

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ЧЕРКАСЬКИЙ ДЕРЖАВНИЙ ТЕХНОЛОГІЧНИЙ УНІВЕРСИТЕТ  
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ І СИСТЕМ

Нечипоренко О. В.  
Корпань Я. В.

## **МЕТОДИЧНІ РЕКОМЕНДАЦІЇ**

### **ДО ЛАБОРАТОРНИХ РОБІТ З ДИСЦИПЛІНИ «СИСТЕМИ УПРАВЛІННЯ БАЗАМИ ДАНИХ»**

для здобувачів освітнього ступеня бакалавра  
зі спеціальності 151 «Автоматизація та комп'ютерно-інтегровані  
технології»  
денної форми навчання

Черкаси



2017

УДК 004.65(07)

Н59

*Затверджено на засіданні кафедри спеціалізованих комп'ютерних систем, протокол № 10 від 19.04.2017 р. та Вченою радою ФІТІС протокол № 10 від 22.05.2017 р.*

Автори:

Нечипоренко Ольга Володимирівна, к.т.н., доцент;  
Корпань Ярослав Васильович, к.т.н., доцент.

Рецензент

Уткіна Т.Ю., к.т.н., доцент.

Н 59 Нечипоренко О.В. Методичні рекомендації до лабораторних робіт з дисципліни «Системи управління базами даних» для здобувачів освітнього ступеня бакалавра зі спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології» денної форми навчання [Електронний ресурс] / О.В. Нечипоренко, Я.В. Корпань; М-во освіти і науки України, Черкас. держ. технол. ун-т. – Черкаси: ЧДТУ, 2017. – 62 с.

Викладено матеріали навчально-методичного характеру, наведено завдання з дисципліни «Системи управління базами даних», подано вимоги щодо їх виконання, тематика робіт та перелік рекомендованої літератури.

Для студентів освітнього ступеня бакалавра спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології» денної форми навчання.

УДК 004.65(07)

Виробничо-практичне  
електронне видання комбінованого використання

Нечипоренко Ольга Володимирівна,  
Корпань Ярослав Васильович

МЕТОДИЧНІ РЕКОМЕНДАЦІЇ  
до лабораторних робіт  
з дисципліни «Системи управління базами даних»  
для здобувачів освітнього ступеня бакалавра  
зі спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології»

*В авторській редакції*

© Нечипоренко О. В., Корпань Я.В., 2017

## ЗМІСТ

ВСТУП.....	4
Лабораторна робота №1 - <i>Знайомство з інтерфейсом СУБД VISUAL FOXPRO, створення та ведення бази даних</i> .....	6
Лабораторна робота №2 - <i>Створення таблиць і обробка існуючої інформації в інструментальному середовищі VISUAL FOXPRO</i> .....	24
Лабораторна робота №3 - <i>Створення проекту, побудова запитів та форм в інструментальному середовищі VISUAL FOXPRO</i> .....	31
Лабораторна робота №4 - <i>Створення звітів і етикеток в інструментальному середовищі VISUAL FOXPRO</i> .....	38
СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ .....	43
ДОДАТКИ.....	44
Додаток А – <i>Індивідуальні завдання до лабораторної роботи №2</i> .....	44
Додаток Б – <i>Індивідуальні завдання до лабораторної роботи №3</i> .....	50
Додаток В – <i>Індивідуальні завдання до лабораторної роботи №4</i> .....	57

## ВСТУП

В сучасному світі об'єми інформації, що потребують зберігання, збільшуються посекундно. Тому використання баз даних (БД) є невід'ємною частиною професійної діяльності людини.

Найпростіші БД можна створювати, не використовуючи спеціальні програмні засоби: наприклад, в електронних таблицях Excel. Але існує декілька відмінностей між ними. По-перше, системи управління базами даних (СУБД) розробляються з метою забезпечення ефективної обробки великих об'ємів інформації, набагато більших, ніж ті, з як справляються електронні таблиці. По-друге, СУБД може легко зв'язувати дві таблиці так, що для користувача вони будуть представлятися одною таблицею. Реалізувати таку можливість в електронних таблицях практично неможливо. І по-третє, СУБД мінімізує загальний об'єм бази даних. Для цієї ж мети, таблиці, що містять дані, які повторюються, розділяються на декілька пов'язаних між собою таблиць.

Лабораторний практикум є складовою частиною методичного забезпечення дисципліни «Системи управління базами даних» і призначений для студентів, що навчаються за спеціальністю 151 «Автоматизація та комп'ютерно-інтегровані технології» денної форми навчання.

До його складу входять лабораторні роботи для вивчення системи управління базами даних Visual FoxPro. Кожна лабораторна робота присвячена певній темі і побудована за єдиним сценарієм.

При підготовці до лабораторних робіт студенти повинні вивчити методичні вказівки до їх виконання, відповідну рекомендовану літературу у відповідності до теми завдання та дати повні відповіді на наведені контрольні запитання. Глибоке вивчення теоретичного матеріалу допоможе студентам успішно виконати роботу.

Завдання до лабораторних виконуються за варіантами. Кожне завдання пов'язане зі створенням певної частини бази даних. Для частини лабораторних робіт використовується одна і та ж предметна область, а результати попередньої роботи використовуються як база для наступних. Необхідно також мати на увазі, що лабораторні роботи йдуть у порядку зростання складності, і у кожній наступній роботі використовуються знання та навички, отримані у попередніх роботах.

Результати навчання. Згідно з вимогами освітньо-професійної програми студенти повинні знати:

- концепції моделювання предметного середовища;
- організацію реляційних баз даних;
- методи забезпечення, контролю та відновлення цілісності даних;
- фізичні та логічні методи захисту даних;
- системи автоматизованого проектування.

Згідно з вимогами освітньо-професійної програми студенти повинні уміти:

– розробляти та реалізовувати логічну та фізичну моделі представлення даних за допомогою спеціалізованих CASE-засобів в умовах автоматизованого робочого місця;

– розробляти запити на роботу з даними (введення, модифікація, вилучення, пошук та вибірка) з використанням спеціалізованих інструментальних засобів для програмування баз даних

– оптимізувати плани виконання запитів та виконувати базові функції адміністрування СУБД (управління ролями, користувачами та правилами на роботу з даними тощо) за допомогою стандартних компонентів СУБД та спеціалізованих CASE-засобів.

У результаті виконання описаного у практикумі циклу лабораторних робіт студенти повинні систематизувати і закріпити теоретичні і практичні знання в області організації, проектування і створення БД та придбати навички практичної роботи в СУБД Visual FoxPro.

В кінці лабораторного практикуму наведені варіанти завдань лабораторних робіт. Варіанти завдань співпадають з нумерацією студентів у журналі академічної групи.

## Лабораторна робота № 1

**Тема:** Знайомство з інтерфейсом СУБД VISUAL FOXPRO, створення та ведення бази даних.

**Мета роботи:** набути практичних навичок створення, ведення та використання реляційної бази даних.

### Теоретичні відомості

Одне з важливих вимог до БД - забезпечення швидкої вибірки інформації, що в ній зберігається. СУБД забезпечують високу швидкість пошуку необхідної інформації, незважаючи на її величезний обсяг. Користувач таких БД може виконати пошук будь-якого його параметру. Крім того, БД компактні - БД, що містить кілька тисяч записів, займає малий обсяг пам'яті.

База даних — файл, у якому зберігається інформація про зв'язки між таблицями та інша інформація про таблиці, що не зберігається безпосередньо у самих таблицях. Цей файл дає змогу зв'язувати таблиці і давати їм довгі назви.

СУБД Visual FoxPro складається з окремих компонентів, які використовуються для зберігання інформації, її відображення і редагування.

В Visual FoxPro все дані зберігаються в БД, яка складається з таблиць, відношень між таблицями, індексів, тригерів і збережених процедур. Кожна таблиця має унікальне ім'я і зберігається в окремому файлі, найменування якого збігається з ім'ям таблиці. Створений файл має розширення DBF.

Кожна створювана таблиця може мати декілька пов'язаних з нею індексів, використовуваних для впорядкування даних і швидкого пошуку необхідних записів.

Надзвичайно зручним і корисним засобом доступу до БД є представлення даних. Представлення даних дозволяє об'єднувати дані таблиць і відображати їх в більш зручному вигляді. Можна вибрати тільки окремі поля, об'єднати декілька полів в одне поле, обчислити підсумкові значення, задати нові імена полів таблиці.

Як правило, кількість записів в БД набагато перевершує кількість таблиць. У міру експлуатації БД їх число неперервно зростає. У багатьох інформаційних системах доступ до даних, включаючи перегляд, створення і редагування, здійснюється тільки за допомогою представлень даних. Такий підхід дозволяє здійснити гнучке управління доступом до інформації.

При використанні представлень для вибірки даних в формах, звітах, при створенні запитів, а також в програмах застосовуються ті ж правила, що і для таблиць. Редагування даних, включених в представлення, можливо тільки за певних умов. Наприклад, в тому випадку, якщо представлення створене на основі тільки однієї таблиці.

Visual FoxPro надає можливість зберігати параметри основного вікна Visual FoxPro, налаштування таблиць, параметри вікон діалогу і панелі інструментів з

допомогою файлу параметрів налаштування.

На рис. 1.1 показано головне вікно Visual FoxPro, яким воно виглядає після запуску програми. У ньому можна виділити шість основних областей: заголовок вікна, рядок меню, панель інструментів, рядок стану, робоча область і командне вікно Command.

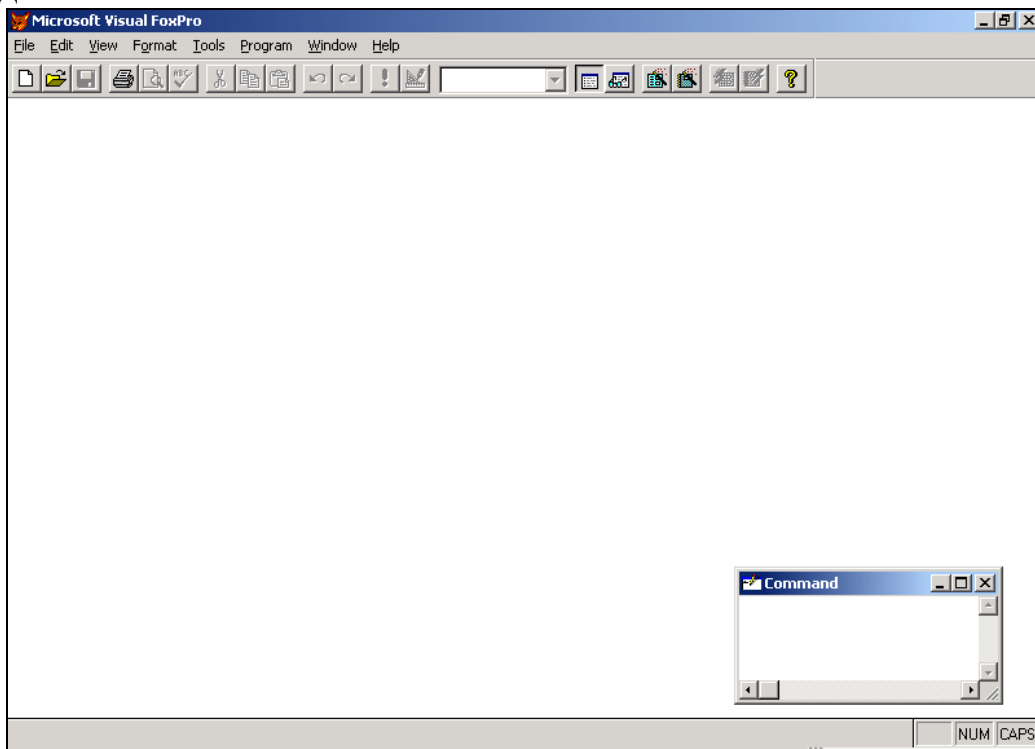


Рисунок 1.1 – Головне вікно Visual FoxPro

Заголовок знаходиться у верхній частині головного вікна і містить три об'єкти: копію значка програми Visual FoxPro, заголовок головного вікна і значки управління цим вікном.

Рядок меню містить ієрархічне меню Visual FoxPro. Список пунктів меню залежить від типу активного вікна.

Visual FoxPro надає в розпорядження користувача багато самих різноманітних панелей інструментів для роботи з БД, формами, звітами і запитами. Ці панелі інструментів містять набір кнопок, причому цей набір залежить від призначення конкретної панелі інструментів. При першому запуску Visual FoxPro після його установки на екрані відображається стандартна панель інструментів.

У нижній частині головного вікна розташований рядок стану, в який виводиться інформація про поточний стан активного вікна, короткий опис обраної кнопки панелі інструментів або пункту меню. Вид інформації, яка відображається в рядку стану, залежить від типу активного вікна. Наприклад, у вікні конструктора форм в рядку стану виводиться найменування активного об'єкта форми, а під час перегляду таблиці в режимі Browse в лівій частині рядка стану відображається найменування відкритої таблиці, в середині рядка стану - номер поточного запису і індикатор режиму роботи.

У процесі роботи в головному вікні Visual FoxPro відкриваються вікна конструкторів форм, звітів, запитів, проекту і інші вікна. Всі ці вікна є дочірніми по відношенню до головного вікна і закриваються при його закритті.

Працюючи з Visual FoxPro, можна використовувати як кнопки панелі інструментів, так і рядок меню. Всі команди рядка меню є ієрархічними і при виборі команди відкривається її підміну.

При відсутності в головному вікні Visual FoxPro інших відкритих вікон рядок меню містить пункти File (Файл), Edit (Правка), View (Вид), Tools (Інструменти), Program (Програма), Windows (Вікна) і Help (Довідка). При відкритті вікон в меню додаються команди, застосовні тільки до об'єктів даного вікна. Наприклад, при роботі з таблицями в головне меню додається пункт меню Table (Таблиця), при відкритті на екрані форми в меню з'являється пункт Form (Форма).

#### Команди меню File

Меню File (Файл) містить команди, пов'язані з доступом до файлів. Ці команди дозволяють створювати нові файли різних типів, відкривати існуючі файли, закривати, зберігати і друкувати їх.

Команда Save As HTML (Зберегти файл у форматі HTML), що з'явилася в Visual FoxPro 6.0, відкриває діалогове вікно Save As HTML, що дозволяє зберегти поточну форму, меню, звіт або таблицю в форматі HTML (HyperText Markup Language).

Дуже зручна команда Revert (Повернути), що дозволяє скасувати всі зміни файлу, зроблені в поточному сеансі редагування. У нижній частині меню розташовуються імена останніх чотирьох проектів, які надають можливість швидкого відкриття будь-якого з них. Останньою командою цього меню є команда Exit (Вихід). Вона аналогічна дії команди QUIT в командному вікні і призначена для виходу з Visual FoxPro.

#### Команди меню Edit

Меню Edit (Правка) є в багатьох додатках Windows. У Visual FoxPro команди цього меню використовуються при створенні форм і звітів, редагуванні програм.

Команди Paste Special (Спеціальна вставка), Insert Object (Вставити об'єкт), Object (Об'єкт) і Links (Зв'язки) застосовні тільки до полів типу General і OLE-об'єктів. Решта команд застосовуються при редагуванні як програм, так і полів таблиць.

Команда Go to Line (Перехід на рядок), призначена в Visual FoxPro для роботи з програмами. За допомогою даної команди можна переходити в тексті програми до рядка з вказаним номером.

Остання в меню - команда Properties (Властивості), при виборі якої відкривається діалогове вікно Edit Properties (Редагування властивостей), що дозволяє задавати параметри збереження програмних файлів, переглядати їх властивості, дату останнього редагування і який він обіймав обсяг на диску.

#### Команди меню View



Цей пункт меню містить команди, що керують переглядом поточного відкритого файлу. Якщо ви не переглядаєте файл, то меню містить всього одну команду Toolbars (Панелі інструментів). Ця команда відкриває діалогове вікно, в якому перераховані панелі інструментів, що використовуються при роботі в Visual FoxPro.

Команда Properties (Властивості) з'являється в даному меню тоді, коли ви знаходитесь в конструкторі форм і конструкторі класів. При виборі цієї команди відкривається діалогове вікно Properties, що дозволяє переглянути і змінити властивості об'єктів, розташованих у формі.

При перегляді таблиць або редагуванні форм, меню або звітів в меню з'являються додаткові команди. Наприклад, якщо ви працюєте з таблицями, в меню з'являється кілька нових команд. Команди Edit (Редагування) і Browse (Огляд) переводять таблицю в режим перегляду і редагування містяться в ній записів. Відмінність між цими двома командами полягає в стилі відображення вмісту таблиці.

#### Команди меню Format

Меню Format (Формат) містить команди, що управляють характеристиками шрифту, вирівнюванням тексту і об'єктів, визначенням інтервалів між об'єктами. Однак при роботі з різними конструкторами стають доступними і додаткові команди.

#### Команди меню Tools

Це меню містить майстрів і засоби налагодження програм.

Першою в меню розташовується команда Wizards (Майстри), що відкриває список майстрів Visual FoxPro. Майстер - це програма, яка допомагає користувачеві у вирішенні таких завдань, як створення форм, звітів, таблиць. Процес створення при цьому полягає у відповідях на питання майстра відносно створюваного об'єкта.

Далі в меню розташована команда Spelling (Орфографія). Її можна використати для перевірки правильності написання будь-якого тексту.

Команда Debugger (Відладчик) призначена для налагодження. Вона відкриває діалогове вікно Visual FoxPro Debugger, в якому ви можете переглянути значення змінних, елементів масиву, полів, значення, які повертаються функціями Visual FoxPro.

В Visual FoxPro в меню Tools (Інструменти) з'явилися нові команди Component Gallery (Галерея компонентів), Coverage Profiler (Аналіз виконання) і Run Active Document (Створення активне документа). Команда Component Gallery відкриває вікно Catalogs - Component Gallery для роботи в Галереї компонентів Visual FoxPro. Команда Coverage Profiler дозволяє продивитися інформацію про виконану програмою. Run Active Document відкриває діалогове вікно Run Active Document для вибору активного документа і його запуску. Остання команда в меню Tools - Options (Параметри). Використавши її можна налаштувати конфігурацію Visual FoxPro.

#### Команди меню Program

Меню Program (Програма) містить команди, пов'язані з виконанням і компіляцією програм.

### Команди меню Windows

У меню Windows (Вікна) є команди, які керують відкритими на екрані вікнами. За допомогою цих команд можна впорядковувати вікна і переходити з одного вікна в інше. Крім того, команди даного меню дозволяють активізувати відкрите вікно.

Команда Command Window (Командне вікно) з меню Windows активізує вікно Command (Команда), що дозволяє використовувати команди при роботі з Visual FoxPro.

Для відкриття вікна Data Session, яке містить список всіх відкритих в даному сеансі таблиць і відношень між ними (рис. 1.2), використовується команда Data Session (Вікно даних).

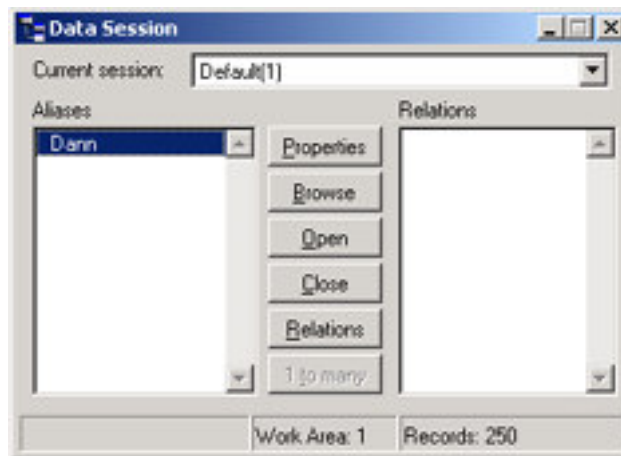


Рисунок 1.2 – Вікно Data Session

У центрі вікна Data Session знаходяться кнопки управління таблицями. Кнопка Properties відкриває вікно, що дозволяє переглянути і змінити поточні властивості таблиці. Для відкриття таблиці в новій робочій області використовується Open. Кнопка Browse (табличний режим) відкриває таблицю для перегляду містяться в ній записів. Кнопка Close (Закрити) вікна Data Session закриває таблицю в поточній робочій області, а кнопка Relations (Відношення) дозволяє визначити відносини між відкритими таблицями.

### Команди меню Help

Help (Довідка) - остання команда меню головного вікна.














В Visual FoxPro є багато панелей інструментів, які використовуються при створенні форм, звітів і запитів. Стандартна панель інструментів, що використовується в усіх режимах роботи наведена на рис. 1.3.



Рисунок 1.3 – Головне вікно Visual FoxPro

Призначення кнопок стандартної панелі інструментів наведено в табл. 1.1.

Таблиця 1.1 – Призначення кнопок стандартної панелі інструментів

Кнопка	Назва	Призначення
	New (Новий)	Створює новий файл
	Open (Відкрити)	Відкриває існуючий файл
	Save (Зберегти)	Зберігає відкритий файл
	Print One Copy (Друкувати одну копію)	Друкує вміст активного вікна
	Print Preview (Попередній перегляд)	Відкриває вікно попереднього перегляду друкованого документа
	Spelling (Орфографія)	Перевіряє орфографію
	Cut (Вирізати)	Видаляє виділений текст або виділені об'єкти і помещає їх в буфер
	Copy (Копіювати)	Копіює в буфер виділений текст або виділені об'єкти, не видаляя їх.
	Paste (Вставити)	Вставляє зміст буфера. Дія цієї кнопки залежить від режиму роботи
	Undo (Відмінити)	Відміняє останню виконану дію
	Redo (Відновити)	Відновлює відмінені дії
	Run (Виконати)	Запускає на виконання
	Modify Form (Модифікувати форму)	Дозволяє модифікувати форму
	Database (База даних)	Містить перелік усіх відкритих баз даних
	Command Window (Командне вікно)	Відкриває командне вікно
	Data Session Window (Вікно сеанса даних)	Відкриває вікно Data Session
	Form (Форма)	Викликає майстер форм
	Report (Звіт)	Викликає майстер звітів
	AutoForm Wizard (Майстер автоформи)	Створює стандартну форму для поточної таблиці
	AutoReport Wizard (Майстер автозвіту)	Створює стандартний звіт для поточної таблиці
	Help (Довідка)	Викликає довідкову систему

## Завдання

- I. Ознайомитися з основними елементами інтерфейсу Visual FoxPro.
- II. Вивчити основні типи даних.
- III. Створити проект (назва проекту – прізвище студента) та базу даних, що складається з трьох таблиць: таблиця «ДОВІДНИК ВІДДІЛІВ», таблиця «ДОВІДНИК ТЕМ» та таблиця «НДІ». Структура таблиць наведена в таблицях 1.2-1.4, а зміст їх наповнення – в таблицях 1.5-1.7.

Таблиця 1.2 – Структура таблиці «ДОВІДНИК ВІДДІЛІВ»

Найменування	Ідентифікатор	Тип	Загальна довжина	Довжина дробової частини
Код підрозділу	k_podr	N	3	
Назва	name_p	C	25	

Індекс до поля <k\_podr>, тип індекса — PRIMARY (ПЕРВИННИЙ)

Таблиця 1.3 – Структура таблиці «ДОВІДНИК ТЕМ»

Найменування	Ідентифікатор	Тип	Загальна довжина	Довжина дробової частини
Номер теми	n_tema	N	3	
Назва	name_t	C	30	

Індекс до поля <n\_tema>, тип індекса — PRIMARY (ПЕРВИННИЙ)

Таблиця 1.4 – Структура таблиці «НДІ»

Найменування	Ідентифікатор	Тип	Загальна довжина	Довжина дробової частини
Код підрозділу	k_podr	N	3	
Табельний номер	tn	N	4	
Номер теми	n_tema	N	3	
Тривалість роботи (міс)	chas	N	5	2

Індекс до поля <k\_podr>, тип індекса — REGULAR (ЗВИЧАЙНИЙ)

Індекс до поля <n\_tema>, тип індекса — REGULAR (ЗВИЧАЙНИЙ)

Таблиця 1.5 – Зміст таблиці «ДОВІДНИК ВІДДІЛІВ»

Код підрозділу	Найменування
11	Інформаційні системи
13	Інформаційний менеджмент
14	Методи та моделі

Таблиця 1.6 – Зміст таблиці «ДОВІДНИК ТЕМ»

Номер теми	Найменування
114	Розробка засобів автоматизації
215	Інформаційні технології
314	Моделі подання даних

Таблиця 1.7 – Зміст таблиці «НДІ»

Код структурного підрозділу	Табельний номер	Номер теми	Тривалість роботи (міс)
11	101	215	20,00
11	102	314	12,00
11	103	215	6,00
11	104	215	15,00
13	302	314	8,60
13	303	314	10,00
13	304	215	4,50
14	401	114	12,00
14	402	114	6,00
14	403	215	14,00
14	404	215	5,50
14	405	314	12,00
14	406	114	18,00
14	407	314	22,40
14	408	215	20,00

**IV. Виконати дії в інтерактивному режимі.**

Перелік дій: додання нового запису; вилучення записів на логічному рівні; поновлення позначених на вилучення записів; рух по таблиці; перегляд даних таблиці; пошук записів; неявне введення даних до таблиці (заміщення записів); обчислення.

**V. Вікно Command вивести на друк.**

**Порядок виконання роботи**

**I. Знайомство з основними елементами інтерфейсу.**

Ознайомитися з основними елементами інтерфейсу користувача СУБД Visual FoxPro (система меню, багатовіконне середовище, блоки діалогу, засоби управління).

**II. Вивчити основні типи даних.**

Розглянути основні типи даних, особливості їх використання та основні команди діалогової обробки даних.

**III. Створити проект** (назва проекту – прізвище студента) та **базу даних**, що складається з трьох таблиць: таблиця «ДОВІДНИК ВІДДІЛІВ», таблиця «ДОВІДНИК ТЕМ» та таблиця «НДІ». Структура таблиць наведена в таблицях 1.2-1.4, а зміст їх наповнення – в таблицях 1.5-1.7.

Створення нового проекту

FILE (ФАЙЛ)

NEW (СТВОРИТИ)

PROJECT (ПРОЕКТ)

<назва>.PJX [у нашому прикладі INSTUTYT.PJX, а за завданням –

прізвище студента].

Після виконання цих дій з'явиться вікно PROJECT MANAGER (МЕНЕДЖЕР ПРОЕКТУ), вигляд якого наведено на рис. 1.4.

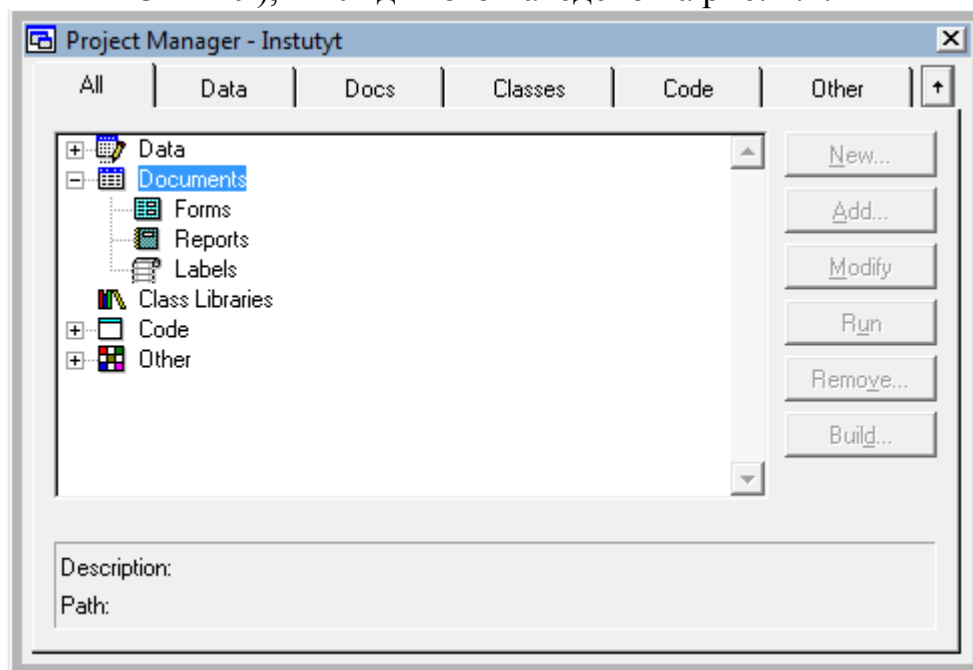


Рисунок 1.4 – Вікно МЕНЕДЖЕР ПРОЕКТУ

#### Створення бази даних

DATABASE (БАЗУ ДАНИХ)

NEW (СТВОРИТИ)

<назва>.DBC [у нашому прикладі INSTUTYT.DBC, а за завданням – прізвище студента].

Після виконання цих дій з'явиться вікно DATABASE DESIGNER (КОНСТРУКТОР БАЗИ ДАНИХ). Вигляд вікна подано на рис. 1.5.

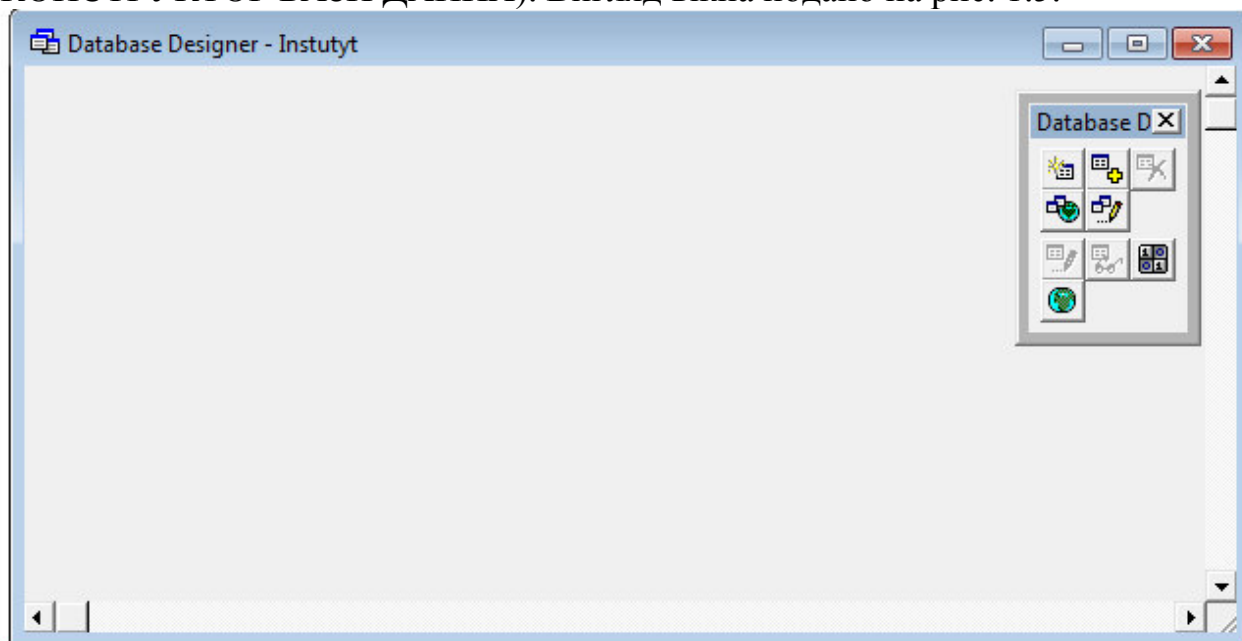


Рисунок 1.5 – Вікно КОНСТРУКТОР БАЗИ ДАНИХ

*Зауваження:* Звернення до бази даних проекту у подальшій роботі можна виконати двома шляхами:

а) звернутися до вікна PROJECT MANAGER (МЕНЕДЖЕР ПРОЕКТУ) (див рис. 6.), вибрати закладку DATA (ДАНИ), вибрати базу даних INSTUTYT та натиснути кнопку MODIFY (ЗМІНИТИ), як показано на рис.1.6;

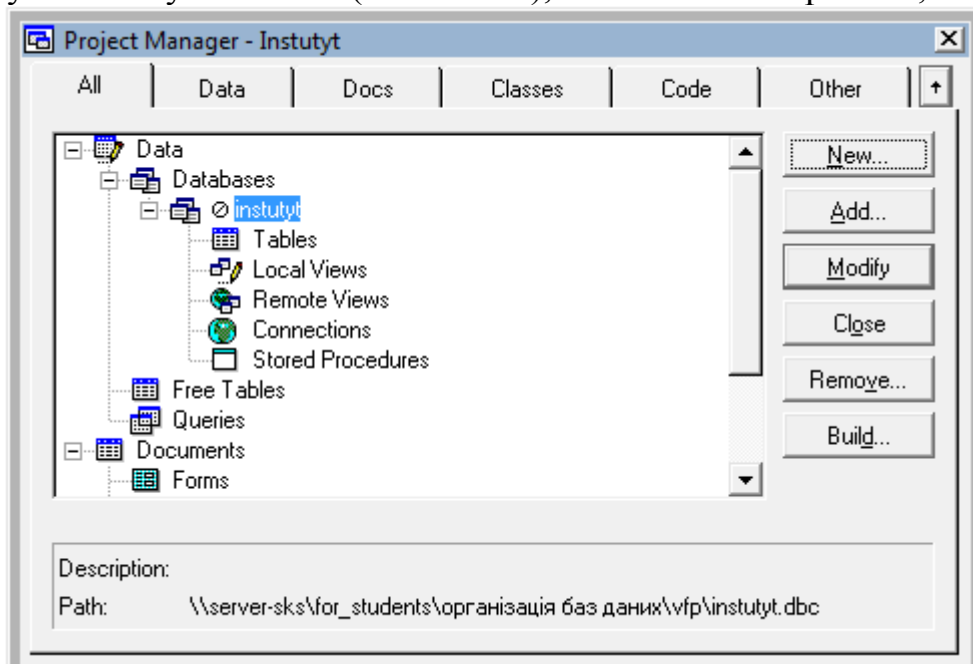


Рисунок 1.6 – Вікно МЕНЕДЖЕР ПРОЕКТУ

б) з використанням системи меню:

VIEW (ВИД)

DATABASE DESIGNER (КОНСТРУКТОР БАЗИ ДАНИХ)

### Створення таблиць

Створення таблиці НДІ

Таблиці можна створити двома шляхами:

а) перейти до вікна PROJECT MANAGER (МЕНЕДЖЕР ПРОЕКТУ), вибрати закладку DATA (ДАНИ), вибрати у переліку об'єктів TABLES (ТАБЛИЦІ) й натиснути кнопку NEW (СТВОРИТИ);

б) перейти до вікна DATABASE DESIGNER (КОНСТРУКТОР БАЗИ ДАНИХ) (див. рис. 1.4) і натиснути кнопку (створення нової таблиці) або за допомогою меню:

DATABASE (БАЗА ДАНИХ)

NEW TABLE (НОВА ТАБЛИЦЯ)

У діалоговому вікні, що з'явиться (незалежно від варіанта виконання), потрібно ввести назву таблиці, яка створюється — <Назва>.dbf [у нашому прикладі NDI.DBF], і натиснути <ОК>.

У вікні опису таблиці TABLE DESIGNER (КОНСТРУКТОР ТАБЛИЦІ) (рис. 1.7) виконати опис структури таблиці НДІ, поданої у таблиці 1.4 завдання.

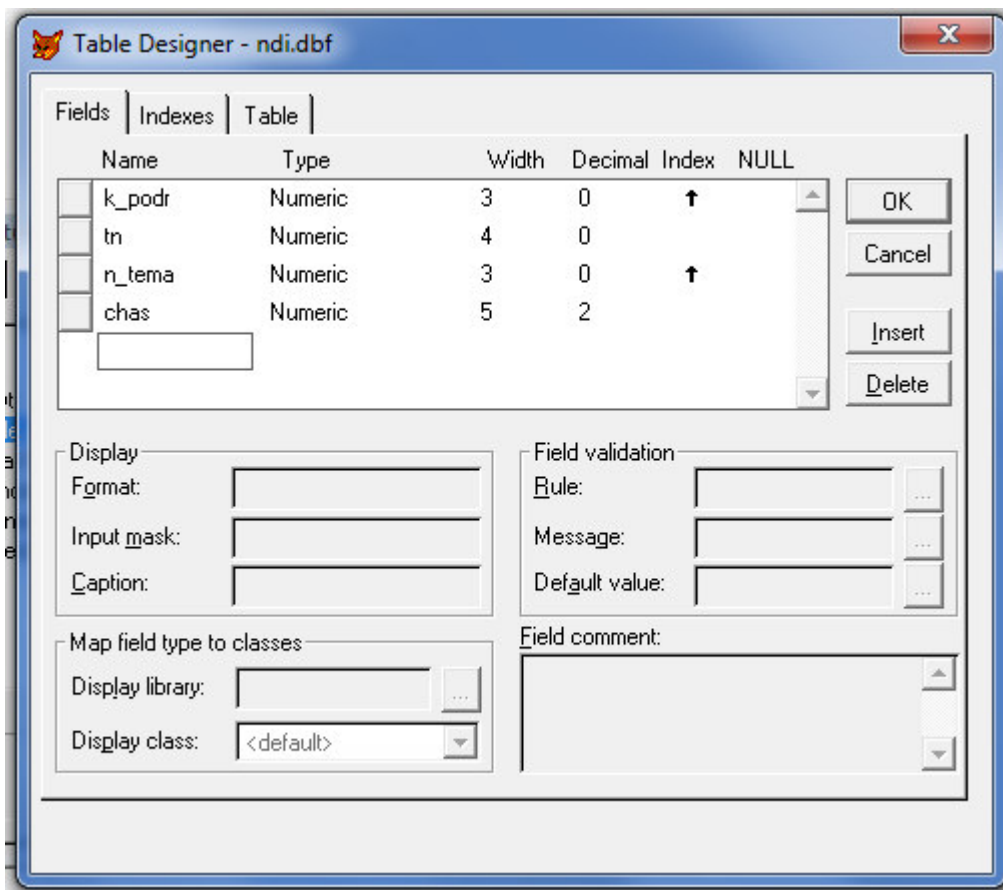


Рисунок 1.7 – Вікно КОНСТРУКТОР ТАБЛИЦІ

Пояснення щодо призначення елементів вікна TABLE DESIGNER (КОНСТРУКТОР ТАБЛИЦІ):

NAME (ІМ'Я) — ідентифікатор поля таблиці, що вводиться;

TYPE (ТИП) — тип поля (цифровий, дата та ін.);

WIDTH (ДОВЖИНА) — загальна довжина поля;

DECIMAL (ДОВЖИНА ДРОБОВОЇ ЧАСТИНИ) — кількість знаків після десятинної коми;

INDEX (ІНДЕКС) — тип індекса (відсутній, за зростанням (ascending) чи за спаданням (descending));

NULL (НУЛЬ) — чи може поле мати нульове значення.

Пояснення щодо групи властивостей, пов'язаних з відображенням полів:

FORMAT (ФОРМАТ) — формат поля;

INPUT MASK (МАСКА ВВОДУ) — маска для вводу поля (можна задати кількість та значення символів, що будуть вводитися у це поле);

CAPTION (НАЗВА) — розширена назва поля. Надає можливість вводити повну довгу назву поля будь якою мовою.

Пояснення щодо групи властивостей, пов'язаних з властивостями полів (Field validation):

RULE (ПРАВИЛА) — правила для вводу полів, за допомогою яких можна визначити обмеження, що накладаються на значення поля;

MESSAGE (ПОВІДОМЛЕННЯ) — повідомлення, яке з'явиться на екрані



при невиконанні правил, заданих у попередньо описаній властивості;

DEFAULT VALUE (ЗНАЧЕННЯ ЗА ЗМОВЧУВАННЯМ) — визначає початкове значення поля, яке приймається за змовчуванням.

FIELD COMMENT (КОМЕНТАР) — коментар до поля. Доступ до заданого коментаря користувач має тільки під час перегляду структури таблиці.

В результаті з'явиться вікно для вводу даних (див. рис. 1.8).

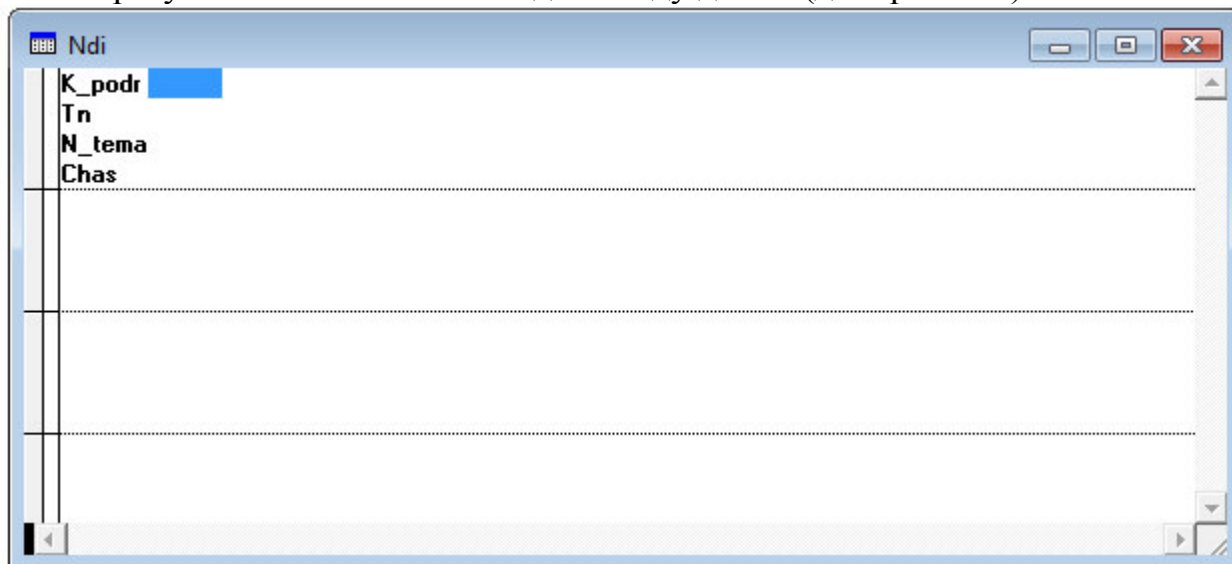


Рисунок 1.8 – Вікно ПЕРЕГЛЯДУ ТА ДОДАВАННЯ ДАНИХ

Ввести відповідно до табл. 1.7 завдання дані у таблицю. Під час введення даних слід звернути увагу, що якщо довжина даних, що вводяться, така сама, як вказано в конструкторі таблиць, то після введення даних курсор автоматично перейде на наступне поле, в іншому разі, після введення даних у поле, для переходу до наступного поля потрібно натиснути кнопку ENTER на клавіатурі.

Аналогічно створити таблиці «Довідник відділів» [Dov\_vid] та «Довідник тем» [Dov\_tem] за даними, наведеними у табл. 1.2-1.3, та табл. 1.5-1.6.

#### Встановлення зв'язків між таблицями

Для встановлення зв'язків перейти у вікно DATABASE DESIGNER (КОНСТРУКТОР БАЗИ ДАНИХ) (див. рис. 1.4). Для цього використати меню:

VIEW (ВИД)

DATABASE DESIGNER (КОНСТРУКТОР БАЗИ ДАНИХ)

У вікні за допомогою маніпулятора "Миша" потрібно встановити зв'язки між таблицями. Для цього слід перемістити курсор на ключове поле таблиці власника, натиснути ліву кнопку миші і, не відпускаючи кнопку, перетягнути поле на відповідне поле таблиці підлеглого.

Встановлення зв'язку між таблицями «Довідник тем» та "НДІ". Перемістити курсор на поле <n\_tema> таблиці «Довідник тем» ("Dov\_tem"), натиснути ліву кнопку миші і, не відпускаючи її, перетягнути виділене поле на поле <n\_tema> таблиці «НДІ» ("NDI"). Таким чином встановиться зв'язок «один

до багатьох» між обраними таблицями.

Аналогічно встановити зв'язок між таблицями «Довідник відділів» та «НДІ».

#### Задання правил цілісності даних

Задання правил цілісності даних дає змогу задати правила, за якими буде здійснюватися додання нових записів, вилучення записів та редагування записів.

Для задання правил цілісності потрібно виконати такі дії:

а) виділити той зв'язок, для якого потрібно задати правила цілісності;

б) перейти в меню і вибрати: DATABASE (БАЗА ДАНИХ)

EDIT REFERENTIAL INTEGRITY (ЗАДАТИ ЦІЛІСНІСТЬ ЗВ'ЯЗКУ)

В результаті з'явиться діалогове вікно REFERENTIAL INTEGRITY BUILDER (УКЛАДАЧ ЦІЛІСНОСТІ ЗВ'ЯЗКУ).

На закладці RULES OF UPDATING (ПРАВИЛА ЗМІНИ) встановлюються правила на дії, що будуть виконуватися при зміні записів у таблицях:

CASCADE (КАСКАД) — автоматично змінити всі зв'язані записи у підлеглий таблиці відповідно до зміненого запису у таблиці-власника;

RESTRICT (ЗАБОРОНИТИ ЗМІНИ) — видати запит на зміну записів у підлеглий таблиці (якщо такі записи присутні);

IGNORE (ІГНОРУВАТИ) — не змінювати записи у підлеглий таблиці.

Такі самі правила присутні і в закладці RULES OF DELETING (ПРАВИЛА ВИДАЛЕННЯ), що задає правила на винищення записів, та в закладці RULES OF INSERTING (ПРАВИЛА ВСТАВКИ), що задає правила на додання нових записів (за винятком опції CASCADE (КАСКАД), що відсутня).

Для всіх зв'язків потрібно задати опції RESTRICT (ЗАБОРОНИТИ ЗМІНИ), щоб була можливість контролювати подальшу роботу з таблицями.

#### **IV. Виконати дії в інтерактивному режимі**

Перелік дій: додання нового запису; вилучення записів на логічному рівні; поновлення позначених на вилучення записів; рух по таблиці; перегляд даних таблиці; пошук записів; неявне введення даних до таблиці (заміщення записів); обчислення.

Загальні зауваження:

1) для виконання дій в інтерактивному режимі таблиця, з якою необхідно працювати, повинна бути активною (відкритою);

2) поширення певної команди на підмножину записів таблиці може бути задане на фізичному рівні — SCOPE (ДІАПАЗОН) та логічному FOR (ДЛЯ) та WHILE (ДОКИ). Можливі границі дії команди:

ALL (УСІ) — усі записи таблиці;

NEXT (НАСТУПНІ) — задана кількість записів, починаючи з поточного (на якому встановлено покажчик);

RECORD (ЗАПИС) — конкретний запис, для якого задається його

системний номер (системний номер запису надається автоматично під час створення запису);

REST (РЕШТА) — підмножина записів до кінця таблиці, починаючи з поточного.

Задання логічної умови FOR (ДЛЯ) поширює дію команди на всі записи таблиці, що відповідають умові.

Задання логічної умови WHILE (ПОКИ) поширює дію команди на множину записів, починаючи з поточного, доки не зміниться логічна умова.

Приклад виконання дії в інтерактивному режимі.

Додати новий запис до таблиці NDI, що містить такі дані:

код підрозділу — номер групи;

табельний номер — номер залікової книжки;

номер теми — номер студента за списком;

тривалість роботи — 24.00.

Для додання нового запису перейти в режим перегляду таблиці: у вікні PROJECT MANAGER (МЕНЕДЖЕР ПРОЕКТУ) (див. рис. 1.5) у закладці DATA (ДАНИ) перемістити курсор на таблицю "НДІ" ("NDI") і натиснути командну кнопку BROWSE (ПЕРЕГЛЯНУТИ).

Відкриється вікно, у якому будуть представлені вже введені записи. Для того щоб додати нові записи, потрібно вибрати з меню: VIEW (ВИД)

APPEND MODE (РЕЖИМ ДОДАТКА) У цьому режимі потрібно додати записи

Видалити на логічному рівні записи таблиці НДІ

Видалити записи до кінця таблиці, починаючи з 7-го:

а) перейти в режим перегляду таблиці;

б) перейти на 7-й запис. TABLE (ТАБЛИЦЯ)

GO TO RECORD (ПЕРЕЙТИ НА ЗАПИС)

RECORD # (# ЗАПИСИ)

В результаті з'явиться вікно GO TO RECORD (ПЕРЕХІД НА ЗАПИС), вигляд якого наведено на рис.1.9. У вікні потрібно ввести <7> для переходу на 7-й запис.



Рисунок 1.9 – Вікно ПЕРЕХІД НА ЗАПИС

в) задати дію, що виконується, видалення: TABLE (ТАБЛИЦЯ)

DELETE RECORD (ВИДАЛИТИ ЗАПИС)

В результаті з'явиться вікно DELETE (ВИДАЛЕННЯ), вигляд якого

наведено на рис. 1.10.

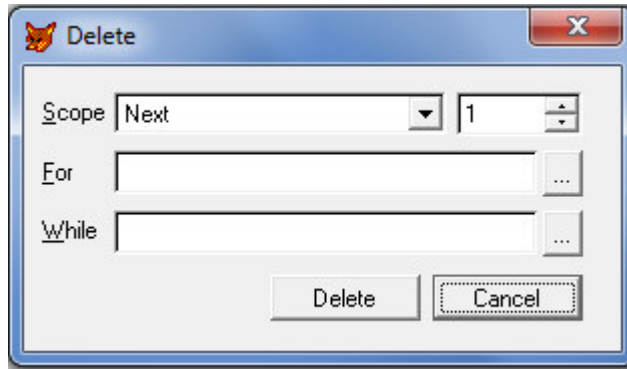


Рис. 1.10 – Вікно ВИДАЛЕННЯ

г) задати діапазон дії команди: SCOPE (ДІАПАЗОН)  
REST (РЕШТА) і натиснути кнопку DELETE (ВИДАЛИТИ)

*Примітка:* Візуальною ознакою того, що записи видалені на логічному рівні, є чорна позначка ліворуч від запису.

Самостійно виконати видалення:

- а) другого запису;
- б) трьох записів, починаючи з 5-го.

Поновлення позначених на видалення записів:

а) відкрити таблицю, записи якої потрібно відновити;

б) задати дію, що виконується: TABLE (ТАБЛИЦЯ)

RECALL RECORDS (ВІДНОВИТИ ЗАПИСИ). В результаті виконання зазначених дій з'явиться вікно, вигляд якого наведено на рис. 1.11.

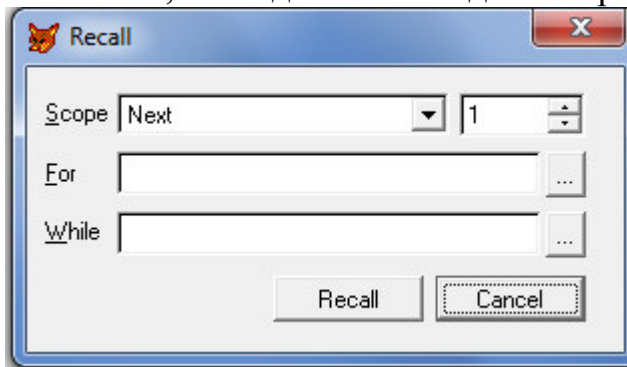


Рисунок 1.11 – Вікно ВІДНОВЛЕННЯ ЗАПИСІВ

Для поновлення всіх позначених на видалення записів задати діапазон команди:

SCOPE (ДІАПАЗОН)

ALL (УСІ) і натиснути кнопку RECALL (ВІДНОВИТИ).

*Примітка:* Результати виконання дій відображаються в інформаційному рядку, якому вказується кількість поновлених записів.

Здійснити рух по таблиці:

TABLE (ТАБЛИЦЯ)

TOP (ПОЧАТОК) — на перший запис BOTTOM (КІНЕЦЬ) — на останній запис GO TO RECORD (ПЕРЕЙТИ НА ЗАПИС) # RECORD (ЗАПИСИ) — на певний запис, номер якого задається у вікні GO TO RECORD (ПЕРЕЙТИ НА ЗАПИС).

#### Вивести для перегляду записи

Для перегляду записів використовується команда BROWSE, яка в явному вигляді задається у командному вікні

Перейти до командного вікна COMMAND.

Якщо вікно на робочому полі відсутнє, активізувати: WINDOW (ВІКНО)  
COMMAND WINDOW (КОМАНДНЕ ВІКНО)

а) всі записи:

BROWSE ALL

б) записи, що мають код підрозділу 11 або 14:

BROWSE FOR K\_PODR=11 OR K\_PODR=14

в) записи, що мають номер теми 215:

BROWSE FOR N\_TEMA=215

г) Самостійно:

записи, що мають табельний номер 102;

записи, що мають час тривання більше 10 місяців.

#### Пошук записів

Знайти перший запис з номером теми 114:

TABLE (ТАБЛИЦЯ)

GO TO RECORD (ПЕРЕЙТИ НА ЗАПИС)

LOCATE... (ЗНАЙТИ)

В результаті з'явиться вікно, у якому потрібно ввести умову пошуку. Вигляд вікна наведено на рис. 1.12.

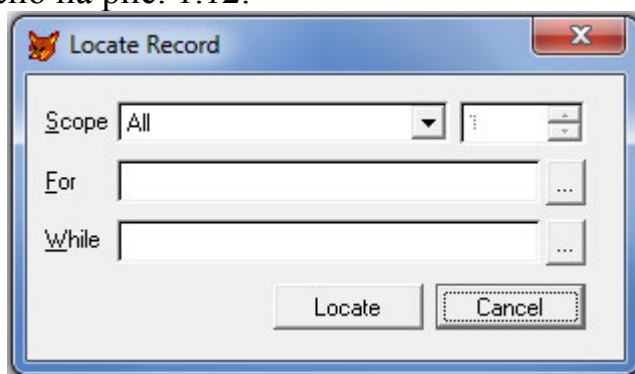


Рисунок 1.12 – Вікно ПОШУК ЗАПИСІВ

Для побудови умови пошуку потрібно натиснути кнопку, що знаходиться праворуч від опції FOR (ДЛЯ). В результаті з'явиться вікно EXPRESION BUILDER (УКЛАДАЧ ВИРАЗІВ).

Це вікно призначене для спрощення побудови різного роду виразів Для задання умови «Номер теми дорівнює 14» потрібно виконати такі дії

— вибрати із списку полів таблиці поле `n_tema`. Для цього у списку FIELDS (ПОЛЯ) клацнути мишкою два рази на полі `n_tema`. В результаті назва поля з'явиться у текстовому полі LOCATE RECORD FOR (ЗНАЙТИ ЗАПИС ДЛЯ),

— додати умову, щоб вираз набув вигляду. `n_tema=1 14`, і натиснути кнопку ОК

*Примітка* Для перевірки правильності задання виразу треба натиснути кнопку VERIFY (ПЕРЕВІРКА) За наявності помилок у виразі отримаєте системне повідомлення «Expression not valid»

В результаті з'явиться вікно із записами таблиці NDI, і покажчик переміститься на потрібний запис. В інформаційному рядку з'явиться повідомлення про номер знайденого запису в таблиці.

Знайти перший запис, у якому номер теми 215 та код підрозділу 14.

Вираз для пошуку повинен мати такий вигляд:

`n_tema=215 AND k_podr=14`

Для задання логічної операції (AND) вибрати зі списку LOGICAL (ЛОГІЧНІ)

Самостійно знайти перший запис з кодом підрозділу 13 і тривалістю робіт більше 10 місяців.

#### **V. Меню FILE – PRINT – роздрукувати вікно Command**

**УВАГА!** Зміст вікна COMMAND зберігається лише в поточному сеансі роботи. Тому в кінці кожного заняття перед закриттям програми потрібно зберігати команди, що є у командному вікні в спеціальному файлі.

### **Вимоги до звіту**

У звіті навести:

- тему та мету лабораторної роботи,
- опис конфігурації середовища розробки, інтерфейс СУБД Visual FOXPRO,
- відповідь на індивідуальне питання<sup>1</sup>,
- текст завдання,
- порядок виконання /алгоритм/ кожного пункту завдання та його результати,
- аналіз команд вікна Command,
- висновки по отриманим результатам роботи,
- список використаних джерел.

Звіт повинен бути оформленим українською мовою на аркушах паперу формату А4. Основний текст повинен бути оформлений з наступними параметрами: Шрифт – TimesNewRoman, кегль – 14, міжрядковий інтервал – 1.

---

<sup>1</sup> Обирається з переліку контрольних питань за номером прізвища студента зі списку в журналі академічної групи

Звіт оформлюється згідно ДСТУ 3008:2015 “Звіти у сфері науки і техніки. Структура та правила оформлювання”, чинний від 2017-07-01.

### **Контрольні питання**

1. Сучасна СУБД як комплекс інструментів, додатків і послуг.
2. Загальні етапи технології роботи з СУБД.
3. Основні функції СУБД. Організація обробки даних засобами СУБД.
4. Проведіть загальний огляд СУБД. Приведіть класифікацію СУБД.
5. Розкрийте характеристики і основні функції СУБД.
6. Розповсюдженні настільні СУБД, основні характеристики і порівняльний аналіз.
7. Розповсюдженні промислові СУБД, основні характеристики і порівняльний аналіз.
8. Технічні характеристики СУБД.
9. Типова структура інтерфейсу СУБД.
10. Об'єктні СУБД, основні характеристики і порівняльний аналіз.
11. Основні поняття розподілених БД.
12. Які властивості і переваги мають настільні СУБД?
13. Які властивості і переваги мають професіональні СУБД?
14. Застосування середовищі Web як платформи додатків БД.
15. Використання WWW-технологій для доступу до існуючих БД. Основні поняття.

## Лабораторна робота № 2

**Тема:** Створення таблиць і обробка існуючої інформації в інструментальному середовищі VISUAL FOXPRO

**Мета роботи:** придбати практичні навички роботи з СУБД Visual FOXPRO, створення таблиць, редагування, використання для різнопланових розрахунків.

### Теоретичні відомості

Реляційна модель даних заснована на математичному понятті відношення і представленні відношень у вигляді таблиць. Запропонована на початку 70-х років американським вченим Е.Коддом. В будь-якій реляційній СУБД припускається, що користувач сприймає БД як набір таблиць.

Це стосується тільки логічної структури БД, тобто відноситься до концептуального і зовнішнього представлень. На фізичному рівні БД реалізується за допомогою різних структур зберігання.

В табл. 2.1 наведені елементи реляційної моделі, а на рис. 2.1 – приклад відношення студент.

Таблиця 2.1 – Елементи реляційної моделі

Елементи реляційної моделі	Форма представлення
Відношення	Таблиця
Кортеж	Рядок таблиці
Атрибут	Заголовок стовпця таблиці
Ключ	Сукупність атрибутів, які унікально визначають кожен рядок таблиці, або виконують функції зв'язування таблиць, або дозволяють прискорити операції над таблицями
Домен	Множина значень атрибута
Схема відношення	Рядок заголовків стовпців таблиці

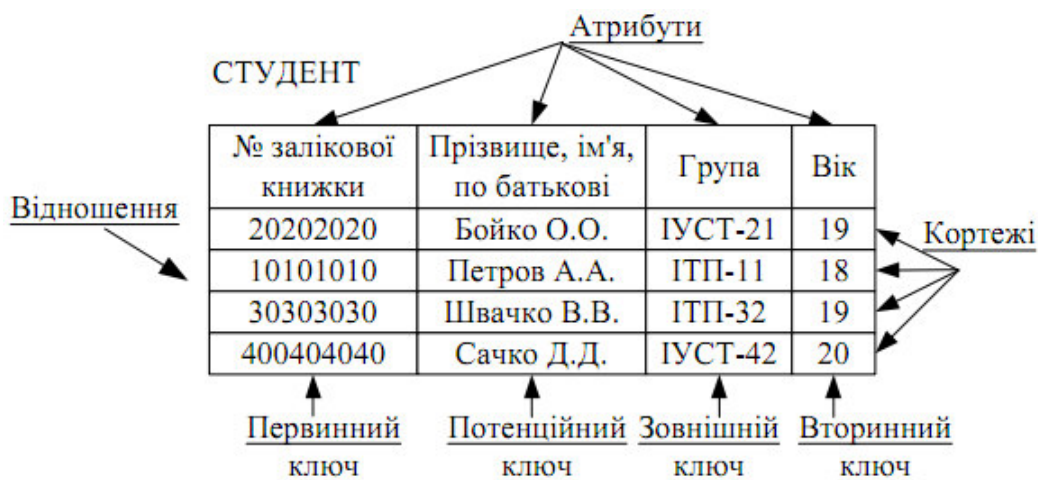


Рисунок 2.1 – Приклад відношення «Студент»



Таблиці складають основу вашої бази даних. У них буде зберігатися вся необхідна інформація. Надалі дані в таблиці будуть доповнюватися новими даними, редагуватися або виключатися з таблиці. Ви зможете переглядати дані таблиць за допомогою форм або впорядковувати їх за заданим критерієм. Інформація, що міститься в таблицях, може бути використана для складання звітів. Крім того, використовуючи діаграми, ви можете графічно представити інформацію, що міститься в базі даних.

Таблиця складається з рядків і стовпців і має унікальне ім'я в базі даних. У кожній з таблиць міститься інформація про будь-які об'єкти одного типу.

В Visual FoxPro ви можете створювати як таблиці, що входять в базу даних, так і окремі таблиці, звані вільними, аналогічні створюваним в попередніх версіях FoxPro.

Таблиці, як і будь-які інші об'єкти в Visual FoxPro, мають імена. Таблиці бувають вільні і входять в базу даних. Найменування таблиці може містити літери, цифри і знак підкреслення.

Створюючи нову таблицю, необхідно пам'ятати, що в базі даних не може бути двох таблиць, що мають однакові імена. Якщо в базі даних вже є таблиця з таким ім'ям, на екрані з'являється запит, замінити чи існуючу таблицю новою. У найменуванні таблиці неприпустимо використання прогалін і спеціальних символів.

Для створення таблиці за допомогою майстра з конструктора бази даних натисніть кнопку New Table (Нова таблиця) на панелі інструментів Database Designer (Конструктор бази даних). У діалоговому вікні натисніть кнопку New Table.

Майстер таблиць (Table Wizard) дозволяє полегшити процес створення типових таблиць за рахунок використання раніше підібраних шаблонів. Шаблони таблиць містять набір полів з певними характеристиками, вибираючи які можна швидко формувати таблицю без звернень до Конструктора для опису атрибутів кожного поля. Майстер таблиць містить більше 40 різновидів шаблонів, що охоплює широкий діапазон інтересів, починаючи від музики до фінансових документів.

Майстер зведеної таблиці (Pivot Table Wizard) дозволяє створити інтерактивну електронну таблицю, за допомогою якої легко сумувати і виконувати аналіз різних даних, одержаних з таблиці Visual FoxPro. Зведену таблицю можна зберігати у форматі Microsoft Excel або як впроваджений об'єкт форми.

Розглянемо структуру таблиці. Таблиця в БД складається з рядків (*записів*) і стовпців. Запис містить інформацію про один елемент бази даних: одну людину, книжку, продукцію, рейс тощо. Запис складається з *полів*, які формують структуру запису.

*Структура запису* фактично визначає структуру таблиці і всієї БД, якщо в ній є лише одна таблиця. *Поле* — це мінімальна (але найважливіша) порція інформації в записі, над якою визначені операції введення, виведення, перетворення тощо. Воно має ім'я, значення, характеризується типом і низкою

додаткових властивостей.

Назви полям дає користувач (при створенні імен полів бажано не використовувати пропусків), назви типів є стандартні, а значення полів впливають зі змісту конкретної задачі.

Отже, *структура таблиці* — це структура запису, тобто сукупність назв полів, їхніх типів та властивостей, визначених користувачем під час аналізу конкретної задачі. Структура визначає послідовність розташування даних у записі на фізичному носії і вигляд даних на екрані.

СУБД VFP — це реляційна база даних. Кожна таблиця зберігається в окремому файлі з розширенням dbf. Усі інші об'єкти — форми (form), запити (query), звіти (report), програми (program), меню (menu), уявлення (view) теж зберігаються в окремих файлах з відповідними типами.

#### Типи даних

Дані поділяються на змінні бази даних (поля), змінні пам'яті (використовуються для проміжного зберігання да них) та масиви змінних пам'яті. Ім'я змінної може мати довжину до 10 символів, містити літери від А до Z, всі цифри та знак підкреслювання (—). У таблиці 2.2 перелічені типи даних, які можуть приймати змінні.

Таблиця 2.2 – Типи даних та їх характеристика

Тип даних	Характеристика
Character	Може містити всі символи клавіатури, максимальна довжина — 254
Currency	Грошовий тип, може приймати значення від -900Е8 до +900Е8, містить 4 дробові розряди
Float	Може містити цифри, десяткову крапку. Максимальна довжина поля — 20 символів
Numeric	Може містити цифри, десяткову крапку. Максимальна довжина поля — 20 символів (ціла частина + дробова частина + 1, якщо є десяткова крапка)
Date	Містить дату в такому вигляді: місяць/число/рік, наприклад, 10/31/01
Date Time	Містить дату та час, наприклад, 10/31/01 11:59 PM
Double	Може містити числові дані, але обчислення виконуються з більшою точністю, ніж з даними типу Numeric
Logical	Логічний тип даних. Може приймати два значення Т (True) та F (False)
Memo	Дозволяє зберігати текст необмеженого розміру. Дані у цьому випадку зберігаються в іншому файлі
General	Може містити OLE-об'єкти, компоненти Windows, об'єкти, що створені в інших додатках
Character (binary)	Може містити будь-які 8-бітні значення та символ null (0)
Memo (binary)	Дозволяє зберігати відскановані зображення, оцифровану музику тощо.

Серед вимог, що пред'являються до СУБД, основне місце займає можливість швидкого пошуку необхідної інформації. Прекрасним засобом, що дозволяє вирішити цю проблему, є використання індексів, які, в залежності від кількості полів, що застосовуються при їх створенні, поділяються на прості і складові. На відміну від багатьох реляційних баз даних, при визначенні складеного індексу в Visual FoxPro можна використовувати не просто список

індексних полів, а індексний вираз, при формуванні якого застосовуваних найменування полів таблиці.

Індекси в Visual FoxPro призначаються також для створення первинних ключів, використовуваних при визначенні відношень між таблицями і умов цілісності даних. В цьому випадку індекси повинні бути унікальними. Часто для створення унікального індексу використовується складовий індекс, тобто він може складатися з декількох полів. У цьому випадку кожне поле окремо може містити повторювані значення, але індексний вираз, до складу якого входять ці поля, має бути унікальним.

Значення індексів зберігаються в індексному файлі, в якому для кожного значення індексу існує унікальне посилання, яке вказує на місцезнаходження в таблиці запису, відповідного індексу, що забезпечує при пошуку потрібної інформації прямий доступ до запису на підставі упорядкованих значень індексу, а не послідовний перегляд всієї таблиці. Це природно, прискорює пошук.

Для створення індексу таблиці використовується вкладка Indexes (Індекси) вікна конструктора таблиці Table Designer (Конструктор таблиці).

Всі індекси в Visual FoxPro мають імена, що задаються в полі Name (Ім'я). Зліва від імені індексу в стовпці Order (Упорядкування) розташовується перемикач, який визначає порядок, в якому будуть вибудовуватися значення індексного виразу. За замовчуванням при створенні індексу в даному полі з'являється стрілка, спрямована вгору. Це означає, що значення індексного вираження впорядковані за зростанням. Якщо стрілка спрямована вниз, це говорить про те, що значення впорядковані за спаданням.

Список Type (Тип) використовується для завдання типу створюваного індексу і містить значення, описані в табл. 2.3.

Таблиця 2.3 – Опис типів індексу

Тип індексу	Опис
Regular (Звичайний)	Створюється індекс, в якому для кожного запису таблиці зберігається значення індексного виразу. Якщо кілька записів мають однакове значення індексного виразу, то кожне значення зберігається окремо і містить посилання на пов'язаний з нею запис
Unique (Унікальний)	Створюється індекс, в якому зберігаються тільки неповторним значення індексного виразу. Якщо дві або більше записів містять однакове значення індексного виразу, то буде зберігатися тільки одне значення і посилання на першу із записів з однаковим значенням індексного виразу. Таблиця може мати кілька унікальних індексів
Candidate (Кандидат)	Створюється унікальний індекс, який не містить полів з порожніми значеннями. Цей індекс має всі якості первинного ключа і не є ним тільки з тієї причини, що таблиця не може містити більше одного первинного ключа
Primary (Первинний)	Створюється унікальний індекс, який використовується для зв'язування таблиць і визначення умов цілісності даних. Поля, що входять до первинного ключа, не повинні допускати введення порожніх значень. На відміну від унікального індексу, таблиця може мати тільки один первинний ключ

## Завдання

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. Відкрити БД SKLAD.dbf. Проглянути зміст і структуру. Описати структуру і використаний тип даних. Змінити структуру згідно варіанту. Роздрукувати структуру особистої БД.
- III. Роздрукувати зміст особистої БД. На друк і дисплей вивести мінімальне число записів, що відповідають умовам згідно варіанту.
- IV. Створити в структурному індексному файлі індекси згідно варіанту.
- V. Активізувати кожний індекс. Проглянути командою BROWSE результати активізації кожного індексу.
- VI. Помітити на видалення записи за умов згідно варіанту. Відновити видалені записи. Видалити не більше 3 записів фізично.
- VII. Додати декілька нових записів. Ввести в нові рядки інформацію по змісту.
- VIII. Самостійно вивчити команду BROWSE. Уміти виводити обмежене число полів, виводити розрахункове поле. Озаглавити кожне поле, що виводиться, заборонити редагування поля, проглядати записи за умовою.
- IX. Вікно Command вивести на друк.  
**УВАГА!** Індивідуальні варіанти завдань до лабораторної роботи наведені в Додатку А.

## Порядок виконання роботи

- I. В свою особисту папку скопіювати файл SKLAD.dbf.
- II. Для виконання основної частини роботи доцільно буде використати вікно DATA SESSION з меню WINDOWS:  
OPEN – відкрити БД  
BROWSE – проглянути зміст БД  
MODIFY – проглянути і змінити структуру БД  
DISPLAY STRUCTURE TO PRINT – роздрукувати структуру БД
- III. Для того, щоб вивести на дисплей записи, що відповідають умові, скористуємося вікном DATA SESSION:  
PROPERTIES – проглянути і відредагувати поточні властивості таблиці  
DATA FILTER – фільтр даних  
EXPRESSION BUILDER – будівник виразів (умов).  
Для того, щоб роздрукувати записи, що відповідають умові, скористуємося вікном COMMAND WINDOW:  
LIST FOR <умова> TO PRINT  
Для того, щоб вивести в файл записи, що відповідають умові, скористуємося вікном COMMAND WINDOW:  
LIST FOR <умова> TO FILE D:\СУБД\<особиста папка>\<назва файлу>  
**УВАГА!** Якщо для запису умови використовуються значення, що складається з літер російської або української абетки, їх потрібно копіювати з БД і при побудові виразу брати у лапки.

- IV. Для створення індексів скористуємося вікном DATA SESSION:  
 PROPERTIES – проглянути і відредагувати поточні властивості таблиці  
 MODIFY – проглянути і змінити структуру БД
- V. Для активізації і перегляду результатів скористуємося вікном DATA SESSION:  
 PROPERTIES – проглянути і відредагувати поточні властивості таблиці  
 INDEX ORDER – активізація індексу  
 BROWSE – проглянути результат
- VI. Для виконання двох наступних пунктів завдання SKLAD.dbf потрібно відкрити в режимі перегляду:  
 DATA SESSION - BROWSE.  
 Потім потрібно скористатися командами з меню TABLE:  
 DELETE RECORDS – помітити на видалення записи  
 RECALL RECORDS – відновити записи, що помічені на видалення  
 REMOVE DELETED RECORDS – видалити фізично записи, що помічені на видалення
- VII. Меню TABLE - APPEND NEW RECORD – додати новий запис або  
 МенюVIEW – APPEND MODE – режим додатку записів
- VIII. Синтаксис команди BROWSE вивчаємо з інформаційних та літературних джерел.
- IX. Меню FILE – PRINT – роздрукувати вікно Command  
**УВАГА!** Зміст вікна COMMAND зберігається лише в поточному сеансі роботи. Тому в кінці кожного заняття перед закриттям програми потрібно зберігати команди, що є у командному вікні в спеціальному файлі.

### Вимоги до звіту

У звіті навести:

- тему та мету лабораторної роботи,
- визначення, класифікація та структурні елементи бази даних, типи даних;
- відповідь на індивідуальне питання<sup>2</sup>,
- текст завдання,
- порядок виконання /алгоритм/ кожного пункту завдання та його результати,
- аналіз команд вікна Command,
- висновки по отриманим результатам роботи,
- список використаних джерел.

Звіт повинен бути оформленим українською мовою на аркушах паперу формату А4. Основний текст повинен бути оформлений з наступними параметрами: Шрифт – TimesNewRoman, кегль – 14, міжрядковий інтервал – 1.

---

<sup>2</sup> Обирається з переліку контрольних питань за номером прізвища студента зі списку в журналі академічної групи

Звіт оформлюється згідно ДСТУ 3008:2015 “Звіти у сфері науки і техніки. Структура та правила оформлювання”, чинний від 2017-07-01.

### **Контрольні питання**

1. Типова організація сучасної СУБД. Ядро СУБД. Основні характеристики СУБД.
2. Буферизація даних в оперативній пам'яті. Призначення Журнала в системі організації роботи з інформаційними засобами СУБД.
3. Поняття м'якого збою і простого відновлення БД. Створення і призначення архівної копії БД засобами СУБД.
4. Послідовність дій при організації зчитування та запису в СУБД.
5. Команди додавання і вилучення записів /APPEND, INSERT, DELETE, RECALL/.
6. Команди створення структури БД /CREATE, MODIFY STRUCTURE, COPY STRUCTURE/.
7. Загальні характеристики розподіленої та паралельної БД.
8. Індксація інформації в БД. Простий індксний і мультиіндксний файл. Активізація індксу.
9. Редагування записів БД. Команда BROWSE. Опції команди BROWSE.
10. Які команди існують для виконання типових операцій в СУБД.

## Лабораторна робота №3

**Тема:** Створення проекту, побудова запитів та форм в інструментальному середовищі VISUAL FOXPRO

**Мета роботи:** освоєння засобів створення проекту, формування запитів до БД та форм засобами VISUAL FOXPRO. Уміти створювати запит на вибірку, вводити параметри в запит, виводити результати запиту. Уміти створювати форму, вводити, переглядати і редагувати дані за допомогою форми.

### Теоретичні відомості

Для відображення і редагування даних використовуються форми, звіти, запити і програми. Щоб створювати форми, звіти і запити, застосовуються конструктори. Тому ці компоненти часто називають конструкторськими об'єктами. Форми і звіти є складовими об'єктами, так як вони складаються з більш дрібних об'єктів (таких, як поля, кнопки, діаграми, рамки, OLE-компоненти і т. п.), які називаються об'єктами інтерфейсу.

Форми використовуються для перегляду або введення даних в таблиці. Дані можна вводити в таблиці і безпосередньо, але, застосовуючи форми, можна значно прискорити цей процес і зробити його більш ефективним. Форма містить деякі або всі поля таблиць, в які вводиться інформація. Щоб створювати форми, можна використовувати майстер форм або конструктор форм. Майстер форм надає цілий ряд шаблонів. Конструктор форм призначений для створення складних форм.

Процес створення форми складається з наступних дій:

- налаштування параметрів форми;
- визначення середовища оточення, тобто вибір використовуваних у формі таблиць і установка зв'язків між ними;
- розміщення у формі об'єктів: тексту, полів для введення, ліній, малюнків, кнопок управління;
- настройка властивостей розміщених у формі об'єктів.

Для створення середовища оточення форми призначено діалогове вікно Data Environment. Розміщення об'єктів у формі здійснюється за допомогою будівника форми або за допомогою вікна FormControls.

Форма, що проектується розміщується у вікні Конструктора форм, а його головним інструментом є вікно Properties. Конструктор форм завантажується автоматично при створенні нової або модернізації існуючої форми. В головному вікні Visual FoxPro з'являється меню Form, за допомогою якого можна виконати команди:

- New Property - дозволяє додати нову властивість для форми, що створюється;
- New Method - дозволяє додати новий метод для форми;
- Edit Property/Method - знищити додані властивості і методи з опису форми або змінити коментар;
- Include File - визначити файл, що містить директиви компілятора;

- Create Form Set - створити набір форм - об'єкт, який буде контейнером відносно кількох форм і панелей інструментів;
- Remove Form Set - знищити набір форм;
- Add New Form - додати форм у набір форм;
- Remove Form - знищити форм у набору форм;
- Quick Form - запускає будівельника форм, за допомогою якого можна швидко створювати просту форм у для роботи з даними з однієї таблиці і в подальшому використовувати як заготовку в Конструкторі форм;
- Run Form - запустити форм у, що розробляється.

Запити є засобом вибірки даних з однієї або декількох таблиць. В Visual FoxPro для створення запиту можна використовувати як конструктор запитів, так і спеціалізовану мову Structured Query Language (SQL). Результати виконання запиту можуть відображатися у формі, виводитись у вигляді звітів і діаграм або зберігатися в зазначеній таблиці.

Майстер запитів (Query Wizard) дозволяє в інтерактивному режимі встановити умови для вибірки даних з кількох таблиць. За його допомогою можна легко встановити умови зв'язку між таблицями. прості умови відбору даних і задати порядок розміщення записів в результаті, що виводиться у вікні Browse.

За допомогою конструктора запитів Visual FoxPro ви можете формувати різної складності критерії для вибору записів з однієї або декількох таблиць, вказуючи при цьому, які поля повинні бути відображені в запиті. Над полями, що обираються з таблиць за допомогою запитів, можна виконувати різні обчислення. Результатом запиту є таблиця, яку ви можете зберегти в масиві, в створюваної новій таблиці, відобразити на екрані в режимі Browse (Перегляд) або вивести у вигляді звіту. Призначення вкладок конструктора запитів наведено в табл. 3.1.

Таблиця 3.1 – Призначення вкладок вікна конструктора запитів

Вкладка	Призначення
Fields (Поля)	Дозволяє вказати поля вихідних таблиць, які обираються в результуючий запит
Join (Об'єднання)	Дозволяє задати умови об'єднання таблиць
Filter (Фільтр)	Дозволяє визначити фільтри, що накладаються для вибору записів
Order By (Впорядкування)	Дозволяє задати критерії впорядкування даних
Group By (Групування)	Дозволяє задати умови угруповання даних
Miscellaneous (Різне)	Дозволяє задати додаткові умови, такі як ознака вибірки повторюваних значень, кількість або відсоток вибору даних

Майстер перехресних запитів (Cross-Tab Wizard) дає змогу вибрати таблицю або представлення і на їх основі збудувати спеціальну таблицю у вигляді "шахматки". Як правило для її побудови використовується три



поля. Дані з першого поля розміщуються в першій колонці створюваного курсору, дані другого поля утворюють колонки, що залишилися, а дані третього поля розміщуються у відповідних комірках. Можна сумувати дані, задати представлення їх у вигляді процентів або підрахунок числа комірок, що відповідають даним в колонках і рядках.

Програми, написані мовою Visual FoxPro, є об'єктно-орієнтованими. За допомогою них обробляються форми, створюються об'єкти, здійснюються різні обчислення, проводиться керування БД. Для зручності роботи можна об'єднати програми в бібліотеки.

Щоб створювати форми в Visual FoxPro, можна використовувати не тільки базові класи, але і створювати власні, наприклад, визначити клас форм, в якому задано певний колір фону і стандартний набір кнопок для управління даними. Щоб стандартизувати розробку, корисно мати один або кілька користувацьких класів для кожного базового класу. Класи, створені в Visual FoxPro, зберігаються в бібліотеках класів.

Для об'єднання компонентів створюваного додатка використовується проект, в який включаються всі перелічені вище компоненти. Використання проекту спрощує розробку програми і її супровід.

Інформація про проект зберігається в спеціальній таблиці, яка, на відміну від звичайних таблиць Visual FoxPro, має розширення PJX. Мето-поля таблиці містять найменування елемента проекту, його опис і інші атрибути тексту. Файл з Мето-полями таблиці має розширення PJT. Використання проекту спрощує розробку додатки, т. к. в проекті бази даних, програми, форми, звіти, запити і інші елементи програми розташовуються у відповідних розділах, а також запам'ятовується розташування кожного включеного в проект елемента (вкладки та можливі файли наведені в таблиці 3.2).

Створивши проект і визначивши елементи, що входять до нього, ви можете використовувати його для збірки додатку, побудувавши файл з розширенням APP, або для створення виконуваного файлу з розширенням EXE. При побудові програми з проекту здійснюється пошук та збирання файлів, на які посилаються елементи додатка, відстежуються версії файлів, що входять в проект.

Таблиця 3.2 – Вкладки вікна Project Manager

Вкладка	Відображені файли
All (Все)	Всі файли
Data (Дані)	Бази даних, таблиці, запити, представлення даних, збережені процедури
Documents (Документи)	Форми, звіти і етикетки
Classes (Класи)	Класи
Code (Коди)	Програми і бібліотеки
Other (Решта)	Меню, клавишні макрокоманди

## Завдання

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
  - II. У особистій теці створити проект. Включити туди БД SKLAD.dbf.
  - III. Створити запити (включаючи їх у проект), використовуючи **МАЙСТЕР ЗАПИТІВ** згідно індивідуального завдання.
  - IV. Виконати перехресний запит.
  - V. Роздрукувати текст і властивості запитів.
  - VI. Самостійно вивчити команду **SELECT**, її призначення і синтаксис.
  - VII. Створити форму (включивши її до проекту), використовуючи **МАЙСТЕР ФОРМ** згідно індивідуального варіанту. Зберегти форму і відкрити її для внесення змін в Конструкторі форм. Проглянути результат.
  - VIII. Проглянути наступні 5 записів.
  - IX. Знайти і проглянути всі записи за умовою згідно індивідуального завдання.
  - X. Роздрукувати поточний запис.
  - XI. Додати запис про новий товар, який поступив на склад.
  - XII. Вікно Command вивести на друк.
- УВАГА!** Індивідуальні варіанти завдань до лабораторної роботи наведені в Додатку Б.

## Порядок виконання роботи

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. Створення проекту:  
FILE – NEW – PROJECT.  
Для додавання до проекту особової бази даних виконуємо наступні дії:  
PROJECT MANAGER – DATA- FREE TABLES – ADD
- III. Створення запитів через МАЙСТЕР ЗАПИТІВ:  
FILE – NEW – QUERY – WIZARD – QUERY WIZARD.  
Кроки QUERY WIZARD (МАЙСТРА ЗАПИТІВ):  
Step 1. Вибір початкової таблиці (початкових таблиць) та полів.  
Step 2. Зв'язки між таблицями (якщо запит ставиться до однієї таблиці, цей крок відсутній).  
Step 3. Введення умов вибору.  
Step 4. Сортування записів.  
Step 5. Фініш.  
Призначення вкладок QUERY DESIGNER (КОНСТРУКТОРА ЗАПИТІВ):  
Fields (Поля) – дозволяє вказати поля початкових таблиць, вибрані в результуючий запит;  
Join (Об'єднання) – дозволяє задати умови об'єднання таблиць;  
Filter (Фільтр) – дозволяє визначити фільтри, що накладаються для вибору записів;

Order (Впорядкування) – дозволяє задати критерії впорядкування даних;  
Group (Групування) – дозволяє задати умови групування даних;  
Miscellaneous (Різне) – дозволяє задати додаткові умови, такі, як ознака вибірки значень, що повторюються, кількість або відсоток вибору даних.  
Для того, щоб продивитись результат виконання запиту в меню QUERY потрібно вибрати команду RAN QUERY.

Для того, щоб переглянути текст і властивості запиту в меню QUERY потрібно вибрати команду VIEW SQL.

IV. Створення перехресного запиту:

FILE – NEW – QUERY – WIZARD – CROSS TAB WIZARD.

Кроки CROSS TAB WIZARD (МАЙСТРА СТВОРЕННЯ ПЕРЕХРЕСНИХ ТАБЛИЦЬ):

Step 1. Вибір початкової таблиці та полів.

Step 2. Визначення рядків, колонок та поля даних.

Step 3. Визначення типу групування та підсумкової колонки.

Step 4. Фініш.

V. SQL-оператори всіх створених запитів потрібно послідовно додавати в один, спеціально створений, текстовий файл, який необхідно роздрукувати.

VI. Синтаксис команди SELECT вивчаємо з навчальної літератури.

VII. Створення форм через МАЙСТЕР ФОРМ:

FILE – NEW – FORM – WIZARD – FORM WIZARD.

Кроки FORM WIZARD (МАЙСТРА ФОРМ):

Step 1. Вибір початкової таблиці та полів.

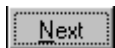
Step 2. Вибір стилю форми.

Step 3. Сортування записів.

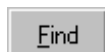
Step 4. Фініш.

Для того, щоб продивитись результат виконання форми в меню FORM потрібно вибрати команду RAN FORM.

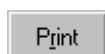
VIII. Для того, щоб продивитись наступні n кількість записів, потрібно використати наступні кнопки (залежно від виду кнопок):



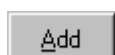
IX. Для того, щоб знайти і проглянути всі записи за умовою, потрібно використати наступні кнопки (залежно від виду кнопок):



X. Для того, щоб роздрукувати поточний запис, потрібно використати наступні кнопки (залежно від виду кнопок):



XI. Для того, щоб додати новий запис, потрібно використати наступні кнопки (залежно від виду кнопок):



## XII. Меню FILE – PRINT – роздрукувати вікно Command

**УВАГА!** Зміст вікна COMMAND зберігається лише в поточному сеансі роботи. Тому в кінці кожного заняття перед закриттям програми потрібно зберігати команди, що є у командному вікні, в спеціальному файлі.

### **Вимоги до звіту**

У звіті навести:

- тему та мету лабораторної роботи,
- поняття форми та розширені засоби введення даних: відображення даних у формі в табличному вигляді, створення багатотабличних форм для таблиць, що мають відношення "один-до-багатьох", створення форми з вкладками, створення форми з використанням таймера;
- відповідь на індивідуальне питання<sup>3</sup>,
- текст завдання,
- порядок виконання /алгоритм/ кожного пункту завдання та його результати,
- аналіз команд вікна Command,
- висновки по отриманим результатам дослідження,
- список використаних джерел.

Звіт повинен бути оформленим українською мовою на аркушах паперу формату А4. Основний текст повинен бути оформлений з наступними параметрами: Шрифт – TimesNewRoman, кегль – 14, міжрядковий інтервал – 1.

Звіт оформлюється згідно ДСТУ 3008:2015 “Звіти у сфері науки і техніки. Структура та правила оформлювання”, чинний від 2017-07-01.

### **Контрольні питання**

1. Запис SQL-операторів.
2. Мова SQL. Визначення схеми даних. Підтримка баз даних і доменів.
3. Мова SQL. Об'єднання таблиць.
4. Мова SQL. Підтримка курсорів та збережених процедур.
5. Мова SQL. Підтримка таблиць. Підтримка індексів та представлень.
6. Мова SQL. Підтримка тригерів. Підтримка системи захисту.
7. Мова SQL. Специфікатор ORDER BY.
8. Мова SQL. Специфікатори GROUP BY та HAVING.
9. Мова SQL. Специфікатори INTO та WHERE.
10. Мова SQL. Стандартизація мови SQL. Оператори SQL.
11. Мова SQL. Типи даних мови SQL.
12. Мова SQL. Формування запитів до бази даних.
13. Мова SQL. Функції агрегування.
14. Оператор SELECT та його специфікатори.
15. Призначення мови SQL. Типи даних SQL. Визначення схеми БД.

---

<sup>3</sup> Обирається з переліку контрольних питань за номером прізвища студента зі списку в журналі академічної групи

16. Призначення мови SQL. Визначення даних. Ідентифікатори мови SQL.
17. Призначення мови SQL. Відбір рядків даних результуючої таблиці запиту.
18. Призначення мови SQL. Маніпулювання даними. Групування результатів (GROUP BY).
19. Призначення мови SQL. Маніпулювання даними. Зміна вмісту бази даних.
20. Призначення мови SQL. Маніпулювання даними. Об'єднання таблиць.
21. Призначення мови SQL. Маніпулювання даними. Підзапити.
22. Призначення мови SQL. Маніпулювання даними. Предикати мови SQL.
23. Призначення мови SQL. Маніпулювання даними. Прості запити. Сортування за значеннями одного стовпця (ORDER BY).
24. Призначення мови SQL. Маніпулювання даними. Узагальнюючі функції мови SQL.
25. Призначення мови SQL. Підсумовування інформації результуючої таблиці запиту.
26. Призначення мови SQL. Підтримка тригерів та збережених процедур.
27. Призначення мови SQL. Створення баз даних. Створення таблиць (оператор CREATE TABLE). Видалення таблиць (оператор DROP TABLE).
28. Призначення мови SQL. Створення індексу (оператор CREATE INDEX). Видалення індексу (оператор DROP INDEX).
29. Призначення мови SQL. Визначення даних. Типи даних мови SQL, визначені стандартом ISO.

## Лабораторна робота №4

**Тема:** Створення звітів і етикеток в інструментальному середовищі VISUAL FOXPRO

**Мета роботи:** освоєння засобів створення звітів і етикеток засобами FOXPRO. Знати і використовувати засоби побудови звітних вихідних форм. Уміти формувати швидкий звіт, редагувати його в конструкторі звіту, включати в звіт обчислюване поле. Навчитись переглядати, друкувати, зберігати і модифікувати звіт. Освоїти засоби побудови етикеток. Уміти формувати етикетки, переглядати, друкувати, зберігати і модифікувати етикетки.

### Теоретичні відомості

Звіти використовуються для друку інформації, що міститься в БД. Прикладами звітів є прайс-лист товарів, список покупців, складська відомість, тощо. Як правило, звіти створюються в тому випадку, якщо інформацію необхідно передавати будь-кому в друкованому вигляді.

В Visual FoxPro для створення звітів можна використовувати такі засоби.

- Report Wizard (Майстер звіту).
- Report Designer (Конструктор звіту).
- Quick Report (Швидкий звіт).

Report Wizard (Майстер звіту) дозволяє досить швидко створити звіт, застосовуючи сортування, угруповання даних і заданий вами стиль оформлення. Для створення готового звіту за допомогою майстра досить відповісти на ряд питань Report Wizard (Майстер звіту) Visual FoxPro. У конструкторі звітів ви можете розробляти або модифікувати звіти, створені за допомогою майстра. Quick Report (Швидкий звіт) призначений для розміщення в конструкторі звіту полів і завдання середовища оточення.

За допомогою майстра звітів можна швидко створити власний звіт на основі наявних шаблонів. Застосовуючи, конструктор звітів, можна створювати звіти довільної складності, включаючи багаторівневе угруповання даних і розміщення обчислюваних полів.

Для розробки звітів можна вибрати одну з таких підпрограм-Майстрів:

- Майстер звітів (Report Wizard) дозволяє створити простий звіт, що використовує дані з однієї таблиці, локального або зовнішнього представлення.
- Майстер звітів з групуванням даних і друком підсумків (Group/Total Report Wizard) дозволяє створити підсумковий звіт.
- Майстер звітів “один-до-багатьох” (One-to-Many Report Wizard) дозволяє створити звіт, який буде ґрунтуватися на даних двох таблиць, що пов’язані відношенням “один-до-багатьох”.

Розглянемо розширені засоби створення звітів. На практиці часто виникає необхідність в створенні звітів, призначених для розсилки інформації рекламного характеру потенційним клієнтам або розсилки відомостей про результати діяльності підприємства за звітний період. В Visual FoxPro для цих

цілей є спеціальний тип звітів і відповідний майстер для створення поштових етикеток.

Етикетка - це спеціальний вид звіту, що як правило, роздруковується на клейкій стрічці. Найчастіше етикетки використовуються для друку адрес і відомостей про адресатів, які надалі наклеюються на конверти або для друку цінників і наклейок. При створенні етикеток доцільно спочатку використовувати Label Wizard (Майстер етикеток), а потім Label Designer (Конструктор етикеток) - для додавання рисунків, графіки та інших елементів оформлення.

Майстер етикеток допомагає створити етикетку на основі даних з однієї таблиці і правильно розмістити етикетки на листі для наступного виведення на друк.

Вікно Конструктора етикеток виглядає приблизно так само, як вікно конструктора звітів. Відмінність полягає в тому, що для етикетки задані розміри звіту і кількість колонок в звіті. В іншому створення етикетки аналогічно створенню звіту.

Основна перевага етикеток, в порівнянні зі звітами, полягає в автоматичному виборі розмірів етикетки і відстані між ними. Не має необхідності самостійно розраховувати розмір одержуваного звіту і піклуватися про те, щоб він розмістився на папері. При друку на звичайному папері ця перевага не цілком очевидно, але як тільки ви почнете використовувати для цих цілей спеціальні листи з наклейками, ви відчуєте зручність вибору типу наклейки.

### Завдання

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. Створити звіт згідно індивідуального варіанту, використовуючи **МАЙСТЕР ЗВІТІВ**.
- III. Зберегти звіт у свій проект і вивести на друк першу і останню сторінки.
- IV. Самостійно вивчити **QUICK REPORT** і його можливості.
- V. Створити етикетки згідно індивідуального варіанту, використовуючи **МАЙСТЕР ЕТИКЕТОК**. Зберегти етикетки в особистому проекті і відкрити для коректування в **КОНСТРУКТОРІ ЕТИКЕТОК**. Переглянути і роздрукувати результат.
- VI. Вікно Command вивести на друк.
- VII. **УВАГА!** Індивідуальні варіанти завдань до лабораторної роботи наведені в Додатку В.

### Порядок виконання роботи

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. Створення звітів через **МАЙСТЕР ЗВІТІВ**:  
FILE – NEW – REPORT – WIZARD – REPORT WIZARD.

## Кроки REPORT WIZARD (МАЙСТРА ЗВІТІВ):

Step 1. Вибір початкової таблиці та полів.

Step 2. Визначення полів для групування даних у звіті. Формування у звіті сумарних значень.









Step 3. Вибір стилю відображення об'єктів у звіті.

Step 4. Встановлення порядку розміщення об'єктів у звіті.

Step 5. Встановлення критерію впорядкованості даних.

Step 6. Фініш.

## Кнопки панелі інструментів REPORT CONTROLS

	Select Objects (Вибір об'єктів)	Є вказівником вибору об'єктів звіту
	Label (Мітка)	Розміщує текст
	Field (Поле)	Розміщує поля
	Line (Лінія)	Малює лінії
	Rectangle (Прямокутник)	Малює прямокутники
	Rounded Rectangle (Скруглений прямокутник)	Малює прямокутники зі скругленими кряями
	Picture/ActivX Bound Control (Зображення/ActivX об'єкт)	Розміщує в звіт рисунок
	Button Lock (Закріплювач кнопки)	Закріплює вибір кнопки

## Типи смуг звіту:

Title (Титул)	У цій смузі розміщується інформація, що з'являється перед основним звітом і звана титульною. Це може бути ім'я звіту, супровідний лист або будь-які дані, які необхідно помістити на першій сторінці звіту.
Page Header (Верхній колонтитул)	Ця смуга в звіті називається верхнім колонтитулом. Дані, вміщені в смугу, друкуються на початку кожної сторінки. Прикладами елементів управління можуть бути назва звіту, поточна дата, номер сторінки і т.д.
Group Header (Група зверху)	У цій смузі друкуються інформація, яка використовується при групуванні. При групуванні даних група може мати верхні смуги, що друкуються до неї. Вони допомагають ідентифікувати інформацію, що міститься на кожному рівні групування
Detail (Деталі)	Ця смуга містить дані полів з таблиці або результат обчислень над ними
Group Footer (Група знизу)	У смузі розміщується підсумкова інформація по групі
Page Footer (Нижній колонтитул)	У нижньому колонтитулі друкуються назва звіту, дата, номер сторінки і підсумкові значення за даними поточної сторінки
Summary (Підсумки)	У підсумковій частині звіту міститься інформація, що з'являється один раз після основного звіту і містить підсумкове значення або заключний текст



III. Меню FILE – PRINT

IV. **QUICK REPORT** вивчаємо з навчальної літератури.

V. Створення етикеток через МАЙСТЕР ЕТИКЕТОК:

FILE – NEW – LABEL – WIZARD – REPORT LABEL.

Кроки REPORT LABEL (МАЙСТРА ЕТИКЕТОК):

Step 1. Вибір початкової таблиці.

Step 2. Вибір типу етикетки.

Стовпець Avery містить найменування етикеток. В стовпці Dimensions (Розміри) відображається розмір етикетки, а в стовпці Columns (Колонки) - кількість колонок етикеток, що поміщаються на сторінці. За умовчанням розмір етикеток указується в дюймах, а при установці опції Metric - в міліметрах.

Step 4. Визначення розташування елементів етикетки.

На третьому кроці створення етикетки необхідно розташувати поля, знаки пунктуації (крапка, кома, дефіс, пропуск, двокрапка), сформувавши текст етикетки. Етикетка заповнюється послідовно, тому перед перенесенням в неї об'єктів ретельно сплануйте їх розміщення в етикетці. Для формування етикетки виконайте наступні дії:

- Перенесіть із списку Available Fields (Наявні поля) в список Selected Fields (Вибрані поля) необхідне поле. Для цього двічі клацніть на полі або натисніть кнопку із стрілкою.
- Щоб розмістити кому, натисніть кнопку із зображенням коми. Для розміщення пропуску перед назвою країни натисніть кнопку Space (Пропуск) і т.д.
- Якщо необхідно розташувати дані з нового рядка, натисніть кнопку із зображенням зігнутої стрілки.
- Аналогічно розташуєте решту полів етикетки.

Step 5. Встановлення критерію впорядкованості даних.

Step 6. Фініш.

VI. Меню FILE – PRINT – роздрукувати вікно Command

**УВАГА!** Зміст вікна COMMAND зберігається лише в поточному сеансі роботи. Тому в кінці кожного заняття перед закриттям програми потрібно зберігати команди, що є у командному вікні, в спеціальному файлі.

### Вимоги до звіту

У звіті навести:

- тему та мету лабораторної роботи,
- відповідь на індивідуальне питання<sup>4</sup>,
- текст завдання,
- порядок виконання /алгоритм/ кожного пункту завдання та його

---

<sup>4</sup> Обирається з переліку контрольних питань за номером прізвища студента зі списку в журналі академічної групи

результати,

- аналіз команд вікна Command,
- висновки по отриманим результатам дослідження,
- список використаних джерел.

Звіт повинен бути оформленим українською мовою на аркушах паперу формату А4. Основний текст повинен бути оформлений з наступними параметрами: Шрифт – TimesNewRoman, кегль – 14, міжрядковий інтервал – 1.

Звіт оформлюється згідно ДСТУ 3008:2015 “Звіти у сфері науки і техніки. Структура та правила оформлювання”, чинний від 2017-07-01.

### **Контрольні запитання**

1. Засоби створення звітів
2. Використання майстра для створення однотобличного звіту
3. Вікно конструктора звітів. Типи смуг вікна конструктора звіту
4. Використання команди Quick Report для розміщення полів
5. Розміщення об'єктів в звіті
6. Розміщення в звіті текстової інформації
7. Додавання в звіт областей заголовка і підсумків
8. Розміщення в звіті полів, ліній і прямокутників
9. Панель інструментів Color Palette
10. Розміщення в звіті рисунків
11. Створення звіту у довільній формі
12. Створення табличного звіту
13. Угруповання даних в звіті
14. Розмітка сторінки звіту
15. Створення багатоколлонкового звіту
16. Використання в звітах представлень даних
17. Використання майстра при створенні етикеток
18. Створення етикетки за допомогою конструктора
19. Використання в звітах відношень "один-до-багатьох"
20. Підготовка до створення меню. Дії, необхідні для створення меню
21. Запуск конструктора меню, створення меню
22. Завдання параметрів пункту меню в діалоговому вікні Prompt Options
23. Роздільники елементів меню
24. Завдання графічних зображень пунктам меню
25. Визначення параметрів меню
26. Збереження, генерація і запуск меню
27. Створення спливаючого меню

## СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

1. Конноли Т. Бегг К. Базы данных. Проектирование, реализация и сопровождение. Пер. с англ. – М.: Изд. дом «Вильямс», 2003
2. Дейт К. Введение в системы баз данных. – М., С-Пб., К.: Вильямс, 2005
3. Гайна Г. А. Основи проектування баз даних: Навч. посібник. – К.: Кондор, 2007. – 208 с.
4. Зрюмов Е.А., Зрюмова А.Г. Базы данных для инженеров: Учебное пособие. - Барнаул: Изд-во АлтГТУ, 2010. – 131 с.
5. Илюшечкин, В. М. Основы использования и проектирования баз данных / В. М. Илюшечкин. – М.: Высшее образование, 2009. – 213 с.
6. Кузин А. В. Базы данных / А. В Кузин, С. В. Левонисова. – М.: Академия, 2008. – 320 с.
7. Кузнецов, С. Д. Основы баз данных: учебное пособие / С. Д. Кузнецов. – М.: Бином, 2007. – 484 с.
8. Гайдаржи В.І., Дацюк О.А. Основи проектування та використання баз даних: Навч. посіб. – 2-ге вид., виправл. і доповн. – К.: ІВЦ «Видавництво «Політехніка»», ТОВ «Фірма «Періодика»», 2004
9. Омельченко Л. Н. Самоучитель Visual Foxpro 8. — СПб.: БХВ-Петербург, 2003. — 688 с.
10. Грицунов О.В. Інформаційні системи та технології: навч. Посіб Х.: ХНАМГ, 2010.
11. Завадський І.О. Основи баз даних: навч. посіб. – К.: Видавець І.О. Завадський, 2011.

## Додаток А

### Індивідуальні завдання до лабораторної роботи № 2

#### ВАРІАНТ 1

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. Відкрити БД SKLAD.dbf. Проглянути зміст і структуру. Описати структуру і використаний тип даних. Змінити структуру: додати поле G1, тип Logical . Роздрукувати структуру особистої БД.
- III. Роздрукувати зміст особистої БД. На друк і дисплей вивести мінімальне число записів, що відповідають умовам:  
NAME= МИКРОВОЛНОВАЯ ПЕЧЬ МК-1,  
CENA>100000,  
KOL=15.
- IV. Створити в структурному індексному файлі наступні індекси: NAME, CENA, NOM, FIRMA+NAME.
- V. Активізувати кожний індекс. Проглянути командою BROWSE результати активізації кожного індексу.
- VI. Помітити на видалення записи за умов:  
наступні 21  
FIRMA=УКРОПТХОЗТОВАРЫ  
kol<=131.  
Відновити видалені записи.  
Видалити не більше 3 записів фізично.
- VII. Додати декілька нових записів. Ввести в нові рядки інформацію по змісту.
- VIII. **Самостійно вивчити команду BROWSE.** Уміти виводити обмежене число полів, виводити розрахункове поле. Озаглавити кожне поле, що виводиться, заборонити редагування поля, проглядати записи за умовою.
- IX. Вікно Command вивести на друк.

#### ВАРІАНТ 2

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. Відкрити БД SKLAD.dbf. Проглянути зміст і структуру. Описати структуру і використаний тип даних. Змінити структуру: додати поле G2, тип Logical . Роздрукувати структуру особистої БД.
- III. Роздрукувати зміст особистої БД. На друк і дисплей вивести мінімальне число записів, що відповідають умовам:  
NAME= СИГНАЛЬНОЕ УСТРОЙСТВО,  
CENA>180000,  
KOL<5.
- IV. Створити в структурному індексному файлі наступні індекси: GOR, NAME, CENA, FIRMA+NAME.
- V. Активізувати кожний індекс. Проглянути командою BROWSE результати активізації кожного індексу.
- VI. Помітити на видалення записи за умов:  
наступні 7  
FIRMA=РКФ 'МАГИСТР',  
cena>=13100  
Відновити видалені записи.  
Видалити не більше 3 записів фізично.

- VII. Додати декілька нових записів. Ввести в нові рядки інформацію по змісту.
- VIII. **Самостійно вивчити команду BROWSE.** Уміти виводити обмежене число полів, виводити розрахункове поле. Озаглавити кожне поле, що виводиться, заборонити редагування поля, проглядати записи за умовою.
- IX. Вікно Command вивести на друк.

### ВАРІАНТ 3

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. Відкрити БД SKLAD.dbf. Проглянути зміст і структуру. Описати структуру і використаний тип даних. Змінити структуру: додати поле G3, тип Logical . Роздрукувати структуру особистої БД.
- III. Роздрукувати зміст особистої БД. На друк і дисплей вивести мінімальне число записів, що відповідають умовам:  
NAME= ИГРА `ВОЛШЕБНЫЙ КУБИК`,  
CENA=40000,  
KOL>=1000.
- IV. Створити в структурному індексному файлі наступні індекси: NAME, CENA, NOM, FIRMA+NAME.
- V. Активізувати кожний індекс. Проглянути командою BROWSE результати активізації кожного індексу.
- VI. Помітити на видалення записи за умов:  
наступні 11  
FIRMA=УКРОПТХОЗТОВАРЫ  
kol<=100.  
Відновити видалені записи.  
Видалити не більше 3 записів фізично.
- VII. Додати декілька нових записів. Ввести в нові рядки інформацію по змісту.
- VIII. **Самостійно вивчити команду BROWSE.** Уміти виводити обмежене число полів, виводити розрахункове поле. Озаглавити кожне поле, що виводиться, заборонити редагування поля, проглядати записи за умовою.
- IX. Вікно Command вивести на друк.

### ВАРІАНТ 4

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. Відкрити БД SKLAD.dbf. Проглянути зміст і структуру. Описати структуру і використаний тип даних. Змінити структуру: додати поле G4, тип Num, довжина поля 5, число десяткових знаків 2. Роздрукувати структуру особистої БД.
- III. Роздрукувати зміст особистої БД. На друк і дисплей вивести мінімальне число записів, що відповідають умовам:  
NAME= НАБОР ПЕРВОКЛАСНИКА,  
CENA<25,  
KOL=500.
- IV. Створити в структурному індексному файлі наступні індекси: NAME, CENA, NOM, GOR+NAME.
- V. Активізувати кожний індекс. Проглянути командою BROWSE результати активізації кожного індексу.
- VI. Помітити на видалення записи за умов:  
наступні 20  
FIRMA= ЦУМ

цена<=110

Відновити видалені записи.

Видалити не більше 3 записів фізично.

- VII. Додати декілька нових записів. Ввести в нові рядки інформацію по змісту.
- VIII. **Самостійно вивчити команду BROWSE.** Уміти виводити обмежене число полів, виводити розрахункове поле. Озаглавити кожне поле, що виводиться, заборонити редагування поля, проглядати записи за умовою.
- IX. Вікно Command вивести на друк.

#### ВАРІАНТ 5

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. Відкрити БД SKLAD.dbf. Проглянути зміст і структуру. Описати структуру і використаний тип даних. Змінити структуру: додати поле G5, тип Logical . Роздрукувати структуру особистої БД.
- III. Роздрукувати зміст особистої БД. На друк і дисплей вивести мінімальне число записів, що відповідають умовам:  
NAME= ПП-10,  
CENA=2500,  
KOL>=1500.
- IV. Створити в структурному індексному файлі наступні індекси: NAME, CENA, NOM, FIRMA+NAME.
- V. Активізувати кожний індекс. Проглянути командою BROWSE результати активізації кожного індексу.
- VI. Помітити на видалення записи за умов:  
наступні 5  
FIRMA=ПРИБОРОСТРОИТЕЛЬНЫЙ ЗАВОД  
kol=5.  
Відновити видалені записи.  
Видалити не більше 3 записів фізично.
- VII. Додати декілька нових записів. Ввести в нові рядки інформацію по змісту.
- VIII. **Самостійно вивчити команду BROWSE.** Уміти виводити обмежене число полів, виводити розрахункове поле. Озаглавити кожне поле, що виводиться, заборонити редагування поля, проглядати записи за умовою.
- IX. Вікно Command вивести на друк.

#### ВАРІАНТ 6

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. Відкрити БД SKLAD.dbf. Проглянути зміст і структуру. Описати структуру і використаний тип даних. Змінити структуру: додати поле G6, тип Character, довжина поля 5. Роздрукувати структуру особистої БД.
- III. Роздрукувати зміст особистої БД. На друк і дисплей вивести мінімальне число записів, що відповідають умовам:  
NAME=НКС-96М,  
CENA=150  
KOL<=3
- IV. Створити в структурному індексному файлі наступні індекси: NAME, CENA, KOL, GOR+NAME.
- V. Активізувати кожний індекс. Проглянути командою BROWSE результати активізації кожного індексу.

- VI. Помітити на видалення записи за умов:  
наступні 17  
FIRMA=НПТФ'АГРОПРОМ'  
kol<=4.  
Відновити видалені записи.  
Видалити не більше 3 записів фізично.
- VII. Додати декілька нових записів. Ввести в нові рядки інформацію по змісту.
- VIII. **Самостійно вивчити команду BROWSE.** Уміти виводити обмежене число полів, виводити розрахункове поле. Озаглавити кожне поле, що виводиться, заборонити редагування поля, проглядати записи за умовою.
- IX. Вікно Command вивести на друк.

#### ВАРІАНТ 7

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. Відкрити БД SKLAD.dbf. Проглянути зміст і структуру. Описати структуру і використаний тип даних. Змінити структуру: додати поле G7, тип Char, довжина поля 7. Роздрукувати структуру особистої БД.
- III. Роздрукувати зміст особистої БД. На друк і дисплей вивести мінімальне число записів, що відповідають умовам:  
NAME= ГРИЛЬ,  
CENA>=200000,  
KOL=400.
- IV. Створити в структурному індексному файлі наступні індекси: CENA, NOM, KOL, NNK, FIRMA+NAME.
- V. Активізувати кожний індекс. Проглянути командою BROWSE результати активізації кожного індексу.
- VI. Помітити на видалення записи за умов:  
наступні 15  
FIRMA=МАГАЗИН 'ОДЕЖДА',  
kol<=30  
Відновити видалені записи.  
Видалити не більше 3 записів фізично.
- VII. Додати декілька нових записів. Ввести в нові рядки інформацію по змісту.
- VIII. **Самостійно вивчити команду BROWSE.** Уміти виводити обмежене число полів, виводити розрахункове поле. Озаглавити кожне поле, що виводиться, заборонити редагування поля, проглядати записи за умовою.
- IX. Вікно Command вивести на друк.

#### ВАРІАНТ 8

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. Відкрити БД SKLAD.dbf. Проглянути зміст і структуру. Описати структуру і використаний тип даних. Змінити структуру: додати поле G8, тип Logical . Роздрукувати структуру особистої БД.
- III. Роздрукувати зміст особистої БД. На друк і дисплей вивести мінімальне число записів, що відповідають умовам:  
NAME=СИСТЕМА ЗАЩИТЫ СИРЕНА,  
CENA<=26,  
KOL=90.

- IV. Створити в структурному індексному файлі наступні індекси: NAME, CENA, NOM, FIRMA+NAME.
- V. Активізувати кожний індекс. Проглянути командою BROWSE результати активізації кожного індексу.
- VI. Помітити на видалення записи за умов:  
     наступні 10  
     FIRMA=УНИВЕРМАГ  
     Kol>6.  
     Відновити видалені записи.  
     Видалити не більше 3 записів фізично.
- VII. Додати декілька нових записів. Ввести в нові рядки інформацію по змісту.
- VIII. **Самостійно вивчити команду BROWSE.** Уміти виводити обмежене число полів, виводити розрахункове поле. Озаглавити кожне поле, що виводиться, заборонити редагування поля, проглядати записи за умовою.
- IX. Вікно Command вивести на друк.

### ВАРІАНТ 9

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. Відкрити БД SKLAD.dbf. Проглянути зміст і структуру. Описати структуру і використаний тип даних. Змінити структуру: додати поле G9, тип Num, довжина поля 6, число десяткових знаків 3. Роздрукувати структуру особистої БД.
- III. Роздрукувати зміст особистої БД. На друк і дисплей вивести мінімальне число записів, що відповідають умовам:  
     NAME= ЗАЩИТНЫЙ ЭКРАН,  
     CENA=2100,  
     KOL<3.
- IV. Створити в структурному індексному файлі наступні індекси: NAME, CENA, KOL, GOR+NAME.
- V. Активізувати кожний індекс. Проглянути командою BROWSE результати активізації кожного індексу.
- VI. Помітити на видалення записи за умов:  
     наступні 4  
     FIRMA=КРЫМКУЛЬТТОВАРЫ  
     CENA>=35000.  
     Відновити видалені записи.  
     Видалити не більше 3 записів фізично.
- VII. Додати декілька нових записів. Ввести в нові рядки інформацію по змісту.
- VIII. **Самостійно вивчити команду BROWSE.** Уміти виводити обмежене число полів, виводити розрахункове поле. Озаглавити кожне поле, що виводиться, заборонити редагування поля, проглядати записи за умовою.
- IX. Вікно Command вивести на друк.

### ВАРІАНТ 10

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. Відкрити БД SKLAD.dbf. Проглянути зміст і структуру. Описати структуру і використаний тип даних. Змінити структуру: додати поле G10, тип Logical. Роздрукувати структуру особистої БД.
- III. Роздрукувати зміст особистої БД. На друк і дисплей вивести мінімальне число записів, що відповідають умовам:



NAME=ЭЛЕКТРОДУХОВКА МЕЧТА,  
CENA=5200,  
KOL>850.

- IV. Створити в структурному індексному файлі наступні індекси: NAME, NNK, NOM, KOL, FIRMA+NAME.
- V. Активізувати кожний індекс. Проглянути командою BROWSE результати активізації кожного індексу.
- VI. Помітити на видалення записи за умов:  
наступні 22  
FIRMA=ООО `КВАРЦ`  
CENA<=50.  
Відновити видалені записи.  
Видалити не більше 3 записів фізично.
- VII. Додати декілька нових записів. Ввести в нові рядки інформацію по змісту.
- VIII. **Самостійно вивчити команду BROWSE.** Уміти виводити обмежене число полів, виводити розрахункове поле. Озаглавити кожне поле, що виводиться, заборонити редагування поля, проглядати записи за умовою.
- IX. Вікно Command вивести на друк.

## Додаток Б

### Індивідуальні завдання до лабораторної роботи № 3

#### ВАРІАНТ 1

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. У особистій теці створити проект. Включити туди БД SKLAD.dbf.
- III. Створити запити (включити їх у проект), використовуючи **МАЙСТЕР ЗАПИТІВ**:
  - ✓ Вибрати всі товари, доставлені з Кіровограда і Житомира.
  - ✓ Вибрати всі товари в алфавітному порядку, кількість яких на складі більше 500 шт. і ціна менше 1000 грн.
  - ✓ Вибрати всі товари, що є на складі, згрупувавши їх по найменуванню.
  - ✓ Вибрати товари, закупівельна вартість яких не менше 25000 грн.
  - ✓ Вибрати фірми, від яких були хоча б раз одержані товари з вказівкою останньої дати поставки.
  - ✓ Вибрати всі партії товару ГРИЛЬ, які є на складі в хронологічному порядку їх надходження.
  - ✓ Вибрати товари, які поступили на склад за накладною № 1088.
- IV. Виконати перехресний запит – згрупувати дані по найменуванню товару і найменуванню фірми-постачальника товару з вказівкою середньої ціни за товар.
- V. Роздрукувати текст і властивості запитів.
- VI. Самостійно вивчити команду **SELECT**, її призначення і синтаксис.
- VII. Створити форму (включивши її до проекту), використовуючи **МАЙСТЕР ФОРМ**.
  - ✓ Вибрати поля: Nnk, Name, Kol, Cena, Firma, Gog.
  - ✓ Вибрати стиль форми – Embossed.
  - ✓ Вибрати стиль кнопок – Text Buttons.
  - ✓ Сортунання – по полю Name (за збільшенням).
  - ✓ Зберегти форму і відкрити її для внесення змін в Конструкторі форм.Проглянути результат.
- VIII. Проглянути наступні 5 записів.
- IX. Знайти і проглянути всі записи за умовою:
  - Kol>10 and Cena<1500
  - Name= «Н» or Cena>10 (кожна умова окремо).
- X. Роздрукувати поточний запис.
- XI. Додати запис про новий товар, який поступив на склад.
- XII. Вікно Command вивести на друк.

#### ВАРІАНТ 2

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. У особистій теці створити проект. Включити туди БД SKLAD.dbf.
- III. Створити запити (включити їх у проект), використовуючи **МАЙСТЕР ЗАПИТІВ**:
  - ✓ Вибрати всі товари, доставлені з Харкова та Сімферополя.
  - ✓ Вибрати всі товари в алфавітному порядку, кількість яких на складі більше 50 шт. і ціна менше 100 грн.
  - ✓ Вибрати всі товари, що є на складі, згрупувавши їх по кількості.
  - ✓ Вибрати товари, закупівельна вартість яких не менше 2500 грн.
  - ✓ Вибрати фірми, від яких були хоча б раз одержані товари з вказівкою останньої дати поставки.
  - ✓ Вибрати всі партії товару ЗАЩИТНЫЙ ЭКРАН, які є на складі в хронологічному порядку їх надходження.

- ✓ Вибрати товари, які поступили на склад за накладною № 65.
- IV. Виконати перехресний запит – згрупувати дані по найменуванню товару і найменуванню фірми-постачальника товару з вказівкою мінімальної ціни за товар.
- V. Роздрукувати текст і властивості запитів.
- VI. Самостійно вивчити команду **SELECT**, її призначення і синтаксис.
- VII. Створити форму (включивши її до проекту), використовуючи **МАЙСТЕР ФОРМ**.
  - ✓ Вибрати поля: Nnk, Nom, Name, Kol, Cena, Firma.
  - ✓ Вибрати стиль форми – Standard.
  - ✓ Вибрати стиль кнопок – Picture Buttons.
  - ✓ Сортування – по полю Cena (за збільшенням).
  - ✓ Зберегти форму і відкрити її для внесення змін в Конструкторі форм.  
Проглянути результат.
- VIII. Проглянути наступні 5 записів.
- IX. Знайти і проглянути всі записи за умовою:
  - Kol>100 and Cena<1500
  - Name= «Г» і Cena>100 (кожна умова окремо).
- X. Роздрукувати поточний запис.
- XI. Додати запис про новий товар, який поступив на склад.
- XII. Вікно Command вивести на друк.

### ВАРІАНТ 3

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. У особистій теці створити проект. Включити туди БД SKLAD.dbf.
- III. Створити запити (включити їх у проект), використовуючи **МАЙСТЕР ЗАПИТІВ**:
  - ✓ Вибрати всі товари, доставлені з Черкас і Сум.
  - ✓ Вибрати всі товари в алфавітному порядку, кількість яких на складі більше 100 шт. і ціна менше 5000 грн.
  - ✓ Вибрати всі товари, що є на складі, згрупувавши їх по найменуванню фірми-постачальника.
  - ✓ Вибрати товари, закупівельна вартість яких не менше 4000 грн.
  - ✓ Вибрати міста, від яких були хоча б раз одержані товари з вказівкою останньої дати поставки.
  - ✓ Вибрати всі партії товару МИКРОВОЛНОВАЯ ПЕЧЬ МК-1, які є на складі в хронологічному порядку їх надходження.
  - ✓ Вибрати товари, які поступили на склад за накладною № 21.
- IV. Виконати перехресний запит – згрупувати дані по найменуванню товару і найменуванню фірми-постачальника товару з вказівкою максимальної ціни за товар.
- V. Роздрукувати текст і властивості запитів.
- VI. Самостійно вивчити команду **SELECT**, її призначення і синтаксис.
- VII. Створити форму (включивши її до проекту), використовуючи **МАЙСТЕР ФОРМ**.
  - ✓ Вибрати поля: -й, Name, Kol, Cena, Gor.
  - ✓ Вибрати стиль форми – Embossed.
  - ✓ Вибрати стиль кнопок – Text Buttons.
  - ✓ Сортування – по полю Gor (за збільшенням).
  - ✓ Зберегти форму і відкрити її для внесення змін в Конструкторі форм.  
Проглянути результат.
- VIII. Переглянути наступних 4 записи.
- IX. Знайти і проглянути всі записи за умовою:
  - Kol<50 and Cena<100

- Gor =КИРОВОГРАД or Name= РП (кожна умова окремо).
- X. Роздрукувати поточний запис.
- XI. Додати запис про новий товар, який поступив на склад.
- XII. Вікно Command вивести на друк.

#### ВАРІАНТ 4

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. У особистій теці створити проект. Включити туди БД SKLAD.dbf.
- III. Створити запити (включити їх у проект), використовуючи **МАЙСТЕР ЗАПИТІВ**:
  - ✓ Вибрати всі товари, доставлені з Шостки та Одеси.
  - ✓ Вибрати всі товари в алфавітному порядку, кількість яких на складі більше 300 шт. і ціна менше 100 грн.
  - ✓ Вибрати всі товари, що є на складі, згрупувавши їх за датою поставки.
  - ✓ Вибрати товари, закупівельна вартість яких не менше 250 грн.
  - ✓ Вибрати фірми, від яких були хоча б раз одержані товари з вказівкою останньої дати поставки.
  - ✓ Вибрати всі партії товару НАБОР ПЕРВОКЛАСНИКА, які є на складі в хронологічному порядку їх надходження.
  - ✓ Вибрати товари, які поступили на склад за накладною № 22.
- IV. Виконати перехресний запит – згрупувати дані по найменуванню товару і найменуванню фірми-постачальника товару з вказівкою середньої кількості товару.
- V. Роздрукувати текст і властивості запитів.
- VI. Самостійно вивчити команду **SELECT**, її призначення і синтаксис.
- VII. Створити форму (включивши її до проекту), використовуючи **МАЙСТЕР ФОРМ**.
  - ✓ Вибрати поля: Nnk, Nom, Name, Kol, Cena.
  - ✓ Вибрати стиль форми – Standard.
  - ✓ Вибрати стиль кнопок – Picture Buttons.
  - ✓ Сортування – по полю Kol (по убутанню).
  - ✓ Зберегти форму і відкрити її для внесення змін в Конструкторі форм.  
Проглянути результат.
- VIII. Переглянути наступних 4 записи.
- IX. Знайти і проглянути всі записи за умовою:
  - Kol>500 and Cena<1200
  - Name= ГРИЛЬ or Cena>100 (кожна умова окремо).
- X. Роздрукувати поточний запис.
- XI. Додати запис про новий товар, який поступив на склад.
- XII. Вікно Command вивести на друк.

#### ВАРІАНТ 5

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. У особистій теці створити проект. Включити туди БД SKLAD.dbf.
- III. Створити запити (включити їх у проект), використовуючи **МАЙСТЕР ЗАПИТІВ**:
  - ✓ Вибрати всі товари, доставлені з Києва і Дніпропетровська.
  - ✓ Вибрати всі товари в алфавітному порядку, кількість яких на складі менше 500 шт. і ціна більше 10000 грн.
  - ✓ Вибрати всі товари, що є на складі, згрупувавши їх по найменуванню міста-постачальника.
  - ✓ Вибрати товари, закупівельна вартість яких не більше 6000 грн.

- ✓ Вибрати фірми, від яких були хоча б раз одержані товари з вказівкою останньої дати поставки.
  - ✓ Вибрати всі партії товару СВЕТИЛЬНИК-БРА `ВЕЛЕНА`, які є на складі в хронологічному порядку їх надходження.
  - ✓ Вибрати товари, які поступили на склад за накладною № 55.
- IV. Виконати перехресний запит – згрупувати дані по найменуванню товару і найменуванню фірми-постачальника товару з вказівкою мінімальної кількості товару.
- V. Роздрукувати текст і властивості запитів.
- VI. Самостійно вивчити команду **SELECT**, її призначення і синтаксис.
- I. Створити форму (включивши її до проекту), використовуючи **МАЙСТЕР ФОРМ**.
- ✓ Вибрати поля: Nnk, Name, Kol, Cena, Firma.
  - ✓ Вибрати стиль форми – Embossed.
  - ✓ Вибрати стиль кнопок – Text Buttons.
  - ✓ Сортування – по полю Firma (по убуванню).
  - ✓ Зберегти форму і відкрити її для внесення змін в Конструкторі форм.  
Проглянути результат.
- II. Проглянути наступних 3 записи.
- III. Знайти і проглянути всі записи за умовою:
- Kol>220 and Cena<2500
  - Name= ГРИЛЬ or ЗАЩИТНЫЙ ЭКРАН (кожна умова окремо).
- IV. Роздрукувати поточний запис.
- V. Додати запис про новий товар, який поступив на склад.
- VII. Вікно Command вивести на друк.

#### ВАРІАНТ 6

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. У особистій теці створити проект. Включити туди БД SKLAD.dbf.
- III. Створити запити (включити їх у проект), використовуючи **МАЙСТЕР ЗАПИТІВ**:
- ✓ Вибрати всі товари, доставлені з Вінниці та Лохвиць.
  - ✓ Вибрати всі товари в алфавітному порядку, кількість яких на складі менше 70 шт. і ціна менше 1000 грн.
  - ✓ Вибрати всі товари, що є на складі, згрупувавши їх по найменуванню.
  - ✓ Вибрати товари, закупівельна вартість яких не менше 51000 грн.
  - ✓ Вибрати фірми, від яких були хоча б раз одержані товари з вказівкою останньої дати поставки.
  - ✓ Вибрати всі партії товару ЕЛЕКТРОКАМИН УЮТ, які є на складі в хронологічному порядку їх надходження.
  - ✓ Вибрати товари, які поступили на склад за накладною № 987.
- IV. Виконати перехресний запит – згрупувати дані по найменуванню товару і найменуванню фірми-постачальника товару з вказівкою максимальної кількості товару.
- V. Роздрукувати текст і властивості запитів.
- VI. Самостійно вивчити команду **SELECT**, її призначення і синтаксис.
- VII. Створити форму (включивши її до проекту), використовуючи **МАЙСТЕР ФОРМ**.
- ✓ Вибрати поля: Nnk, Nom, Name, Kol, Cena, Firma.
  - ✓ Вибрати стиль форми – Standard.
  - ✓ Вибрати стиль кнопок – Picture Buttons.
  - ✓ Сортування – по полю Cena (по убуванню).

- ✓ Зберегти форму і відкрити її для внесення змін в Конструкторі форм.  
Проглянути результат.
- VIII. Проглянути наступних 3 записи.
- IX. Знайти і проглянути всі записи за умовою:
  - Kol<10 and Cena<1200
  - Name= ЕЛЕКТРОУТЮГ or УСГГ (кожна умова окремо).
- X. Роздрукувати поточний запис.
- XI. Додати запис про новий товар, який поступив на склад.
- XII. Вікно Command вивести на друк.

### ВАРІАНТ 7

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. У особистій теці створити проект. Включити туди БД SKLAD.dbf.
- III. Створити запити (включити їх у проект), використовуючи **МАЙСТЕР ЗАПИТІВ**:
  - ✓ Вибрати всі товари, доставлені з Кіровограда і Черкас.
  - ✓ Вибрати всі товари в алфавітному порядку, кількість яких на складі більше 600 шт. і ціна менше 400 грн.
  - ✓ Вибрати всі товари, що є на складі, згрупувавши їх за номером накладної.
  - ✓ Вибрати товари, закупівельна вартість яких дорівнює 25 грн.
  - ✓ Вибрати міста, від яких були хоча б раз одержані товари з вказівкою останньої дати поставки.
  - ✓ Вибрати всі партії товару ЕЛЕКТРОНАСОС `ВИХРЬ`, які є на складі в хронологічному порядку їх надходження.
  - ✓ Вибрати товари, які поступили на склад за накладною № 107.
- IV. Виконати перехресний запит – згрупувати дані за найменуванням товару, датою поставки і кількістю товару з вказівкою сумарної кількості за кожним видом товару.
- V. Роздрукувати текст і властивості запитів.
- VI. Самостійно вивчити команду **SELECT**, її призначення і синтаксис.
- VII. Створити форму (включивши її до проекту), використовуючи **МАЙСТЕР ФОРМ**.
  - ✓ Вибрати поля: Name, Kol, Cena, Data, Firma, Gor.
  - ✓ Вибрати стиль форми – Embossed.
  - ✓ Вибрати стиль кнопок – Text Buttons.
  - ✓ Сортування – по полю Data (за збільшенням).
  - ✓ Зберегти форму і відкрити її для внесення змін в Конструкторі форм.  
Проглянути результат.
- VIII. Проглянути наступні 6 записів.
- IX. Знайти і проглянути всі записи за умовою:
  - Cena<21 and kol=30
  - Name= ДЕРЖАТЕЛЬ БУМАГИ / КХ/ or ГРИЛЬ (кожна умова окремо).
- X. Роздрукувати поточний запис.
- XI. Додати запис про новий товар, який поступив на склад.
- XII. Вікно Command вивести на друк.

### ВАРІАНТ 8

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. У особистій теці створити проект. Включити туди БД SKLAD.dbf.
- III. Створити запити (включити їх у проект), використовуючи **МАЙСТЕР ЗАПИТІВ**:
  - ✓ Вибрати всі товари, доставлені з Сум та Сімферополя.

- ✓ Вибрати всі товари в алфавітному порядку, кількість яких на складі більше 30 шт. і ціна більше 450 грн.
  - ✓ Вибрати всі товари, що є на складі, згрупувавши їх по найменуванню.
  - ✓ Вибрати товари, закупівельна вартість яких менше ніж 100 грн.
  - ✓ Вибрати фірми, від яких були хоча б раз одержані товари з вказівкою останньої дати поставки.
  - ✓ Вибрати всі партії товару ИГРА `ВОЛШЕБНЫЙ КУБИК`, які є на складі в хронологічному порядку їх надходження.
  - ✓ Вибрати товари, які поступили на склад за накладною № 30.
- IV. Виконати перехресний запит – згрупувати дані за найменуванням товару, датою поставки і кількістю товару з вказівкою процентного співвідношення кількості за кожним видом товару.
- V. Роздрукувати текст і властивості запитів.
- VI. Самостійно вивчити команду **SELECT**, її призначення і синтаксис.
- VII. Створити форму (включивши її до проекту), використовуючи **МАЙСТЕР ФОРМ**.
- ✓ Вибрати поля: Nom, Name, Kol, Cena, Data, Firma.
  - ✓ Вибрати стиль форми – Standard.
  - ✓ Вибрати стиль кнопок – Picture Buttons.
  - ✓ Сортування – по полю Name (за збільшенням).
  - ✓ Зберегти форму і відкрити її для внесення змін в Конструкторі форм.  
Проглянути результат.
- VIII. Проглянути наступні 6 записів.
- IX. Знайти і проглянути всі записи за умовою:
- Kol>100 and Cena<1500
  - Name= «Б» or Cena>100 (кожна умова окремо).
- X. Роздрукувати поточний запис.
- XI. Додати запис про новий товар, який поступив на склад.
- XII. Вікно Command вивести на друк.

### ВАРІАНТ 9

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. У особистій теці створити проект. Включити туди БД SKLAD.dbf.
- III. Створити запити (включити їх у проект), використовуючи **МАЙСТЕР ЗАПИТІВ**:
- ✓ Вибрати всі товари, доставлені з Херсону і Житомира.
  - ✓ Вибрати всі товари в алфавітному порядку, кількість яких на складі більше 70 шт. і ціна не менше 90 грн.
  - ✓ Вибрати всі товари, що є на складі, згрупувавши їх по кількості.
  - ✓ Вибрати товари, закупівельна вартість яких не більше 40 грн.
  - ✓ Вибрати фірми, від яких були хоча б раз одержані товари з вказівкою останньої дати поставки.
  - ✓ Вибрати всі партії товару УСГГ-91, які є на складі в хронологічному порядку їх надходження.
  - ✓ Вибрати товари, які поступили на склад за накладною № 57.
- IV. Виконати перехресний запит – згрупувати дані за найменуванням товару, датою поставки і середньою ціною товару з вказівкою сумарної кількості поставок за кожним видом товару.
- V. Роздрукувати текст і властивості запитів.
- VI. Самостійно вивчити команду **SELECT**, її призначення і синтаксис.
- VII. Створити форму (включивши її до проекту), використовуючи **МАЙСТЕР ФОРМ**.
- ✓ Вибрати поля: Nnk, Name, Kol, Cena, Data, Firma, Gor.

- ✓ Вибрати стиль форми – Embossed.
- ✓ Вибрати стиль кнопок – Text Buttons.
- ✓ Сортування – по полю Name (за збільшенням).
- ✓ Зберегти форму і відкрити її для внесення змін в Конструкторі форм.  
Проглянути результат.

VIII. Проглянути наступних 3 записи.

IX. Знайти і проглянути всі записи за умовою:

- Kol >= 320 and Cena < 60
- Name = РП or НКС (кожна умова окремо).

X. Роздрукувати поточний запис.

XI. Додати запис про новий товар, який поступив на склад.

XII. Вікно Command вивести на друк.

### ВАРІАНТ 10

I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.

II. У особистій теці створити проект. Включити туди БД SKLAD.dbf.

III. Створити запити (включити їх у проект), використовуючи **МАЙСТЕР ЗАПИТІВ:**

- ✓ Вибрати всі товари, доставлені з Шостки та Києва.
- ✓ Вибрати всі товари в алфавітному порядку, кількість яких на складі не більше 60 шт. і ціна менше 450 грн.
- ✓ Вибрати всі товари, що є на складі, згрупувавши їх за датою поставки.
- ✓ Вибрати товари, закупівельна вартість яких більше 110 грн.
- ✓ Вибрати міста, від яких були хоча б раз одержані товари з вказівкою останньої дати поставки.
- ✓ Вибрати всі партії товару РП-10, які є на складі в хронологічному порядку їх надходження.
- ✓ Вибрати товари, які поступили на склад за накладною № 1194.

IV. Виконати перехресний запит – згрупувати дані за найменуванням товару, датою поставки і мінімальною ціною товару з вказівкою сумарної кількості поставок за кожним видом товару.

V. Роздрукувати текст і властивості запитів.

VI. Самостійно вивчити команду **SELECT**, її призначення і синтаксис.

VII. Створити форму (включивши її до проекту), використовуючи **МАЙСТЕР ФОРМ.**

- ✓ Вибрати поля: Nom, Name, Kol, Cena, Data, Gor.
- ✓ Вибрати стиль форми – Standard.
- ✓ Вибрати стиль кнопок – Picture Buttons.
- ✓ Сортування – по полю Data (за збільшенням).
- ✓ Зберегти форму і відкрити її для внесення змін в Конструкторі форм.  
Проглянути результат.

VIII. Проглянути наступні 5 записів.

IX. Знайти і проглянути всі записи за умовою:

- Kol < 100 and Cena < 120
- Gor = КИРОВОГРАД or ЖИТОМИР (кожна умова окремо).

X. Роздрукувати поточний запис.

XI. Додати запис про новий товар, який поступив на склад.

XII. Вікно Command вивести на друк.



## Додаток В

### Індивідуальні завдання до лабораторної роботи № 4

#### ВАРІАНТ 1

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. Створити звіт, приведений нижче, використовуючи **МАЙСТЕР ЗВІТІВ**:

#### ТОВАРИ НА СКЛАДІ

	Артикул	Дата поставки	Назва товару	Фірма-постачальник	Кількість на складі	Вартість покупки	Сума поставки
		Сортування за спаданням ознаки					
<b>Середнє значення:</b>					=	=	=
<b>Всього:</b>					Σ		Σ

- III. Зберегти звіт у свій проект і вивести на друк першу і останню сторінки.
- IV. Самостійно вивчити **QUICK REPORT** і його можливості.
- V. Створити етикетки, використовуючи **МАЙСТЕР ЕТИКЕТОК**.
- ✓ Вибрати стиль етикеток – Every 4143.
  - ✓ Оформити етикетки за зразком:

ТОВАРИ НА СКЛАДІ	
Найменування - _____	Name _____
Оптова ціна: _Cena_ грн, кількість на складі - _Kol_ шт.	
Постачальник - _____	Firma _____, Gor _____
Дата поставки: _____	Data _____, артикул Nom _____

- ✓ Упорядкувати по полю Name.
  - ✓ Зберегти етикетки в особистому проекті і відкрити для коректування в конструкторі етикеток. Переглянути і роздрукувати результат.
- VI. Вікно Command вивести на друк.

#### ВАРІАНТ 2

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. Створити звіт, приведений нижче, використовуючи **МАЙСТЕР ЗВІТІВ**:

#### ФІРМИ-ПОСТАЧАЛЬНИКИ ТОВАРІВ

	Фірма-постачальник	Місто	Назва товару	Дата поставки	Кількість на складі	Вартість покупки	Сума поставки
	Групування	Групування	Сортування за зростанням ознаки				
<b>Середнє значення:</b>					=	=	=
<b>Всього:</b>					Σ		Σ

- III. Зберегти звіт у свій проект і вивести на друк.
- IV. Самостійно вивчити **QUICK REPORT** і його можливості.
- V. Створити етикетки, використовуючи **МАЙСТЕР ЕТИКЕТОК**.
- ✓ Вибрати стиль етикеток – Every 5198.
  - ✓ Оформити етикетки за зразком:

<b>ФІРМИ – ПОСТАЧАЛЬНИКИ ТОВАРІВ</b>	
Постачальник - _____	Firma _____, Gor _____
Найменування - _____ Name _____	
Оптова ціна: _Cena_ грн, кількість на складі - _Kol_ шт.	
Дата поставки: _____ Data _____, артикул _____ Nom _____	

- ✓ Упорядкувати по полю Firma.
- ✓ Зберегти етикетки в особистому проекті і відкрити для коректування в конструкторі етикеток. Переглянути і роздрукувати результат.

VI. Вікно Command вивести на друк.

### ВАРІАНТ 3

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. Створити звіт, приведений нижче, використовуючи **МАЙСТЕР ЗВІТІВ:**

#### НАДХОДЖЕННЯ ТОВАРІВ НА СКЛАД

VII.3 б е р е р т и	Дата поставки	Фірма- постачальник	Назва товару	Кількість на складі	Ціна без ПДВ	Оптова ціна	Сума поставки
	Групування	Групування	Сортування за зростанням ознаки			Cena*1.2	
<b>Середнє значення:</b>					=	=	=
<b>Всього:</b>				Σ			Σ

- III. Зберегти звіт у свій проект і вивести на друк.
- IV. Самостійно вивчити **QUICK REPORT** і його можливості.
- V. Створити етикетки, використовуючи **МАЙСТЕР ЕТИКЕТОК.**
  - ✓ Вибрати стиль етикеток – Every 5199-F.
  - ✓ Оформити етикетки за зразком:

<b>НАДХОДЖЕННЯ ТОВАРІВ НА СКЛАД</b>	
Дата поставки: _____	Data _____
Найменування - _____ Name _____	
Оптова ціна: _____Cena_____ грн	
Кількість на складі - _____Kol_____ шт.	
Артикул _____ Nom _____	

- ✓ Упорядкувати по полю Data.
- ✓ Зберегти етикетки в особистому проекті і відкрити для коректування в конструкторі етикеток. Переглянути і роздрукувати результат.

VI. Вікно Command вивести на друк.

### ВАРІАНТ 4

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. Створити звіт, приведений нижче, використовуючи **МАЙСТЕР ЗВІТІВ:**

#### НАКЛАДНІ НА ТОВАРИ

	Дата поставки	Номер накладної	Назва товару	Кількість на складі	Вартість покупки	ПДВ	Сума поставки
	Групування	Сортування за зростанням ознаки				Cena*0.2	
<b>Середнє значення:</b>					=	=	=
<b>Всього:</b>				Σ			Σ

- III. Зберегти звіт у свій проект і вивести на друк.  
 IV. Самостійно вивчити **QUICK REPORT** і його можливості.  
 V. Створити етикетки, використовуючи **МАЙСТЕР ЕТИКЕТОК**.  
 ✓ Вибрати стиль етикеток – Every 5197.  
 ✓ Оформити етикетки за зразком:

НАКЛАДНІ НА ТОВАРИ	
Номер накладної -	<u>Nnk</u> , Постачальник - <u>Firma</u>
Найменування -	<u>Name</u>
Оптова ціна: <u>Cena</u> грн, кількість на складі -	<u>Kol</u> шт.
Дата поставки: <u>Data</u> , артикул	<u>Nom</u>

- ✓ Упорядкувати по полю Nnk.  
 ✓ Зберегти етикетки в особистому проекті і відкрити для коректування в конструкторі етикеток. Переглянути і роздрукувати результат.  
 VI. Вікно Command вивести на друк.

### ВАРІАНТ 5

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.  
 II. Створити звіт, приведений нижче, використовуючи **МАЙСТЕР ЗВІТІВ**:

#### ПРАЙС – ЛИСТ ТОВАРІВ

	Артикул	Назва товару	Кількість поставки	Вартість покупки	Оптова ціна	Сума поставки
		Сортування за зростанням ознаки			Cena*1.2	
<b>Середнє значення:</b>				=	=	=
<b>Всього:</b>			Σ			Σ

- III. Зберегти звіт у свій проект і вивести на друк.  
 IV. Самостійно вивчити **QUICK REPORT** і його можливості.  
 V. Створити етикетки, використовуючи **МАЙСТЕР ЕТИКЕТОК**.  
 ✓ Вибрати стиль етикеток – Every 5198.  
 ✓ Оформити етикетки за зразком:

ПРАЙС – ЛИСТ ТОВАРІВ	
Найменування -	<u>Name</u>
Оптова ціна: <u>Cena</u> грн, артикул	<u>Nom</u>
Максимальна кількість замовлення -	<u>Kol</u> шт.
Замовлення № <u>Nd</u> , партія №	<u>Nnk</u>

- ✓ Упорядкувати по полю Name.  
 ✓ Зберегти етикетки в особистому проекті і відкрити для коректування в конструкторі етикеток. Переглянути і роздрукувати результат.  
 VI. Вікно Command вивести на друк.

### ВАРІАНТ 6

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.  
 II. Створити звіт, приведений нижче, використовуючи **МАЙСТЕР ЗВІТІВ**:

### ПРАЙС – ЛИСТ ТОВАРІВ

	Артикул	Назва товару	Кількість поставки	Вартість покупки	Роздрібна ціна	Сума поставки
	Групування	Сортування за спаданням ознаки			Cena*1.5	
<b>Середнє значення:</b>				=	=	=
<b>Всього:</b>			Σ			Σ

- III. Зберегти звіт у свій проект і вивести на друк.
- IV. Самостійно вивчити **QUICK REPORT** і його можливості.
- V. Створити етикетки, використовуючи **МАЙСТЕР ЕТИКЕТОК**.
  - ✓ Вибрати стиль етикеток – Every 5261.
  - ✓ Оформити етикетки за зразком:

ТОВАРИ НА СКЛАДІ	
Найменування - _____	<b>Name</b>
Оптова ціна: _Cena_ грн, кількість на складі - _Kol_ шт.	
Постачальник - _____, _____	
Дата поставки: _____, артикул _____	
Data _____, Nom _____	

- ✓ Упорядкувати по полю Name.
  - ✓ Зберегти етикетки в особистому проекті і відкрити для коректування в конструкторі етикеток. Переглянути і роздрукувати результат.
- VI. Вікно Command вивести на друк.

### ВАРІАНТ 7

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.
- II. Створити звіт, приведений нижче, використовуючи **МАЙСТЕР ЗВІТІВ**:

#### ТОВАРИ НА СКЛАДІ

	Артикул	Дата поставки	Назва товару	Фірма-постачальник	Кількість на складі	Вартість покупки	Сума поставки
	Групування	Сортування за спаданням ознаки					
<b>Середнє значення:</b>					=	=	=
<b>Всього:</b>					Σ		Σ

- III. Зберегти звіт у свій проект і вивести на друк.
- IV. Самостійно вивчити **QUICK REPORT** і його можливості.
- V. Створити етикетки, використовуючи **МАЙСТЕР ЕТИКЕТОК**.
  - ✓ Вибрати стиль етикеток – Every 5262.
  - ✓ Оформити етикетки за зразком:

ФІРМИ – ПОСТАЧАЛЬНИКИ ТОВАРІВ	
Постачальник - _____	<b>Firma</b>
Найменування - _____	
Оптова ціна: _Cena_ грн, кількість на складі - _Kol_ шт.	
Дата поставки: _____, артикул _____	
Data _____, Nom _____	

- ✓ Упорядкувати по полю Firma.
  - ✓ Зберегти етикетки в особистому проекті і відкрити для коректування в конструкторі етикеток. Переглянути і роздрукувати результат.
- VI. Вікно Command вивести на друк.

### ВАРІАНТ 8

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.  
II. Створити звіт, приведений нижче, використовуючи **МАЙСТЕР ЗВІТІВ**:

#### ФІРМИ-ПОСТАЧАЛЬНИКИ ТОВАРІВ

	Фірма-постачальник	Місто	Назва товару	Дата поставки	Кількість на складі	Вартість покупки	Сума поставки
	Групування	Групування	Сортування за зростанням ознаки				
<b>Середнє значення:</b>					=	=	=
<b>Всього:</b>					Σ		Σ

- III. Зберегти звіт у свій проект і вивести на друк.  
IV. Самостійно вивчити **QUICK REPORT** і його можливості.  
V. Створити етикетки, використовуючи **МАЙСТЕР ЕТИКЕТОК**.  
✓ Вибрати стиль етикеток – Every 5897.  
✓ Оформити етикетки за зразком:

<b>ПРИХІД ТОВАРІВ НА СКЛАД</b>	
Дата поставки: _____	<b>Data</b> _____
Найменування - _____	<b>Name</b> _____
Оптова ціна: _____	<b>Cena</b> _____ грн
Кількість на складі - _____	<b>Kol</b> _____ шт.
Артикул _____	<b>Nom</b> _____

- ✓ Упорядкувати по полю Data.  
✓ Зберегти етикетки в особистому проекті і відкрити для коректування в конструкторі етикеток. Переглянути і роздрукувати результат.  
VI. Вікно Command вивести на друк.

### ВАРІАНТ 9

- I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.  
II. Створити звіт, приведений нижче, використовуючи **МАЙСТЕР ЗВІТІВ**:

#### НАДХОДЖЕННЯ ТОВАРІВ НА СКЛАД

	Дата поставки	Фірма-постачальник	Назва товару	Кількість на складі	Ціна без ПДВ	Оптова ціна	Сума поставки
	Групування	Групування	Сортування за зростанням ознаки			Cena*1.2	
<b>Середнє значення:</b>					=	=	=
<b>Всього:</b>				Σ			Σ

- III. Зберегти звіт у свій проект і вивести на друк  
IV. Самостійно вивчити **QUICK REPORT** і його можливості.  
V. Створити етикетки, використовуючи **МАЙСТЕР ЕТИКЕТОК**.  
✓ Вибрати стиль етикеток – Every 5663.  
✓ Оформити етикетки за зразком:

<b>НАКЛАДНІ НА ТОВАРИ</b>	
Номер накладної -	<u>    Nnk    </u> , Постачальник - <u>    Firma    </u>
Найменування -	<u>                    Name                    </u>
Оптова ціна: <u>    Cena    </u> грн, кількість на складі - <u>    Kol    </u> шт.	
Дата поставки: <u>    Data    </u> , артикул <u>    Nom    </u>	

- ✓ Упорядкувати по полю Nnk.
- ✓ Зберегти етикетки в особистому проекті і відкрити для коректування в конструкторі етикеток. Переглянути і роздрукувати результат.

VI. Вікно Command вивести на друк.

### ВАРІАНТ 10

I. **Початкові дані:** SKLAD.dbf - інформація про товари, одержані від постачальників, що є на складі.

II. Створити звіт, приведений нижче, використовуючи **МАЙСТЕР ЗВІТІВ:**

#### НАКЛАДНІ НА ТОВАРИ

	Дата поставки	Номер накладної	Назва товару	Кількість на складі	Ціна покупки	ПДВ	Сума поставки
	Групування	Сортування за спаданням ознаки				Cena*0.2	
<b>Середнє значення:</b>					=	=	=
<b>Всього:</b>				Σ			Σ

III. Зберегти звіт у свій проект і вивести на друк.

IV. Самостійно вивчити **QUICK REPORT** і його можливості.

V. Створити етикетки, використовуючи **МАЙСТЕР ЕТИКЕТОК.**

- ✓ Вибрати стиль етикеток – Every 5662.
- ✓ Оформити етикетки за зразком:

<b>ПРАЙС – ЛИСТ ТОВАРІВ</b>	
Найменування -	<u>                    Name                    </u>
Оптова ціна: <u>    Cena    </u> грн, артикул <u>    Nom    </u>	
Максимальна кількість замовлення - <u>    Kol    </u> шт.	
Замовлення № <u>    Nd    </u> , партія № <u>    Nnk    </u>	

- ✓ Упорядкувати по полю Name.
- ✓ Зберегти етикетки в особистому проекті і відкрити для коректування в конструкторі етикеток. Переглянути і роздрукувати результат.

VI. Вікно Command вивести на друк.