Проектування і створення інформаційної пошукової системи для автоматизації роботи в аптеці

Зміст.
1. Вступ З
2. Аналіз предметної області та структура інформаційно-пошукової системи.....4
3. Забезпечення цілісності посилань в Базі Даних...................................................7
4. Забезпечення обробки даних...........................................................................................9
5. Організація введення даних...........................................................................................11
6. Формування вихідної документації......................................................................16
7. Розробка інтерфейсу................................................................................................17
8. Висновок..............................................................................................................................19
9. Список використаної літератури................................................................................20
10. Додатки.

1. Вступ.

В наш час жодну велику проблему не можна вирішити без обробки значного об'єму інформації і налагоджених комунікаційних процесів. Фірми, підприємства, організації створюють свої основні стратегічні переваги за допомогою інформаційних технологій, унікальних знань експертів та кваліфікації персоналу.
Світ інформаційних ресурсів продовжує стрімко зростати. З метою вдосконалення своєї інтелектуальної діяльності людині необхідно орієнтуватися в сучасному інформаційному просторі. Для цього необхідно знати можливості комп'ютерних систем та інформаційних технологій, вміти користуватися інформаційним багатством сучасного суспільства.
Сучасні інформаційні технології дозволяють задовольнити потреби як окремої людини так і суспільства в цілому. Сьогодні інформаційні системи є основою розвитку економіки, освіти, практично всіх галузей народного господарства. Людина створює собі платформу майбутнього існування через таке поняття як інформаційна культура, котра починається з розуміння природи та властивостей інформації, знань способів її формалізації з метою комп'ютерної обробки. Саме інформаційний потенціал нації визначає її соціально-економічне положення в світовому суспільстві.
Роль інформації стала фундаментальною для цивілізації загалом, як основним напрямком удосконалення управлінської діяльності є створення та застосування інформаційних систем, які базуються на сучасних апаратно-програмних засобах, інформаційних технологіях, розділеній обробці даних у мережах, на використанні економіко-математичних методів і моделей, систем підтримки прийняття рішень та експертних систем.
Ядром сучасних інформаційних систем, що використовуються на підприємствах, фірмах, організаціях тощо є бази даних, які дозволяють створювати на своїй основі прості і складні інформаційно-пошукові системи для автоматизації найрізноманітніших рутинних операцій, чим сприяють зменшенню витрат часу на прийняття управлінських рішень, замін матеріальних потоків інформаційними, що дуже важливо в сучасних економічних умовах.
Метою даної розрахунково-графічної роботи є формування практичних навиків і вмінь по виконанню комплексу робіт з проектування інформаційно-пошукових систем для автоматизації виконання завдань в аптеці. Освоєння запропонованого комплексу робіт і поглиблене вивчення пов'язаних із ними теоретичних питань будуть сприяти успішному виконанню кожним спеціалістом своїх майбутніх функціональних обов'язків на різних рівнях управління організації. Крім того, самостійне та вдумливе виконання всіх складових розрахункової роботи дозволить майбутньому спеціалісту визначитися щодо ролі сучасних інформаційних систем і технологій в управлінні організації.

2. Аналіз предметної області та структура інформаційно-пошукової системи.

Невід'ємним чинником якісної роботи відділу з продажу медикаментів є якісне і бездоганне обслуговування клієнтів.
Якщо клієнт здійснив купівлю того чи іншого препарату, то дані необхідно занести до реєстраційного журналу, щоб у разі появи певної інформації, що безпосередньо стосується покупця повідомити його про постачальника. Такі дані заносяться до реєстраційного журналу "Дані про постачальника”( Рис 2.1. ).
№ Назва постачальника Адреса постачальника Місто Поштовий індекс Номер телефону Номер факсу Умови оплати Електронна пошта
1 „Рівне-Ліки” Соборна,4 Рівне 33000 72-11-11 72-12-11 передоплата rliki@med.rv.ua
Рис. 2.1. Реєстраційний журнал "Дані про пстачальника".
Дані про продаж ліків клієнту заноситься до іншого журналу "Продаж ліків"(Рис.2.2. ).
№ Код ліків Назва ліків Назва типу ліків Одиниця виміру Опис ліків
1 1 Гель антицилюлітний косметика мг Дуже дієві
Рис. 2.2. Реєстраційний журнал "Продаж ліків".
Звідси маємо, що замовлення у клієнта прийняв певний співробітник, а отже заніс дані до журналів, про які було сказано раніше. Тому відомості про співробітників також необхідно зберігати у певному документі, наприклад, у журналі „Дані про співробітників” (Рис. 2.3.).
№ Код співробітника ПІП співробітника Дата народження Дата найму Сімейний стан Домашній телефон Серія і номер паспорта Ідентифікційний код
1 1 Онищук В.В. 19.09.1984 30.10.2005 неодружена 2-55-91 СР №159357 987654321
Рис. 2.3. Реєстраційний журнал " Дані про співробітників ".
У зв'язку з достатньо великим рухом товарів та обсягом документації аналіз роботи організації завантажений, тому доцільно реалізувати на ПК автоматизовану систему аналізу роботи фірми з продажу ліків.
Оскільки завдання полягає в організації зберігання та обробки певної сукупності інформації у вигляді бази даних і доступу до них. А , враховуючи специфіку задачі, всі дані предметної області можна розбити на два класи: початкові та результуючі. Слід зазначити, що в кожному класі існує ще своя класифікація даних.
Отже, до початкових даних можна віднести:
1. Дані про клієнта.
2. Відомості про продаж ліків.
3. Відомості про постачальника та співробітників фірми.
4. Відомості про ліки та інша інформація, яка буде додаватись і
уточнюватись в процесі розробки інформаційно-пошукової системи.
В якості результуючих даних розв'язуваної задачі повинні виступати відповіді на різноманітні запити користувача, а також звіти про діяльність фірми за певний період. Наприклад, на запит про співробітника повинна видаватись інформація, що стосується: домашньої адреси, номера телефону тощо. Результати пошуку за кожним запитом повинні обов'язково відображатись на екрані монітора. При складанні звітів обов'язково повинні аналізуватися всі дані, які відповідають заданому проміжку часу, проводитися розрахунки..
Початкові дані відповідають різним типам даних, але групуються в окремі інформаційні об'єкти, які зберігаються в таблицях бази даних, які в Access є фізичним вмістилищем даних файла бази даних.
Таблиці - основні об'єкти бази даних, в яких зберігаються дані. Реляційна база даних ( база даних, яка містить взаємозв'язані таблиці, що мають спільні поля) може мати багато взаємозв'язаних таблиць.
Існують такі способи створення таблиць;
Режим таблицы. Видає бланк (форму) абстрактної таблиці. Потім вона може приймати конкретне наповнення і зміст. В режимі таблиці додаються, редагуються або переглядаються табличні дані. Також можна перевірити орфографію і надрукувати табличні дані, фільтрувати і відсортовувати записи, змінювати зовнішній вигляд таблиці або змінювати структуру таблиці, додаючи або знищуючи поля.
Конструктор. Створення таблиці в режимі, коли вводиться ім'я поля, тип даних, властивості поля.
Мастер таблиц. Робота основана на використанні заготовки таблиць, з яких користувач вибирає поле для своєї нової таблиці. Звичайно не передбачувані заготовки полегшують роботу користувача, але вони не вирішують всіх життєвих випадків.
Импорт таблиц. Цей спосіб дозволяє імпортувати дані у таблицю, яка створюється, з інших файлів, у тому числі і з файлів інших форматів даних, які відрізняються від формату, який використовується в СКБД Access.
Связь с таблицами. Цей спосіб дозволяє шляхом встановлення зв'язків з файлом іншого формату інтегрувати дві інформаційні системи.
В зв'язку з тим, що зберігання, обмін, обробка інформації на магнітних носіях (у вигляді БД) не завжди може бути раціональною в тому вигляді, як вона зберігається на папері, тому пропонується структуру БД розробити наступним чином.
Решта інформаційних об'єктів аналогічним чином мають бути трансформовані в таблиці бази даних..
В розрахунковій роботі для створення всіх таблиць використовується "Конструктор". За допомогою нього створюємо таблицю про постачальника, що містить відомості за допомогою, яких можна знайти його для повідомлення певної інформації. Розглянуті вище структури таблиць, відображають тільки необхідні властивості полів. Проте кожне поле в свою чергу може містити ще вісім властивостей, кожна з яких полегшує роботу непрофесійному. В конструкторі таблиця має наступний вигляд (Рис. 2.4.).

Рис. 2.4. Конструктор таблиці „Постачальник”.
Оскільки ліки, що продаються в аптеці постачаються, то досить важливо мати інформацію про постачальника. В яких буде міститися інформація про нього, що дозволить менеджеру з продажу передати замовлення на купівлю автомобіля (телефоном, факсом, електронною поштою), перераховувати кошти за поставку на розрахунковий рахунок. В конструкторі і в режимі таблиці ця таблиця має наступний вигляд.

Рис 2.5. Конструктор таблиці „Аптека”
В цьому конструкторі містяться дані про аптеку в залом. Ця таблиця є головною в аптеці.

Рис 2.6. Конструктор таблиці „Співробітники” .
Ця таблиця містить інформацію про працівника, що здійснює продаж ліків в аптеці. До цих відомостей відносяться особисті дані про працівника і загальні. До особистих відносяться дані про місце проживання, дом.телефон, сімейний стан, до загальних паспортні дані та ідентифікаційний код.

3. Забезпечення цілісності посилань в Базі Даних.

У реальному світі між об'єктами будь-якої предметної області існує велика кількість різноманітних зв'язків, що втілюються у функціональні залежності її інформаційними об'єктами.
Наприклад, покупці купують ліки, причому один покупець може купувати декілька препаратів; співробітники обслуговують клієнтів і отримують за це робітну плату. При цьому кожен з них може обслуговувати багато клієнтів і отримувати зарплату багато разів. Тому реально в базі даних такі функціональні залежності реалізовуються через зв'язки між таблицями, які бувають наступних типів:
- один до одного;
- один до багатьох;
- багато до одного;
- багато до багатьох.
Зв'язок один до одного означає, що кожному запису таблиці А відповідає лише один запис таблиці Б. Зв'язок один до багатьох означає, що одному запису таблиці А може відповідати кілька записів таблиці Б. Зв'язок багато до багатьох означає, що кожний запис таблиці А може відповідати зв'язаний з багатьма записами в таблиці А. Зв'язок багато до одного означає, що декільком записам таблиці А може відповідати лише один запис таблиці Б.
В системі Access зв'язки між таблицями встановлюються за допомогою графічного інтерфейсу "Схеми даних", що виводиться на екран за допомогою відповідної кнопки панелі інструментів (див. Рис 3.1.).

Для того, щоб встановити зв'язки між таблицями необхідно встановити зв'язки між полями в яких міститься загальна інформація. Як правило, зв'язок встановлюється з'єднанням ключових полів таблиць: первинного ключа в одній таблиці і зовнішнього ключа в іншій. Первинні ключі створюються для унікального ідентифікатора кожного запису. Якщо поле зв'язку в другій таблиці не являється первинним ключем, то його називають зовнішнім ключем.
На Рис.3.1. показаний зв’язок таблиці Аптека з таблицею Ліки. Тип зв’язку: один до одного. Зв’язок таблиці Аптека з таблицею Постачальник. Тип зв’язку: один до одного. Зв’язок таблиці Ліки з таблицею Тип ліків. Тип зв’язку: один до одного. В таблиці Ліки зберігаються дані про аптеку, при знищенні даних з цієї таблиці автоматично знищуються дані про ліки і про постачальника з таблиці Ліки та Постачальник.
В Access можна встановлювати не тільки зв'язки між таблицями, але і для забезпечення цілісності даних. Більшість заданих правил визначаються оновленими відношеннями. Крім того, з допомогою відношень можна організувати каскадну зміну і видалення даних.
Встановлення каскадної зміни забезпечує наступне: при зміні значення первинного ключа в таблиці "один", зміна поширюється на всі таблиці "багато". Встановлення каскадного видалення забезпечує наступне: при видаленні icy з таблиці "один" видаляються всі записи з таблиці "багато". Це може як перевагою так і недоліком. При видаленні інформації про клієнта в мі каскадного видалення видаляються всі його рахунки в таблиці "багато". о такий режим не встановлений, видалення записів в таблиці "один" не дозволяється до тих пір, поки в таблиці "багато" не будуть видалені всі зв'язані записи, щоб в ній не залишалось записів з невизначеним значенням зовнішнього ключа.

4. Забезпечення обробки даних.

Під якісною обробкою бази даних, як правило, розуміють можливості пошуку та обробки різних відомостей з бази даних. Під час роботи будь-якої установи потрібно мати певну інформацію про той чи інший об'єкт предметної області. Щоб отримати такі дані, треба витрачати багато часу на пошук їх по таблицях баз даних. Крім того база може містити і конфіденційну інформацію. Для цього у система Access призначені запити. Для однієї і тієї ж таблиці можна створити багато різних запитів, кожний з яких може вибрати із таблиці тільки малу частину інформації, але саме ту частину, яка у дану мить є необхідною.
В результаті роботи запиту із загальної вихідної бази формується
результуюча таблиця, яка містить частину загальної інформації, що відповідає
запиту. Важливою властивістю є те, що при створенні результуючої таблиці
можна не тільки вибирати інформацію із бази, але і опрацьовувати її. При роботі
запиту дані можуть упорядковуватися (сортування), фільтруватися
(відсіюватися), об'єднуватися, розділятися, змінюватися і при цьому ніяких
змін в базових таблицях може не відбуватися.
Результати обробки відбиваються тільки на змісті результуючої таблиці, а вона має тимчасовий характер, і іноді її називають миттєвим знімком.
І ще однією важливою властивістю запитів є їх здатність виконувати підсумкові обчислення. Запит може видавати не тільки результуючу таблицю, але і знайти, наприклад, середнє (найбільше, найменше, сумарне) значення по якому-небудь полю.
Для створення запитів до баз даних існує спеціальна мова запитів. Вона називається SQL ( Structured Query Language - структурована мова запитів). Але для тих, хто користується СУБД Access, замість неї є простий засіб, який називається бланком запиту за зразком. З його допомогою можна сформувати запит простими прийомами, переміщуючи елементи запиту між вікнами.
Існують різні види запитів, але найпростіші з них - це запити на вибір. Вони використовуються найбільш часто. Запит на вибір повертає дані із однієї кількох таблиць, а також результати, які при бажанні може змінити користувач (з деякими обмеженнями). Також запит на вибір можна використовувати, щоб згрупувати записи для обчислення сум, середніх значень, перерахунку та інших дій.
В багатьох випадках користувачеві необхідно надавати можливість вибору того, що він хоче знайти в таблицях бази даних. Для цього існує спеціальний вид запиту - запит з параметрами.
Створення запиту з параметрами із запрошенням на введення умов відбору вимагає виконання таких дій:
- створити запит на вибір або перехресний запит;
- в режимі конструктора запиту перемістити за допомогою миші поля із
списку полів у бланк запиту;
- для кожного поля, яке буде використане як параметр, ввести в комірку
рядка Условие отбора текст запрошення, обмежений квадратними
дужками. Це запрошення буде виводитись під час запуску запиту. Текст
підказки повинен відрізнятися від імені поля, але може включати його;
- для перегляду результатів натиснути кнопку Вид на панелі інструментів і
ввести значення параметру. Для повернення в режим конструктора запиту
знову натиснути на кнопці Вид на панелі інструментів.
Існує ще і особлива група запитів, які називаються запитами на зміну. Вони дозволяють автоматично створювати нові таблиці або змінювати ті, які вже є. Логіка використання запитів на зміну така:
- створюється запит на вибір, який відбирає дані із різних таблиць або сам
створює нові дані шляхом обчислень;
- після запуску запиту утворюється тимчасова результуюча таблиця;
- дані із цієї результуючої таблиці використовуються для створення нових
таблиць або зміни існуючих.
В перехресному запиті відображаються результати статистичних розрахунків (такі як суми, кількість записів і середні значення) виконаних за даними із одного поля таблиці. Ці результати групуються за двома наборами даних, один з яких розміщений у лівому стовпчику таблиці, а другий - у верхньому рядку. Даний розділ описує створення таких запитів:
- Запит Залишки;
- Запит Ліки;
- Запит Одвиміру;
- Запит Постачальник;
- Зарпит Тип ліків;
- Запит Співробітники.
„Запит Залишки” створений на основі таблиць "Аптека.
SELECT Аптека.\*, [Аптека]![Прийнято]-[Аптека]![Продано]-[Аптека]![Списано]-[Аптека]![Повернуто] AS Залишок
FROM Аптека
WHERE ((([Аптека]![Прийнято]-[Аптека]![Продано]-[Аптека]![Списано]-[Аптека]![Повернуто])>0));

„Запит Ліки” створений на основі таблиць „Ліки”
SELECT Ліки.НазваЛіків, Ліки.КодЛіків
FROM Ліки
ORDER BY Ліки.НазваЛіків;

„Запит Одвиміру” створений на основі таблиць „Тара”
SELECT Тара.Одвимір, Тара.КодОдвимір
FROM Тара
ORDER BY Тара.Одвимір;

„Запит Постачальник” створений на основі таблиць „Постачальник”
SELECT Постачальники.НазваПостачальника, Постачальники.КодПостачальника
FROM Постачальники
ORDER BY Постачальники.НазваПостачальника;

„Зарпит Тип ліків” створений на основі таблиць „Тип ліків”
SELECT ТипиЛіків.НазваТипу, ТипиЛіків.КодТипу
FROM ТипиЛіків
ORDER BY ТипиЛіків.НазваТипу;

„Запит Співробітники” створений на основі таблиць „Співробітники”
SELECT Співробітники.\*, Співробітники.[ПІП співробітники]
FROM Співробітники;

5. Організація введення даних.

При введенні даних безпосередньо у таблиці можуть виникнути певні незручності - не всі поля одночасно відображені на екрані тощо. Часто дані необхідно розташовувати у такому вигляді, в якому зберігаються відповідні документи. Система Access дозволяє розташовувати на екрані дані так, як того бажає користувач, тобто у вигляді різноманітних форм. Форма - це своєрідний електронний бланк, в якому наявні поля для введення даних. Форма може використовуватись також для пошуку даних.
Існує три методи створення форми:
1. Знаходячись у вікні бази даних можна вибрати команду Вставка- Форма.
2. Знаходячись у вікні бази даних можна вибрати вкладку Форми і клацнути по кнопці Створити.
3. Знаходячись у вікні бази даних в режимі таблиці або запиту потрібно клацнути по кнопці Новий обєкт, який розташований на панелі інструментів і вибрати опцію Нова форма.
Який би спосіб створення форми не був би обраний, на екрані з'явиться {діалогове вікно Нова форма. Перед тим як про довити роботу потрібно вибрати ім'я таблиці чи запиту зі списку поля "Виберіть в якості джерела даних таблицю а§о запит:". Діалогове вікно Нова форма пропонує сім способів створення форми:
Конструктор відображає абсолютно порожню форму у вікні конструювання форми.
Майстер форм створює форму одного з трьох стандартних типів (в стовпчик, стрічкову або табличну) на основі відповідей користувача на поставлені майстром питання.
Автоформа: в стовпчик автоматично створює форму з полями в один або декілька стовпчиків.
Автоформа '.стрічкова автоматично створює стрічкову форму.
Автоформа: таблична автоматично створює табличну форму.
Діаграма створює форму з діаграми. Звідна таблиця створює зведену таблицю для Excel. У даній роботі форми створюються Конструктором і за допомогою Майстра форм.
Структуру форми складають її розділи, а розділи містять елементи управління.
На панелі інструментів існує спеціальний елемент управління для створення заголовків, який називається Надпись. Натиснувши на ньому, а потім на формі, отримаємо текстову рамку, в яку можна вводити довільний текст. При введенні тексту не потрібно турбуватися про його форматування. Немає значення, який він має вигляд і де розміщений. Закінчивши введення, потрібно натиснути клавішу Enter, після чого можна приступати до оформлення тексту.
Для форматування об'єкта управління його необхідно спочатку виділити. | Для цього служить інструмент Выбор объектов.
Коли об'єкт уже виділений, можна змінювати параметри шрифту, метод І вирівнювання тексту і інші елементи форматування. Це виконують звичайними ! засобами форматування, які є доступними через відповідну панель інструментів Access.
Якщо натиснути на виділеному елементі правою кнопкою миші, відкриється його контекстне меню, в якому є додаткові можливості зміни оформлення, наприклад, примінивши оформлення з тінню.
Щоб елементи управління розміщувались у формі рівними рядками, існують спеціальні команди вирівнювання. Спочатку потрібно виділити групу елементів управління з допомогою інструмента Выбор объектов (група вибирається при натиснутій клавіші Shift ), а потім дати команду Формат -Выровнять і вибрати метод вирівнювання: по левому краю, по правому краю, по верхньому краю, по нижнему краю, по узлам сетки.
Крім розглянутих раніше існують ще декілька корисних елементів
управління:
1. Перемикачі. З ними можна зв'язати команди, наприклад ті, які виконують фільтрацію.
2. Прапорці. Діють аналогічно до перемикачів, але на відміну від них, допускають множинний вибір. Зручні для управління режими сортування даних.
3. Список. Може містити фіксований набір значень або значення із заданого поля однієї з таблиць. Дозволяє не вводити дані, а вибирати їх із списку.
4. Поле зі списком. Використовується так само, як і список, яке займає менше місця у формі, оскільки список відкривається тільки після натискання на відкриваючій кнопці.
5. Командні кнопки. З кожною з них можна зв'язати яку-небудь корисну команду, наприклад, команду пошуку запису, переходу між записами та інші.
6. Вкладки. Дозволяють розмістити багато інформації на обмеженій площині. На вкладках розміщують інші елементи управління.
7. Поле об'єкта OLE. Служить для розміщення зовнішнього об'єкта, який відповідає прийнятій у Windows концепції зв'язування і впровадження об'єктів. Об'єктом, як правило, являється ілюстрація,
наприклад фотографія, але це може бути і відеозапис, і музичний фрагмент, і звукове повідомлення.
Даний розділ описує створення наступних форм:
- Відпуск ліків;
- Залишки;
- Тип ліків;
- Отримання ліків;
- Повна база;
- Тара;
- Повернення;
- Постачальник;
- Списання;
- Лікувальні засоби;
- Старт.
Всі ці форми створені за допомогою Майстра форм з наступним коригуванням в режимі конструктора.
Форма „Випуск ліків” виводить інформацію про назву постачальника, назву ліків, вхідна ціна і кількість проданих препаратів. (Рис. 5.1.)

Слідуюча форма виводить відповідну інформацію про номер накладної, вхідну і вихідну ціну ліку, назва постачальника, назва ліків і залишок. Дана форма називається „Залишок” (Рис. 5.2.)
6. Формування вихідної документації.

В процесі організації роботи будь-якого закладу завжди постає питання звітності на паперових носіях, які утворюються, як правило, на основі певних статистичних підсумків і т.д. для виконання таких завдань в Access призначені
Вони багато в чому схожі на форми і теж дозволяють отримати результати роботи запитів в наочній формі, але не на екрані, а у роздрукованому на принтері вигляді. Таким чином, в результаті роботи звіту створюється паперовий документ.
Більша частина того, що було сказано про форми, відноситься до звітів. Вибравши в діалоговому вікні Бази даних вкладку Отчеты і натиснувши кнопку Создать, ми отримуємо діалогове вікно Новый отчет, яке дозволяє створити звіт автоматично (автозвіт), з допомогою майстра і без допомоги майстра.
Автозвіт являється засобом створення звіту, в якому виводяться всі поля і
записи із базової таблиці або запиту. В діалоговому вікні Новый отчет
необхідно вибрати один із наступних майстрів:
Автоотчет: в столбец - кожне поле утворює окремий рядок із заголовком зліва;
Автоотчет: ленточный - поля кожного запису утворюють окремий рядок; заголовки друкуються зверху на кожній сторінці.
Потім вибирається таблиця або запит, що містять дані, які виводяться у звіті і натискується кнопка ОК.
Для створення звіту з допомогою звіту в діалоговому вікні Новый отчет потрібно вибрати майстра, а потім - ім'я таблиці або запиту, що містять дані, за якими будується звіт і натиснути кнопку ОК.
Якщо створений майстром звіт потребує внесення змін, то це можна зробити в режимі конструктора звіту.
Щоб створити звіт без допомоги майстра в діалоговому вікні Новый отчет потрібно вибрати пункт Конструктор, Потім вибирається ім'я таблиці або
иту, що містять дані, на яких базується звіт і натискається кнопка ОК. Якщо ібно створити звіт, ім'я таблиці чи запиту із списку не вибирається.
Для того, щоб використати у звіті дані із кількох таблиць, звіт юрюється на основі запиту.
Як і форма, звіти складаються із розділів, а розділи можуть містити Іиементи управління. Але, на відміну від форм, розділів у звітах більше, а (лементів управління, навпаки менше.
Структура звіту складається з п'яти розділів:
Розділ заголовку служить для друку загального заголовку звіту.
Розділ верхнього колонтитулу можна використовувати для друку підзаголовків, якщо звіт має складну структуру і займає багато сторінок. Тут ножна також поміщати і колонцифри (номер сторінок), якщо це не зроблено в нижньому колонтитулі.
В області даних розміщують елементи управління, пов'язані із змістом
полів таблиць бази. У ці елементи управління видаються дані із таблиць для
друку на принтері. Порядок розміщення і вирівнювання елементів управління
такий самий, як і при створенні структури форм.
Розділ нижнього колонтитула використовують з тією ж метою, що і розділ верхнього колонтитулу.
Розділ примітки використовують для розміщення додаткової інформації.
Найзручніший спосіб створення звітів - за допомогою майстра, але з наступним коригуванням в режимі конструктора.
Отже, розглянемо звіт, що відображає продаж ліків в аптеці, схема створення якого наступна:
- запускається майстер звітів;
- вибирається головна таблиця і необхідні поля з неї;
- обирається підлеглий запит, в якому міститься вже згрупована інформація
по кількості проданих препаратів;
- задаються рівні групування і сортування;
- проводяться необхідні підрахунки статистичних показників;
підбирається стиль оформлення звіту.
В даному розділі розглядається створення наступних звітів:
- Звіт по аптеці;
- Звіт по залишкам;
- Звіт по постачальникам;
- Звіт по співробітниках;
- Звіт по лікам.
Всі звіти створені з використанням Майстра звітів з подальшим коригуванням в Конструкторі. ( Див. Додатки

7. Розробка інтерфейсу.

При розробці будь-якого програмного продукту виникає потреба організації інтерфейсу між користувачем та програмою. Створення меню, підтримка системи підказок тощо - це обов'язковий атрибут будь-якого розробленого додатку.
Меню можна визначити як своєрідний список можливих функцій програми, оформлений таким чином, що користувач без труднощів може вибрати будь-яку з них.
Для реалізації інтерфейсу в Access можливо використати систему випадаючих меню, кнопочні форми, панелі інструментів і контекстні меню.
В розрахунковій роботі система управління базою даних створена на основі кнопочної форми, що створюються в режимі форм за допомогою майстрів генерації кнопок, має наступну ієрархічну структуру: при запуску на екрані з'являється вікно головного меню (Рис 7.1.), в якому розміщені кнопки: „Відпуск ліків”, „Отримання ліків”, „Повернення ” , „Списання”, „Залишки”, „Повна база”, „Постачальники”, „Лікувальні засоби”, „Тип ліків”, „Тара”, „Про програму”, „Вихід” .

Якщо натиснути на кнопку "Відпуск ліків", то з'явиться вікно однойменна форма (Рис. 5.1.) і так само на будь-якій кнопці буде з’являтися відповідна форма.
Під час натиснення кнопки „Про програму” появиться форма яка вказує хто виконав дану роботу. (Рис.7.2.)

ис.7.2. Форма про програму
По повному закінченню конструювання інформаційно-пошукової системи натискаємо кнопку „Вихід” і робота з програмою буде негайно припинена.

8. Висновок.

Важливого значення для молодого спеціаліста в сфері управління набуває використання ним інформаційних систем в управлінні виробництвом. Оскільки досить часто при веденні управлінської діяльності постає потреба у створенні різноманітної довідкової інформації за різними критеріями; організації зручної і безпечної системи ведення інформаційно-пошукової системи (внесення нової інформації, редагування існуючої, видалення неактуальних даних); забезпеченні цілісності бази даних і взаємозв'язків в ній; отримання вихідної документації та створення простої і зрозумілої системи керування інформаційно-пошуковою системою.
В зв'язку з цим виконання розрахункової роботи по проектуванню інформаційно-пошукової системи стане прикладом найпростішої інформаційної системи в управлінні виробництвом.
Оскільки ведення управлінської діяльності вимагає постійного контакту з різноманітною інформацією, то необхідно забезпечити гнучке і надійне введення, зберігання і оброблення даних. Тому структуру бази даних було ретельно розроблено. Початкові дані були згруповані в окремі інформаційні об'єкти, які містяться в таблицях бази даних, що є основним джерелом їх збереження. Тому ретельне формування таблиць скоротило можливість дублювання вхідних інформаційних потоків, а створення зв'язків між ними забезпечило цілісність даних у базі.
Проте слід відмітити той факт, що введення даних у таблиці дуже стомлююче заняття, вже після кількох годин роботи люди роблять помилки. Вводити дані в форму простіше. Форма - це своєрідний електронний бланк, в якому наявні поля для введення даних. Форма може використовуватись також для пошуку даних. В формах можна багато чого автоматизувати, до того ж , елементи управління форм настроюють таким чином, щоб при введенні даних до них виконувалась їх первинна перевірка. Застосування форм зменшує можливість внесення помилкових даних в базу та забезпечує швидкість відбору необхідних масивів даних, що значно полегшує роботу працівникам. Формування більш наочнішої та інформативнішої вихідної документації можливе за рахунок використання звітів, що дозволяють отримати результати роботи запитів в наочній формі - у роздрукованому на принтері вигляді. Таким чином, в результаті роботи звіту створюється паперовий документ, що відображає підсумки роботи підприємства.

11. Список використаної літератури.
1. М.И. Семенов, И.Т. Трубилин, В.И. Лайко, Т.П.Барановська. „Автоматизирование информационные технологии в экономике”: Учебник – М., 2000.
2. І.В. Гордієнко „Інформаційні системи в менеджменті”: Навчально- методичний посібник для самостійного вивчення дисципліни. – К.: КНЕУ 1999. – 128с.
3. В.Ф. Ситник, Т. А. Писаревська, Н.В. Єрьоміна, О.С. Краєва „Основи інформаційних систем”: Навчальний посібник. – К.:КНЕУ, 1993.-240с.
4. В.Ф. Ситник, О.С. Краєва , „Технологія автоматизованої обробки економічної інформації” – К.:КНЕУ, 1999.-224с.