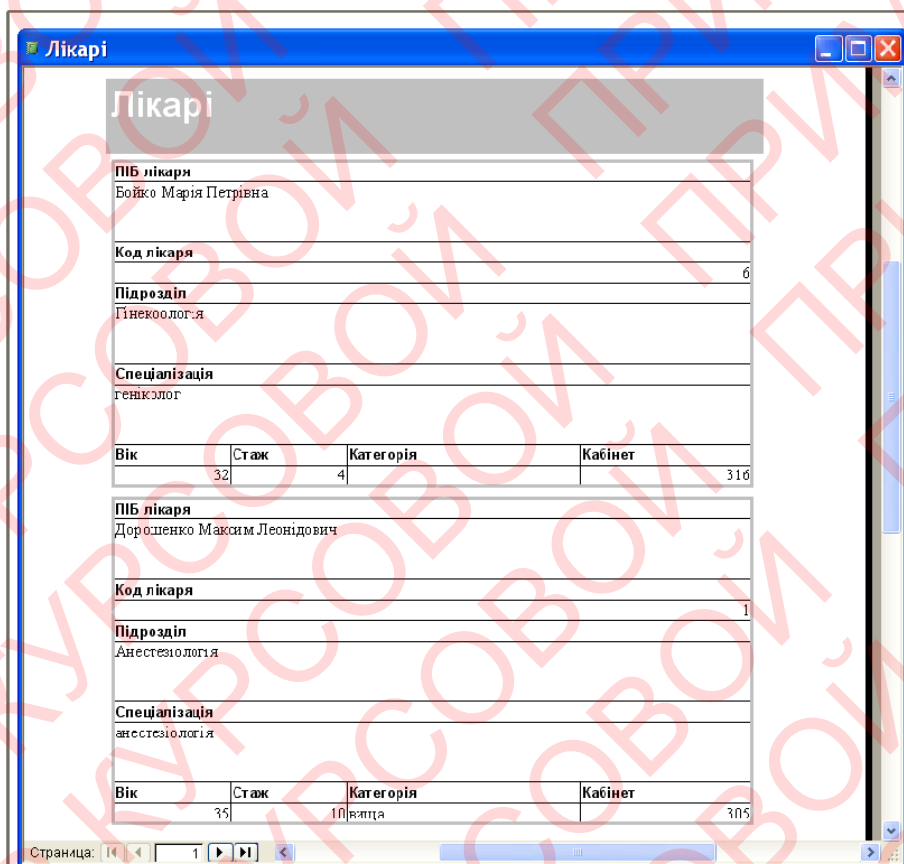
1. Звіти в одну колонку є одним довгим стовпцем тексту, що містить значення всіх полів кожного запису таблиці або запиту. Напис указує ім'я, а праворуч від неї вказується значення поля. Новий засіб Access автозвіт дозволяє створити звіт в одну колонку клацанням по кнопці панелі інструментів автозвіт. Звіти в одну колонку використовуються рідко, оскільки такий формат представлення даних приводить до зайвої витрати паперу.
2. У стрічкових звітах для кожного поля таблиці або запиту виділяється стовпець, а значення всіх полів кожного запису виводяться по строчках, кожне в своєму стовпці. Якщо в записі більше полів, ніж може поміститися на сторінці, то додаткові сторінки друкуватимуться до тих пір, поки не будуть виведені всі дані; потім починається друк наступної групи записів.
3. Многоколоночні звіти створюються із звітів в одну колонку при використуванні колонок "газетного" типу або колонок "змійкою", як це робиться в настільних видавничих системах і текстових процесорах. Інформація, яка не поміщається в першому стовпці, переноситься в початок другого стовпця, і так далі. Формат многоколоночних таблиць дозволяє заощадити частину паперу, але застосуємо не у всіх випадках, оскільки вирівнювання стовпців навряд чи відповідає очікуванням користувача.

В основному звіти найпростіші побудувати за допомогою Майстра звітів. Майстер звітів прагне створити оптимальний варіант остаточного звіту з першої спроби. Звичайно майстер достатньою мірою наближається до закінченого варіанту, так що витратиться набагато менше часу на редагування базового звіту майстра, чим пішло б на створення звіту з незаповненого шаблону.

Створення стрічкового автозвіту на підставі таблиці Лікарі за допомогою Майстра.

Після запуску Майстра необхідно вибрати Таблиця: Лікарі в спливаючому списку; натиснути Далі; додати всі Доступні поля у Вибрані поля; натиснути Далі; натиснути Далі двічі; зробити сортування по Назві; натиснути Далі; вибрати макет (Табличний) і орієнтацію (Книжкова); натиснути Далі; вибрати стиль (Діловий); натиснути Далі і Готово.

Створений звіт показаний на малюнку 29.



The screenshot shows a window titled "Лікарі" (Doctors) with a table of data. The table has columns for "ПІБ лікаря" (Full Name), "Код лікаря" (Code), "Підрозділ" (Department), "Спеціалізація" (Specialization), "Вік" (Age), "Стаж" (Experience), "Категорія" (Category), and "Кабінет" (Cabinet). The data is sorted by name.

ПІБ лікаря	Код лікаря	Підрозділ	Спеціалізація	Вік	Стаж	Категорія	Кабінет
Бойко Марія Петрівна	6	Гінеколог-я	Геншклог	32	4		310
Дорошенко Максим Леонідович	1	Анестезіологія	анестезіологія	35	10	злпца	305

Малюнок 29 – Звіт по лікарям

Таким же образом складаються і решта звітів.

Розділ 5. Описання роботи створеного програмного продукту

База даних «Лікарня» проста у використанні і розрахована на багато користувачів для організації обліку прийому пацієнтів, обстежень і висновків від приватної лікарської практики одного лікаря, до лікувально-діагностичних центрів.

Вона дозволяє працювати з однією базою даних з декількох комп'ютерів підключених до однієї локальної мережі.

Інтерфейс програми інтуїтивно зрозумілий навіть початкуючому користувачеві.

Сфера застосування:

- приватна лікарська практика
- лікувально-діагностичні центри.

Можливості:

- Реєстратура, облік прийому пацієнтів.
- Збереження результатів обстежень і висновків у вигляді реєстраційних карток позмінно для кожного лікаря або групи лікарів під будь-які види лікарської практики.
- Здобуття статистичних даних про захворювання пацієнтів будь-якої складності у вигляді створення спеціальних звітів.
- Необмежена кількість діагнозів захворювань.
- Необмежена кількість ознак прийому пацієнтів.

Результати роботи програмного продукту

Спроектowana і створена база даних «Лікарня» має інтуїтивно зрозумілий інтерфейс, дозволяє працювати з однією базою даних з декількох комп'ютерів підключених до однієї локальної мережі.

Також вона дозволяє вести облік підрозділів та спеціалізації лікарні, облік прийому пацієнтів, відомості про пацієнтів, дані про лікарів, стандартизовані типи лікування та загальноприйняті діагнози захворювань, а також журнали запису на прийом та особові картки пацієнтів, в яких зберігаються усі результати обстежень і висновків у вигляді реєстраційних карток позмінно для кожного лікаря або групи лікарів під будь-які види лікарської практики.

Висновки

В ході виконання роботи я ознайомилася з основними поняттями, використовуваними при створенні баз даних, а також прийомами проектування і механізмами створення баз даних в СУБД Access. Пройшовши всі етапи складання бази даних, починаючи із складання ER-діаграми і закінчуючи складанням звітів, я навчилася проектувати і створювати бази даних, що поза сумнівом згодиться в подальшій практичній діяльності.

Список використаної літератури:

1. Фабричев В. А. Основи інформатики. Система управління базою даних Access. — К.: НАУ, 2003. — 132с.
2. Феддема Елен. Access 2002. — СПб.: Питер, 2003. — 940с.
3. Чаповська Р. Б., Жмуркевич А. Є. Работа с базами данных MICROSOFT ACCESS 2000. — К.: ЦУЛ, 2003. — 323с.
4. Дубнов П. Ю. Access 2002. — М.: ДМК Пресс, 2004. — 313с.