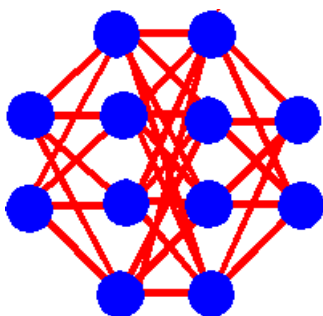


Міністерство освіти і науки, молоді та спорту України

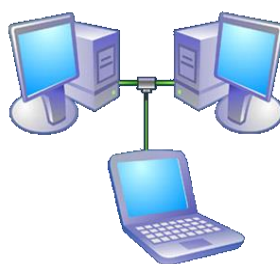
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ  
АВТОМОБІЛЬНО-ДОРОЖНІЙ УНІВЕРСИТЕТ

Л. М. СИМБІРСЬКА  
Г. Д. СИМБІРСЬКИЙ  
А. І. ЛЕВТЕРОВ



## ІНФОРМАЦІЙНІ СИСТЕМИ І ТЕХНОЛОГІЇ

*Навчально-методичний посібник  
(лабораторний практикум)*



Харків  
ХНАДУ  
2011

УДК 004.43 (072)  
ББК 32.97  
С 37

Рецензенти:

**Туркін І. Б.**, проф., д-р техн. наук,  
завідувач кафедру програмного забезпечення  
(Харківський національний аерокосмічний  
університет ім. М. Є. Жуковського "ХАІ")

**Алексєєв О. П.**, проф., д-р техн. наук,  
завідувач кафедру мехатроніки транспортних засобів  
(Харківський національний  
автомобільно-дорожній університет)

**С 37**

**Симбірська Л. М., Симбірський Г. Д., Левтеров А. І.**

Інформаційні системи і технології:

Навчально-методичний посібник (лабораторний практикум). –  
Харків: ХНАДУ, 2011. – 91 с.

**ISBN: 966-303-121-2**

Навчально-методичний посібник (лабораторний практикум) містить теоретичні та практичні відомості для користувачів-початківців про застосування існуючих інформаційних систем і про методи проектування нових із метою їх подальшої практичної реалізації.

Призначений для студентів технічних спеціальностей.

Іл. 57. Табл. 13. Бібліогр. 9 найм.

УДК 004.43 (072)

ББК 32.97

© ХНАДУ, 2011

© Симбірська Л. М.,  
Симбірський Г. Д.,  
Левтеров А. І.

**ISBN: 966-303-121-2**

## ВСТУП

Навчально-методичний посібник (лабораторний практикум) для дисципліни "Інформаційні системи і технології" напряму підготовки "Транспортні технології" передбачає підготовку користувачів для ефективного використання сучасної обчислювальної техніки і програмного забезпечення у професійній діяльності.

Цей посібник розроблений згідно із програмою дисципліни і містить теоретичні відомості за темами, завдання для практичного засвоєння користувачами теоретичних знань та запитання для самоконтролю вивченого матеріалу.

Метою навчально-методичного посібника є сприяння користувачам-початківцям набуття умінь й навичок проектування інформаційних систем та їх практичної реалізації із урахуванням особливостей функціонування об'єктів автомобільного транспорту.

Для кожного завдання наведені детальні пояснення до виконання відповідних операцій, що дає можливість користувачам за певних умов самостійно вивчати кожен тему.

У лабораторному практикумі розглядаються питання комунікаційних можливостей інформаційних систем, технології аналізу предметної області, створення реляційної моделі даних, побудови інформаційних об'єктів і зв'язків між ними, проектування інформаційної структури бази даних, розробки простої, але діючої інформаційної системи із використанням системи управління базами даних Microsoft Access. Для результативного дослідження і засвоєння принципів створення і використання інформаційних систем і технологій рекомендується точно виконувати усі наведені у текстах вказівки, уважно й детально досліджувати у процесі роботи стан вікон на екрані дисплею.

Рекомендується у процесі виконання завдань за темами посібника прагнути досягати подібності відображень результатів роботи наведеним у посібнику ілюстраціям прикладів. У багатьох випадках при відкритті або закритті вікон певних середовищ на екран виводяться діалогові вікна із запитом або переліками можливих дій. Необхідно уважно аналізувати повідомлення у цих вікнах, виконати усвідомлений вибір і виконати відповідні команди.

На порталі ХНАДУ знаходиться електронна версія лабораторного практикуму. Для оптимізації навігації по сторінках тексту нумерація розділів **Змісту** у цій версії виконана у вигляді гіпертекстових посилань. Повернення до **Змісту** із будь-якого місця у тексті виконується натисканням клавіш [Ctrl] + [Home].

## Тема 1. ДОСЛІДЖЕННЯ ОГЛЯДАЧА INTERNET EXPLORER (IE)

**Завдання 1.1.** Засвоїти призначення і можливості оглядача IE.

**1.1.1.** Викликати на екран менеджер файлів **Total Commander** і створити на диску **C:** папку із назвою, що відповідає назві групи, використовуючи прописні українські літери (наприклад, **ТД21**). Усередині папки групи створити папку користувача із назвою, що відповідає **прізвищу** користувача (українською мовою); наприклад, **Іванов**.

Надалі, при необхідності збереження будь-якого файлу (створеного користувачем), слід розташовувати його у папці користувача і призначати файлу ім'я, яке має починатись прізвищем користувача, далі - нижня риска, а за нею - назва, що зазначена у завданнях.

Через те, що у наступних дослідженнях звертання до файлів буде виконуватись за допомогою програми **Total Commander**, слід за допомогою кнопки **Свернуть** (яка знаходиться у рядку заголовку, праворуч, позначена нижньою рисою) згорнути вікно цієї програми на панель задач для швидкого та зручного повторного виклику.

**1.1.2.** Всесвітня мережа Інтернет є поєднанням великих і малих комп'ютерних мереж і сукупністю декількох служб, які функціонують в єдиному світовому інформаційному просторі. Ці служби базуються на різних протоколах (правилах) обміну інформацією і на відповідному програмному забезпеченні. Інтернет у більшості випадків ототожнюється із його найпопулярнішою службою **World Wide Web (WWW)** або просто **Web** (веб). Інформація, опублікована у мережі у форматі цієї служби, надається у вигляді веб-сторінок, тобто комплексних документів із даними будь-яких типів: текстів, графіків, звуків, відео, анімації. Для прийому веб-сторінок із Інтернету і для їх відображення на екрані використовують спеціальні програми-оглядачі (браузери, від browser). Найбільш популярними і стандартними засобами перегляду веб-сторінок є оглядачі **Internet Explorer (IE)** та **Opera**.

**1.1.3.** Місце розташування веб-сторінки в Інтернеті визначається адресою URL (Uniform Resource Locator) - уніфікованою формою запису адреси документу, що починається префіксом **http://** (аббревіатура від Hypertext Transfer Protocol - тип протоколу зв'язку). Правила запису адреси у цій формі дозволяють точно зазначити місце розташування будь-якого документу. Адреса **URL** (рис. 1.1) може складатись із таких чотирьох частин:

- 1) назви протоколу (правила і стандарти) обміну інформацією);
- 2) шляху до домену (частини мережі), де знаходиться веб-сервер;
- 3) шляху пошуку документа на веб-сервері;
- 4) посилання із документа на назву іншого файлу, в якому зазначено файл із типом **.html** (Hyper Text Markup Language).

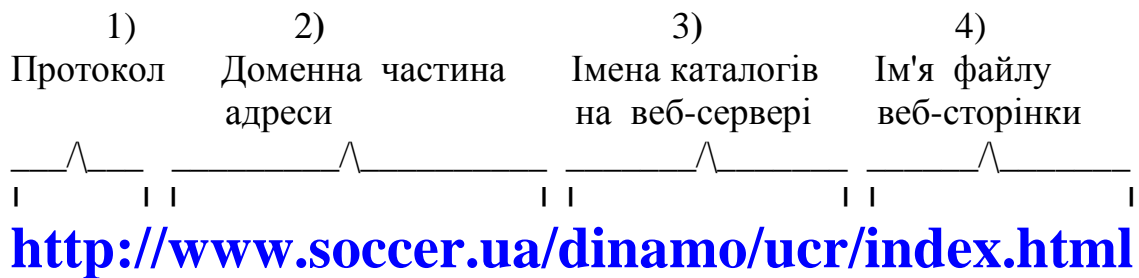


Рис. 1.1. Приклад складових частин адреси веб-сторінки.

## Завдання 1.2. Вивчити склад панелей інструментів оглядача ІЕ.

**1.2.1.** Викликати оглядач на екран дисплею натисканням ЛК миші на ярлику оглядача ІЕ на робочому столі комп'ютера або виконанням наступної команди: **Пуск→Программы→Internet Explorer.**

Результатом однієї із зазначених дій буде відображення на екрані домашньої сторінки ХНАДУ (рис. 1.2), на яку налаштований оглядач.

**1.2.2.** У вікні оглядача ІЕ **перший рядок** (вгорі, синього кольору) є рядком **заголовку**, в якому ліворуч виводиться інформація про програму, а праворуч - традиційні для програмних продуктів корпорації Microsoft кнопки управління вікном - **Свернуть** (на **Панель задач**), **Свернуть в окно** (меншого розміру) або **Развернуть** (на весь екран) і **Закреть**.

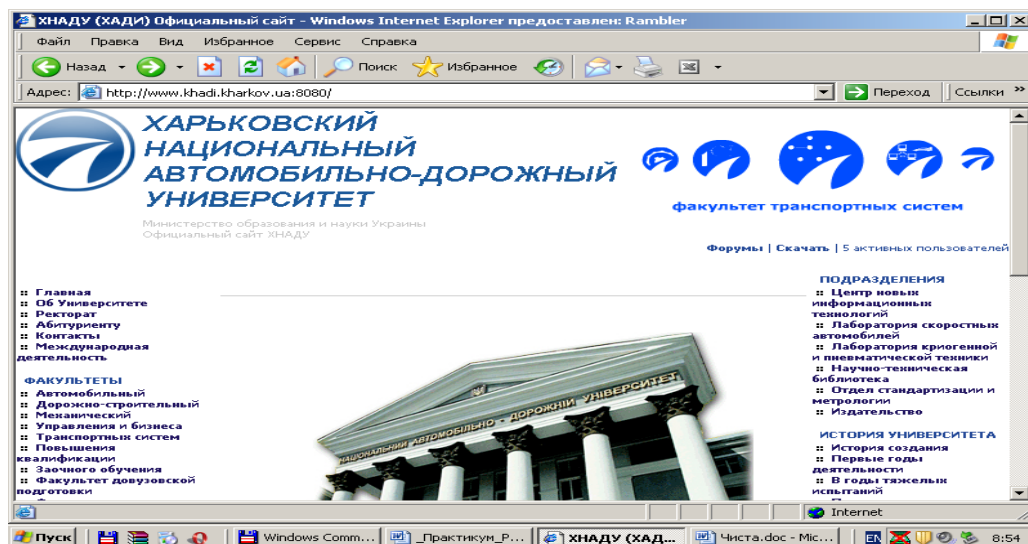


Рис. 1.2. Вікно оглядача **Internet Explorer.**

**Другий** рядок являє собою панель головних команд; натискання на будь-якій із них лівої клавіші (ЛК) миші сприяє виведенню на екран вторинних команд цієї головної команди. Деякі із вторинних команд у свою чергу містять додаткові команди, які викликаються натисканням трикутників - кнопок розгортання списків команд.

**Третій** рядок відображає панель інструментів **Стандартная**, що складається із декількох груп кнопок-інструментів.

Перша ліворуч група (відокремлена вертикальною рисою) використовується для переходів до попередньої (**Назад**) або наступної (**Вперед**) веб-сторінки, для зупинки виклику сторінки (**Остановить**) або для оновлення (**Обновить**) розташованої на ній інформації, для переходу на домашню сторінку (**Домой**).

Другу групу кнопок цього рядка використовують для пошуку інформації в Інтернеті (**Поиск**), доступу до обраних веб-сторінок (**Избранное**), повернення до однієї із переглянутих сторінок (**Журнал**). При натисканні зазначених кнопок на екран виводяться відповідні додаткові панелі. Зазначені команди дублюються командою:

#### **Вид→Панели обозревателя.**

Остання група кнопок призначена для відправки повідомлення електронною поштою, роздруківки веб-сторінок на папері (**Печать**), редагування тексту веб-сторінок (**Править**) із можливістю звернення до середовищ Microsoft Excel, Microsoft Word або Блокнот і пошуку довідок за допомогою окремої панелі **Справочные материалы**.

У **четвертому** рядку розташована адресна панель; текстове поле цієї панелі призначене для введення потрібних адрес. Натискання кнопки списку (в адресному полі праворуч) сприяє виведенню на екран списку адрес, до яких звертались у попередніх сеансах роботи. При натисканні ЛК миші на вибраній адресі відбувається заміна поточної адреси на зазначену. Завершується введення нової адреси обов'язковим натисканням клавіші [**Enter**] на клавіатурі

Оглядач ІЕ налагоджений таким чином, що при звертанні до нього на екран виводиться домашня сторінка Харківського національного автомобільно-дорожнього університету (ХНАДУ).

Якщо у вікні оглядача ІЕ виводиться інша інформація, потрібно у текстовому полі адресного рядка ввести **URL**-адресу:

<http://www.khadi.kharkov.ua>

**1.2.3.** В інформаційному полі оглядача ІЕ (що розташоване під панелями) у вигляді кнопок і коротких пропозицій (так званих **гіпертекстових посилань** або скорочено - **гіперпосилань**) наданий перелік усіх видів діяльності університету.

У вікні домашньої сторінки ХНАДУ майже усі пропозиції є гіперпосиланнями, яким відповідають певні веб-сторінки. Як було зазначено вище, перехід до веб-сторінки виконується введенням за допомогою клавіатури її **URL**-адреси, але введення довгих **URL**-адрес за допомогою клавіатури незручний і схильний до помилок.

Альтернативою друкуванню адрес є використання гіперпосилань, які містять інформацію про адресу розшукуваного об'єкта (веб-сторінки): інших гіперпосилань, фрагментів тексту, графічних ілюстрацій, рисунків, схем, відео. Гіперпосилання можуть відображатись (за бажанням автора

проекту) іншим кольором шрифту - **синім** до звертання за конкретною адресою і **малиновим** після звертання. У більшості випадків речення гіперпосилань підкреслені **нижньою рискою**.

При розташуванні покажчика миші на гіперпосиланні зображення стрілки покажчика змінюється на зображення **долоні**, а натискання ЛК миші сприяє переходу на веб-сторінку із зазначеною адресою.

**1.2.4.** Користувачам рекомендується самостійно у довільному порядку ознайомитись із складовими домашньої сторінки ХНАДУ: історією, діяльністю факультетів і кафедр університету, переліком дисциплін, тощо.

### **Завдання 1.3.** Засвоїти способи пошуку інформації в Інтернеті.

**1.3.1.** Пошук в Інтернеті веб-сторінок із потрібною інформацією виконують спеціальні пошукові системи (іноді їх називають "пошуковими машинами" і "спайдерами", тобто "павуками"). Одною із найбільш популярних в Україні є система із адресою <http://www.meta.ua>, а у Росії - із адресою <http://www.rambler.ru>.

Ввести в адресному полі оглядача ІЕ адресу пошукової системи <http://www.rambler.ru> і натиснути [Enter]. Через декілька секунд на екран виводиться вікно головної сторінки (рис. 1.3) цієї пошукової системи.

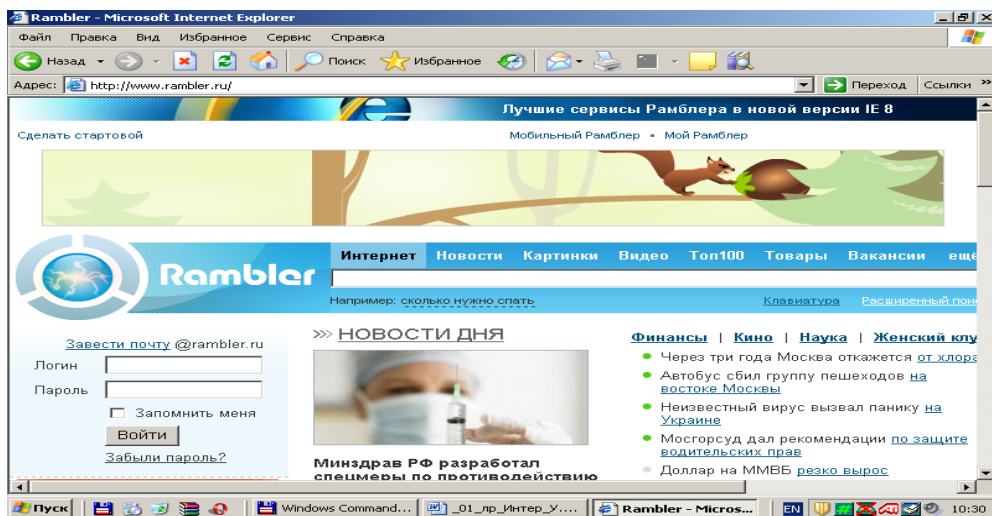


Рис. 1.3. Вікно пошукової системи **Rambler**.

У вікні пошукової системи Rambler розташована **Форма** запиту (синього кольору) із полем для введення **ключових слів** (прізвищ, назв різних типів) для пошуку інформації. Слід відзначити, що адміністратори оглядачів і пошукових систем прагнуть відображати у вікнах новітню інформацію і тому зміст і оформлення вікон постійно змінюються.

Ввести у вікні **Форми** (рис. 1.4) ключові слова **Информационные системы на автотранспорте** і завершити введення натисканням [Enter].



**1.3.2** Для виведення на екран змісту веб-сторінки (для перегляду із метою знайти потрібну інформацію) слід натиснути ЛК миші на її гіперпосиланні; як правило, це декілька слів після номера сторінки.

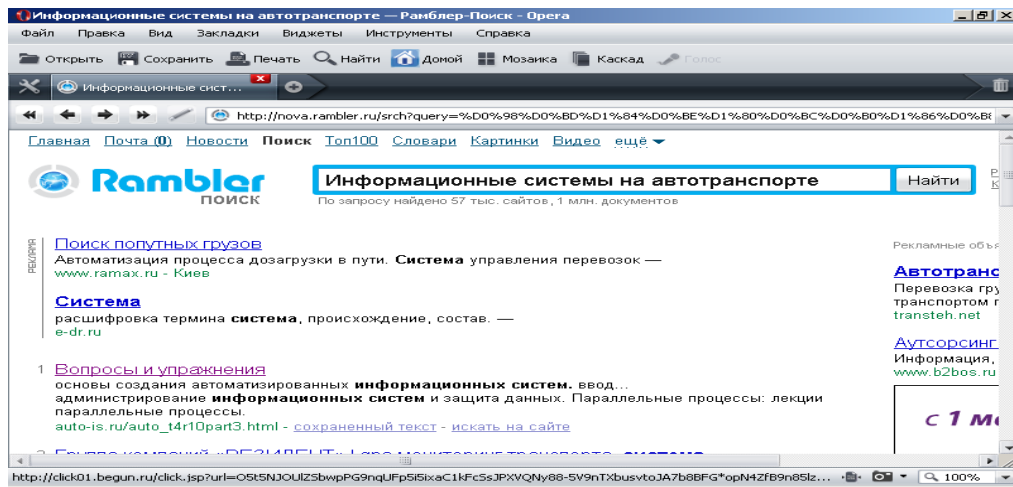


Рис. 1.4. Надання інформації за ключовими словами.

Веб-сторінки є складними комбінованими документами, що містять як текстову інформацію, так і вбудовані об'єкти; вони можуть бути ілюстровані статичними зображеннями, а також звуками та об'єктами, створеними за мультимедійними технологіями.

Веб-сторінки є також форматуваними електронними документами. Багато у чому вони нагадують документи, що створюються текстовим процесором. Проте, у текстовому процесорі готують документи, для яких заздалегідь відомо, в якому вигляді вони відтворюватимуться (наприклад, друкуватися на аркуші паперу певного розміру). Веб-документи є більше універсальними: вони легко відтворюються незалежно від розміру, кількості пікселів і кольорів екрану користувача.

**1.3.3.** За замовчуванням оглядач ІЕ відображає усю інформацію веб-сторінки. Якщо за якоїсь причини відображення об'єктів веб-документу не відбувається, програма позначає відповідні місця порожніми рамками або спеціальними позначками. Якщо засіб перегляду не може відобразити вбудований об'єкт, то замість об'єкта відображається пов'язаний із ним альтернативний текст.

**1.3.4.** Підвести покажчик миші до гіперпосилання на другій сторінці (рис. 1.4) "[13. Проблемы проектирования и внедрения информационной системы учёта работы автотранспорта](#)". Для переходу за гіперпосиланням слід на ньому натиснути один раз ЛК миші - оглядач ІЕ викличе веб-сторінку, URL-адреса якої буде відображена в адресному полі. Знайти на цій сторінці гіперпосилання Автопарк і виконати її - на екран виводиться вікно сайту **SIKE Корпоративные системы** (рис. 1.5).

Нова веб-сторінка може містити нові гіперпосилання; це дозволяє продовжувати перегляд документів, які відносяться до зазначеної теми.



Механізм гіперпосилань є однією із тих особливостей World Wide Web, які найбільшою мірою сприяли зростанню популярності цієї служби.

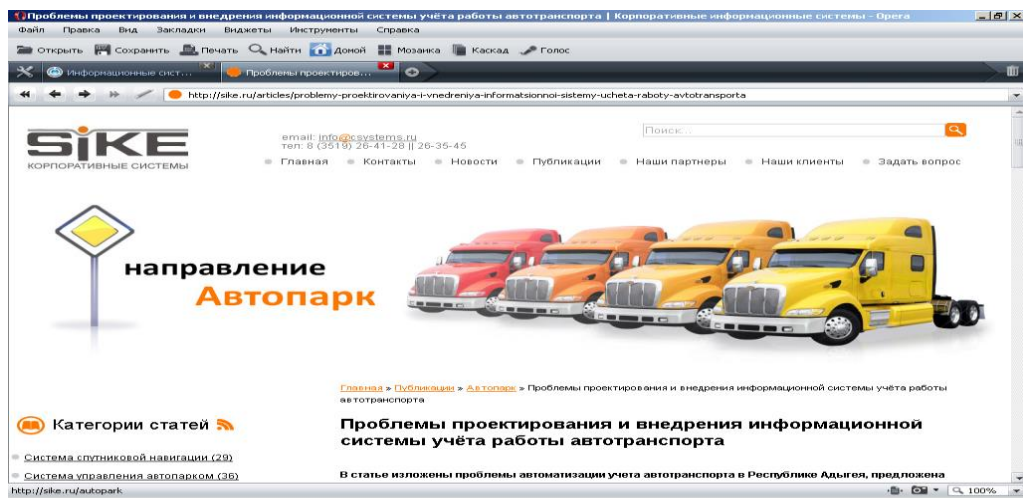


Рис. 1.5. Одна із сторінок сайту "SIKE Корпоративные системы".

**1.3.5.** Рекомендується продовжити подорож по сторінках зазначеного сайту для повного ознайомлення із його змістом. Якщо тематична сфера розшукуваної інформації відома, то, почавши із одного-двох стандартних гіперпосилань, теоретично можна знайти будь-яку інформацію за цією темою, користуючись тільки гіпертекстовими посиланнями.

В Інтернеті знаходиться багато спеціальних сторінки-показчиків, що містять тільки великі списки гіперпосилань на веб-сторінки, присвячені певним питанням.

Для повернення (після серії переходів по гіперпосиланнях) на будь-яку із раніше переглянутих сторінок слід скористатися або кнопкою **Назад** на панелі інструментів оглядача або кнопкою **Вперед**. На цих кнопках є кнопка-трикутник, натискання якої викликає на екран список сторінок за темами поточного сеансу роботи.

**1.3.6.** Для дослідження інших можливостей оглядача ІЕ завершити цей етап натисканням кнопки **Закрить** і знову викликати оглядач на екран натисканням ЛК миші на відповідному ярлику.

#### **Завдання 1.4.** Дослідити способи налаштування оглядача ІЕ.

**1.4.1.** Користувачам рекомендується уважно ознайомитись (але не змінювати) із зазначеними нижче налаштуваннями параметрів відображення веб-сторінок та значеннями параметрів. Непередбачена зміна будь-якого параметру може сприяти пошкодженню програми оглядача.

Деякі налаштування оглядача виконують за допомогою команди **Вид**. Перша група складових цієї команди дозволяє змінити зовнішній вигляд вікна ІЕ. Вони управляють відображенням панелей інструментів, рядка стану і додаткових панелей.

Команду **Шрифты** використовують для управління шрифтами веб-сторінки - зазначення розміру шрифту (від найменшого до найбільшого) та кодування використовуваних символів на інших мовах.

Команди **Остановить** і **Обновить** належать до поточної веб-сторінки. Результати дії обох команд відповідають значенню цих слів. Команда **Просмотр HTML-кода** дозволяє переглянути коди HTML, якими утворюється зміст будь-якої веб-сторінки. Команда **На весь экран** сприяє переведенню ІЕ до повноекранного режиму, який розрахований на перегляд веб-сторінок із збільшеною кількістю рядків.

**1.4.2.** Незважаючи на значну кількість елементів управління на панелях, можливості налаштування оглядача із їх допомогою обмежені. Для повного налаштування програми використовується команда **Сервис→Свойства обозревателя**, за якою відкривається відповідне вікно із декількома картками (рис. 1.6). Елементами управління картки **Общие** уводяться основні параметри роботи програми. Область **Временные файлы Интернета** визначає термін зберігання на жорсткому диску тимчасових файлів, прийнятих із Інтернету. Ці файли використовуються при повторних зверненнях до веб-сторінок.

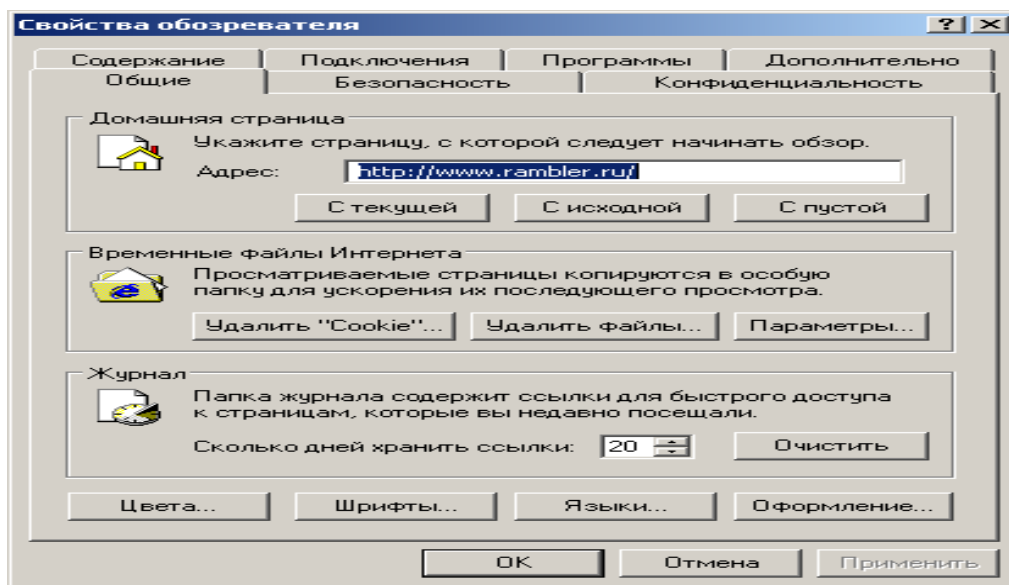


Рис. 1.6. Вікно налаштування властивостей оглядача.

Оглядач може не приймати їх із Мережі, а прочитувати з жорсткого диску, що значно швидше. В області **Журнал** відзначається термін зберігання інформації прочитаних веб-сторінок. Кнопки у нижній частині сторінки дозволяють зазначати **Цвета**, **Шрифты**, **Языки** і **Оформление**, використовувані при відображенні веб-сторінок.

**1.4.3.** Картка **Подключение** дозволяє змінити або додати параметри поєднання із Інтернетом. Інші області цієї сторінки забезпечують спеціальні можливості налаштування. Картка **Программы** дозволяє вибрати

додаткові програми для виконання операцій, що відсутні у використовуваних програмах. Картка **Безопасность** забезпечує налаштування параметрів управління мірою ризику прийому із Інтернету інформації, небезпечної для комп'ютера. Обмеження, які зазначаються на цій сторінці, зменшують вірогідність ушкодження інформації комп'ютерним вірусом, але обмежують свободу навігації в Інтернеті. Засоби картки **Содержание** дозволяють обмежити доступ до мережної інформації. Принцип обмежень заснований на системі рейтингових оцінок веб-вузлів. Налаштування обмежень за змістом потрібне у тих випадках, коли одним комп'ютером користуються декілька користувачів. Картка **Дополнительно** містить великий комплект перемикачів для зазначення параметрів, відсутніх на загальних картках. Особливо часто доводиться налаштовувати елементи управління категорії **Мультимедиа**, які дозволяють або забороняють завантаження графічних або мультимедійних об'єктів.

### **Завдання 1.5.** Засвоїти способи зберігання інформації веб-сторінок

**1.5.1.** У процесі роботи в Інтернеті можуть визначитись веб-сторінки постійного звертання, тобто утворюється список найчастіше запитуваних URL-адрес. Такі адреси називають **закладками**; вони дозволяють швидше починати роботу в Інтернеті від обраної сторінки. Закладки створюються у папці **Избранное**.

Щоб надіслати ярлик веб-сторінки до папки **Избранное**, потрібно або натиснути ПК миші у межах веб-сторінки і у контекстному списку команд виконати команду **Добавить в избранное** або виконати команду **Избранное→Добавить в избранное**.

**1.5.2.** Для доступу до ярликів, які знаходяться у папці **Избранное**, використовується область **Избранное**, яку відкривають за допомогою відповідної кнопки на панелі інструментів або командою **Избранное**. При зростанні кількості обраних сторінок використовувати команду **Избранное** стає незручно.

Стандартний засіб упорядкування у структурі адрес полягає у створенні вкладених тематичних папок із наступним розміщенням ярликів безпосередньо у них. Провести цю роботу дозволяє команда **Избранное→Упорядочить избранное**. При цьому відкривається діалогове вікно, за допомогою елементів управління якого можна створити додаткові папки і розташувати у них ярлики веб-сторінок. Недоліком папки **Избранное** є введення інформації за допомогою клавіатури.

**1.5.3.** За допомогою пошукової системи [www.meta.ua](http://www.meta.ua) виконати пошук інформації для кожної із зазначених тем:

- 1) "комп'ютерні мережі";
- 2) "технологія побудови локальних мереж";
- 3) "стандарты і протоколи мережі Інтернет".

У кожній із тем переглянути декілька сайтів і вибрати по одному

сайту, який, на думку користувача, найбільше відповідає темі.

Виконати за допомогою команди **Файл→Сохранить как** збереження вибраних веб-сторінок у папці користувача. Ім'я файлу має починатись **прізвищем** користувача, далі розташувати символ нижньої риски і відповідну темі назву; наприклад, **Іванов\_Сети**, або **Іванов\_Технологии**, або **Іванов\_Стандарти**.

### **Завдання 1.6.** Ознайомитись із призначенням Журналу.

Інструмент **Журнал** зберігає адреси усіх сайтів, які відвідувались у сеансі роботи, і дозволяє автоматично (без клавіатурного набору) відновити дії користувача за певний період часу. Звернення до журналу виконується натисканням ЛК миші на кнопку **Журнал** на панелі інструментів **Стандартная** або виконанням команди **Вид→Панели обозревателя→Журнал**. У вікні, що відкрилося, відображаються відсортовані за датами папки поточного і минулих тижнів. Папка із відповідним веб-вузлом містить тільки ярлики відвіданих веб-сторінок цього вузла і дозволяє повернутись до них натисканням ЛК миші на відповідній сторінці.

Негативною властивістю журналу є доступність його інформації сторонній людині. Крім того, упродовж роботи в Інтернеті могла бути переглянута значна кількість сторінок. В обох випадках зайві сторінки рекомендується вилучити. Для цього слід викликати журнал командою **Вид→Панели обозревателя→Журнал**; позначити запис або папку у Журналі; натисканням ПК миші викликати контекстний список команд; виконати команду **Удалить**.

### **Завдання 1.7.** Дослідити довідкову систему оглядача ІЕ.

Довідка викликається із середовища оглядача за допомогою команди **Справка→Вызов справки**. Якщо на екран виводиться вікно довідкової системи у згорнутому форматі, то слід натиснути у заголовку вікна кнопку **Развернуть** для розгортання цього вікна (рис. 1.7) на весь екран. У текстовому полі ліворуч розташовані чотири картки: **Содержание**, **Указатель**, **Поиск**, **Избранное**.

Картка **Содержание** побудована за методикою пошуку інформації із застосуванням гіперпосилань. Для відкриття складових змісту слід натиснути ЛК миші на відповідному гіперпосиланні. Зміст позначеного розділу відображається у полі праворуч, де, у свою чергу, також зустрічаються пояснення у стилі технології гіперпосилань.

Картка **Указатель** надає довідку за окремими словами або поняттями. Слід перевести покажчик миші на рядок із потрібним словом і натиснути ЛК миші - пояснення виводяться у полі праворуч.

Картка **Поиск** надає можливість знайти інформацію тоді, коли виникають труднощі у пошуку гіперпосилання для певного слова. У цьому

випадку слід надрукувати слово у полі, яке знаходиться під заголовком **Введите ключевое слово для поиска**, і натиснути [Enter].

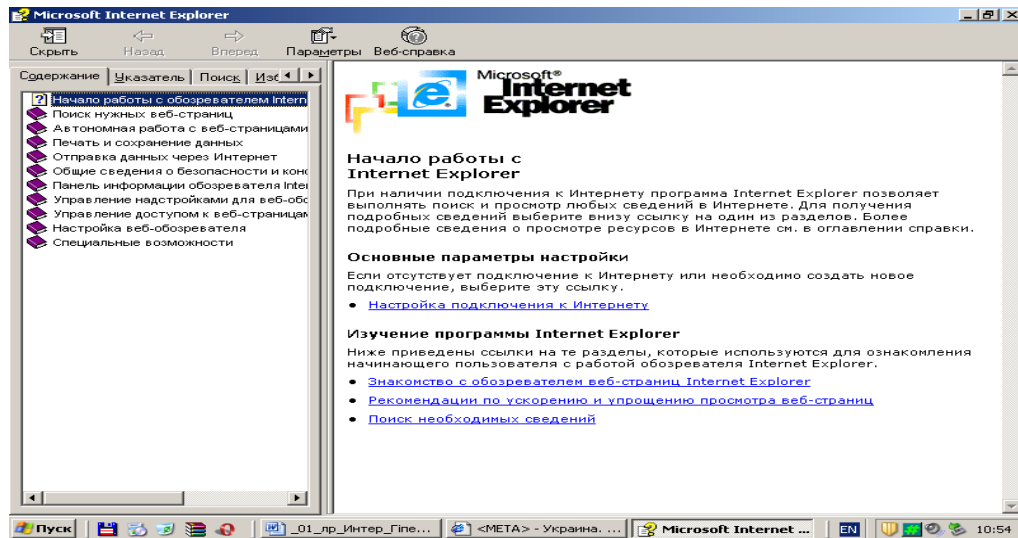


Рис. 1.7. Початкове вікно довідки оглядача ІЕ.

Картка **Избранное** надає довідку про зібрані сторінки для швидкого доступу. Користувачам рекомендується знайти довідку про можливості, методи і способи друкування інформації знайдених у процесі заняття веб-сторінок. Записати зазначену інформацію у конспект.

Завершити дослідження середовища Internet Explorer натисканням кнопки **Закорыть** у заголовку вікна.

### **Завдання 1.8.** Засвоїти методику створення гіперпосилань.

**1.8.1.** Викликати на екран **Total Commander**, відкрити у каталозі диску **C:** папку **2Г** і скопіювати у папку користувача файл **1\_лр\_ІЕ\_студ.doc**. Цей файл у власній папці користувач буде використовувати для проектування гіперпосилань, застосованих у **Змісті** для прискореного переходу до кожного завдання. Повернення до змісту із будь-якого місця тексту виконується натисканням [Ctrl] + [Home]. Як відзначалося у попередніх завданнях, для нелінійних переходів (тобто без дотримання формальної послідовності) застосовуються гіперпосилання, що сприяють переходу на окремі інформаційні вузли, на інший файл, на фрагмент файлу, на певне місце у документі.

**1.8.2.** Створення гіперпосилань виконується наступними діями.

- 1) Відкрити у папці користувача файл **1\_лр\_ІЕ\_студ.doc**.
- 2) Позначити у рядку формулювання першого завдання номер **1.1.** і натиснути [Ctrl]+[c], тобто переслати копію номера у **Буфер обмена**.
- 3) Натиснути [Ctrl] + [Home] для повернення на початок **Змісту**.
- 4) Розташувати покажчик миші перед цифрами **1.1.** у **Змісті** і на-

тиснути ЛК миші (тобто зазначити місце для розташування гіперпосилання), а потім виконати команду **Правка→Вставить гіперссылку**.

Внаслідок виконаних дій на екран виводиться гіперпосилання (має бути підкресленим та синього кольору ) на зазначений номер завдання.

5) Натисканням клавіші [**Delete**] вилучити старий номер.

6) Виконати аналогічні пунктам 2 - 5 дії для усіх завдань, введення та контрольних питань.

7) Перевірити перехід за гіперпосиланням за допомогою натискання лівою рукою ліворуч клавішу [**Ctrl**], утримування її і натискання ЛК миші на відповідному гіперпосиланні.

8) Після завершення роботи виконати команду **Файл→Сохранить как** для збереження файлу у відредагованому вигляді у папці користувача. Назва файлу має складатись із прізвища користувача та назви теми; наприклад, **Іванов\_Ссылки**.

## 1.9. Контрольні запитання

1. Що являє собою служба World Wide Web (WWW)?
2. Для чого служить програма-оглядач Internet Explorer?
3. Із яких частин складається адреса URL (Uniform Resource Locator)?
4. Яку роль виконує адреса URL (Uniform Resource Locator)?
5. Яка технологія застосовується для перегляду веб-сторінок?
6. Назвати основні інструменти управління середовища Internet Explorer.
7. Що являють собою "пошукові системи" і для чого вони призначені?
8. У чому складається перевага та ефективність використання служб WWW в інформаційних та виробничих процесах?

## Тема 2. АНАЛІЗ ОБМІНУ ІНФОРМАЦІЄЮ У КОМП'ЮТЕРНИХ МЕРЕЖАХ

**Завдання 2.1.** Ознайомитись із теорією комутації інформаційних потоків.

**2.1.1.** Комутація каналів у глобальних мережах є процесом, який за запитом здійснює поєднання двох або більше джерел даних і забезпечує монопольне використання каналу передачі даних до моменту роз'єднання. Комутація каналів полягає в утворенні безперервного складеного фізичного каналу із послідовно сполучених окремих каналних ділянок для прямої передачі даних між вузлами. Окремі канали поєднуються між собою спеціальною апаратурою - **комутаторами**, які можуть впроваджува-

ти зв'язки між будь-якими кінцевими вузлами мережі.

**2.1.2.** Комутація повідомлень у глобальних мережах є процесом пересилання даних, який складається із приймання, зберігання, вибору початкового напрямку і наступної передачі повідомлень без порушення їх цілісності.

**Повідомленнями** називаються дані, які об'єднані за змістом, мають певну структуру і придатні для опрацювання, пересилання або використання. Повідомлення пересилаються між транзитними комп'ютерами мережі із тимчасовою **буферизацією** їх на дисках кожного комп'ютера. Джерелами повідомлень можуть бути голос, зображення, текст, дані. Поєднання відправника і одержувача із можливістю обміну повідомленнями без помітних тимчасових затримок характеризує режим роботи **online**. Істотні затримки із запам'ятовуванням інформації у проміжних вузлах властиві режиму **offline**.

**2.1.3.** Комутацією пакетів у глобальних мережах є передача повідомлень, сформованих у вигляді адресованих пакетів. Канал передачі, завантажений під час передачі цим пакетом, потім звільняється для інших пакетів. Комутатори мережі, у ролі яких виступають шлюзи і маршрутизатори, приймають пакети від кінцевих вузлів і за адресною інформацією передають їх один одному до станції призначення.

**2.1.4.** У глобальних мережах для передачі інформації використовують такі види комутації:

- 1) комутація каналів - для передачі аудіо інформації по звичних телефонних лініях зв'язку;
- 2) комутація повідомлень - для передачі електронної пошти, у телеконференціях, в електронних новинах;
- 3) комутація пакетів - для передачі даних, аудіо- та відео інформації.

Перевагою мереж комутації каналів є простота реалізації (утворення безперервного складеного фізичного каналу), а недоліком - низький коефіцієнт використання каналів, висока вартість передачі даних, підвищений час очікування звільнення каналу іншими користувачами.

При комутації повідомлень передача даних здійснюється після звільнення каналу і до моменту досягнення даними адресату. Кожний сервер виконує приймання, перевірку, збирання, маршрутизацію і передачу повідомлення.

До переваги можна віднести зменшення вартості передачі даних. Недоліком цього способу є низька швидкість передачі інформації тобто неможливість ведення діалогу між користувачами.

Пакетна комутація полягає в обміні невеликими пакетами (частинами повідомлень) фіксованої структури і не сприяє утворенню черг у вузлах комутації.

Перевага такої комутації полягає у швидкому поєднанні, надійності та ефективності використання мережі.



## Завдання 2.2. Вивчити методи адресації у комп'ютерних мережах.

**2.2.1.** Передача повідомлень у комп'ютерних мережах здійснюється за допомогою спеціальних правил, званих **протоколом передачі даних**. У протоколі **ТСР/ІР** (Transmission Control Protocol / Internet Protocol) адресацію забезпечує протокол **ІР**. Згідно стандарту **ІР**, кожному хосту призначається **ІР-адреса** - унікальне число довжиною у 32 біти (4 байти), яке за своєю структурою поділяється (рис. 2.1) на дві логічні частини - префікс і суфікс.

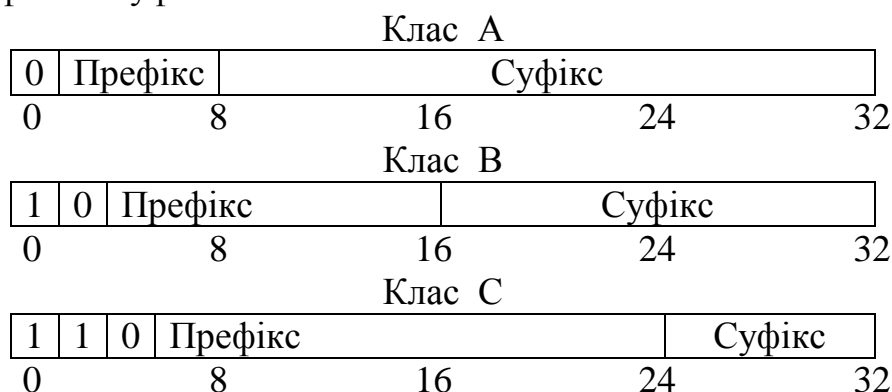


Рис. 2.1. Три основні класи ІР-адрес.

Такий розподіл потрібний для ефективної маршрутизації. **Префікс** адреси характеризує фізичну мережу, до якої підключений комп'ютер, а **суфікс** - окремий **хост** (вузол) цієї мережі.

Схема, яка названа ІР-адресацією на основі класів, передбачає розподіл простору ІР-адрес на три основні класи; кожен клас має префікс і суфікс відмінних розмірів у порівнянні із іншими класами. Перші біти адреси визначають частини адреси, що належать до номерів мережі і до номера вузла. Префікс довжиною у **n** біт дозволяє призначити **2<sup>n</sup>** унікальних номерів мережі, а суфікс завдовжки **m** біт дозволяє призначити **2<sup>m</sup>** номерів хостів у цій мережі.

Якщо адреса починається з нуля (**0**), то мережа належить класу **А**, а номер мережі займає **один** байт; інші 3 байти інтерпретуються як номер вузла у мережі.

Мережі класу **А** мають номери у діапазоні від **1** до **126**; кількість вузлів у них має бути більше 216, але не перевищувати **224**.

Якщо перші два біта адреси починаються послідовністю **10**, то мережа належить класу **В** і є мережею середніх розмірів із числом вузлів від **28** до **216**. У мережах класу **В** для адреси мережі і адреси вузла відводиться по **16** бітів, тобто по **2** байти.

Якщо адреса починається послідовністю **110**, то мережа належить класу **С**, а число вузлів не перевищує **28**. Для адреси мережі відводиться **24** біти, а для вузла - **8** бітів.

**2.2.2.** Для взаємодії із користувачем у програмному забезпеченні використовується **крапково-десятькова** система позначень IP-адрес. У цьому випадку IP-адреса поділяється на чотири відокремлених крапками октети (ділянки по 8 біт). Наприклад, 32-розрядне двійкове число **10000000 10000000 11111111 00000000** має у цій системі вигляд **128.128.255.0** і належить класу **B**.

Перетворення символічних адрес в IP-адреси виконує служба **DNS** (Domain Name System). Вона є розподіленою базою даних, яка підтримує ієрархічну систему імен для ідентифікації вузлів у мережі Інтернет. База даних **DNS** має структуру дерева, званого **доменним простором імен**, де кожен **домен** (вузол дерева) має назву (ім'я) і може містити **піддомени**.

**Назва** домену ідентифікує його **місце** у цій базі даних за відношенням до батьківського домену, причому крапки у назві відокремлюють частини, які відповідають **вузлам** домену. Кожний хост у мережі Інтернет однозначно визначається своїм **повним доменним ім'ям**, що містить імена усіх доменів за напрямом від хосту до кореня.

### **Завдання 2.3.** Дослідити принцип дії утиліти **ping.exe**

**2.3.1.** Моніторинг та аналіз мережі є важливими етапами контролю роботи мережі. Для виконання цих завдань регулярно виконується збирання даних, які є базою для виміру реакції мережі на зміни і перевантаження. Для того, щоб здійснити мережну передачу, необхідно перевірити коректність приєднання клієнта до мережі, наявність у клієнта хоча б одного протоколу сервера, знати IP-адреси комп'ютерів мережі.

Тому у мережних операційних системах, і зокрема у Windows XP, існують численні потужні службові програми (утиліти) для пересилання текстових повідомлень, управління загальними ресурсами, діагностування мережних поєднань, пошуку та опрацювання помилок. Використання утиліт відбувається у режимі дискової операційної системи (DOS).

**2.3.2.** Утиліта **ping** використовується для перевірки досяжності віддаленого хосту. У ній зазначаються запити і відлуння повтору за протоколом **ICMP** (Internet Control Message Protocol - протокол міжмережних управляючих повідомлень). Для транспортування повідомлень **ICMP** використовується протокол **IP**. Запити **ICMP інкапсулюються** (групуються усередині) у створений пакет **IP-дейтаграм**. Після відправки запиту програма упродовж певного короткого часу чекає відповідь віддаленого комп'ютеру. Формат утиліти **ping** має такий вигляд:

**ping [ - n ][ - w ][ - t ]** IP-адреса або **DNS-ім'я** віддаленого хосту, де **n** - число посланих запитів; **w** - час очікування відповіді віддаленого хосту, яке за замовчуванням дорівнює 1000 мсек.

Для того, щоб визначити час досягнення пакетом певного вузла (час транспортування інформації), необхідно ураховувати, що параметр часу в утиліті **ping** зазначає час відправки повідомлення і очікування ві-

дгуку від вузла призначення - тобто подвоєний час передачі пакету.

**2.3.3.** Для дослідження дії зазначеної утиліти слід перейти у середовище DOS: викликати на екран **Total Commander**, викликати в обох його вікнах кореневий каталог диску **C:** і виконати команду **Команды→Запустить сеанс DOS** - виводиться вікно середовища DOS. У верхній частині вікна наведені системні повідомлення, а під ними зазначений повний шлях до місця, звідки виконано звернення до DOS - у даному випадку - **C:\>**, праворуч якого знаходиться текстовий курсор (рис. 2.2).

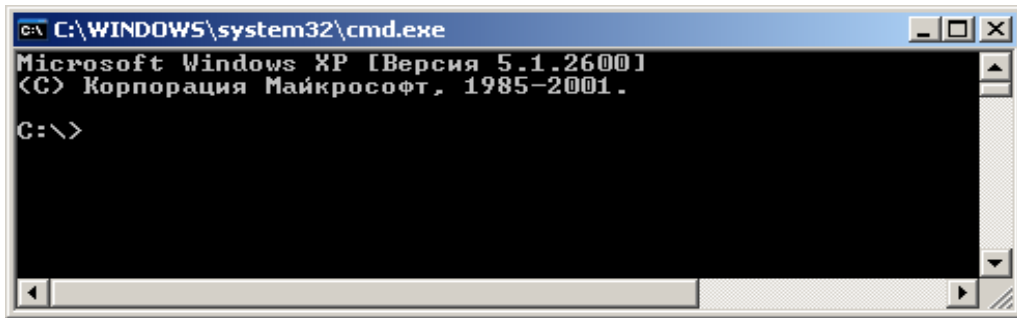


Рис. 2.2. Вікно введення команд DOS.

**2.3.4.** Ввести по черзі чотири варіанти утиліти **ping**:

<b>ping kpi.ua</b>	<b>i</b>	<b>ping -n 1 kpi.ua</b>
<b>ping 77.47.133.2</b>	<b>i</b>	<b>ping -n 1 77.47.133.2</b>

Введення команди кожного варіанта починається після імені диска і закінчується натисканням [**Enter**]. Через декілька секунд на екран виводиться повідомлення відповіді, яке автоматично завершується символами **C:\>**. Виконання команд відображене на рис. 2.3 та 2.4.

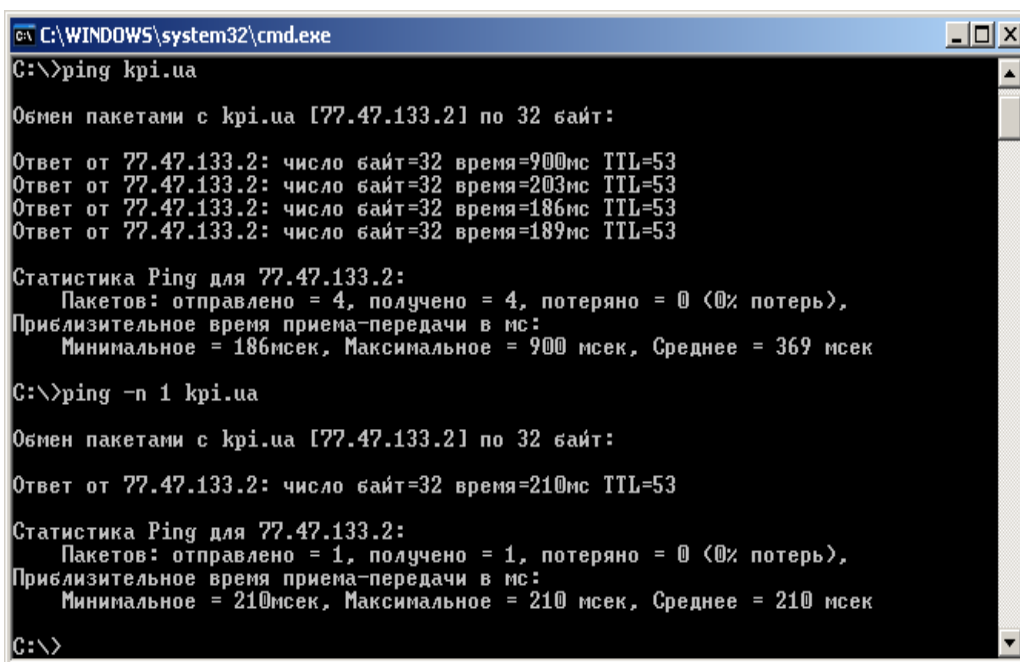


Рис. 2.3. Виконання команди **ping** за двома варіантами.

```
C:\WINDOWS\system32\cmd.exe
C:\>ping 77.47.133.2
Обмен пакетами с 77.47.133.2 по 32 байт:
Ответ от 77.47.133.2: число байт=32 время=188мс TTL=53
Ответ от 77.47.133.2: число байт=32 время=212мс TTL=53
Ответ от 77.47.133.2: число байт=32 время=186мс TTL=53
Ответ от 77.47.133.2: число байт=32 время=190мс TTL=53
Статистика Ping для 77.47.133.2:
  Пакетов: отправлено = 4, получено = 4, потеряно = 0 (0% потерь),
Приблизительное время приема-передачи в мс:
  Минимальное = 186мсек, Максимальное = 212 мсек, Среднее = 194 мсек
C:\>ping -n 1 77.47.133.2
Обмен пакетами с 77.47.133.2 по 32 байт:
Ответ от 77.47.133.2: число байт=32 время=211мс TTL=53
Статистика Ping для 77.47.133.2:
  Пакетов: отправлено = 1, получено = 1, потеряно = 0 (0% потерь),
Приблизительное время приема-передачи в мс:
  Минимальное = 211мсек, Максимальное = 211 мсек, Среднее = 211 мсек
C:\>
```

Рис. 2.4. Виконання команди ping за двома варіантами.

**2.3.5.** Записати у конспекті відображення виконання команд; порівняти результати і пояснити різниці у відображеннях.

**Завдання 2.4.** Дослідити принцип дії утиліти **tracert.exe**.

**2.4.1.** Утиліта **tracert** призначена для відстежування маршруту проходження пакету до зазначеного хосту. Утиліта передає декілька дейтаграм і чекає відповіді на кожну із них. Перед відправленням першої дейтаграми значення **TTL** (Time To Live - час мандрування пакету мережею, тобто "час його життя") для неї дорівнює **1**. Час "життя" пакету вимірюється у секундах і визначається джерелом передачі засобами протоколу **IP**. Перший маршрутизатор, який виявиться на шляху проходження цієї дейтаграми, зменшить значення **TTL** на одиницю і, якщо це значення стане рівним **0**, поверне помилку **ICMP** про закінчення **TTL**. Оскільки повідомлення **ICMP** передається також у вигляді дейтаграми **IP**, то **tracert** може визначити **IP**-адресу джерела і вивести на екран адресу маршрутизатора.

Для наступної дейтаграми значення **TTL** буде збільшено на одиницю; це повторюватиметься доти, поки не буде одержано запит від комп'ютера призначення. Формат команди **tracert** має такий вигляд:

**tracert [- h] [- w] [- d] IP-адреса (або DNS-ім'я кінцевого комп'ютеру),**

де **h** визначає максимальне значення **TTL** (за замовчуванням дорівнює **30**); **w** зазначає інтервал у мсек, упродовж якого очікуватиметься відповідь; параметр **d** використовується для відмови визначення **DNS**-імен за **IP**-адресами маршрутизаторів.

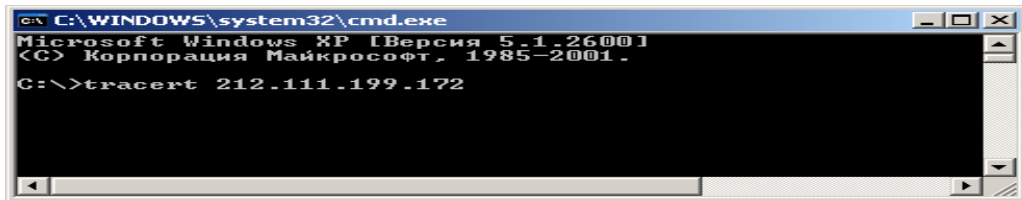
**2.4.2.** Визначити доменне ім'я вузла **212.111.199.172** такими діями.

1) Викликати на екран середовище DOS (див. п. 2.3.3).

2) Ввести команду **tracert 212.111.199.172** (рис. 2.5) і натиснути [Enter]; результат виконання зазначеної команди відображено на рис. 2.6.

3) Записати у конспект зміст останнього зображення екрану.

2.4.3. Виконати команду **tracert** для вузлів **194.0.131.18** та **18.19.70.3**. Записати у конспект усю інформацію, яка буде відображена на екрані після виконання команд.

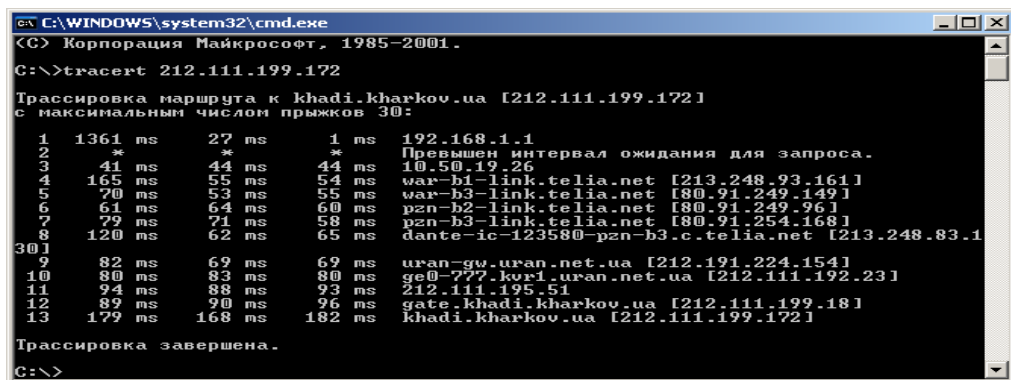


```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Версия 5.1.2600]
(C) Корпорация Майкрософт, 1985-2001.

C:\>tracert 212.111.199.172
```

Рис. 2.5. Вікно DOS із відображенням введення команди.

На екран виводиться перелік виконаних кроків (стрибків) для досягнення вузла призначення (рис. 2.6).



```
C:\WINDOWS\system32\cmd.exe
(C) Корпорация Майкрософт, 1985-2001.

C:\>tracert 212.111.199.172

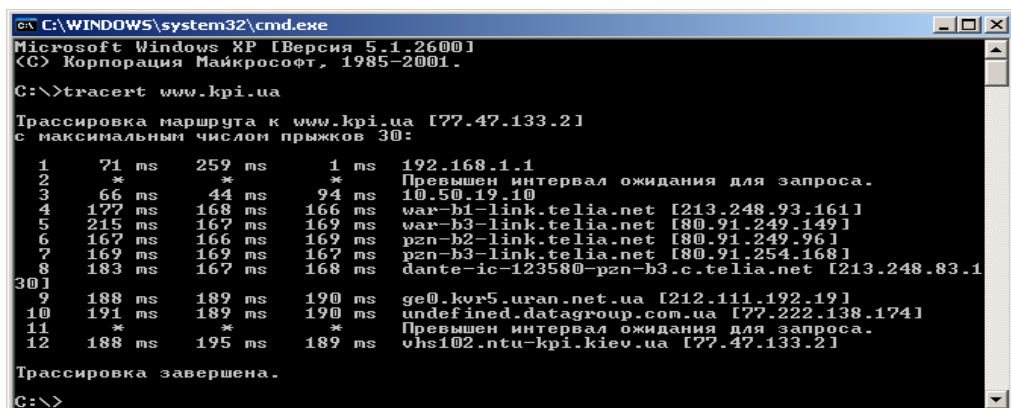
Трассировка маршрута к khadi.kharkov.ua [212.111.199.172]
с максимальным числом прыжков 30:
  0  1361 ms    27 ms      1 ms     192.168.1.1
  1  *          *          *          Превышен интервал ожидания для запроса.
  2  *          *          *          10.50.19.26
  3  41 ms     44 ms     44 ms     war-b1-link.telia.net [213.248.93.161]
  4  165 ms    55 ms     54 ms     war-b3-link.telia.net [80.91.249.149]
  5  70 ms     53 ms     55 ms     pzn-b2-link.telia.net [80.91.249.96]
  6  61 ms     64 ms     60 ms     pzn-b3-link.telia.net [80.91.254.168]
  7  79 ms     71 ms     58 ms     dante-ic-123580-pzn-b3.c.telia.net [213.248.83.130]
  8  120 ms    62 ms     65 ms
30  9  82 ms     69 ms     69 ms     uran-gw.uran.net.ua [212.191.224.154]
 10  80 ms     83 ms     80 ms     ge0-777.kvr1.uran.net.ua [212.111.192.23]
 11  94 ms     88 ms     93 ms     212.111.195.51
 12  89 ms     90 ms     96 ms     gate.khadi.kharkov.ua [212.111.199.18]
 13  179 ms    168 ms    182 ms    khadi.kharkov.ua [212.111.199.172]

Трассировка завершена.

C:\>
```

Рис. 2.6. Виконання команди **tracert** із IP-адресою.

2.4.4. Визначити IP-адресу Національного технічного університету України (**kpi** - Київського політехнічного інституту), використовуючи доменне ім'я, тобто ввести утиліту **tracert www.kpi.ua** (рис. 2.7).



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Версия 5.1.2600]
(C) Корпорация Майкрософт, 1985-2001.

C:\>tracert www.kpi.ua

Трассировка маршрута к www.kpi.ua [77.47.133.2]
с максимальным числом прыжков 30:
  1  71 ms     259 ms     1 ms     192.168.1.1
  2  *          *          *          Превышен интервал ожидания для запроса.
  3  66 ms     44 ms     94 ms     10.50.19.10
  4  177 ms    168 ms    166 ms    war-b1-link.telia.net [213.248.93.161]
  5  215 ms    167 ms    169 ms    war-b3-link.telia.net [80.91.249.149]
  6  167 ms    166 ms    169 ms    pzn-b2-link.telia.net [80.91.249.96]
  7  169 ms    169 ms    167 ms    pzn-b3-link.telia.net [80.91.254.168]
  8  183 ms    167 ms    168 ms    dante-ic-123580-pzn-b3.c.telia.net [213.248.83.130]
30  9  188 ms    189 ms    190 ms    ge0.kvr5.uran.net.ua [212.111.192.19]
 10  191 ms    189 ms    190 ms    undefined.datagroup.com.ua [77.222.138.174]
 11  *          *          *          Превышен интервал ожидания для запроса.
 12  188 ms    195 ms    189 ms    vhs102.ntu-kpi.kiev.ua [77.47.133.2]

Трассировка завершена.

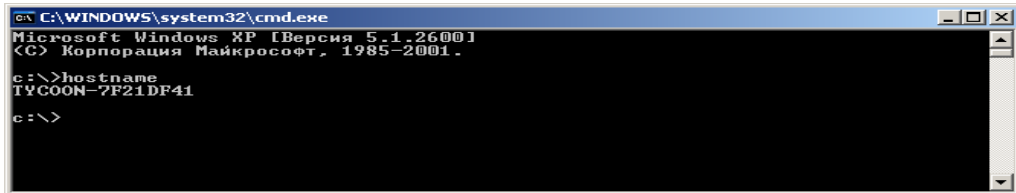
C:\>
```

Рис. 2.7. Виконання команди **tracert** із доменним ім'ям.

Порівняти зображення рис. 2.6 та 2.7; записати у конспект висновки у чому полягає різниця.

**Завдання 2.5.** Дослідити утиліти аналізу стану комп'ютера.

**2.5.1.** Утиліта **hostname** (рис. 2.8) сприяє виведенню на екран імені локального комп'ютера (хосту). Вона доступна у присутності підтримки протоколу **TCP/IP**.



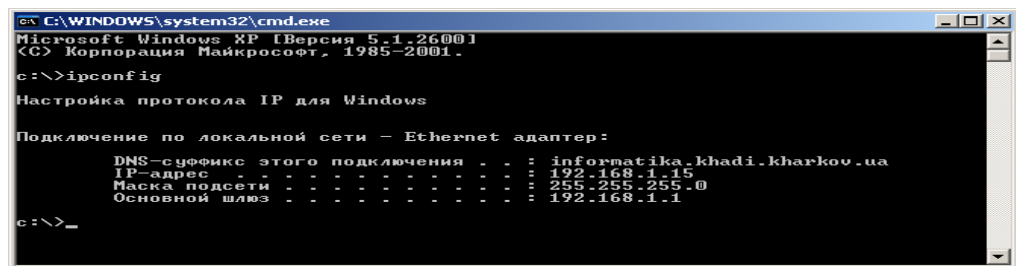
```
cmd C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Версия 5.1.26001]
(C) Корпорация Майкрософт, 1985-2001.

c:\>hostname
TUCOON-7F21DF41

c:\>
```

Рис. 2.8. Приклад виконання утиліти **hostname**.

**2.5.2.** Утиліта **ipconfig** (рис. 2.9) сприяє виведенню на екран діагностичної інформації про поточну конфігурацію мережі **TCP/IP** і про поточні конфігурації **IP**-адрес комп'ютерів мережі.



```
cmd C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Версия 5.1.26001]
(C) Корпорация Майкрософт, 1985-2001.

c:\>ipconfig

Настройка протокола IP для Windows

Подключение по локальной сети - Ethernet адаптер:

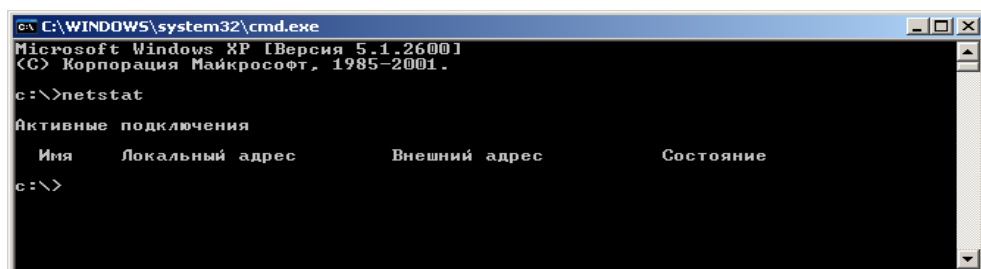
DNS-суффикс этого подключения . . . : informatika_khadi.kharkov.ua
IP-адрес . . . . . : 192.168.1.15
Маска подсети . . . . . : 255.255.255.0
Основной шлюз . . . . . : 192.168.1.1

c:\>_
```

Рис. 2.9. Приклад виконання утиліти **ipconfig**.

Використання команд **ipconfig** без параметрів сприяє виведенню повної конфігурації **TCP/IP** разом із **IP**-адресами і маскою підмережі.

**2.5.3.** Утиліта **netstat** (рис. 2.10) виводить статистику протоколу і поточних поєднань мережі **TCP/IP**. Ця команда доступна тільки після зазначення підтримки протоколу **TCP/IP**.



```
cmd C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Версия 5.1.26001]
(C) Корпорация Майкрософт, 1985-2001.

c:\>netstat

Активные подключения

Имя          Локальный адрес          Внешний адрес          Состояние

c:\>
```

Рис. 2.10. Приклад виконання утиліти **netstat**.

## Завдання 2.6. Вивчити принцип дії електронної пошти.

**2.6.1.** Електронні поштові скриньки створюються на поштових серверах. У ці ящики надходять листи на адресу користувача, який може також переадресувати пошту із сервера на свій комп'ютер за допомогою програм, які називаються **поштовими клієнтами**. Формат адреси електронної пошти має такий вигляд:

**ім'я користувача@ім'я домену.зона**

[ivanii@rambler.ru](mailto:ivanii@rambler.ru)

У цій адресі **ivan** - ім'я адресату; **@** - службовий символом, що відокремлює ім'я користувача від імені домену; **rambler** - ім'я домену, **ru** - ім'я зони Інтернету, в якій розташований домен.

У прикладі ім'я домену (домен другого рівня) є звичайною назвою фірми, організації, провайдера; зона (домен першого рівня) призначається для країни.

**2.6.2.** Окрім зазначеної вище, використовується інша (тематична) система імен, в якій домен верхнього рівня складається із трьох літер і визначає належність адреси до одного із таких класів:

- 1) com - комерційні організації;
- 2) edu - освітні установи;
- 3) gov - урядові організації;
- 4) org - некомерційні організації;
- 5) net - мережі.

Крім того, домени Інтернету укладаються один у другий; чим дрібніше домен, тим із більшої кількості сегментів складається його позначення. Наприклад, домен **khadi.kharkov.ua** належить одночасно домену університету ХНАДУ, домену міста і домену держави України.

**2.6.3.** Адреса електронної поштової скриньки надається **провайдером** (організація, що забезпечує доступ до Інтернету). На теперішній час багато провайдерів пропонують дешеві пакети доступу до Інтернету, але без надання поштових скриньок.

Проте, в Інтернеті існує також безліч серверів, які надають безкоштовну поштову скриньку усім бажаючим. Адреса електронної пошти є повністю незалежною. Можна скільки завгодно міняти організацію і навіть місце проживання - це ніяк не відображається на мережній адресі.

Електронна адреса довговічна - знайти людину за електронною адресою можна навіть через багато років. Найважливіше, що фізично знайти людину за такою адресою набагато важче (практично неможливо), ніж, наприклад, за номером телефону. Це означає, що електронна адреса забезпечує надійний захист. Усі листи зберігаються в одному місці - на сервері; вони доступні власникові поштової скриньки із будь-якого комп'ютера, підключеного до мережі.



## Завдання 2.7. Створити поштову скриньку в ukr.net.

2.7.1. Для створення електронної поштової скриньки слід в адресному рядку оглядача ІЕ ввести адресу сайту <http://freemail.ukr.net> і натиснути [Enter]. У відповідь на екран виводиться головне вікно української мережної поштової служби (рис. 2.11). Рекомендується користувачу самостійно уважно ознайомитись із змістом головного вікна.

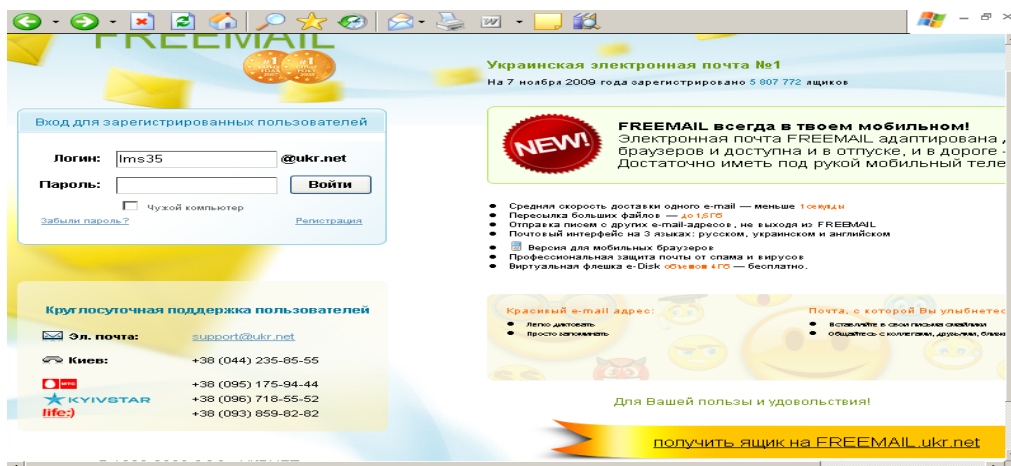


Рис. 2.11. Вікно поштової служби УКРNET.

Для користування поштовою службою цього сайту (як і інших світових сайтів) необхідно виконати реєстрацію особи користувача. Це виконується натисканням ЛК миші на гіперпосиланні **Регистрация** у правому нижньому кутку прямокутника реєстрації. У відповідь відкривається чергове вікно (рис. 2.12) із емблемою і назвою **Создать аккаунт УКРNET**, призначене для відповідей на питання анкети реєстрації.

Позначки "i" є інформаційними повідомленнями - при підведенні до них покажчика миші виводиться відповідна порада.

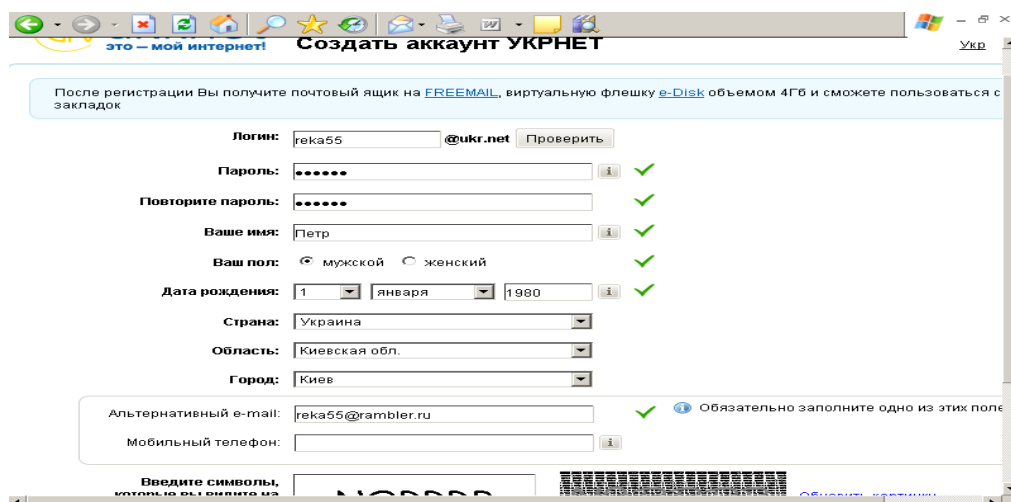


Рис. 2.12. Вікно введення параметрів реєстрації в УКРNET.

У зазначеній анкеті у полі **Логин:** користувач створює із комбінації літер і цифр поштове **ім'я-звернення**. Натисканням ЛК миші в області **Проверить** необхідно перевірити винятковість свого поштового імені. Якщо в Інтернеті знайдеться **подібне** ім'я, то на екран виводиться повідомлення про необхідність змінити **Логин:**; слід вибрати інше ім'я і повторити введення.

У полі **Пароль:** ввести для електронної поштової скриньки пароль (від шести до восьми латинських літер і цифр). У паролі заборонено використовувати кирилицю; не слід вибирати прості паролі - їх легко підібрати і скористатись. Інші поля анкети заповнюються за допомогою наведених на екрані пояснень і порад. Завершується реєстрація натисканням ЛК миші у полі (зеленого кольору) **Регистрация**. Через декілька секунд відкривається чергове вікно для роботи у середовищі електронної пошти (рис. 2.13).

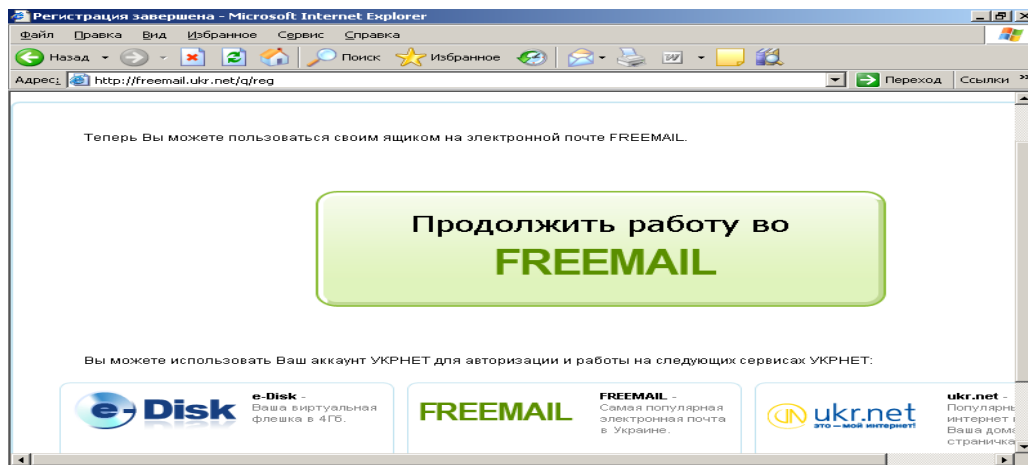


Рис. 2.13. Вікно для зазначення роботи із поштовою програмою **УКРNET**.

**2.7.2.** При натисканні ЛК миші в області **Продолжить работу во FREEMAIL** виконується перехід до первинного вікна (рис. 2.11), в якому треба надрукувати поштове ім'я і пароль і натиснути ЛК миші на гіперпосиланні **Войти**.

**2.7.3.** Відкривається наступне поштове вікно (рис. 2.14), у верхній частині якого розташований перелік можливих операцій.

Для вивчення можливостей цієї поштової служби слід виконати гіперпосилання **Помощь**.

Користувачам рекомендується самостійно вивчити відомості, наведені на сторінках зазначеної допомоги.

**2.7.4.** У вікні листування (рис. 2.14) натиснути ЛК миші на гіперпосиланні **Написать письмо** - на екран виводиться стандартна форма із відповідними текстовими полями, в які користувач вводить свою інформацію. У полі під рядком вибору шрифтів вводиться текст листа.

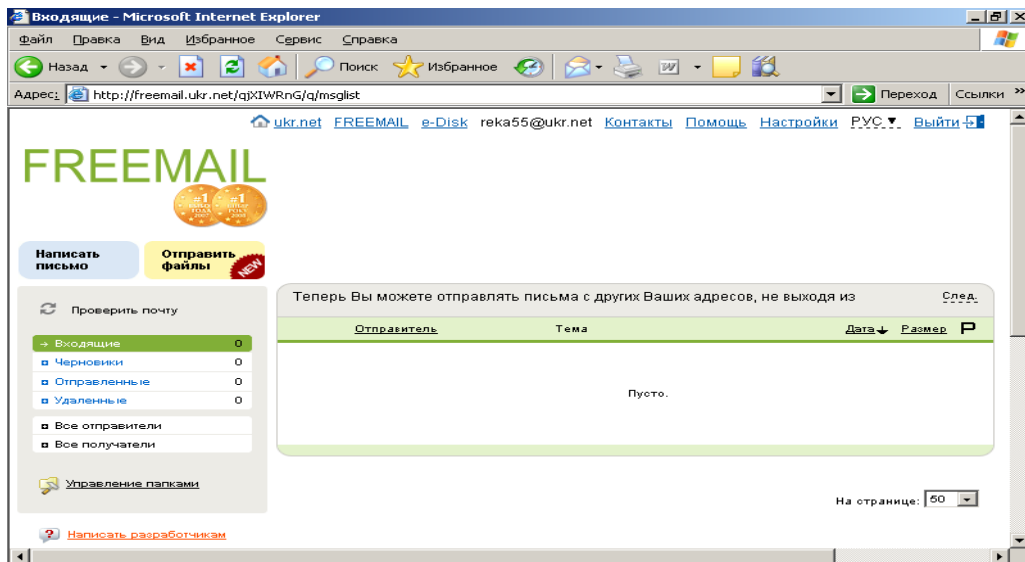


Рис. 2.14. Вікно поштових операцій програми УКРНЕТ.

Якщо автору листа необхідно надіслати графічні матеріали (креслення, рисунки, фотографії, таблиці, тощо), то є можливість додати до листа зазначені файли за допомогою команди **Прикріпить**. В окремому полі ліворуч відображається дійсний стан із листами у цій поштової скринці.

**2.7.5.** Після завершення знайомства із можливостями і способами користування поштовою програмою виконати гіперпосилання [ukr.net](http://ukr.net) (вгорі, перше ліворуч), яке сприяє переходу на головну сторінку цієї поштової програми (рис. 2.15).

Користувачам рекомендується **самостійно** ознайомитись із інформацією, наданою у цьому і у підлеглих йому вікнах.

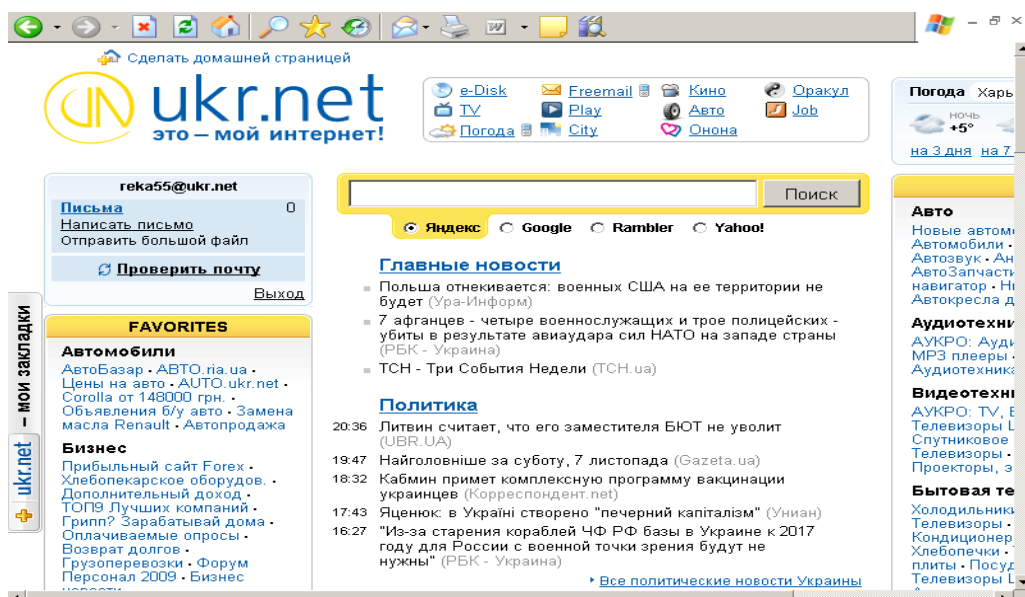


Рис. 2.15. Головне вікно поштової програми УКРНЕТ.

## 2.8. Контрольні запитання

1. Що являє собою комутація каналів у глобальних мережах ?
2. Які пристрої застосовуються для поєднання вузлів мережі ?
3. Що являють собою "пакети повідомлень" ?
4. Як реалізується передача інформації у глобальних мережах ?
5. Чим є та як улаштовані "протоколи передачі даних" ?
6. Що являють собою службові програми - утиліти ?
7. Яку роль виконують команди ping.exe та tracert.exe ?
8. Який принцип дії електронної пошти ?
9. Які елементи складають формат адреси електронної пошти ?
10. Які дії необхідно виконувати для створення поштової скриньки в ukr.net ?

## Тема 3. ДОСЛІДЖЕННЯ ІНФОРМАЦІЙНИХ МОЖЛИВОСТЕЙ ВЕБ-ДОКУМЕНТІВ

### Завдання 3.1. Дослідити структуру веб-документів.

**3.1.1.** Сучасний етап розвитку Інтернету почався на початку 1990-х років із появою протоколу обміну інформацією **НТТР** (Hyper Text Transfer Protocol - протокол передачі гіпертексту). Разом із цим протоколом з'явилась і служба **World Wide Web** (WWW), що є великою мережею серверів **НТТР**, які передають файли через Інтернет.

Основну частину цих файлів складають **веб-сторінки** - спеціальні файли, написані мовою **HTML** (Hyper Text Markup Language - мова розмітки гіпертексту). Веб-сторінка є окремим веб-документом (або html-документом) служби WWW і може містити текст, графіку, звуковий супровід, анімацію та інші мультимедійні об'єкти, а також гіпертекстові посилання. Веб-сторінки публікуються в Інтернеті шляхом розміщення **веб-файлів** на серверах **НТТР** (веб-вузлах).

**3.1.2.** Для створення html-документів використовується будь-який текстовий редактор. Текст документів такого типу розмічується за допомогою **тегів** (**tag** - ознака). В html-документі зазначаються синтаксис і розташування тегів, відповідно до яких Web-браузер відображатиме інформацію, але текст самих тегів буде прихований. Файлам html-документів привласнюється тип **.htm**. Усі теги обмежуються символами "<" та ">". Значна частина тегів укладають пару - **стартовий** (що відкриває) та **кінцевий** (що закриває); наприклад,

**<p>Информационная система</p>**

Тут стартовим тегом є **<p>**, а кінцевим - тег **</p>**, що відрізняється від стартового додаванням символу **" / "** (прямий слеш).

**3.1.3.** Теги можуть мати атрибути та значення атрибутів. Атрибути додають для розширення або модифікації дій тегу; деякими із таких атрибутів є такі:

**src** - атрибут, значенням якого є розміри зображення;

**width** та **height** - зазначають ширину і висоту (у пікселях);

**alt** - альтернативний текст, який виводиться, якщо браузер не у змозі вивести зображення.

Правила запису атрибутів і значень наступні:

1) атрибути зазначають після імені тегу і пропуску;

2) атрибути відділяють один від одного пропусками;

3) порядок дотримання атрибутів довільний;

4) атрибути не треба удруге описувати у кінцевому тегу;

5) значення атрибутів записують у лапках " ". після символу =;

6) назви тегів не мають містити пропуски.

Наступний приклад ілюструє використання атрибутів:

```
<h1 align="center">Моя первая Web-страница</h1>
```

Цим тегом визначений заголовок, який на екрані буде розташований у центрі. Для мови HTML байдуже, в якому регістрі записані назви тегів та їх атрибути; наприклад,

```
<h1 align="center">Моя первая Web-страница</h1>
```

```
<H1 ALIGN="center">Моя первая Web-страница</H1>
```

Результати виконання наведених тегів будуть ідентичні, причому у цьому прикладі обидва записи коректні. Крім того, в одному документі можна використовувати для запису тегів і верхній і нижній регістри. Проте, щоб уникнути помилок і прискорити роботу із текстом за допомогою клавіатури, рекомендується використовувати **нижній регістр**.

**3.1.4.** Відображення html-документу у вікні Web-браузера виконується відповідно до структури html-тегів. У кожному html-документі мають бути присутні три головні частини: оголошення HTML, заголовок і тіло документу. Основні теги та їх призначення надані у табл. 3.1. Рекомендується уважно вивчити зміст таблиці.

1) Оголошення HTML забезпечує пара <html> і </html>. Вони повідомляють Web-браузеру, що між ними знаходиться документ у форматі HTML, причому першим тегом у документі має бути тег <html> (на самому початку документу), а останнім - </html> (у самому кінці документу).

2) Заголовок визначає пара тегів <head> і </head>. Між цими тегами розташовується інформація про документ (назва, ключові слова для пошуку, опис, тощо). Найважливішою із них є назва документу, яка знаходиться на екрані у першому (верхньою) рядку вікна браузера. Спеціальні **програми-спайдери** пошукових систем використовують назву документа для побудови своїх баз даних.

Назва html-документа у вигляді текстового рядка розміщується між тегами <title> і </title>. Це єдиний обов'язковий елемент заголовної частини документа; він відображається у **першому** рядку вікна браузера.

Назва документа є посиланням на сторінку сайту. У вікні браузера у полі пошуку вводяться ключові слова відшукуваної теми. Пошукова система відповідно до запиту виводить деяку кількість посилань на різні документи. Одним із таких посилань стає текст, який знаходиться між тегами **<title>** та **</title>**.

3) Головна частина документа (**тіло** html-документа) розташовується відразу за заголовком і знаходиться між тегами **<body>** та **</body>**. Перший із них має бути розташований відразу після тегу **</head>**, а другий - перед тегом **</html>**. Тіло htm-документа складає інформативну частину документа, яка має бути відформатована засобами HTML.

Таблиця 3.1. Основні теги мови HTML

Теги	Призначення
<code>&lt;html&gt;</code> і <code>&lt;/html&gt;</code>	Початок/Кінець документа
<code>&lt;head&gt;</code> і <code>&lt;/head&gt;</code>	Зазначають початок і кінець службової області html-документа
<code>&lt;title&gt;</code> і <code>&lt;/title&gt;</code>	Усю інформацію, розташовану між ними, браузер сприймає як назву html-документа. Ця інформація виводиться у рядку заголовка вікна браузера
<code>&lt;body&gt;</code> і <code>&lt;/body&gt;</code>	Зазначають початок і кінець тіла (основної області) html-документа
<code>&lt;body bgcolor=?&gt;</code>	Зазначає колір фону html-документа, використовуючи значення кольорів у вигляді поєднання символів RRGGBB
<code>&lt;body text=?&gt;</code>	Зазначає колір тексту html-документу, використовуючи значення кольорів у вигляді поєднання символів RRGGBB
<code>&lt;body link=?&gt;</code>	Встановлює колір гіперпосилань, використовуючи значення кольорів у вигляді поєднання символів RRGGBB
від <code>&lt;h1&gt;</code> і <code>&lt;/h1&gt;</code> до <code>&lt;h6&gt;</code> і <code>&lt;/h6&gt;</code>	Заголовки від першого до шостого рівнів
<code>&lt;p&gt;</code> і <code>&lt;/p&gt;</code>	Початок / Кінець абзаца
<code>&lt;p align=?&gt;</code>	Вирівнює параграф відносно однієї із сторін html-документа: left, right, або center
<code>&lt;br&gt;</code>	Кінець рядка
<code>&lt;marquee&gt;</code> і <code>&lt;/marquee&gt;</code>	Зазначають виведення тексту у вигляді рухливого рядка
<code>&lt;img src="[ім'я файлу]"&gt;</code>	Вставка графічного зображення
<code>&lt;center&gt;</code> і <code>&lt;/center&gt;</code>	Зазначення розташування тексту між тегами, по центру
<code>&lt;hr&gt;</code>	Виведення на екран горизонтальної лінії

**3.1.6.** Уважно ознайомитись із текстом html-документа на рис. 3.1.

Визначити за допомогою табл. 3.1 призначення усіх тегів, які утворюють html-документ; переписати текст рис. 3.1 до робочого зошита (на розвороті двох чистих сторінок), супроводжуючи коментарями щодо призначення кожного рядка. Рекомендується усі досягнуті результати у процесі досліджень також відображати у зошиті.

Зазвичай запис кожного рядка html-документа починається у першій позиції. На рис. 3.1 html-документ має структурований вигляд із метою звернути увагу користувача на симетрію розташування тегів і на змістовну інформацію окремих фрагментів html-документа. Лініями із стрілками позначена зона дії тих або інших контейнерних тегів html-документа.

	Рядки
<html> .....	1
<head> .....	2
<title> .....	3
Страница Иванова И. И. ....	4
</title> .....	5
</head> .....	6
<body bgcolor=blue text=red link=white>.....	7
<center>.....	8
<h1> Web-страница Иванова И. И .....	9
</h1> .....	10
</center> .....	11
<p align=center> .....	12
студента группы ТД21 факультета транспортных систем ... 13	
Харьковского национального автомобильно-дорожного..... 14	
университета   .....	15
  .....	16
<img width=50% height=250 src= .....	17
<a href="http://www.khadi.kharkov.ua/media%5Ckhadi.jpg">http://www.khadi.kharkov.ua/media%5Ckhadi.jpg</a> > .....	18
<marquee height=20 width=90% bgcolor=#ffff00> .....	19
Желаю успехов в учебе !!! .....	20
</marquee> .....	21
</p> .....	22
</body> .....	23
</html> .....	24

Рис. 3.1. Структуроване подання html-документа.



### Завдання 3.2. Дослідити процес створення html-документа.

**3.2.1.** Викликати на екран менеджер файлів Total Commander і згорнути його вікно на панель задач для швидких і зручних повторних викликів.

Відкрити стандартне доповнення **Блокнот** виконанням команди:

**Пуск→Программы→Стандартные→Блокнот**

Ввести за допомогою клавіатури у текстових полях **Блокнота** текст рис. 3.1, в якому відобразити прізвище та ініціали користувача і назву групи; введення усіх рядків рекомендується починати з першої позиції.

**3.2.2.** Зберегти створений текст у папці автора за допомогою команди **Файл→Сохранить как**. Ім'я текстового файлу, що зберігається, має складатись із прізвища автора, нижньої риски, символів **Веб** і номера модифікації завдання; наприклад: **Іванов\_Веб\_1** (перший варіант дослідження). При збереженні програма **Блокнот** самостійно надасть тип файлу (**.txt**). Згорнути усі відкриті програми на **Панель задач**.

**3.2.3.** За допомогою ярлика на **Рабочем столе** відкрити браузер **Opera**. Для відкриття у середовищі браузера створеного html-документу натиснути на панелі інструментів браузера кнопку **Открыть**.

На екран виводиться вікно **Открыть**; у полі **Папка** має знаходитись ім'я папки користувача, а в інформаційних полях - перелік змісту цієї папки. Для досягнення такого стану у полі папка повторними натисканнями кнопки списку (розташована у полі праворуч) досягти виведення переліку дисків и папок комп'ютеру, у тому числі диску **C:**; відкрити його натисканням ЛК миші - виводиться список вмісту диску **C:**. Позначити папку групи (наприклад, **ТД21**) і відкрити її подвійним натисканням ЛК миші; аналогічно відкрити папку користувача.

**3.2.4.** У даному випадку у вікні **Открыть** за замовчуванням передбачається відкриття файлу html-документа через те, що у полі **Тип файлов** зазначений тип html-файлу. Але html-документ (рис. 3.1) збережений як текстовий файл **Іванов\_Веб\_1.txt**. Для виклику цього файлу у вікні **Открыть** слід натиснути кнопку списку у полі **Тип файлов** і двічі натиснути ЛК миші на типі **Все файлы [ \*.\* ]** - в інформаційному полі виводиться ім'я створеного текстового файлу.

Позначити ім'я створеного файлу і двічі натиснути ЛК миші - у вікні браузера **Opera** виводиться текст створеного html-документа.

**3.2.5.** Для збереження тексту як веб-сторінки слід на панелі **Стандартная** браузера **Opera** натиснути кнопку **Сохранить** - виводиться вікно **Сохранить как**. У полі папки має знаходитись ім'я папки користувача. Для цього у полі **Тип файлов** натиснути кнопку списку і у списку типів файлів натисканням ЛК миші позначити тип **html-файл**, а потім натиснути кнопку **Сохранить**. У результаті виконаних дій створений документ буде збережений як веб-сторінка із типом файлу **.htm**; цей файл може бути відтворений будь-яким браузером.

Завершити створення веб-сторінки користувача закриттям браузеру за допомогою команди **Файл→Выход**.

**Завдання 3.3.** Дослідити методи зміни інформації в html-документі.

**3.3.1.** У html-документі (рис. 3.1) можна редагувати і форматувати текст, змінювати і додавати рисунки, звуки, кольори, мультимедійні ефекти. Після вивчення кожного із наступних завдань (3.3.4, 3.3.5, 3.3.6) користувачам рекомендується самостійно визначити зміну тих або інших тегів і ввести їх у текст html-документа.

Кожну зміну html-документа слід зберігати у папці користувача із черговим порядковим номером (див. п. п. 3.2.5 і 3.2.6).

**3.3.2.** Зміни в html-документі виконуються таким чином.

1) У середовищі Total Commander у папці користувача зазначити файл html-документа (наприклад, файл **Іванов\_Веб\_1.htm**) і двічі натиснути ЛК миші. Через декілька секунд на екран виводиться вікно браузера Opera, а ще через декілька секунд - результат розпізнавання браузером html-документа.

2) Для відкриття веб-сторінки слід виконати команду **Вид→Начальный код** - у полі браузера виводиться текст веб-документа. Виконати у html-файлі необхідні зміни, варіанти яких розглядаються у наступних завданнях цієї теми.

3) Після завершення редагування елемента html-документа слід на панелі **Стандартная** браузер Opera натиснути кнопку **Сохранить** - виводиться вікно **Сохранить как**. У полі **Имя файла** замінити цифру на чергове значення (наприклад, **Іванов\_Веб\_2.htm**) для збереження чергової модифікації цього файлу, а потім натиснути кнопку **Сохранить**.

4) Завершити роботу у вікні браузера командою **Файл→Выход**.

5) Перевірити результат модифікації веб-сторінки (наприклад, **Іванов\_Веб\_2.htm**): у середовищі Total Commander у папці користувача зазначити нову модифікацію html-файла і двічі натиснути ЛК миші - на екран виводиться результат розпізнавання позначеного файлу.

**Завдання 3.4.** Дослідити способи використання графічних зображень.

**3.4.1.** При використанні графічних зображень на веб-сторінках потрібно пам'ятати, що в Інтернеті важлива не художня якість картинки, а швидкість її завантаження, тобто мінімізація розміру картинки. Для розташування картинок на веб-сторінках в основному використовуються два графічні формати (типи): **.gif** та **.jpg**. Основна відмінність між ними у тому, що вони застосовуються для зберігання різних видів графіки.

На рис. 3.2 наведені URL-адреси картинок обох типів. Для зручності вибору користувачами варіантів у кінці рядка праворуч зазначені їх порядкові номери.

<a href="http://s48.radikal.ru/i120/0909/2f/87fbe63a9a27.jpg">http://s48.radikal.ru/i120/0909/2f/87fbe63a9a27.jpg</a>	00
<a href="http://gifanimation.ru/images/pticy/ptici10.gif">http://gifanimation.ru/images/pticy/ptici10.gif</a>	01
<a href="http://gifanimation.ru/images/pticy/ptici06.gif">http://gifanimation.ru/images/pticy/ptici06.gif</a>	02
<a href="http://gifanimation.ru/images/cvety/8036.gif">http://gifanimation.ru/images/cvety/8036.gif</a>	03
<a href="http://gifanimation.ru/images/cvety/flower01.gif">http://gifanimation.ru/images/cvety/flower01.gif</a>	04
<a href="http://gifanimation.ru/images/email/email22.gif">http://gifanimation.ru/images/email/email22.gif</a>	05
<a href="http://gifanimation.ru/images/knopki/button78.gif">http://gifanimation.ru/images/knopki/button78.gif</a>	06
<a href="http://gifanimation.ru/images/komputer/hw30.gif">http://gifanimation.ru/images/komputer/hw30.gif</a>	07
<a href="http://gifanimation.ru/images/komputer/hw24.gif">http://gifanimation.ru/images/komputer/hw24.gif</a>	08
<a href="http://gifanimation.ru/images/mehanizmy/peopls167.gif">http://gifanimation.ru/images/mehanizmy/peopls167.gif</a>	09
<a href="http://gifanimation.ru/images/mehanizmy/peopls33.gif">http://gifanimation.ru/images/mehanizmy/peopls33.gif</a>	10
<a href="http://gifanimation.ru/images/planety/b97.gif">http://gifanimation.ru/images/planety/b97.gif</a>	11
<a href="http://gifanimation.ru/images/predmety/14.gif">http://gifanimation.ru/images/predmety/14.gif</a>	12
<a href="http://gifanimation.ru/images/derevo/fish17.gif">http://gifanimation.ru/images/derevo/fish17.gif</a>	13
<a href="http://gifanimation.ru/images/derevo/fish24.gif">http://gifanimation.ru/images/derevo/fish24.gif</a>	14
<a href="http://gifanimation.ru/images/derevo/79.gif">http://gifanimation.ru/images/derevo/79.gif</a>	15
<a href="http://gifanimation.ru/images/derevo/54terg.gif">http://gifanimation.ru/images/derevo/54terg.gif</a>	16
<a href="http://gifanimation.ru/images/strelki/strelki09.gif">http://gifanimation.ru/images/strelki/strelki09.gif</a>	17
<a href="http://gifanimation.ru/images/strelki/strelki05.gif">http://gifanimation.ru/images/strelki/strelki05.gif</a>	18
<a href="http://gifanimation.ru/images/raznoe/80.gif">http://gifanimation.ru/images/raznoe/80.gif</a>	19
<a href="http://gifanimation.ru/images/raznoe/105.gif">http://gifanimation.ru/images/raznoe/105.gif</a>	20
<a href="http://gifanimation.ru/images/raznoe/animal01.gif">http://gifanimation.ru/images/raznoe/animal01.gif</a>	21
<a href="http://gifanimation.ru/images/raznoe/129.gif">http://gifanimation.ru/images/raznoe/129.gif</a>	22
<a href="http://gifanimation.ru/images/lica/77.gif">http://gifanimation.ru/images/lica/77.gif</a>	23
<a href="http://gifanimation.ru/images/lica/174.gif">http://gifanimation.ru/images/lica/174.gif</a>	24
<a href="http://i075.radikal.ru/1003/ee/a7c022c1f226.jpg">http://i075.radikal.ru/1003/ee/a7c022c1f226.jpg</a>	25
<a href="http://s46.radikal.ru/i113/1003/4d/7d9f9b4e092b.jpg">http://s46.radikal.ru/i113/1003/4d/7d9f9b4e092b.jpg</a>	26
<a href="http://i3.wall-papers.in.ua/uploads/gallery_photo/photo/0080/56_thumb_80.jpg">http://i3.wall-papers.in.ua/uploads/gallery_photo/photo/0080/56_thumb_80.jpg</a>	27
<a href="http://i1.wall-papers.in.ua/uploads/gallery_photo/photo/0079/92_thumb_80.jpg">http://i1.wall-papers.in.ua/uploads/gallery_photo/photo/0079/92_thumb_80.jpg</a>	28
<a href="http://i2.wall-papers.in.ua/uploads/gallery_photo/photo/0096/00_thumb_80.jpg">http://i2.wall-papers.in.ua/uploads/gallery_photo/photo/0096/00_thumb_80.jpg</a>	29
<a href="http://i1.wall-papers.in.ua/uploads/gallery_photo/photo/0098/73_thumb_80.jpg">http://i1.wall-papers.in.ua/uploads/gallery_photo/photo/0098/73_thumb_80.jpg</a>	30

Рис. 3.2. URL-адреси рисунків.

Формат **.gif** переважно використовується для збереження рисованих зображень, векторної графіки, картинок без півтонів, градієнтів (переходів від кольору до кольору) і великої кількості дрібних деталей різних кольорів. Формат **.gif** застосовується для фотозображень і півтонової графіки, живопису, градієнтів і зображень із множиною дрібних різнокольорових деталей.

Це пояснюється тим, що формат **.gif** дозволяє зберігати зображення, які містять не більше **256** кольорів. Кольори можуть бути будь-якими і у будь-якій комбінації, але загальна їх кількість обмежена цим числом. Тому у цьому форматі зручно зберігати зображення із невеликою кількістю кольорів. Це дозволяє за рахунок скорочення об'єму інформації про кольори значно зменшити розмір файлу і час завантаження. Крім того, формат **.gif** використовується для збереження анімованих картинок і зображень із прозорими частинами; у **.jpg** такі можливості відсутні.

Сфера застосування формату **.jpg** визначається тим, що він зберігає повну колірну палітру 24-бітового зображення, яка може містити мільйони кольорів. При такому об'ємі інформації розмір файлу значно зростає. Але головною особливістю формату **.jpg** є його здатність зберігати зображення у стислому (ніби в архівному) виді. Міра архівації зображення, збереженого у форматі **.jpg**, може бути дуже значною: нормальним вважається стиснення картинки у 10 - 20 разів без втрати якості.

**3.4.2.** Ілюстрації відіграють найважливішу роль в оформленні веб-сторінок. Самі рисунки зберігаються в окремих файлах поза документом HTML, але відображаються вони браузером усередині веб-сторінки. Для розташування рисунків на веб-сторінці у html-документі застосовується поодинокий тег **<img>**. Цей тег завжди має містити обов'язковий атрибут **src=**, значенням для якого є URL-адреса файлу зображення, записана в абсолютній або відносній формах.

При завантаженні документа разом із ним завантажується рисунок і відображається у тому місці документа, де розташований тег **<img>**. Зображення переноситься на веб-сторінку із збереженням розміру. Якщо при компонованні зображення необхідно змінити його масштаб, потрібні розміри рисунка можна задати у пікселях за допомогою атрибутів **width=** (ширина) і **height=** (висота).

**3.4.3.** Використання ілюстрацій на веб-сторінках пов'язане із деякими складнощами. По-перше, веб-сторінка може відобразитись браузером, який не має засобів для показу зображень. По-друге, користувачі часто вимикають відображення рисунків для прискорення прийому веб-сторінки.

В обох випадках бажано знати, що зображено на невидимій картинці. Для цієї мети використовують альтернативний текст, який надає більш-менш детальне описування зображення. Альтернативний текст визначається тегом **<img>** і значенням спеціального атрибуту **alt=**.

**3.4.4.** На рис. 3.2 вибрати URL-адресу, порядковий номер якої відповідає номеру користувача у списку групи, переписати URL-адресу до робочого зошита.

Відкрити вікно браузера **Opera**. Натисканням на панелі інструментів **Стандартная** кнопки **Открыть** (див. Завдання 3.3) викликати на екран папку користувача, а у ній - текстовий файл html-документа. Вилу-

чити попередню URL-адресу зображення (рядок 18) і ввести нову URL-адресу за варіантом користувача. Зберегти нову редакцію файла із назвою, наприклад, **Іванов\_Веб\_2** і завершити роботу браузера.

У вікні **Total Commander** подвійним натисканням ЛК миші перевірити результат розпізнавання веб-сторінки із зміненою адресою рисунка. У разі невідповідності розпізнавання повторити редагування.

### Завдання 3.5. Дослідити способи використання кольорової палітри.

Кольори тексту, посилань, заголовків html-документа можна зазначити усередині html-тегів: наприклад,

```
<font color="колір">                                <body text="колір">
```

Будь-який колір на екрані монітора визначається змішенням трьох кольорів - червоного (red), зеленого (green) і синього (blue). Перші букви цих слів визначають колірну модель RGB. Зазначити колір можна його назвою; наприклад `<font color="red">Текст буде червоним</font>`.

Іншим способом зазначення кольору є застосування цифрового коду, в якому використовуються шість символів шістнадцятиричної системи числення; наприклад, `<font color= # FF0000>` - за цим позначенням текст буде червоного кольору.

Перед кодом ставиться символ # (дієз); перші два символи (FF) зазначають складову червоного кольору, друга пара (00) - зеленого кольору і третя пара (00) - синього кольору (рис. 3.3).



	<code>=#000000</code>	black (чорний)		<code>=#008000</code>	green (зелений)
	<code>=#C0C0C0</code>	silver (срібло)		<code>=#00FF00</code>	lime (св.-зелений)
	<code>=#808080</code>	gray (сірий)		<code>=#808000</code>	olive (оливковий)
	<code>=#FFFFFF</code>	white (білий)		<code>=#FFFF00</code>	yellow (жовтий)
	<code>=#800000</code>	maroon (малиновий)		<code>=#000080</code>	navy (темно-синій)
	<code>=#FF0000</code>	red (червоний)		<code>=#0000FF</code>	blue (синій)
	<code>=#800080</code>	purple (фіолетовий)		<code>=#008080</code>	teal (сизий)
	<code>=#FF00FF</code>	fuchsia (ліловий)		<code>=#00FFFF</code>	aqua (бірюзовий)

Рис. 3.3. Назви кольорів і значення **RGB**.

Кольори можуть істотно додавати інформації до документа, підвищуючи зручність читання і сприйняття; проте при використанні кольорів необхідно урахувувати наступні особливості:

1) використання елементів і атрибутів HTML для зазначення кольору є небажаним; замість цього слід використовувати таблиці стилів;

2) не слід використовувати комбінації кольорів, які викликають ускладнення у відображенні;

3) при використанні зображення як фону або при зазначенні кольорів фону не слід забувати і про визначення кольору тексту;

4) кольори, зазначені у тегах **body** і **font** та у **bgcolor**, виглядають по-різному у таблицях, у різних комп'ютерах і системах.

### Завдання 3.6. Дослідити способи управління стилем шрифту.

**3.6.1.** Для створення розміру, кольору і відображення шрифту застосовується парний тег **<font>**, який впливає на увесь текст, розташований усередині нього. Тег **<font>** повинен мати хоч б один із трьох можливих атрибутів : **size=**; **color=**; **face=**.

Атрибут **size=** визначає розмір шрифту. Передбачається, що можливі сім задалегідь зазначених розмірів шрифту. Ці значення не відповідають якимсь одиницям виміру, але чим більше значення, тим більше шрифт. За замовчуванням використовується значення **3**.

**3.6.2.** Атрибут **color=** визначає колір шрифту, який може бути зазначений або ключовим словом (наприклад, **red** - червоний), або у системі RGB шістнадцятиричним значенням (наприклад, **#FF0000** - червоний).

**3.6.3.** Атрибут **face=** визначає вид шрифту. Значенням цього атрибуту має бути назва одного із шрифтів, установлених на комп'ютері. Але для документа, розміщеного в Інтернеті, не можна передбачити які шрифти доступні на комп'ютері користувача, тому цей атрибут краще не використовувати.

**3.6.4.** Для зазначення наведених параметрів для усього документа у цілому використовують поодинокий тег **<basefont>**. Він містить аналогічні атрибути і зазначає вид, колір і розмір шрифту, використовуваних за замовчуванням.

**3.6.5.** Певна група тегів використовується для зображення шрифту:

**<b>** та **</b>** роблять текст між ними напівжирним;

**<i>** та **</i>** визначають курсивне зображення;

**<u>** та **</u>** визначають підкреслення;

**<s>** та **</s>** визначають викреслювання тексту.

**3.6.6.** Вивчити зазначені нижче приклади тегів.

```
<html>
  <head>
    <title> Управління стилем шрифту </title>
  </head>
<body>
<basefont size=4 face="Arial">
```

Цей текст використовує нестандартний стиль шрифту заданий за

замовчуванням.

```
<p><font size=-2 face="Times New Roman" color="green">
```

Цей текст дрібніше і використовує інший шрифт і інший колір.

```
</font>
```

```
<p><b> Напівжирний шрифт </b> і <i> курсив </i> використовують  
для виділення фрагментів тексту.
```

```
<p> Використання <u> підкреслення </u> не рекомендується  
оскільки підкреслений текст легко переплутати із посиланням.
```

```
<p><s> Викреслювання тексту </s> іноді застосовують для  
розмітки фрагментів, які втратили актуальність, але по якихось  
причинам зберігаються у документі.
```

```
</body>
```

```
</html>
```

**Завдання 3.7.** Засвоїти принципи якісного подання html-документів.

**3.7.1.** Творчий характер створення html-документів подібний до програмування; при цьому можливі помилки, які можуть привести до труднощів або навіть повної неможливості розпізнати за допомогою браузера створений html-документ. Веб-сторінки призначаються для широкого кола користувачів і тому при підготовці html-документів слід урахувати ці особливості.

**3.7.2.** Зміст документа значно важливіший за зовнішній вигляд. Зміст веб-сторінки або групи веб-сторінок має бути пов'язаним логічно. Добре продумана система посилань повинна дозволити переходити від сторінки до сторінки і повертатись назад практично без використання кнопок навігації **Вперед** і **Назад** на панелі інструментів браузера.

**3.7.3.** Для того, щоб html-документ був доступний найширшій аудиторії, слід відставати на один крок від останніх досягнень у розвитку мови HTML. Останні нововведення не завжди відразу реалізуються у браузерах, а новим версіям потрібний час для того, щоб досягти більшості користувачів. Документи із використанням самих останніх досягнень доступні лише обмеженій аудиторії.

**3.7.4.** Помилка, властива користувачам-початківцям, полягає у зловживанні елементами оформлення, особливо різноманітними кольорами і шрифтами. Така сторінка у кращому разі виглядатиме кричущо. У гіршому разі, текст взагалі не можна буде прочитати на комп'ютерах із нестандартною колірною гаммою або за відсутності необхідних шрифтів.

**3.7.5.** Не слід зловживати графічними зображеннями і мультимедійними файлами. За наявності таких об'єктів збільшується час читання веб-сторінки і користувач втрачає до неї інтерес ще до того, як дістане можливість щось прочитати.

**3.7.6.** Через те, що створення html-документів подібне до програмування, процес пошуку і виправлення помилок потрібний і при створенні



веб-сторінки. Більшість помилок можна знайти ще до того, як сторінка стане доступна стороннім читачам. До помилок, які треба виправляти, належать також граматичні і друкарські помилки.

**3.7.7.** Інтернет містить мільйони веб-сторінок, здатних запропонувати ідеї оформлення, якими можна скористатись. Гідність цього підходу у тому, що у такий спосіб можна побачити що і як зроблене.

**3.7.8.** Розміщення найважливішої інформації у верхній частині сторінки дозволяє із користю витратити час, упродовж якого відбувається завантаження документа. Супровід ілюстрацій альтернативним текстом не вимагає значної праці, але високо цінується. Наявність великої кількості гіперпосилань на сторінці не лише допомагає користувачам плідно подорожувати по просторах Інтернету, але й спонукає їх повертатись на сторінку, яка надала їм таку можливість.

### **3.8. Контрольні запитання**

1. Що являють собою html-документи? В якому середовищі вони створюються?
2. Які елементи складають синтаксис тексту html-документа?
3. У чому полягає особливість структури html-документа?
4. Яким чином вносяться зміни до html-документа?
5. Якого типу графічні елементи використовуються в html-документі і у чому полягає різниця та єдність між цими типами?
6. Як використовується кольорова палітра в html-документі?
7. Якими способами виконується управління шрифтами?

## **Тема 4. ДОСЛІДЖЕННЯ МЕТОДІВ ПРОГНОЗУВАННЯ ТРАНСПОРТНИХ ПЕРЕВЕЗЕНЬ**

**Завдання 4.1.** Ознайомитись із призначенням статистичних методів досліджень.

**4.1.1.** Статистичне дослідження закономірностей розвитку вантажних і пасажирських перевезень, їх сезонних нерівномірностей і конкурентної здатності складає базу для побудови прогнозів, необхідних на стадії розробки планів автотранспортного підприємства (АТП).

Сезонними коливаннями є порівняно стійкі коливання перевезень протягом року. Вони обумовлені специфічними явищами і багатьма чинниками, у тому числі - кліматичними. Сезонні коливання перевезень негативно впливають на економічні показники АТП, сприяють недостатньому (або надмірному) використанню рухомого складу і нерівномірному використанню трудових ресурсів.

Дослідження цих об'ємів перевезень за допомогою застосування статистичних методів згладжування, побудови графіків і наступного порівняння результатів дає можливість виявити тенденцію максимальних і мінімальних об'ємів перевезень у цілому. При моделюванні робочих процесів перевезень автомобільним транспортом використовуються зростаючі можливості сучасних комп'ютерних технологій, а також ефективні математичні методи прогнозування - ковзного середнього, експоненціального згладжування, побудови регресій.

**4.1.2.** Дослідження і прогнозування зазначених процесів можна здійснити засобами середовища Excel із застосуванням математичного методу ковзного середнього, що використовується для згладжування і прогнозування часових рядів.

Нагадаємо, що часовий ряд є деякою кількістю пар даних  $(X, Y)$ , в яких  $X$  відображає моменти або періоди часу (незалежна змінна), а  $Y$  є параметром, що характеризує величину досліджуваного процесу (залежна змінна). Метод ковзного середнього дозволяє виявити тенденції зміни фактичних величин параметра  $Y$  у часі і прогнозувати майбутні величини  $Y$ .

Створену модель можна ефективно використовувати у випадках, коли для величин прогнозованого параметра спостерігається постійна тенденція у динаміці. Але цей метод неефективний у випадках, коли така тенденція порушується; наприклад, при стихійній лихах, військових діях, громадському безладі, при різкій зміні параметрів внутрішньої або зовнішньої ситуації (рівня інфляції, цін на сировину); при корінній зміні плану діяльності підприємства, що терпить збитки.

Основна ідея методу ковзного середнього полягає у заміні **фактичних рівнів** досліджуваного часового ряду їх **середніми величинами**, які урівноважують випадкові коливання. Таким чином, у результаті обчислюється згладжений ряд величин досліджуваного параметра, що дозволяє чіткіше виділити основну тенденцію його зміни.

Метод ковзного середнього є відносно простим методом згладжування і прогнозування часових рядів, заснованим на поданні прогнозу  $y_{t-1}^*$  у вигляді середньої величини  $m$  попередніх спостережуваних даних  $y_{t-1}$ , де  $i$  змінюється від  $1$  до  $m$ ; тобто

$$y_t^* = \frac{1}{m} \sum_{i=1}^m y_{t-i}$$

Якщо, наприклад, при дослідженні часового ряду даних про прибуток від реалізованих об'ємів перевезень АТП за місяцями у якості прогнозу вибрати ковзне середнє за три місяці ( $m=3$ ), то прогнозом на червень буде середня величина показників за три попередні місяці (березень, квітень, травень).

Якщо ж вибрати ковзне середнє за чотири місяці ( $m=4$ ), то прогнозом на червень буде середня величина показників за чотири попередні місяці (лютий, березень, квітень, травень).

Часто, наприклад, при розрахунку прогнозу прибутку від об'єму перевезень АТП метод ковзного середнього, заснованого на спостереженнях за три або чотири попередні місяці, буває ефективнішим (дозволяє відстежувати фактичний об'єм перевезень із більшою точністю), ніж методи, засновані на довгострокових спостереженнях (за 12 місяців і більше).

Це пояснюється тим, що у результаті використання 3-місячного ковзного середнього кожна із трьох величин показника (за ці три місяці) відповідає за одну третину величини прогнозу.

При тому, що дванадцяти-місячній ковзній середній величині кожний із показників цих же останніх трьох місяців відповідає лише за одну дванадцятину частину прогнозу. На жаль, немає правила, яке дозволяло б підібрати оптимальне число  $m$  членів ковзного середнього.

Проте можна зазначити, що чим менше  $m$ , тим сильніше прогноз реагує на коливання часового ряду; і навпаки, чим більше  $m$ , тим процес прогнозування стає більш інертним.

**Завдання 4.2.** Дослідити процес транспортних перевезень методом ковзного середнього.

**4.2.1.** Викликати на екран табличний процесор MS Excel. Створити таблицю даних подібну таблиці рис. 4.1, в якій у комірки A1 - A4 ввести заголовки стовпців. У комірки A2 - A13 ввести назви місяців року.

За даними рис. 4.1 обчислити і записати у зошит відхилення (додатні або від'ємні) перевезень  $Q_1$  між сусідніми місяцями.

У комірку B2 таблиці користувача для створення власного варіанта кількості перевезеного вантажу  $Q_1$  (тис. тонн) ввести кількість тонн вантажу, що відповідає двом останнім цифрам залікової книжки.

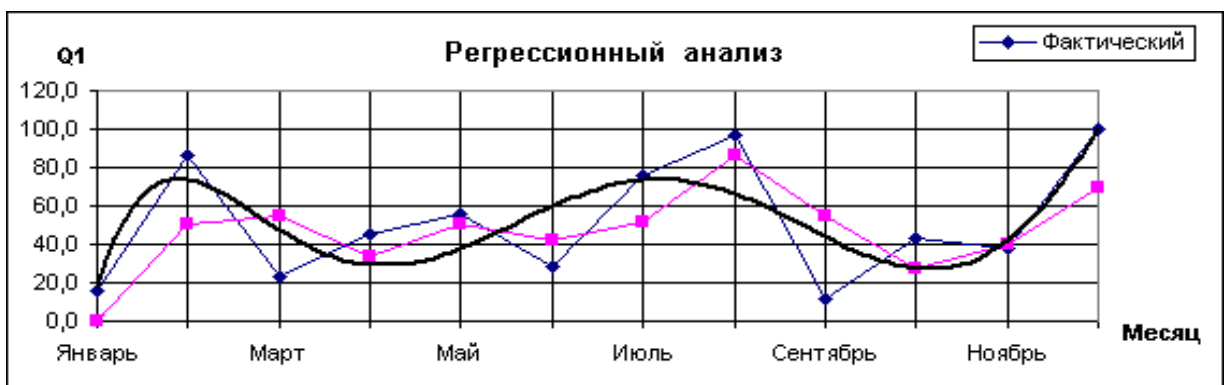
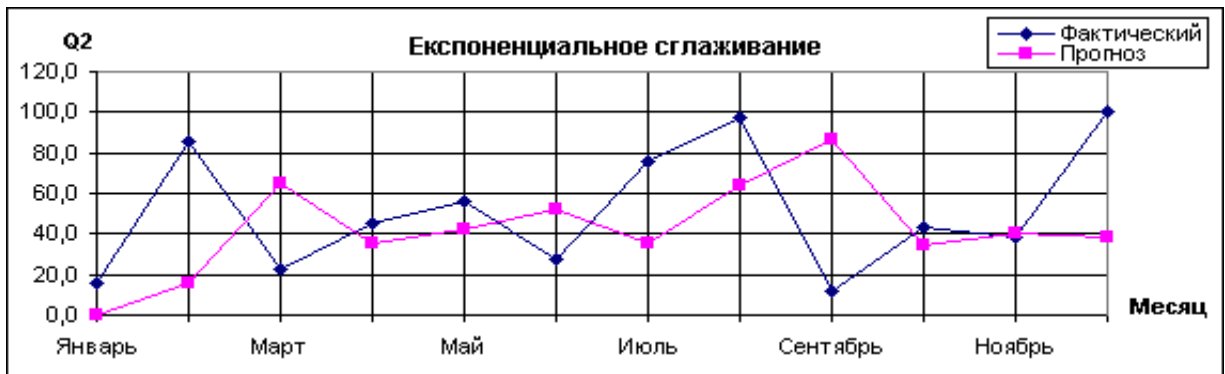
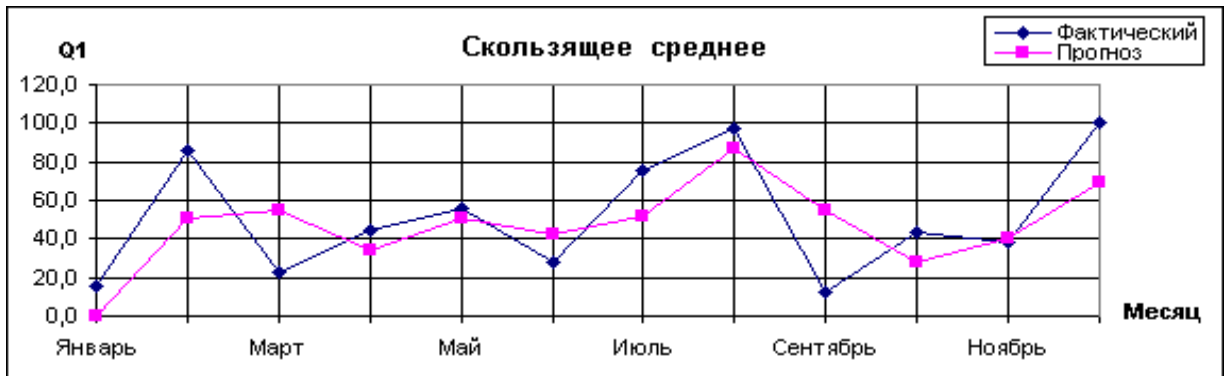
У наступні комірки (B3 - B13) ввести кількості тонн вантажу, які дорівнюватимуть кількості попереднього місяця із додаванням або відніманням обчислених відхилень. Комірки стовпців C та D (у заголовку таблиці ці стовпці названі Q2 та Q3) будуть заповнюватись програмою у процесі реалізації відповідного метода.

**4.2.2.** Звернути увагу на той факт, що графічне відображення результатів обчислень виконується за стандартною методикою побудови діаграм у середовищі MS Excel за допомогою інструмента **Мастер диаграмм**, розташованого на панелі інструментів **Стандартная**. Усі необхідні відомості про роботу зазначеного інструмента можна одержати у результаті виконання складної команди: **Справка→F1→Оглавление→Работа с данными→Работа с диаграммами и рисунками→Работа с диаграммами→Создание диаграмм→Диаграммы**.

**4.2.3.** Для візуального подання результатів досліджень слід позначити комірки A2 - B13, натиснути ЛК миші на панелі **Стандартная** на інструменті **Мастер диаграмм** для побудови графічного відображення фактичного розподілу  $Q_1$  за місяцями (рис. 4.1).

Місяці	Q1	Q2	Q3
Січень	16,0	##Н/Д	##Н/Д
Лютий	86,0	51,0	16,0
Березень	23,0	54,5	65,0
Квітень	45,0	34,0	35,6
Травень	56,0	50,5	42,2
Червень	28,0	42,0	51,9
Липень	76,0	52,0	35,2
Серпень	97,0	86,5	63,7
Вересень	12,0	54,5	87,0
Жовтень	43,0	27,5	34,5
Листопад	38,0	40,5	40,5
Грудень	100,0	69,0	38,7

Рис. 4.1.  
 Приклад табличного і графічного подання результатів дослідження перевезень вантажів автомобільним транспортом (у таблиці зазначити шрифт величиною 12 пунктів; у діаграмах зазначити шрифт величиною 8 пунктів).



Вибрати тип діаграми **График** і виконати усі етапи її побудови.

**4.2.4.** Виконати команду **Сервис→Анализ данных** - відкривається діалогове вікно **Анализ данных**, в якому наведені інструменти для аналізу, прогнозування, зміни та опрацювання даних середовищу MS Excel.

У діалоговому вікні **Анализ данных** в області **Инструменты анализа** позначити **Скользящее среднее** і натиснути **ОК** - відкривається вікно **Скользящее среднее** для введення додаткових параметрів.

Натиснути кнопку **Справка** і уважно прочитати довідку про даний метод; після ознайомлення закрити довідку.

**4.2.5.** У полі **Входной интервал** ввести адресу **\$B\$2:\$B\$13** - область розташування даних, що аналізуються.

У полі **Интервал** ввести цифру **2**, тобто зазначити кількість середніх величин, які розраховуються на конкретному інтервалі. Чим більше даних, тим точніше буде згладжена крива. Якщо кількість даних занадто велика, то виникає небезпека повністю згладити певні зміни і аналізувати їх у спотвореному виді.

**4.2.6.** У полі **Выходной интервал** ввести адресу комірки розташування результатів - **\$C\$2**. Позначити опцію (параметр) **Вывод графика** і натиснути **ОК** - програма виводить рисунок діаграми, а у третьому стовпці таблиці - числові величини кривих діаграми.

За допомогою **Мастера диаграмм** виконати форматування для відповідного графічного відображення результатів роботи.

Користувачам, при виконанні побудови таблиці та діаграм за власним варіантом, необхідно досягти повної подібності зображенням, наведеним на рис. 4.1.

При цьому розташувати діаграми під таблицею; розтягнути їх на ширину сторінки; висоту вибрати так, щоб усі три діаграми розташувались на одній сторінці.

Зберегти результати роботи у папці користувача у файлі із назвою, яка складається із прізвища користувача, назви завдання і номера дослідження, наприклад, **Іванов\_Аналіз\_1**.

**Завдання 4.3.** Дослідити процес транспортних перевезень методом експоненціального згладжування.

**4.3.1.** У методі експоненціального згладжування коливань початкових даних для кожного значення використовується попереднє середнє із урахуванням зазначеного вагового коефіцієнта. Цей метод застосовується до даних, які характеризуються коливаннями великої амплітуди.

**4.3.2.** Виконати таку команду:

**Сервис→Анализ данных→Экспоненциальное сглаживание**

Натиснути кнопку **Справка** і уважно прочитати довідку про метод, що застосовується. Закрити довідку і у вікні **Экспоненциальное сглаживание** натиснути **ОК**.

**4.3.3.** Ввести **\$B\$2:\$B\$13** у поле **Входной интервал**; у полі **Выходной интервал** ввести **\$D\$2** і позначити опцію **Вывод графика**. За допомогою **Мастера диаграмм** виконати усі необхідні дії для відповідного графічного відображення результатів роботи.

**4.3.4.** У вікні **Фактор затухания** ввести одне значення із інтервалу від **0** до **1**. Чим вище це значення, тим більше програма згладжує окремі коливання. Ввести, наприклад, величину **0,3** і закрити діалогове вікно натисканням **ОК**.

**4.3.5.** Вилучити зазначеним вище способом криву **Фактический** і залишити криву **Прогноз**. Порівняти криву **Прогноз** із двома попередніми методами; дати їм якісні характеристики.

**4.3.6.** Зберегти результати роботи із назвою, наприклад, **Іванов\_Аналіз\_2** (за розташуванням, для зручності, вважатимемо цю діаграму другою).

**Завдання 4.4.** Дослідити процес транспортних перевезень методом регресійного аналізу.

**4.4.1.** При реалізації регресійного аналізу програма не будує криву діаграми від одної точки до наступної. На основі даних і вибраного типу регресії програма розраховує ідеальну криву лінії тренду і рисує її між точками діаграми.

**4.4.2.** Скопіювати діаграму ковзного середнього у буфер обміну. Позначити відповідну комірку під діаграмою експоненціального згладжування і натиснути кнопку **Вставить** на панелі інструментів **Стандартная** (за розташуванням, для зручності, вважатимемо цю діаграму третьою).

Вибрати будь-яку точку на третій діаграмі, натиснути ПК миші і виконати команду **Добавить линию тренда**.

**4.4.3.** У відповідному діалоговому вікні для визначення нової кривої слід, за бажанням користувача, вибрати один із шести типів регресії.

Натиснути ЛК миші на сторінці **Тип**; серед типів регресії позначити **Полиномиальная**; в опції **Степень** зазначити **6** і натиснути **ОК**.

**4.4.4.** Математичні формули, за якими будується крива відповідного типу регресії, можна викликати на екран за допомогою наступних дій.

1) Натиснути ЛК миші на **знаку питання** (кнопка праворуч на панелі **Стандартная**) - на екран виводиться вікно **Справка Excel**.

2) у полі **Искать** надрукувати ключові слова визначення типу лінії тренду (наприклад, поліноміальну лінію) і натиснути зелену кнопку із стрілкою **Начать поиск**.

3) У результаті попередньої дії на екран виводиться чергове вікно довідки. У полі **Результаты (16)** натиснути ЛК миші на гіперпосиланні **Формулы для вычислений линий тренда** - у черговому вікні виводиться перелік зазначених ліній.

4. Натискання ЛК миші на назві потрібної лінії викликає на екран

відображення її математичної формули із поясненнями.

5) Записати у конспект усі типи ліній тренду, їх математичні вирази і пояснення до них.

**4.4.5.** Для подання результатів досліджень у належному науковому вигляді необхідно ретельно виконати форматування таблиці і діаграм.

Зберегти результати роботи із зазначеною вище назвою, але із черговим номером дослідження; наприклад, **Іванов\_Аналіз\_3**.

#### **4.5. Контрольні запитання**

1. Які можливості та переваги статистичних методів досліджень?
2. Яка мета моделювання процесів перевезень вантажів?
3. У чому укладається метод ковзного середнього?
4. У чому укладається метод експоненціального згладжування?
5. У чому укладається метод регресійного аналізу?
6. Які вимоги до графічного відображення результатів моделювання фактичних процесів та прогнозів?

### **Тема 5. АНАЛІТИЧНЕ І СТРУКТУРНЕ ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ БАЗИ ДАНИХ АТП**

**Завдання 5.1.** Ознайомитись із основними поняттями баз даних.

**5.1.1.** Особливостями сучасних інформаційних систем є великі об'єми даних та їх складна організація, а умови конкуренції змушують надавати до швидкості і якості опрацювання інформації найвищі вимоги. Великі об'єми інформації найефективніше опрацьовуються системами управління базами даних, однією із яких є MS Access - складова програма пакету Microsoft Office. Основними термінами і поняттями СУБД є наступні.

**База даних** є структурованою сукупністю взаємопов'язаних даних, що організована за певними правилами, відображає стан конкретної предметної області і підкоряється централізованому програмному управлінню.

**Система управління базами даних** є комплексом програмних і мовних засобів, необхідних для створення та опрацювання баз даних, централізованого управління ними і доступу до даних.

**Моделі даних** використовуються для ефективної організації даних і адекватного відображення предметної області; на етапі проектування баз даних застосовують інформаційні моделі даних.

**Моделювання** є способом дослідження систем, який полягає у ви-

користанні моделі досліджуваного об'єкта, що зберігає основні властивості реального об'єкта, але є зручнішим для досліджень. Так, наприклад, при моделюванні ділянки авторемонтного підприємства у процесі рішення задачі обліку основних засобів необхідно враховувати склад і вартість устаткування; при рішенні задачі розміщення устаткування - враховувати його склад, розміри, норми відстаней.

**Клас об'єктів** складає сукупність об'єктів із своїми властивостями. Наприклад, у базі даних "Постачання автозапчастинами" класами об'єктів будуть "Постачальники" і "Автозапчастини". Для кожного окремого об'єкта із цього класу об'єктів у таблиці створюється окремий запис. У кожній таблиці мають зберігатись дані тільки про один клас об'єктів, тобто в одній таблиці не можна зберігати дані про "Постачальників" і про "Автозапчастини".

**5.1.2.** При використанні у виробничих процесах методів комп'ютерних технологій і засобів інформаційних систем основним об'єктом опрацювання є **інформація**. Кожна галузь, яка має відношення до інформації, має своє суб'єктивне поняття про інформацію: для журналіста це одне, для ученого - інше, для водія - третє. Така ситуація не задовольняє методи комп'ютерних технологій, тому що для роботи із об'єктом необхідно мати чітке уявлення і відомості про його властивості.

**Джерелом** інформації є навколишній матеріальний світ, що складається із різноманітних фізичних об'єктів, які знаходяться у постійному русі та взаємодії. При цьому відбувається обмін інформацією, що супроводжується тими або іншими сигналами, які реєструються іншими об'єктами. Наприклад, під час руху автомобіля нагріваються гальмівні колодки. Відповідні прилади реєструють цей факт і повідомляють водія; такі повідомлення прийнято називати **даними**.

Дані містять у собі **інформацію** про події, проте обидва поняття не є тотожними. Щоб мати інформацію, необхідно (але недостатньо) отримати дані; крім того, ці дані треба уміти інтерпретувати. За цій точки зору інформацію можна характеризувати як **продукт взаємодії даних і адекватних ним методів**. Інформація носить **динамічний** характер, вона з'являється, існує і перетворюється тільки у процесі взаємодії даних і методів, тобто є **інформаційним процесом**. Поза цим процесом існують тільки дані, які є інформацією лише потенційно.

**5.1.3.** До найважливіших властивостей інформації належать повнота, достовірність, адекватність, доступність, актуальність.

**Повнота** інформації характеризує її якість і визначає достатність даних для їх правильної інтерпретації.

**Достовірність** є властивістю, пов'язаною із якістю реєстрації даних. При передачі сигналів разом із "корисними" можуть реєструватися "зайві" сигнали, так звані "інформаційні шуми". Якщо рівень шуму високий, він може спотворювати основні сигнали.

**Адекватність** визначає міру відповідності інформації її реальному



змісту. Неадекватна інформація може з'явитися у результаті свідомого спотворення даних, помилкової інтерпретації, тощо.

**Доступність** свідчить про міру можливості отримання інформації, маючи на увазі доступність даних і методів.

**Актуальність** свідчить про відповідність інформації теперішньому часу. Є багато сфер діяльності, де застаріла інформація не має ніякої цінності. Із актуальністю і повнотою пов'язують цінність інформації у комерції і управлінні, коли за отриманою інформацією приймаються важливі рішення.

Таким чином, із урахуванням зазначених властивостей, процес одержання інформації складається із таких операцій:

- 1) **збирання** даних із метою одержання їх достатньої повноти;
- 2) **формалізації** даних - перетворення даних, одержаних із різних джерел до єдиного, зручного для опрацювання виду;
- 3) **фільтрації** "зайвих" сигналів, тобто підвищення рівня достовірності;
- 4) **сортування** - упорядкування за певною ознакою із метою збільшення міри доступності;
- 5) **захисту** даних - комплексу заходів запобігання вилученню, копіюванню або модифікації даних;
- 6) **транспортування** - обміну даними між віддаленими учасниками інформаційного процесу;
- 7) **перетворення** із однієї форми в іншу; наприклад, із електронної у паперову і навпаки.

Для виконання усіх цих операцій необхідно застосовувати методи, які не вносять додаткових спотворень у дані і не порушують властивостей даних, а також використовувати носії інформації, що дозволяють зберігати великі об'єми даних в єдиному формалізованому вигляді.

Необхідність дотримання зазначених вимог зумовила появу **інформаційних систем (ІС)** та **інформаційних технологій (ІТ)**.

Поняття "**інформаційна система**" передбачає наявність великих масивів даних про об'єкти і явища реального світу, а також програмно-апаратних засобів для опрацювання цих даних.

Поняття "**інформаційна технологія**" передбачає поєднання процедур, що реалізують функції збирання, накопичення, зберігання, опрацювання і передачі даних із застосуванням технічних засобів.

**5.1.4.** Основою будь-якої інформаційної системи є база даних, яка у загальному випадку є організованою структурою із сукупністю відомостей про конкретні об'єкти реального світу у будь-якій предметній області.

**Метою** інформаційної системи є опрацювання даних про об'єкти, які містяться у базах даних, із урахуванням зв'язків між об'єктами.

У теорії інформаційних систем **об'єкти** називають **сутністю**, а **відомості** про них (дані) - **атрибутами**; обидва терміни є **фундаментальними поняттями**.

**Сутність** є чимось існуючим і розрізненим, яке має назву; а також існує спосіб відрізнити цей об'єкт від інших. **Сутністю** можуть виступати як матеріальні предмети, так і абстрактні поняття, які відображають реальний світ. Наприклад, філософські теорії, певні події (автомобільні перегони), правові норми, тощо. Група подібних **сутностей** складає **набір**; конкретний **об'єкт** називають **екземпляром**.

**Атрибут** характеризує **об'єкт** і набуває у конкретного екземпляра **об'єкта** певного значення. Списки можливих значень атрибутів називають або класифікаторами, або довідниками, або словниками.

**5.1.5.** Опрацювання баз даних виконується за допомогою спеціальних програм - **Систем управління базами даних (СУБД)**; саме такою є програма Access - із пакету програм Microsoft Office.

У СУБД дані зберігаються в упорядкованому вигляді. Залежно від міри упорядкованості даних ІС можна умовно поділити на **фактографічні** (утримують інформацію із чіткою структурою про зареєстровані факти) і **документальні** (клас завдань, який не допускає однозначної відповіді). Існує ще один тип ІС - **експертні**. Такі системи здатні дати оцінку ситуації і спрогнозувати її розвиток на основі закладених у систему фактичних даних і їх загальних закономірностей.

База даних у середовищі Access є засобом організації зберігання великої кількості упорядкованої різномірної інформації і управління нею. У загальному випадку базу даних характеризують жорстка внутрішня структура і взаємозв'язок між окремими елементами, що зберігаються.

Працюючи із базою даних, користувач абстрагується від конкретного чину їх фізичного зберігання на комп'ютері (локальному або віддаленому). Замість того, щоб мати справу із великою кількістю окремих файлів (наприклад, текстових, табличних або графічних), користувач працює із єдиним інтерфейсом, за допомогою якого можна додавати, видаляти або редагувати записи, створювати аналітичні звіти.

**5.1.6.** Програмний продукт Microsoft Access є **локальною** системою управління базами даних, але успішно працює у **комп'ютерних мережах**. Поняття локальної СУБД означає, що усі операції із базою даних здійснюються на локальному комп'ютері користувача. Саме тут знаходиться фізичне місце зберігання інформації, а також працюють засоби управління і організації запитів.

Ідеальним середовищем для упровадження Access є численні і різноманітні автотранспортні підприємства, малий і середній бізнес, пов'язаний із інтенсивним оборотом товарів і документів, облік складського господарства, облік кадрового складу підприємств, обчислення і контроль витрат палива, тощо.

**5.1.7.** Найважливішим складовим бази даних є **ядро**, яке забезпечує збереження введених даних і можливість оперативного реагування на запити, які надходять до цих даних. Маються на увазі приховані механізми, відповідальні за фізичні операції у середовищі Access. Ядра не сто-

сується хто і як ці запити генерує і куди далі посилаються результати вибору, тобто ці дії знаходяться поза компетенцією ядра бази даних.

Для виконання акцій поза ядром використовуються засоби управління, що відповідають за створення і модифікацію баз даних, додавання до них нових структур, типів даних, тощо.

Призначений для користувача інтерфейс допомагає складати запити (без проникнення у тонкості внутрішньої архітектури бази даних), а також належним чином створювати звіти.

Зазначені три рівні (ядро, засоби управління і інтерфейс) можуть існувати незалежно один від одного, але можуть утворювати єдину СУБД. Середовище Access побудоване так, що усі елементи бази (від сховищ даних до об'єктів інтерфейсу користувача) зберігаються усередині єдиного файлу (типу **.mdb**), що значно скорочує час операцій копіювання.

База даних складається із декількох об'єктів, сукупність яких складає **модель бази** даних. Об'єктами бази даних є таблиці, запити, звіти, форми, модулі, макроси. Деякі із зазначених об'єктів можуть бути відсутніми у конкретній базі даних, але одна або декілька таблиць присутні завжди і є основними структурними компонентами.

## **Завдання 5.2.** Ознайомитися із складовими предметної області.

**5.2.1. Предметна область.** На теперішній час на автомобільних дорогах України спостерігається підвищена інтенсивність руху, що є результатом диспропорції між зростанням автомобільного парку і мережею автомобільних доріг. Високий рівень аварійності на транспорті є наслідком диспропорції між рівнем підготовки і транспортною культурою учасників руху. Зростання інтенсивності і швидкісних режимів транспортних потоків, незадовільний стан дорожнього покриття, недостатня освітленість вулиць викликають зростання вимог до методів управління дорожнім рухом і до заходів забезпечення максимально можливої безпеки руху.

Управління рухом транспорту у критичних ситуаціях полягає в оперативному впливі на основні характеристики транспортного потоку і його розподілі по мережах паралельних маршрутів залежно від умов руху, що склалися. Однією із критичних ситуацій вважається виникнення дорожньо-транспортних подій (ДТП) - порушень процесу дорожнього руху, що викликали травмування людей, ушкодження транспортних засобів або дорожніх споруд, матеріальні збитки. Для розробки заходів організації дорожнього руху із метою зниження кількості ДТП необхідно мати повний об'єм інформації про ДТП, забезпечити її зберігання, систематизацію, опрацювання, аналіз і оперативну звітність.

**5.2.2. Інформаційний аналіз.** Основою розробки заходів попередження ДТП є точні відомості про них. За існуючими правилами облік

ДТП виконується Державною автоінспекцією (ДАІ); при цьому усі обставини події реєструються в облікових картках ДТП, складених на підставі первинних документів. Статистика ДТП ведеться місцевими органами ДАІ і автотранспортними підприємствами (АТП). У первинних документах реєструється характер, чинники і обставини ДТП, умови їх виникнення, стан транспортного засобу, кількість потерпілих, наслідки. У свою чергу, АТП мають зберігати повну інформацію про водіїв і періодично її оновлювати. Розвиток теоретичних основ і технічних засобів реєстрації інформації, її опрацювання і транспортування створили передумови для розробки комп'ютерних інформаційних систем контролю, аналізу і прогнозування безпеки руху.

Для відображення зв'язків між зазначеними чинниками, для довгострокового зберігання відомостей і для автоматизації їх опрацювання створюється **інформаційна** база даних СБР (система контролю безпеки руху), структурна схема якої відображена на рис. 5.1.

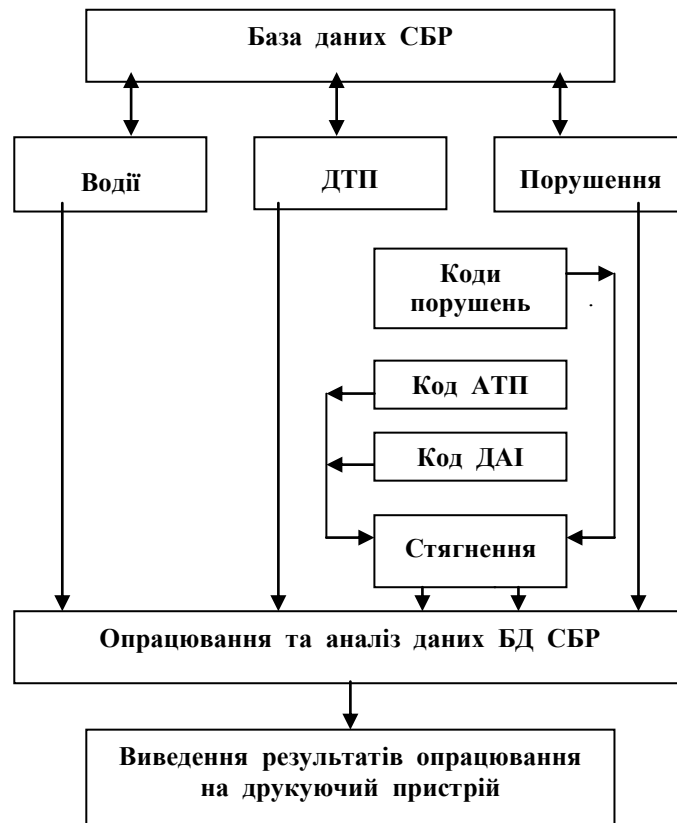


Рис. 5.1. Структурна схема бази даних СБР.

База даних СБР містить інформацію про водіїв (загальну і службову), про ДТП, види порушень і стягнень, їх коди, стан учасників, місце, дату і винних у ДТП, тощо. Система дозволяє переглядати інформацію, редагувати, додавати нові дані, вилучати надмірні або застарілі. Збирання інформації і введення даних у комп'ютер виконується у процесі виробничого функціонування незалежно від періодичності і черговості.

Для вирішення зазначеної інформаційної задачі можуть бути застосовані **реляційні системи управління базами даних**, в яких дані використовуються у вигляді **двовимірних таблиць**. **Рядок** у таблиці дорівнює одному **запису** (про конкретного водія або подію), а **стовпці - полям** записів (складовим параметрам).

Для створення бази даних інформаційної системи контролю безпеки руху застосована система управління базами даних MS Access, яка дозволяє працювати як в індивідуальному режимі, так і у складі локальної або глобальної мереж. При цьому вона є середовищем процедур, в яких передбачено введення і редагування користувачами первинної інформації за допомогою клавіатури, спілкування із персональним комп'ютером у діалоговому режимі, створення тих або інших запитів і отримання на їх основі допоміжної інформації.

**5.2.3. Логічна структура. Таблиці** (двовимірні) є основними об'єктами баз даних; структуру таблиць утворюють **поля** (стовпці) і **записи** (рядки). Якщо записи у таблиці відсутні, то її структура утворена тільки **переліком** полів. Зміна складу полів базової таблиці (або їх властивостей) викликає зміну структури бази даних.

Для роботи із даними, розподіленими по таблицях, необхідно за своїти такі поняття:

1) **ім'я поля** - спосіб звернення до даних цього поля при автоматичних операціях із базою (за замовчуванням імена полів використовуються як заголовки стовпців таблиць);

2) **тип поля** - тип даних; розмір поля визначає граничну довжину (у символах) даних, які можуть розміщуватись у полі;

3) **формат поля** - спосіб форматування даних у комірках поля;

4) **маска введення** - форма, в якій вводяться дані у полі (засіб автоматизації введення даних).

Модель бази даних СБД складають шість наступних таблиць:

1) **Інформація** - облік водіїв АТП із службовою інформацією;

2) **Учет ДТП** - інформація про ДТП;

3) **Нарушення** - зареєстровані порушення водіїв;

4) **Коди порушень** - коди 11 видів порушень;

5) **Код АТП** - коди і види стягнень керівництвом АТП;

6) **Код ГАИ** - коди і види стягнень працівниками ДАІ.

У кожній із наведених нижче табл. 5.1 - 5.6 в якості прикладів скорочено відображається тільки **один** запис. Рядки у цих прикладах мають наступне призначення:

1-й рядок - назви полів (стовпців) відповідних таблиць;

2-й рядок - приклад запису змісту в осередках полів;

3-й рядок - пояснення призначення кожного стовпця;

4-й рядок - суцільна нумерація полів усіх таблиць.

Таблица 1. "Информация"

<b>№</b>	<b>ФИО</b>	<b>№_таб</b>	<b>№_ак</b>	<b>Д_рожд</b>
<b>1</b>	<b>Иванов Г. П.</b>	<b>151</b>	<b>1</b>	<b>27.06.1927</b>
Табличный код водителя	Фамилия и инициалы водителя	Табельный номер	Номер автоколонны	Дата рождения
1	2	3	4	5

<b>Нац</b>	<b>Образ</b>	<b>Стаж</b>	<b>Кл</b>	<b>Д_раб</b>
<b>Украинец</b>	<b>Средняя</b>	<b>46</b>	<b>1</b>	<b>31.10.1955</b>
Национальность	Образование	Общий стаж работы водителем	Класс водителя	Дата зачисления на работу
6	7	8	9	10

<b>№_удост</b>	<b>Кем_выд</b>	<b>Ав_шк.</b>	<b>Д_мед</b>
<b>ААД 809990 ВСД</b>	<b>ГАИ Кировограда</b>	<b>г. Одесса</b>	<b>10.02.1992</b>
Номер свидетельства водителя	Кем выдано удостоверение	Место окончания автошколы (курсов)	Дата медицинского обследования
11	12	13	14

<b>Д_экз</b>	<b>№_экз</b>	<b>Оц_экз</b>	<b>Адрес</b>	<b>ФИО_жены</b>
<b>11.10.1992</b>	<b>23</b>	<b>4</b>	<b>Ялта, Чкалова, 36</b>	<b>Иванова О.М.</b>
Дата экзамена	Номер экзаменационного билета	Оценка на экзамене	Домашний адрес	Фамилия и инициалы жены
15	16	17	18	19

<b>Раб_жены</b>	<b>Д_взыск</b>	<b>Д_аннул</b>	<b>ИО</b>
<b>База культторга</b>	<b>07.07.1993</b>	<b>07.10.1993</b>	<b>Георгий Петрович</b>
Место работы жены	Дата наложения взыскания	Дата снятия взыскание	Имя и отчество водителя
20	21	22	23

Таблица 2. "Нарушения"

<b>№_таб</b>	<b>Д_нар</b>	<b>Вр_нар</b>	<b>Тип_тс</b>	<b>№_тс</b>
<b>669</b>	<b>15.11.1992</b>	<b>22.45</b>	<b>ГАЗ 53</b>	<b>54-76</b>
Табельный номер водителя	Дата нарушения	Время нарушения	Тип транс. средства	Номер транс. средства
24	25	26	27	28

<b>М_нар</b>	<b>Код_Д</b>	<b>Реестр</b>	<b>Код_А</b>	<b>Код_нар</b>
<b>г. Кировоград</b>	<b>3</b>	<b>ГАИ</b>	<b>3</b>	<b>00101</b>
Географ. название места нарушения	Код взыскания ГАИ	Организация регистрации нарушения	Код взыскания АТП	Код вида нарушения
29	30	31	32	33

Таблица 3. "Учет ДТП"

<b>№_таб</b>	<b>Зд_в</b>	<b>Д_дтп</b>	<b>Тип_тс</b>	<b>№_тс</b>
<b>669</b>	<b>Трезвый</b>	<b>09.01.1989</b>	<b>КАМАЗ</b>	<b>76-12м</b>
Табельный номер водителя	Состояние здоровья водителя	Дата ДТП	Тип транспортного средства	Номер транспортного средства
34	35	36	37	38

<b>М_дтп</b>	<b>Код_дтп</b>	<b>Время_дтп</b>	<b>ДТП</b>
<b>Ул. Чкалова</b>	<b>00100</b>	<b>13.15</b>	<b>Наезд на пешехода</b>
Название места ДТП	Код вида ДТП	Время ДТП	Описание ДТП
39	40	41	42

<b>Повр_А</b>	<b>Убыток</b>	<b>В_в</b>	<b>Код_Г</b>	<b>Код_А</b>
<b>2</b>	<b>0</b>	<b>Не виновен</b>	<b>0</b>	<b>3</b>
Количество поврежденных автомобилей	Материальные убытки	Виновность водителя	Код взыскания ГАИ	Код взыскания АТП
43	44	45	46	47

Таблиця 4. "Коды нарушений"

Код_нар	Вид_нар
00101	Управление транспортом в нетрезвом состоянии
Код нарушения	Вид нарушения
48	49

Таблиця 5. "Коды взысканий АТП"

Код	АТП
0	Взыскание отсутствует
Код взыскания	Взыскание на автотранспортном предприятии
50	51

Таблиця 6. "Коды взысканий ГАИ"

Код	ГАИ
0	Взыскание отсутствует
Код взыскания	Взыскание госавтоинспекции
52	53

**5.2.4.** У процесі виконання роботи на основі аналізу завдання, синтезу знань і умінь предметної області і інформаційних систем користувачі за індивідуальними завданнями проектують базу даних і реалізують засоби користування нею. Для цього користувачам необхідно виконати наступне:

- 1) кожному користувачу створити у середовищі MS Access персональну базу даних системи контролю безпеки руху (СБД);
- 2) побудувати схему зв'язків у базі даних СБД;
- 3) створити і дослідити форми для введення інформації у таблиці бази даних СБД;
- 4) створити запити на вибір і за умовами;
- 5) створити на основі запитів звіти.

**Завдання 5.3.** Засвоїти способи звертання до середовища MS Access.

**5.3.1.** Викликати на екран комп'ютера середовище Access за допомогою одного із двох способів:



1) на робочому столі комп'ютера натиснути ЛК миші на ярлику MS Access;

2) виконати наступну команду:

**Пуск→Програми→Microsoft Office→Microsoft Office Access**

У вікні середовища Access у першому (верхньою) рядку ліворуч розташована **назва** вікна, а праворуч - кнопки управління вікном (**Свернуть, Восстановить, Закреть**).

У **другому** рядку розташовані головні команди середовища Access, а у **третьому** - інструменти стандартних операцій (тобто операцій, власних усім програмним продуктам корпорації Microsoft); ці інструменти всі разом утворюють панель інструментів **Стандартная**.

Залежно від режиму роботи склад панелей і головних команд або інструментів, які містяться у них, може бути змінений користувачем або самостійно (за замовчуванням) комп'ютером.

Призначення кожної кнопки панелі інструментів **Стандартная** можна визначити за допомогою пояснення, яке виводиться на екран через 1 - 3 секунди після розташування покажчика миші на інструменті.

У нижній частині вікна знаходиться рядок **поточного стану**, в якому знаходиться середовище Access.

**5.3.2.** Праворуч на екрані виводиться панель області задач, за допомогою якої виконується звернення до відповідних режимів роботи. Звернути увагу на той факт, що деякі записи області задач підкреслені і мають синій колір. У такий спосіб відображаються гіперпосилання - команди, які сприяють переходам за прихованими адресами до потрібної користувачеві інформації.

Для звернення до довідкової системи виконати наступну команду:

**Справка→Справка: Microsoft Office Access→Оглавление**

Через декілька секунд в області задач виводиться вікно із змістом довідкової системи. Користувачам слід самостійно засвоїти методи управління пошуком необхідних відомостей за допомогою довідки. Після ознайомлення із довідковою системою повернутись до початкового стану області задач натисканням у верхньому рядку ліворуч кольорової кнопки **Домашняя страница**.

**5.3.3.** Для початку роботи у нижній частині **Панели задач** натиснути ЛК миші на гіперпосиланні **Создать файл** - на екран виводиться наступне вікно діалогу. В області **Создание** виконати команду **Новая база данных** - на екран виводиться вікно **Файл новой база данных** для зазначення місця розташування файлу майбутньої бази даних. У полі **Папка** натиснути кнопку списку (позначена стрілкою-трикутником) - виводиться перелік носіїв інформації.

Виконати команду **Локальный диск (C:)** - в інформаційних полях виводиться каталог диска **C:**. Відкрити папку із назвою групи, а потім папку користувача, **назва** якої буде відображена у полі **Папка**. У полі **Имя файла:** вилучити усі символи, надрукувати прізвище користувача,

нижню риску, букви СБД, нижню риску, цифру 1 (наприклад, **Іванов\_СБД\_1**) і натиснути кнопку **Создать** - на екран у середовищі Access виводиться первинне вікно майбутньої бази даних (рис. 5.2).

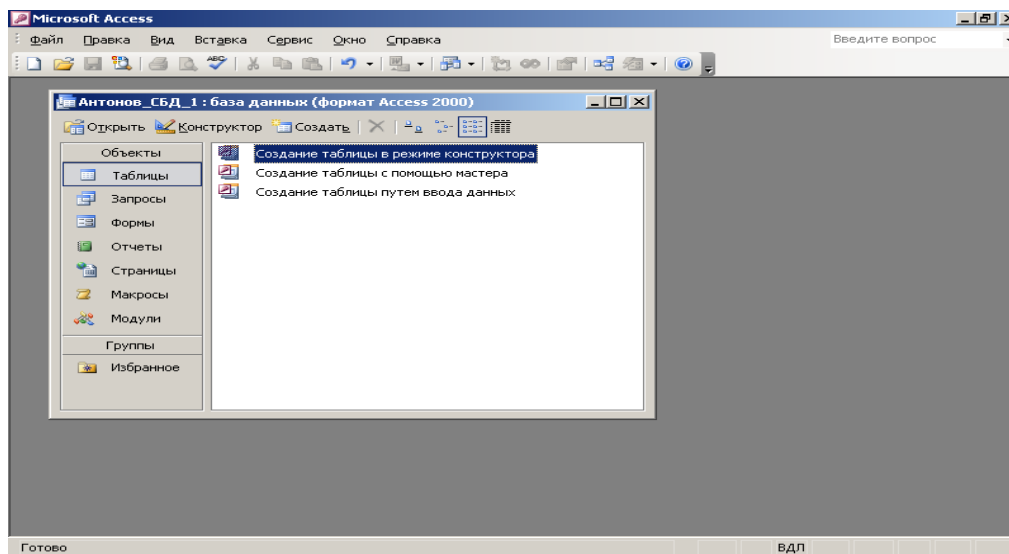


Рис. 5.2. Вікно бази даних MS Access.

**Увага!** Не рекомендується без необхідності пересувати вікна, які виводяться на екран, і приєднувати вікна до панелей інструментів. Іноді при цих надмірних діях втрачається уся напрацьована користувачем інформація. Авторами пакету програм MS Office визначені і надані оптимальні місця розташування вікон та їх розміри.

#### **Завдання 5.4.** Вивчити способи створення таблиць у базі даних.

**5.4.1.** Для створення таблиць із даними необхідно у первинному вікні бази (рис. 5.2) даних натиснути ЛК миші на об'єкті **Таблицы**. Праворуч, у полі списків виводиться перелік способів створення таблиць. Для початку створення таблиці слід натиснути ЛК миші на одному із трьох способів.

Спосіб **Создание таблицы с помощью мастера** передбачає вибір із запропонованих списків (рис. 5.3) вбудованих поширених таблиць для ділового і особистого застосування. Недоліком цього способу є обмеженість вибору.

Спосіб **Создание таблицы путем ввода данных** (рис. 5.4) вимагає безпосереднього введення даних у стандартну таблицю. Цей спосіб викликає деякі складнощі із зазначенням типів полів і їх назв.

Спосіб **Создание таблицы в режиме конструктора** передбачає створення таблиць шляхом зазначення (рис. 5.5) імен полів, типів даних і властивостей полів. Цей спосіб найбільш прийнятний - він сприяє точному відображенню інформації і не обмежує кількість і якість даних.

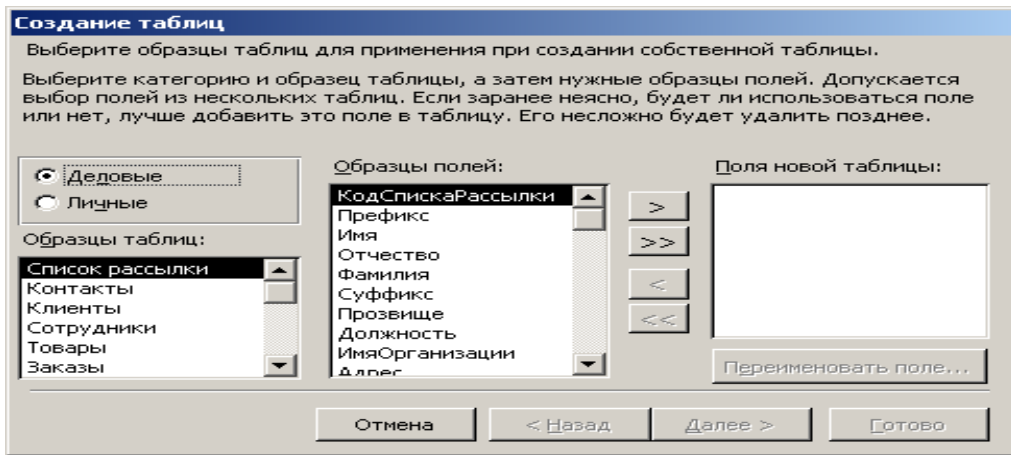


Рис. 5.3. Вікно створення таблиц із застосуванням **Майстра**.

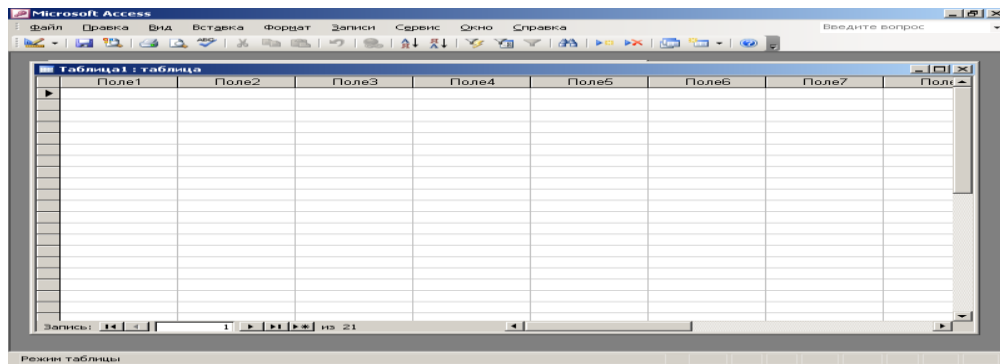


Рис. 5.4. Вікно введення даних у таблицю.

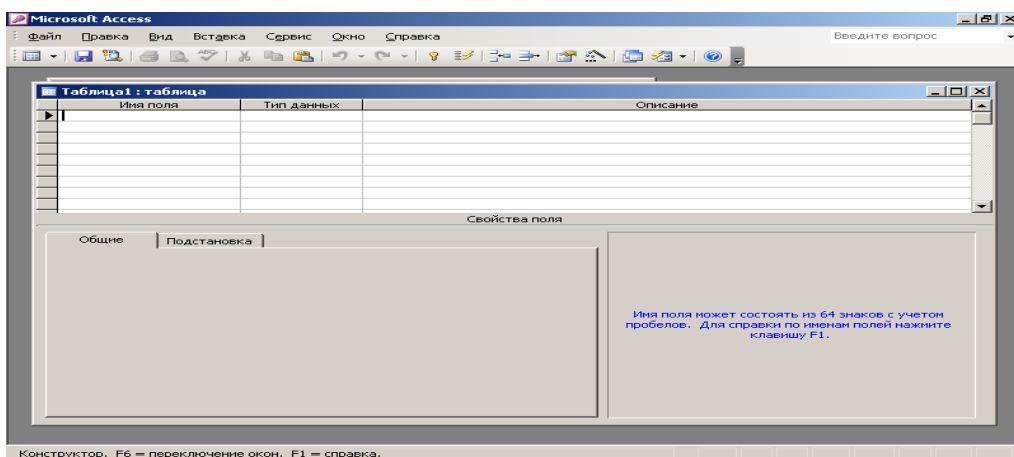


Рис. 5.5. Вікно **Конструктора** таблиць.

**5.4.2.** Користувачам необхідно створити шість таблиць, які укладають базу даних системи контролю безпеки руху. Першою опрацьовується таблиця **Інформація**. Вона складається із 23 полів, скорочені (для зменшення ширини поля) назви яких наведені у першому рядку табл. 5.1.

Перед початком введення значень у вікні **Конструктор** користувачу слід визначити тип даних кожного поля табл. 5.1.

В MS Access використовуються такі типи полів:

1) **Текстовый** - тип даних, призначений для зберігання неформатованого тексту (до 255 символів);

2) **Поле Мемо** - спеціальний тип даних для збереження великих текстів (до 65535 символів); фізично ця інформація зберігається не у полі таблиці, а у іншому місці бази даних; у полі зберігається лише посилання на місце зберігання тексту; проте для користувача це приховано і тому не має істотного значення;

3) **Числовой** - тип даних для зберігання чисел; за замовчуванням цей тип використовується для цілих чисел; якщо необхідно працювати із дійсними числами, то у нижній частині вікна **Конструктор** у рядку **Размер поля** зазначають властивість **Одинарное с плавающей точкой** або **Двойное с плавающей точкой**;

4) **Дата/время** - тип даних для зберігання календарних дат і поточного часу;

5) **Денежный** - тип даних для зберігання грошових сум; цей тип аналогічний типу **Числовой**, але для типу **Денежный** є особливості округлення грошових сум і відображення символу грошової одиниці поряд із числом;

6) **Счетчик** - спеціальний тип для унікальних даних із автоматичним нарощуванням; використовується для порядкової нумерації; поля цього типу заповнюються автоматично;

7) **Логический** - використовується для зберігання логічних даних типу **Да/Нет**;

8) **Поле объекта OLE** - спеціальний тип даних для зберігання мультимедійних даних;

9) **Гиперссылка** - спеціальне поле для зберігання адрес об'єктів Інтернету; при зверненні до значень цього поля автоматично здійснюється виклик браузера і оновлення відображення об'єкта.

**5.4.3.** У вікні **Конструктор** таблиць (рис. 5.5) у першу (верхню) комірку стовпця **Имя поля** ввести назву першого поля таблиці **Информация**, у другу комірку - назву другого поля, і так усі 23 параметри (назви полів) таблиці 5.1. Для переходів по комірках стовпця використовується натискання ЛК миші у потрібній комірці, або натискання клавіш [→] або [←].

Тип даних кожного поля автоматично зазначений праворуч у комірці стовпця **Тип данных**, але для усіх даних однаковий - **Текстовый**, що є помилковим. Необхідно виправити типи даних наступними діями:

1) натиснути ЛК миші у відповідній комірці стовпця **Тип даних** - праворуч виводиться кнопка списку;

2) натиснути кнопку списку - виводиться список типів полів;

3) натиснути ЛК миші на назві потрібного типу.

4) перейти до іншої комірки і виконати аналогічні дії.

**5.4.4.** Після завершення роботи необхідно у рядку заголовка вікна **Конструктора** натиснути кнопку **Закрить** - на екран виводиться діалогове вікно із запитом про збереження; відповісти ствердно, тобто натиснути кнопку **Да**. Далі на екран виводиться вікно для введення назви створюваної таблиці. Слід вилучити запропоновану назву і надрукувати назву **Информация**, а після цього натиснути **ОК** - виконується повернення до первинного вікна бази даних, в якому зазначена назва створеної таблиці.

Слід мати на увазі, що програма Access настільки досконала, що не вимагає виконання **спеціальних** команд для збереження змін і поточного стану. Після натискання кнопки **Закрить** і відповідей на питання у діалоговому вікні результати усіх дій автоматично зберігаються в ядрі програми.

**5.4.5.** Відкрити створену таблицю подвійним натисканням ЛК миші на назві таблиці - відкривається вікно цієї таблиці із порожніми комірками. Позначити комірку першого поля першого запису і ввести за допомогою клавіатури текст, який відображений у табл. 1.1 (у другому рядку першого поля). Для переходів по комірках рядка використовується натискання у потрібній комірці ЛК миші, або натискання таких клавіш:

[→] та [←].

**5.4.6.** Аналогічно створити і заповнити даними п'ять інших таблиць бази даних. Остаточне первинне вікно бази даних СБД матиме вигляд, подібний рис. 5.6. Звернути увагу на той факт, що у цій роботі таблиці бази даних СБД створюються із одним записом, тобто розглядається скорочена кількість записів із метою економії витрат часу на введення реальної кількості даних. Цих записів вистачає для ознайомлення із побудовою структури бази даних, привласненням таблицям імен, зазначенням типів даних і введенням даних у комірки записів.

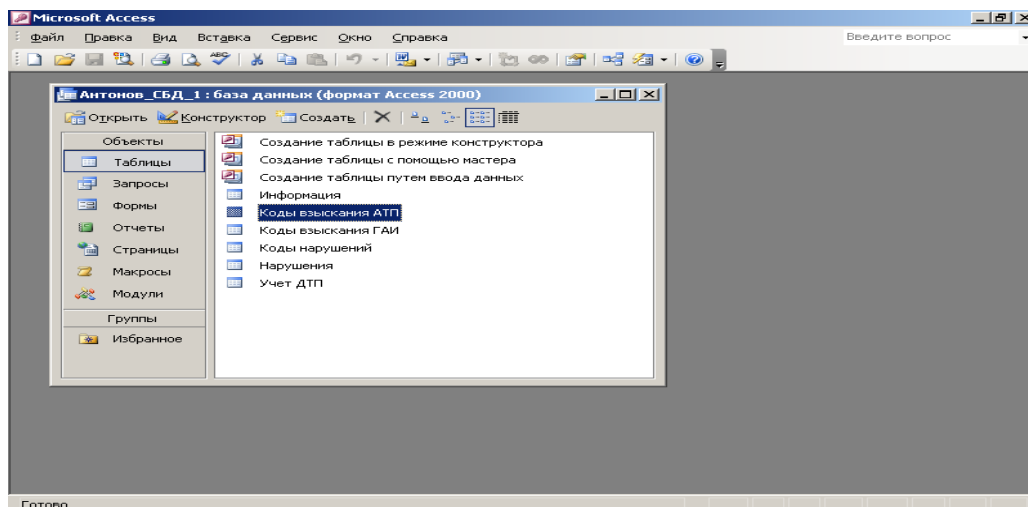


Рис. 5.6. Первинне вікно бази даних СБД.

**5.4.7.** Для закриття бази даних і збереження результатів роботи слід натиснути кнопку **Закрить** у рядку заголовка Access (самий верхній рядок). Створена база даних автоматично буде збережена у файлі із назвою **Іванов\_СБД\_1**.

## **5.5. Контрольні запитання**

1. Що являє собою Система управління базами даних?
2. Які властивості інформації являються найважливішими?
3. Які існують типи моделей даних?
4. Який елемент є найважливішим у СУБД MS Access?
5. Який тип привласнюється файлам MS Access і що означає кожний символ цього типу?
6. Для чого використовується поняття "предметна область"?
7. Із якою метою виконується "інформаційний аналіз"?
8. Що відображає структурна схема бази даних СБД?
9. У чому полягає логічна структура бази даних СБД?
10. Які існують способи створення таблиць у СУБД MS Access?
11. Які типи даних властиві СУБД MS Access?

## **Тема 6. ІНФОРМАЦІЙНЕ НАПОВНЕННЯ БАЗИ ДАНИХ АТП**

**Завдання 6.1.** Дослідити елементи структури бази даних.

**6.1.1.** База даних є файлом спеціального формату, який містить структуровану певним чином інформацію. Більшість баз даних мають табличну структуру, де **адреса** даних визначається перетином рядків і стовпців. **Поля** утворюють **структуру** бази даних, а **записи** складають **інформацію**, що міститься у базі даних. Поля є основними елементами таблиці; вони мають властивості, від яких залежить **тип** даних, що вводяться у поле. Наприклад, дані, що містяться у полі **Кількість даних** можна підсумовувати для визначення підсумкового результату; але ніхто не підсумовуватиме дані поля **Номер телефону**. Зрозуміло, що властивості цих полів відрізняються і належать до різних типів.

**6.1.2.** Основною властивістю кожного поля є його **довжина**, що утворюється символами. Від заданої довжини поля залежить кількість інформації, яку можна у ньому розташувати. Символи кодуються одним або двома байтами і тому можна умовно вважати, що довжина поля вимірюється у **байтах**. Унікальною властивістю будь-якого поля є його **ім'я**; одна таблиця не може мати двох полів із однаковим ім'ям. Різні типи полів мають різне призначення і різні властивості.

Створення бази даних завжди починається із розробки **структури**

її таблиць. Структура має бути такою, щоб при роботі із базою даних вводити мінімальну кількість даних. Якщо введення деяких даних доводиться повторювати, тоді базу даних проектують із декількома пов'язаними таблицями із їх особистими структурами.

Для надійності функціонування зв'язків між таблицями і для однозначного пошуку по запису однієї таблиці запису в іншій таблиці передбачаються **унікальні** поля. При створенні структури таблиць унікальне поле призначають **ключовим** полем. Програма перевіряє унікальність ключових полів і виконує сортування (перегляд) за такими полями із більшою швидкістю.

**Завдання 6.2.** Створити базу даних за варіантом користувача.

**6.2.1.** Викликати на екран менеджер файлів Total Commander і у полі каталогів праворуч відкрити папку користувача. У полі каталогів ліворуч на диску **C:** відкрити папку **2Т**, в якій знаходиться початковий файл **СБД.mdb** бази даних автотранспортного підприємства (АТП) - системи контролю безпеки руху; скопіювати файл у папку користувача. У полі каталогів праворуч у папці користувача змінити ім'я файлу бази даних такими діями:

1) позначити ім'я файлу, тобто перевести на нього покажчик миші і натиснути ЛК миші - усе ім'я файлу оточується рамкою;

2) через одну-дві секунди вдруге натиснути ЛК миші на цьому імені - поле усередині рамки заливається синім кольором (**активізується**);

3) третій раз натиснути ЛК миші - зникає синій колір і виводиться текстовий курсор; далі ім'я файлу можна редагувати як текст;

4) перед літерами СБД ввести прізвище користувача і нижню риску, після літер СБД ввести нижню риску і цифру **2** (крапку необхідно зберегти - вона належить до типу файл, тобто **.mdb**). Таким чином, назва файлу буде мати вигляд, наприклад, **Іванов\_СБД\_2.mdb** (цифра **1** використовувалась у лабораторній роботі **5**); завершити перейменування натисканням [**Enter**].

Подвійним натисканням ЛК миші на імені файлу бази даних користувача викликати на екран первинне вікно бази даних.

**6.2.2.** У вікні бази даних натисканням ЛК миші активізувати об'єкт **Таблиця**; двічі натиснути ЛК миші на імені таблиці **Інформація** - відкривається вікно таблиці, яка містить понад 500 записів.

Упорядкувати таблицю за номерами поля **Счетчик**. Для цього слід розташувати покажчик миші у будь-якій комірці зазначеного поля і натиснути ЛК миші на інструменті **Сортировка по возрастанию** на панелі інструментів **Стандартная**.

**6.2.3.** Надалі користувачу необхідно створити особисту базу із десятьма записами про водіїв. **Першим** у цьому переліку буде водій, **порядковий номер** якого відповідає **трьом останнім цифрам залікової**

**книжки користувача.** Усі попередні записи зайві - їх необхідно вилучити. Для вилучення зайвого запису (рядка) її слід позначити за допомогою натискання ЛК миші на **маркері запису** - перша у рядку кнопка (сірого кольору), а потім натиснути клавішу [**Delete**]. Якщо вилученню підлягають декілька **записів**, то спочатку позначається перший із них; потім слід натиснути клавішу [**Shift**], утримувати її і натискати [↓] для позначення інших. Після позначення декількох записів натиснути клавішу [**Delete**].

Кожен черговий водій вибирається через дев'ять записів; проміжні записи вилучаються.

Вилученню також підлягають останні залишкові записи.

У створену базу даних обов'язково додати **останнім записом прізвище та ініціали користувача**, ввести довільно усі інші параметри подібно до параметрів інших водіїв; комірку табельного номера залишити порожньою. **Звернути увагу** на той факт, що новий запис додається у таблицю бази даних в останньому рядку, і програма самостійно признає йому у полі **Счетчик** останній номер загальної кількості записів цієї таблиці бази даних.

Створення таблиці **Інформація** за варіантом користувача завершується натисканням кнопки **Закричь** у заголовку вікна таблиці - за цією командою в ядрі файлу бази даних буде збережена відредагована таблиця.

**6.2.4.** Записати у конспекті у вигляді таблиці порядкові номери водіїв, їх прізвища, ініціали і табельні номери, роль яких особлива - вони однозначно визначають особу водія в АТП.

**6.2.5.** Відкрити таблицю **Нарушення**, відсортувати її за табельними номерами та відредагувати її так, щоб таблиця складалась із 20-ти записів **виключно** із табельними номерами **усіх** водіїв варіанту користувача; зайві записи слід вилучити.

Якщо записи із відповідними табельними номерами відсутні, то рекомендується використовувати інший запис із заміною у ньому табельного номера на потрібний. Після закінчення формування зазначеної таблиці натиснути кнопку **Закричь** у заголовку таблиці. При цьому в ядрі файлу збережеться відредагована таблиця.

**6.2.6.** Відкрити таблицю **Учет ДТП**, відсортувати її за табельними номерами і сформувати на її основі варіант користувача із 20-ма записами (аналогічно п. 6.2.5).

Після закінчення формування таблиці для її збереження в ядрі файлу бази даних натиснути кнопку **Закричь** у заголовку таблиці.

**6.2.7.** По черзі відкрити створені таблиці, відшукати порожні комірки і самостійно ввести у них дані за зразком інших записів у цих полях. Для збереження змін даних у таблиці обов'язково завершувати редагування кожної таблиці натисканням кнопки **Закричь** у заголовку таблиці.



**6.2.8.** Закінчити формування бази даних АТП за варіантом користувача натисканням кнопки **Закрить** у заголовку вікна середовища Access; при цьому виконується вихід із вікна середовища Access із збереженням в ядрі файлу бази даних користувача усіх виконаних змін.

**Завдання 6.3.** Сформулювати варіанти запитів користувача.

**6.3.1.** Визначити у табл. 6.1 варіант завдання користувача для наступної роботи із базою даних. Звернути увагу на те, що у табл. 6.1 цифри у стовпцях першого, другого та третього запитів відповідають суцільній нумерації стовпців табл. 5.1, 5.2, 5.3, 5.4, 5.5 і 5.6.

**6.3.2.** У конспекті, відповідно до прикладу табл. 6.2, 6.3 і 6.4 і до загальних номерів полів (табл. 5.1 - 5.6), записати особистий варіант завдання користувача.

Таблиця 6.1. **Варіанти завдань**

Дві останні цифри заліковою книжки	Перший запит Номери полів таблиці 1: "Информация"	Другий запит Номери полів таблиць 1 і 2: "Информация" і "Нарушения"	Третій запит Номери полів таблиць 1 і 3: "Информация" і "Учет ДТП"
00	1, 2, 3, 23	1, 2, 3, 26	1, 2, 3, 46
01	1, 2, 3, 4	1, 2, 3, 25	1, 2, 3, 36
02	1, 2, 3, 5	1, 2, 3, 26	1, 2, 3, 37
03	1, 2, 3, 6	1, 2, 3, 27	1, 2, 3, 38
04	1, 2, 3, 7	1, 2, 3, 28	1, 2, 3, 39
05	1, 2, 3, 8	1, 2, 3, 29	1, 2, 3, 40
06	1, 2, 3, 9	1, 2, 3, 31	1, 2, 3, 42
07	1, 2, 3, 10	1, 2, 3, 33	1, 2, 3, 43
08	1, 2, 3, 11	1, 2, 3, 25	1, 2, 3, 44
09	1, 2, 3, 12	1, 2, 3, 26	1, 2, 3, 45
10	1, 2, 3, 13	1, 2, 3, 27	1, 2, 3, 46
11	1, 2, 3, 14	1, 2, 3, 28	1, 2, 3, 36
12	1, 2, 3, 15	1, 2, 3, 29	1, 2, 3, 37
13	1, 2, 3, 16	1, 2, 3, 31	1, 2, 3, 38
14	1, 2, 3, 17	1, 2, 3, 33	1, 2, 3, 39
15	1, 2, 3, 18	1, 2, 3, 25	1, 2, 3, 40
16	1, 2, 3, 19	1, 2, 3, 26	1, 2, 3, 42
17	1, 2, 3, 20	1, 2, 3, 27	1, 2, 3, 43
18	1, 2, 3, 21	1, 2, 3, 28	1, 2, 3, 44
19	1, 2, 3, 22	1, 2, 3, 29	1, 2, 3, 45
20	1, 2, 3, 23	1, 2, 3, 31	1, 2, 3, 46
21	1, 2, 3, 4	1, 2, 3, 33	1, 2, 3, 36
22	1, 2, 3, 5	1, 2, 3, 25	1, 2, 3, 37
23	1, 2, 3, 6	1, 2, 3, 26	1, 2, 3, 38
24	1, 2, 3, 7	1, 2, 3, 27	1, 2, 3, 39

Табл. 6.1 (продовження)

Дві останні цифри заліковою книжки	Перший запит Номери полів таблиці 1: "Інформація"	Другий запит Номери полів таблиць 1 і 2: "Інформація" і "Нарушення"	Третій запит Номери полів таблиць 1 і 3: "Інформація" і "Учет ДТП"
25	1, 2, 3, 8	1, 2, 3, 28	1, 2, 3, 40
26	1, 2, 3, 9	1, 2, 3, 29	1, 2, 3, 42
27	1, 2, 3, 10	1, 2, 3, 31	1, 2, 3, 43
28	1, 2, 3, 11	1, 2, 3, 33	1, 2, 3, 44
29	1, 2, 3, 12	1, 2, 3, 25	1, 2, 3, 45
30	1, 2, 3, 13	1, 2, 3, 26	1, 2, 3, 46
31	1, 2, 3, 14	1, 2, 3, 27	1, 2, 3, 36
32	1, 2, 3, 15	1, 2, 3, 28	1, 2, 3, 37
33	1, 2, 3, 16	1, 2, 3, 29	1, 2, 3, 38
34	1, 2, 3, 17	1, 2, 3, 31	1, 2, 3, 39
35	1, 2, 3, 18	1, 2, 3, 33	1, 2, 3, 40
36	1, 2, 3, 19	1, 2, 3, 25	1, 2, 3, 42
37	1, 2, 3, 20	1, 2, 3, 26	1, 2, 3, 43
38	1, 2, 3, 21	1, 2, 3, 27	1, 2, 3, 44
39	1, 2, 3, 22	1, 2, 3, 28	1, 2, 3, 45
40	1, 2, 3, 23	1, 2, 3, 29	1, 2, 3, 46
41	1, 2, 3, 4	1, 2, 3, 31	1, 2, 3, 36
42	1, 2, 3, 5	1, 2, 3, 33	1, 2, 3, 37
43	1, 2, 3, 6	1, 2, 3, 25	1, 2, 3, 38
44	1, 2, 3, 7	1, 2, 3, 26	1, 2, 3, 39
45	1, 2, 3, 8	1, 2, 3, 27	1, 2, 3, 40
46	1, 2, 3, 9	1, 2, 3, 28	1, 2, 3, 42
47	1, 2, 3, 10	1, 2, 3, 29	1, 2, 3, 43
48	1, 2, 3, 11	1, 2, 3, 31	1, 2, 3, 44
49	1, 2, 3, 12	1, 2, 3, 33	1, 2, 3, 45
50	1, 2, 3, 13	1, 2, 3, 25	1, 2, 3, 46
51	1, 2, 3, 14	1, 2, 3, 26	1, 2, 3, 36
52	1, 2, 3, 15	1, 2, 3, 27	1, 2, 3, 37
53	1, 2, 3, 16	1, 2, 3, 28	1, 2, 3, 38
54	1, 2, 3, 17	1, 2, 3, 29	1, 2, 3, 39
55	1, 2, 3, 18	1, 2, 3, 31	1, 2, 3, 40
56	1, 2, 3, 19	1, 2, 3, 33	1, 2, 3, 42
57	1, 2, 3, 20	1, 2, 3, 25	1, 2, 3, 43
58	1, 2, 3, 21	1, 2, 3, 26	1, 2, 3, 44
59	1, 2, 3, 22	1, 2, 3, 27	1, 2, 3, 45
60	1, 2, 3, 23	1, 2, 3, 28	1, 2, 3, 46
61	1, 2, 3, 4	1, 2, 3, 29	1, 2, 3, 36
62	1, 2, 3, 5	1, 2, 3, 31	1, 2, 3, 37
63	1, 2, 3, 6	1, 2, 3, 33	1, 2, 3, 38
64	1, 2, 3, 7	1, 2, 3, 25	1, 2, 3, 39
65	1, 2, 3, 8	1, 2, 3, 26	1, 2, 3, 40

Таблиця 6.1 (продовження)

Дві останні цифри заліковою книжки	Перший запит Номери полів таблиці 1: "Інформація"	Другий запит Номери полів таблиць 1 і 2: "Інформація" і "Нарушення"	Третій запит Номери полів таблиць 1 і 3: "Інформація" і "Учет ДТП"
66	1, 2, 3, 9	1, 2, 3, 27	1, 2, 3, 42
67	1, 2, 3, 10	1, 2, 3, 28	1, 2, 3, 43
68	1, 2, 3, 11	1, 2, 3, 29	1, 2, 3, 44
69	1, 2, 3, 12	1, 2, 3, 31	1, 2, 3, 45
70	1, 2, 3, 13	1, 2, 3, 33	1, 2, 3, 46
71	1, 2, 3, 14	1, 2, 3, 25	1, 2, 3, 36
72	1, 2, 3, 15	1, 2, 3, 26	1, 2, 3, 37
73	1, 2, 3, 26	1, 2, 3, 27	1, 2, 3, 38
74	1, 2, 3, 17	1, 2, 3, 28	1, 2, 3, 39
75	1, 2, 3, 18	1, 2, 3, 29	1, 2, 3, 40
76	1, 2, 3, 19	1, 2, 3, 31	1, 2, 3, 42
77	1, 2, 3, 20	1, 2, 3, 33	1, 2, 3, 43
78	1, 2, 3, 21	1, 2, 3, 25	1, 2, 3, 44
79	1, 2, 3, 22	1, 2, 3, 26	1, 2, 3, 45
80	1, 2, 3, 23	1, 2, 3, 27	1, 2, 3, 46
81	1, 2, 3, 4	1, 2, 3, 28	1, 2, 3, 36
82	1, 2, 3, 5	1, 2, 3, 29	1, 2, 3, 37
83	1, 2, 3, 6	1, 2, 3, 31	1, 2, 3, 38
84	1, 2, 3, 7	1, 2, 3, 33	1, 2, 3, 39
85	1, 2, 3, 8	1, 2, 3, 25	1, 2, 3, 40
86	1, 2, 3, 9	1, 2, 3, 26	1, 2, 3, 42
87	1, 2, 3, 10	1, 2, 3, 27	1, 2, 3, 43
88	1, 2, 3, 11	1, 2, 3, 28	1, 2, 3, 44
89	1, 2, 3, 12	1, 2, 3, 29	1, 2, 3, 45
90	1, 2, 3, 13	1, 2, 3, 31	1, 2, 3, 46
91	1, 2, 3, 14	1, 2, 3, 33	1, 2, 3, 36
92	1, 2, 3, 15	1, 2, 3, 25	1, 2, 3, 37
93	1, 2, 3, 16	1, 2, 3, 26	1, 2, 3, 38
94	1, 2, 3, 17	1, 2, 3, 27	1, 2, 3, 39
95	1, 2, 3, 18	1, 2, 3, 28	1, 2, 3, 40
96	1, 2, 3, 19	1, 2, 3, 29	1, 2, 3, 42
97	1, 2, 3, 20	1, 2, 3, 31	1, 2, 3, 43
98	1, 2, 3, 21	1, 2, 3, 33	1, 2, 3, 44
99	1, 2, 3, 22	1, 2, 3, 25	1, 2, 3, 45

Таблиця 6.2. Приклад запису варіанта користувача

Дві останні цифри залікової книжки	Перший запит - номери полів таблиці 1: "Інформація"	Другий запит - номери полів таблиць 1, 2: "Інформація" і "Нарушення"	Третій запит - номери полів таблиць 1, 3: "Інформація" і "Учет ДТП"
<b>00</b>	<b>1, 2, 3, 23</b>	<b>1, 2, 3, 26</b>	<b>1, 2, 3, 46</b>

Таблиця 6.3. Номери полів і заголовків таблиці "Інформація"

<b>1</b>	<b>2</b>	<b>3</b>
<b>№</b>	<b>ФІО</b>	<b>№_таб</b>
Табличний код водія	Прізвище і ініціали водія	Табельний номер

Таблиця 6.4. Номери полів і заголовків таблиць для запису умов відбору

Перший запит - номер поля таблиці 1: "Інформація"	Другий запит - номери полів таблиць 1 і 2: "Інформація" "Нарушення"	Третій запит - номери полів таблиць 1 і 3: "Інформація" "Учет ДТП"
<b>23</b>	<b>26</b>	<b>46</b>
<b>ІО</b>	<b>Вр_нар</b>	<b>Код_Г</b>
Ім'я і по батькові водія	Час порушення	Код стягнення ДАІ

#### 6.4. Контрольні запитання

1. У чому полягає інформаційне наповнення бази даних?
2. Які елементи укладають фізичну структуру бази даних?
3. Що є інформаційним вмістом бази даних?
4. У чому полягає зміст поняття "ключевое поле"?
5. Для чого використовується поле "Счетчик" и що для нього характерно?
6. Які дії сприяють переміщенню текстового курсору по комірках бази даних?

## Тема 7. МОДЕЛЮВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ БАЗИ ДАНИХ АТП

**Завдання 7.1.** Виконати аналіз стану вікон таблиць бази даних/

**7.1.1.** Викликати на робочий стіл комп'ютера менеджера файлів **Total Commander**. Відкрити папку користувача і двічі натиснути ЛК миші на назві файлу бази даних - на екран виводиться вікно бази даних СБД користувача.

**7.1.2.** Відкрити подвійним натисканням ЛК миші на назві будь-яку таблицю у вікні бази даних (наприклад, **Коды взысканий ГАИ**). У вікні таблиці (рис. 7.1) кожний запис (рядок) має ліворуч маркер запису - кнопку сірого кольору, натискання якої позначає відповідний запис.

**7.1.3.** Ліворуч, у нижній частині вікна знаходиться поле номера запису із кнопками переміщення по записах таблиці. У цьому полі виводиться номер **активного** запису; трикутники визначають напрям перегляду записів у таблиці, а праворуч цього поля виводиться кількість записів усієї таблиці. Маркер запису, який знаходиться у лівому верхньому кутку таблиці, є маркером **усієї** таблиці. Натискання на ньому ЛК миші позначає усю таблицю, а натискання ЛК миші у будь-якому місці позначеної таблиці знімає її позначення.

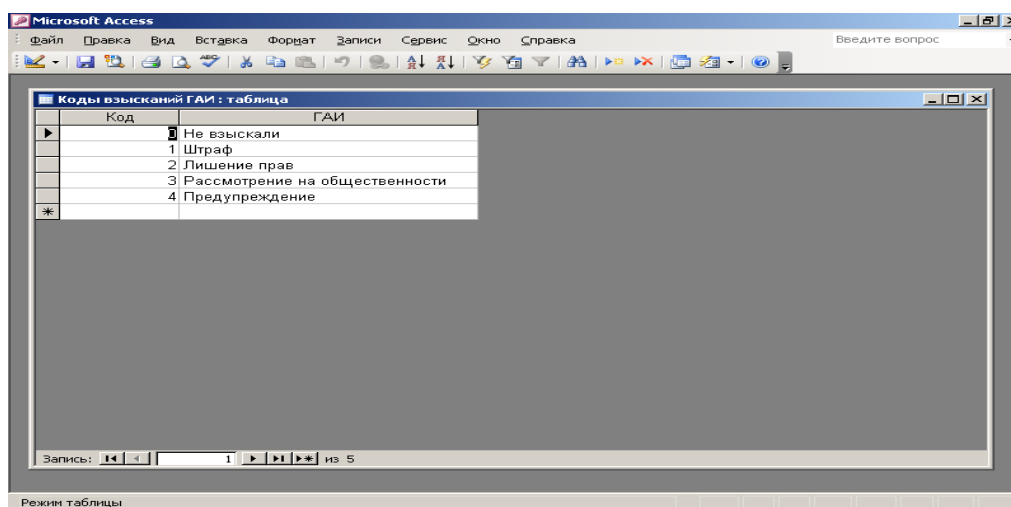


Рис. 7.1. Вікно таблиці **Коды взысканий ГАИ**.

**7.1.4.** Межі (риски) між заголовками полів рухливі - їх можна пересувати ліворуч або праворуч (іноді програма виконує це самостійно). Подвійне натискання ЛК миші на межі автоматично визначає ширину усіх комірок цього поля рівною довжині найдовшого запису.

**7.1.5.** Натисканням ЛК миші на заголовку стовпця виділяється увесь стовпець. Натискання ПК миші у будь-якому місці стовпця викликає контекстний список команд, в якому є команди, що дозволяють сор-

тувати записи у цих полях, вставити новий стовпець, приховати стовпець, тощо. Прихований стовпець залишається у базі даних, але зникає з екрану. При необхідності його відображення слід навести покажчик на межу, де прихований стовпець, і двічі натиснути ЛК миші.

**7.1.6.** Закрити вікно **Коды разысканий ГАИ** натисканням кнопки **Закреть** у рядку заголовку цієї таблиці - програма повертається до первинного вікна бази даних.

## **Завдання 7.2.** Дослідити можливості зазначення типів даних/

**7.2.1.** Для дослідження можливостей модифікації структури таблиць бази даних позначити об'єкт **Таблицы** і у переліку таблиць (у вікні праворуч) позначити таблицю **Информация**. Перевести покажчик миші у рядок інструментів (у вікні бази даних під рядком заголовку) і натиснути ЛК миші на інструменті **Конструктор** - на екран виводиться вікно цього інструменту із назвою **Информация: таблица** (рис. 7.2).

Виведення на екран вікна **Конструктор** можна виконати також розташуванням покажчика миші на назві таблиці у вікні бази даних і натисканням ПК миші - виводиться контекстний список команд. У цьому вікні слід виконати команду **Конструктор** - на екран виводиться вікно із назвою **Информация: таблица** (рис. 7.2).

**7.2.2.** У вікні Конструктор у верхній половині відображена структура бази даних - перелік полів і їх типів; саме тут зазначають або редагують складові частини структури бази даних.

У нижній половині ліворуч зазначені програмою за замовчуванням або змінюються користувачем властивості полів; праворуч автоматично виводяться коментарі до кожної властивості. Вставляється нове поле у вікні **Конструктор** розташуванням покажчика миші на маркері поля і натисканням клавіші **[Insert]**, а далі зазначається назва поля і тип даних.

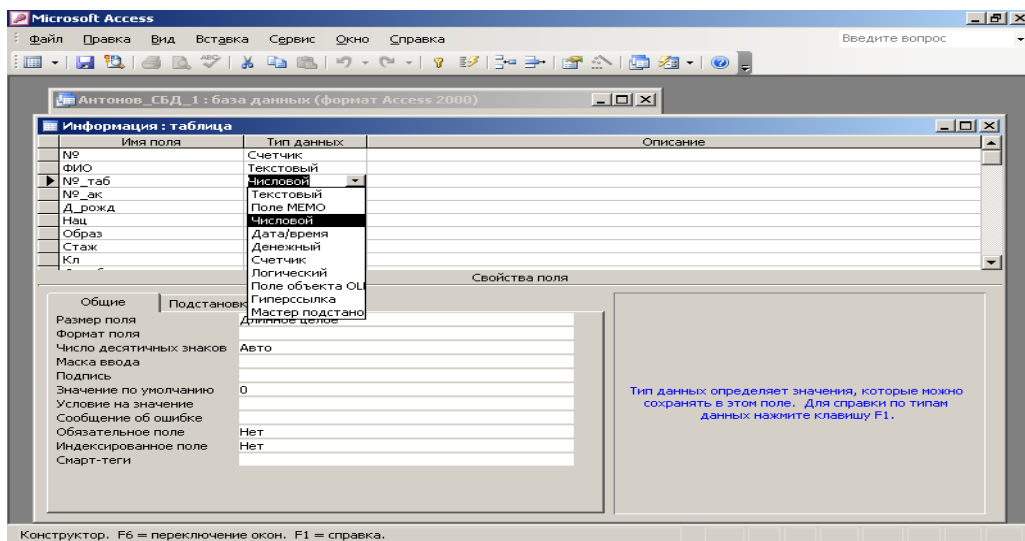


Рис. 7.2. Вікно **Конструктора** таблиці **Информация**.

Вилучення зайвого поля виконується також у вікні **Конструктор**; слід натисканням ЛК миші на маркері позначити зайве поле і натиснути клавішу [**Delete**].

Далі слід натиснути кнопку **Закри́ть** у заголовку цього вікна, а у діалоговому вікні із запитом наступних дій зазначити свій вибір.

**7.2.3.** Якщо у таблицях тип поля не відповідає типу даних, програма повідомляє про помилку. Користувачам рекомендується у вікні **Конструктор** обов'язково перевірити відповідність типів даних у полях усіх таблиць бази даних користувача. При виявленні помилкового типу поля слід виправити помилку такими діями:

- 1) у рядку поля із помилковим типом слід у стовпці **Тип поля** натиснути ЛК миші - з'являється кнопка списку типів;
- 2) натисканням кнопки списку вивести на екран перелік типів полів;
- 3) натиснути ЛК миші на потрібному типі - список закривається, а у комірці розташовується вибраний тип даних.

Після корекції структури слід повернутися до таблиці за допомогою виконання команди **Вид→Режим таблиці**. Повернутись до таблиці можна також за допомогою натискання кнопки **Закри́ть** у заголовку цього вікна.

### **Завдання 7.3.** Вивчити методи створення зв'язків між таблицями/

**7.3.1.** Основні переваги систем управління базами даних реалізуються при роботі із групами взаємопов'язаних таблиць. У лабораторній роботі 5 користувачі виконували завдання із окремими таблицями. Якби інформація зберігалась у таких простих структурах, то для роботи із нею можна було б обійтись без спеціальних систем управління базами даних. Але на практиці доводиться мати справу із значно складнішими структурами, які утворені багатьма **пов'язаними** таблицями.

Бази даних, утворені пов'язаними таблицями, називають **реляційними базами даних** (від англ. **relation** - відношення).

Потужність реляційних баз даних, таких як MS Access, базується на їх здібності швидко знайти дані із різних таблиць за допомогою запитів, форм і звітів. Для цього кожна таблиця має містити одне або декілька полів (наприклад, табельні номери водіїв АТП), які містять у таблиці тільки одне значення кожного запису. Такі поля називають **первинним ключем таблиці**. Поле типу **ключове поле** відіграє особливу роль у таблицях - його використовують для утворення **зв'язків** між таблицями. Усі записи цього поля мають бути **унікальними**. Зв'язки за таким полем згодом сприяють побудові запитів, звітів, тощо.

**7.3.2.** Для створення зв'язків між таблицями у СУБД Access існує спеціальне діалогове вікно, що називається **Схема даних**. Його відкривають натисканням однойменної кнопки панелі інструментів або виконанням команди **Сервіс→Схема даних**.

На початку роботи у цьому вікні відсутні будь-які елементи. Але необхідно перевірити наявність прихованих пов'язаних таблиць. Вони могли бути утворені самою програмою Access за замовчуванням і зберігались в її ядрі. Для перевірки присутності помилкових зв'язків потрібно натиснути на панелі інструментів **Стандартная** кнопку **Отобразить прямые связи** - стан вікна залишається незмінним. Натиснути кнопку **Отобразить все связи** - у вікні можуть з'явитись зменшені копії (у них відображені тільки назви полів) деяких таблиць із лініями зв'язків, що сполучають однойменні поля.

Порівняти виведені у вікні **Схема данных** копії таблиць і зв'язки між ними із зображенням схеми на рис. 7.3.

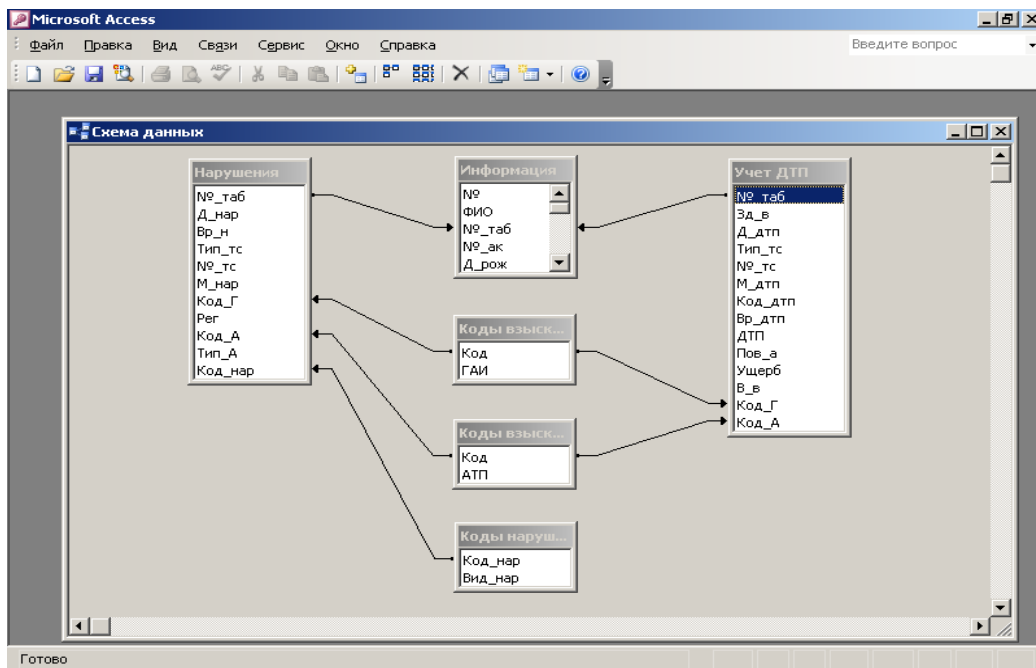


Рис. 7.3. Схема бази даних **Систем безпеки руху**.

Створена програмою (за замовчуванням) схема даних значно відрізнятиметься від наведеної на рис. 7.3. Але у ядрі БД вона зберігається і у майбутньому може викликати значні похибки. Тому слід вилучити спочатку лінії зв'язків (створені за замовчуванням), потім таблиці (створені за замовчуванням) і створити власну схему даних.

Для вилучення лінії зв'язку необхідно навести кінчик покажчика миші на лінію і натиснути ПК миші. У списку контекстних команд, що спливає, виконати **Удалить**. Після вилучення усіх ліній зв'язку необхідно вилучити усі копії таблиць. Активізація (рядок заголовку набуває синього кольору) кожної із них виконується програмою за замовчуванням. Вилучення копій таблиць виконується за допомогою натискання клавіші **[Delete]**. Процедура очищення вікна **Схема данных** завершується натисканням кнопки **Закреть** для збереження в ядрі програми чистого вікна.



**7.3.3.** Знову викликати на екран вікно **Схемы данных**. Натиснути інструмент **Отобразить таблицу** - виводиться вікно **Добавление таблицы**, яке дозволяє вибирати потрібні таблиці для створення міжтабличних зв'язків. У цьому вікні слід зазначити об'єкт, для якого виконується створення **Схемы данных**, тобто натиснути кнопку **Таблицы**.

Для побудови схеми даних для **усіх** таблиць БД їх слід помітити таким чином: натиснути ЛК миші на назві таблиці **Информация**, потім натиснути ліву клавішу [**Shift**], утримувати її і натискати [**↓**].

Далі необхідно натиснути кнопку **Добавить** - у полі **Схема данных** з'являються копії усіх таблиць БД. Натиснути кнопку **Заккрыть** цього вікна - поточне вікно закривається і виконується повернення у вікно **Схема данных**.

**7.3.4.** Розташувати копії таблиць у відповідності рис. 7.3 - перевести покажчик миші на заголовок таблиці, натиснути ЛК миші, утримувати, перемістити таблицю у потрібне місце, відпустити ЛК миші.

Звернути увагу, що поєднуються однакові за змістом поля, хоча назви полів можуть дещо відрізнитись (рис. 7.3). Напрями ліній зв'язків між таблицями бази даних СБД наступні (рис. 7.3):

- 1) від таблиці **Нарушения** до таблиці **Информация**;
- 2) від таблиці **Учет ДТП** до таблиці **Информация**;
- 3) від таблиці **Коды розысканий ГАИ** до таблиці **Нарушения**;
- 4) від таблиці **Коды розысканий АТП** до таблиці **Нарушения**;
- 5) від таблиці **Коды нарушений** до таблиці **Нарушения**;
- 6) від таблиці **Коды розысканий ГАИ** до таблиці **Учет ДТП**;
- 7) від таблиці **Коды розысканий АТП** до таблиці **Учет ДТП**.

**7.3.5.** Для створення зв'язку між полями двох таблиць слід розташувати покажчик миші на ключовому полі однієї таблиці, натиснути ЛК миші, утримувати і перемістити покажчик миші на відповідне йому пов'язуване поле іншої таблиці; відпустити ЛК миші - відкривається діалогове вікно **Изменение связей**, в якому можна задати властивості зв'язку. Для виконання наступних операцій переконатись, що у вікні **Изменение связей** вилучена позначка в опції **Обеспечение целостности данных**.

Присутність позначки у віконці опції (параметра) **Обеспечение целостности данных** дозволяє захистити від випадкового вилучення записів лише із однієї таблиці; при цьому пов'язані із ними дані інших таблиць залишаються без зв'язку.

Для існування умови цілісності поле основної таблиці має бути ключовим і обидва поля повинні мати однаковий тип. Якщо існує позначка в опції **Обеспечение целостности данных**, то активізуються опції **Каскадное обновление связанных полей** та **Каскадное удаление связанных записей**, які також вимагають позначок. Ці опції забезпечують одночасне оновлення або вилучення даних в усіх підлеглих таблицях при зміні цих даних у головній таблиці. Використання зазначеної опції виконуються у режимі редагування таблиць бази даних.

**7.3.6.** У вікні **Изменение связей** натиснути кнопку **Объединение** - відкривається чергове вікно **Параметры объединения**, в якому наведені опції можливих типів з'єднань. Самостійно і уважно ознайомитись із інформацією цього вікна. Надалі для усіх зв'язків схеми даних зазначити опцію із номером **2** і завершувати натисканням **ОК**.

#### **Завдання 7.4.** Дослідити методи проектування форм.

**7.4.1.** Звичайно автор проекту бази даних створює структуру таблиць, але заповненням таблиць інформацією він не займається. Цю рутинну роботу виконують професійні оператори. Для оптимізації їх праці автор проекту може підготувати спеціальні об'єкти - **форми**. Методи проектування різноманітних форм користувачі досліджують у процесі вивчення об'єктно-орієнтованої мови Visual Basic - інструментальної мови Microsoft Office. При проектуванні форм (електронних бланків) у них створюються пов'язані із таблицями поля для введення даних. Завдяки зв'язку, дані із пов'язаних полів пересилаються у таблиці. Також дані у таблицю можна вводити і без допомоги форм. Але існує декілька наведених нижче чинників, через які форми є необхідним засобом введення даних у таблиці.

1) Малокваліфікованому персоналу не можна надавати доступ до таблиць (найціннішому із того, що є у базі даних);

2) Окремим користувачам призначають різні права доступу до інформації у таблицях із метою забезпечення збереження даних та їх конфіденційності. Наприклад, деякий користувач має право вводити тільки імена і адреси клієнтів, інший - тільки номери їх рахунків, третій - тільки грошові суми, які зберігаються на цих рахунках. Змова між цими людьми має бути виключена. Для введення даних їм надають різні форми, але дані із форм будуть надходити до одної таблиці.

3) Введення даних у таблицю надзвичайно кропітка робота; після декількох годин роботи оператори роблять помилки. Введення даних у форму простіше - виконується окремо для кожного запису і не торкається інших записів.

4) Інформація для баз даних витягається із паперових бланків: анкет, заяв, накладних, рахунків, довідок тощо. Екранні форми проектують як точні копії паперових бланків, що значно зменшує стомлюваність оператора і кількість помилок при введенні.

**7.4.2.** Форми складаються із певної кількості елементів управління і від того, наскільки раціонально ці елементи розташовані на екрані, залежить зовнішній вигляд форми. Засоби автоматизації середовища Access дозволяють створювати форми достатньо просто та раціонально.

Для створення **автоформи** у вікні бази даних слід позначити об'єкт **Формы**, а потім виконати команду **Создание форм с помощью мастера** - на екран виводиться вікно **Создание форм**; рекомендується

уважно ознайомитися із змістом вікна (рис. 7.4).

У цій роботі форми створюються лише для таблиць. Тому у зазначеному вікні у полі **Таблицы и запросы** слід натиснути кнопку списку, а потім натиснути ЛК миші на назві таблиці **Информация**.

У відповідь в області **Доступные поля** виводяться назви полів таблиці **Информация**.

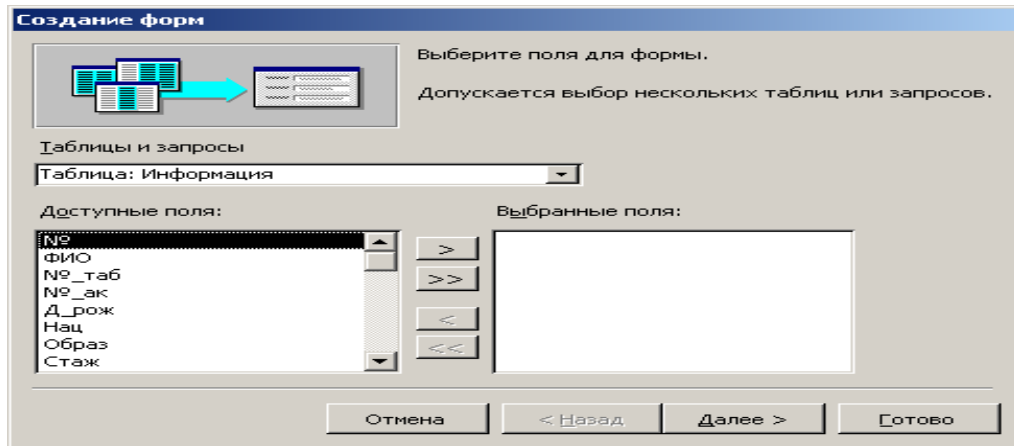


Рис. 7.4. Вікно створення форм.

**7.4.3.** Програма надає користувачу дві можливості вибору переліку полів для введення інформації:

1) позначити назву окремого поля і натиснути [ > ]; при цьому позначене поле переміщується в область **Выбранные поля**;

2) поля не зазначати, а натиснути [ >> ] - в область **Выбранные поля** переміщуються **усі** назви.

Звернути увагу, що натисканням [ < ] та [ << ], переміщення назв виконується у **зворотному** напрямі.

Перевести в область **Выбранные поля** усі назви і натиснути кнопку **Далее** - на екран виводиться вікно **Создание форм**, в якому надані опції (параметри) вибору зовнішнього вигляду форми; ліворуч області опцій відображається зразок, відповідний позначеній опції. Рекомендується вибрати оформлення "в один столбец" і натиснути кнопку **Далее**.

**7.4.4.** У черговому вікні виконати вибір стилю форми і натиснути кнопку **Далее**. У наступному вікні необхідно присвоїти назву формі. Рекомендується залишити назву таблиці, для якої вона створена, але додати до неї пробіл і цифру **1**; наприклад, **Информация 1** (рис. 7.5). Додавання цифри дозволяє уникнути однакових назв об'єктів.

**7.4.5.** Завершується проектування форми натисканням кнопки **Готово** - на екран виводиться вікно форми **Информация 1** (рис. 7.5).

У вікні цієї форми відображається тільки один запис. Гортання записів виконується за допомогою кнопок із стрілками-трикутниками у лівому нижньому кутку форми.

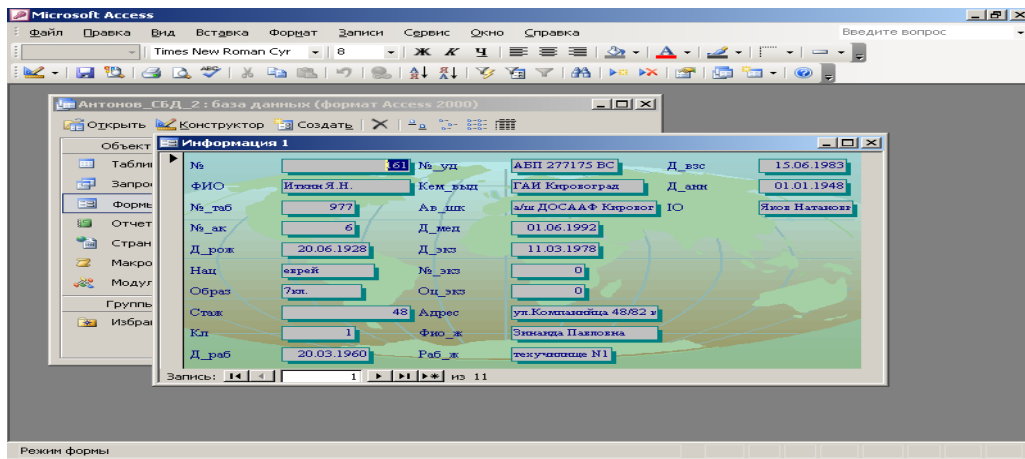


Рис. 7.5. Вікно форми **Информация 1**.

Введення і вилучення даних у полях форми виконується звичайним способом - подібно до роботи у середовищі текстового редактора; натискання [Tab] сприяє переміщенню по полях.

Закрити вікно натисканням кнопки **Закерыть** у вікні форми.

**7.4.6.** Аналогічно побудувати форми для усіх шести таблиць, що укладають базу даних системи контролю безпеки руху (рис. 7.6).

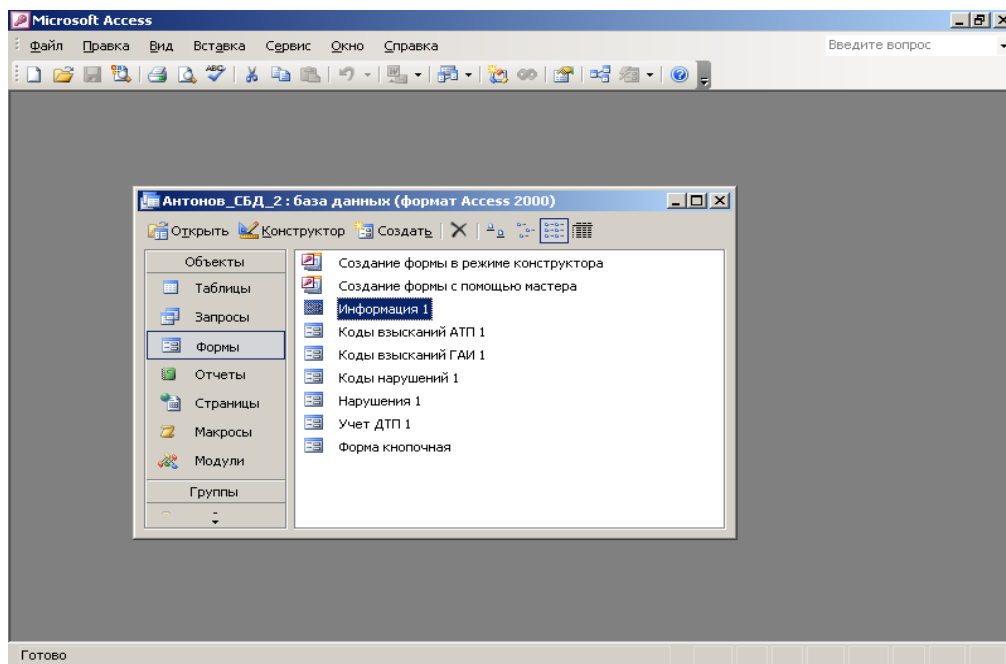


Рис. 7.6. Вид вікна бази даних із назвами форм після закінчення їх проектування.

Кожній формі призначити назву, що складається (рис. 7.6) із назв таблиць, для яких створена форма, пропуску і цифри 1.

Після закінчення роботи закрити вікно середовища Access натисканням кнопки **Закерыть** у рядку заголовка.

## 7.5. Контрольні запитання

1. Які елементи укладають таблиці СУБД MS Access?
2. Які таблиці укладають базу даних СБД?
3. Яка роль інструменту **Конструктор** у СУБД MS Access?
4. Що являє собою **Схема даних** бази даних СБД?
5. Із якою метою створюються зв'язки між таблицями бази даних?
6. Які основні етапи створення таблиць бази даних СБД?
7. Що являють собою об'єкти бази даних форми?
8. Із якою метою проектуються форми?

## Тема 8. ДОСЛІДЖЕННЯ ІНФОРМАЦІЙНИХ МОЖЛИВОСТЕЙ БАЗИ ДАНИХ АТП

**Завдання 8.1.** Ознайомитись із призначенням запитів та їх типами.

**8.1.1. Запит** є гнучким і зручним засобом для опрацювання інформації у базі даних. Для однієї і тієї ж таблиці можна створити декілька запитів, кожний із яких зможе вибирати із таблиці окрему частину інформації, що цікавить у даний момент користувача. У результаті із загальної початкової бази формується таблиця запиту із зазначеною частиною загальної інформації. Важливою властивістю запитів є те, що при створенні таблиці запиту можна не лише вибирати інформацію із бази даних, але й опрацьовувати її: дані можна сортувати, фільтрувати, об'єднувати, розподіляти, змінювати і обчислювати; при цьому базові таблиці залишаються незмінними.

**8.1.2.** У середовищі Access передбачена можливість створювати наступні запити.

1) **Запит на вибір** - вибираються записи, що відповідають зазначеним умовам відбору параметрів.

2) **Запит на зміну** - виконується копіювання або зміна даних. Такий запит може бути створений для виконання деяких операцій: вилучення записів, оновлення записів, додавання таблиці нових записів, створення нової таблиці із певних записів.

3) **Перехресний запит** - дозволяє виконувати статистичні обчислення (суми, кількість записів, середніх величин) за даними одного поля запису. Результати виводяться в окремих певних полях.

4) **Запит із параметрами** - забезпечує введення додаткової інформації, яка згодом використовується як критерії відбору.

5) **Запит на обчислення** - сприяє виконанню обчислень за формулами із вмістом полів.

## Завдання 8.2. Дослідити методи створення запитів.

**8.2.1.** Викликати на екран первинне вікно бази даних користувача і позначити об'єкт **Запросы** - у полі вікна праворуч зазначаються три можливі способи створення запитів (рис. 8.1):

- 1) виконанням команди **Создание запроса в режиме конструктора**;
- 2) виконанням команди **Создание запроса с помощью мастера**.
- 3) натисканням на панелі інструментів кнопки **Создать**.

При зверненні до перших двох способів відкриваються відповідні вікна для проектування запиту. При зверненні до третього способу програма надає наступні п'ять додаткових типів запиту:

- 1) **Конструктор** - із порожнім бланком;
- 2) **Простой запрос** - створення запиту на вибірку із певних полів;
- 3) **Перекрестный запрос** - запит, що виводить дані у компактно-му форматі, подібному формату електронної таблиці;
- 4) **Повторяющиеся записи** - вибираються дубльовані записи;
- 5) **Записи без подчиненных** - із зазначеної таблиці вибираються записи, які не пов'язані із записами іншої таблиці.

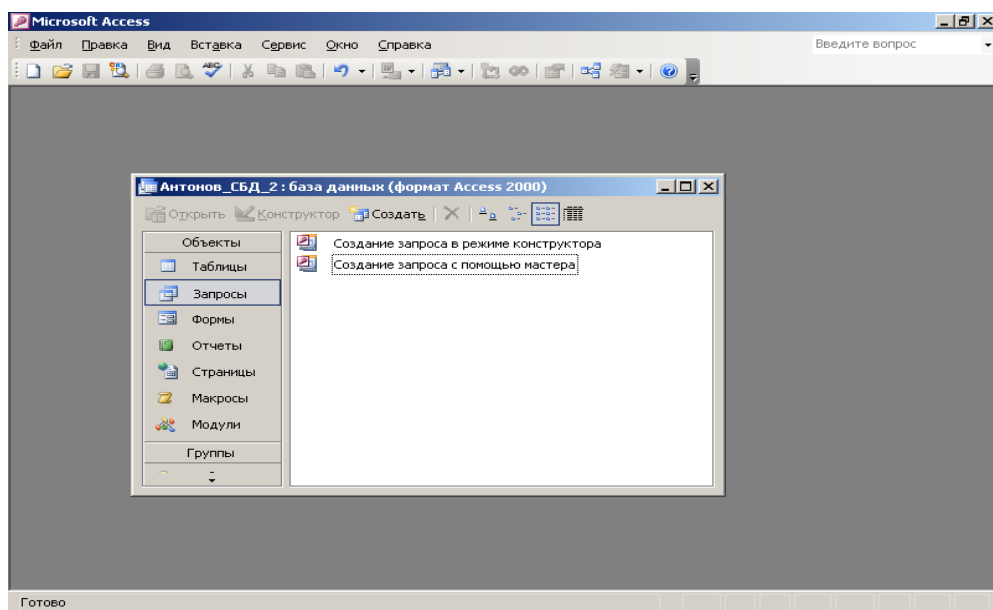


Рис. 8.1. Вікно бази даних у режимі звернення до об'єкту **Запити**.

**8.2.3.** Подвійним натисканням ЛК миші виконати команду **Создание запроса в режиме конструктора**; на екран виводиться вікно конструктора **Запрос1: запрос на выборку**, а поверх нього - вікно **Добавление таблицы** (рис. 8.2).

Вибір таблиць, які братимуть участь у запиті, виконується у діалоговому вікні **Добавление таблицы**. Звернути увагу на наявність у цьому

вікні трьох карток: **Таблицы**, **Запросы** та **Запросы и таблицы**. Вони свідчать про те, що запити створюються не лише за таблицями. Якщо раніше вже був створений запит, то він може бути основою нового самостійного запиту або комбінованого - за іншими запитами і таблицями.

**8.2.4.** Бланк запиту (рис 8.2), який знаходиться під вікном **Добавление таблицы**, має дві частини. **Верхня** частина є областю схеми, в якій відображаються віконця таблиць, що беруть участь у запиті (із назвами усіх стовпців зазначеної таблиці), та лініями зв'язків між полями таблиць.

У відповідності із варіантом користувача (табл. 6.1) перший запит слід створити за таблицею **Информация**. Для цього зазначити у вікні **Добавление таблицы** картку **Таблицы** і позначити назву **Информация**; натиснути кнопку **Добавить** - зазначена таблиця виводиться в області схеми бланка запиту на вибір (рис. 8.3).

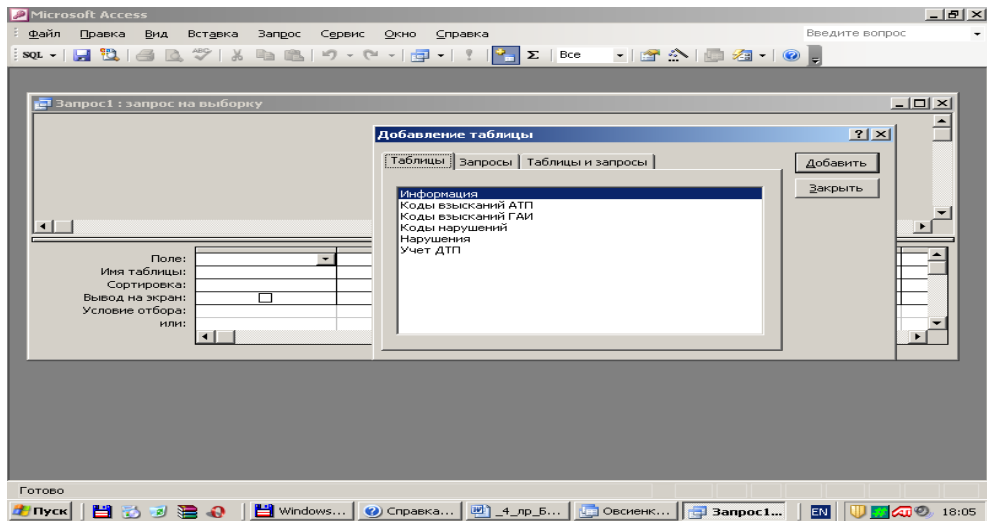


Рис. 8.2. Вікно для створення запиту у режимі **Конструктора**.

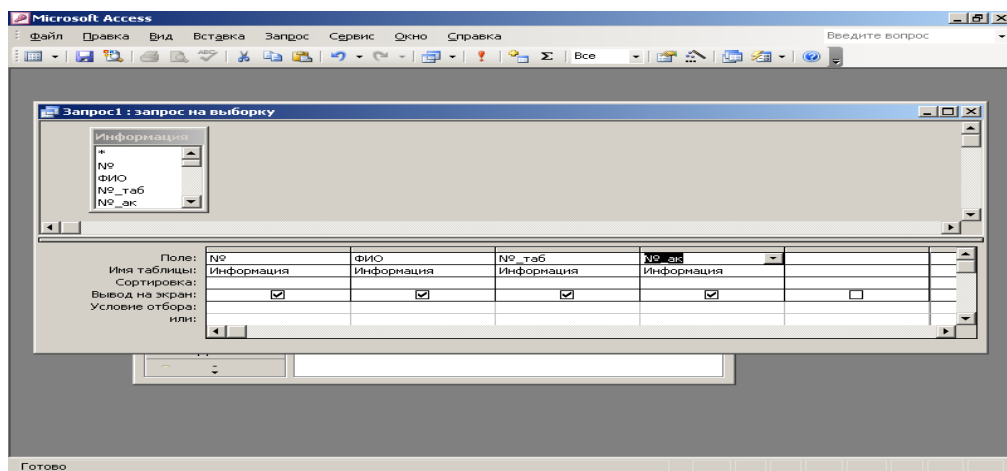


Рис. 8.3. Вікно **Конструктора** при створенні запиту **Информация 1**.

Після завершення вибору таблиць у вікні **Добавление таблицы** слід натиснути кнопку **Закреть** - це вікно зникає з екрану, але залишається бланк запиту.

**8.2.5.** Нижня частина бланку запиту є **областю сітки** і містить порожні комірки, в яких зазначаються параметри запиту: імена стовпців, умови відбору рядків, угруповання записів, тощо. **Стовпці області сітки** визначають **структуру результату** запиту.

Рядок **Поле** в області сітки заповнюють перетягуванням **назви поля** із потрібної таблиці в області схеми у верхню комірку відповідного стовпця в області сітки. Кожному полю майбутньої результуючої таблиці відповідає один стовпець бланку запиту на вибір.

Звернути особливу увагу на те, що у варіантах (табл. 6.1) запитів користувачів повторюються поля із номерами 1, 2 і 3. Це пояснюється тим, що усі запити мають бути **персоніфіковані**; четверте поле у кожному запиті у кожного користувача відрізняється.

Перетягування виконується таким чином. Перевести покажчик миші в область схеми на перше поле, натиснути ЛК миші, утримувати її натиснутою і пересунути в область сітки у верхню комірку першого стовпця ліворуч; відпустити клавішу миші - у цій комірці виводиться назва зазначеного поля таблиці, а рядок **Имя таблицы** заповнюється автоматично при перетягуванні назви поля.

У рядку **Сортировка** визначають тип сортування. Натисканням у відповідній комірці цього рядка ЛК миші на екран викликається кнопка списку. Наступне натискання кнопки списку сприяє вивченню на екран можливих варіантів сортування записів цього поля у результуючій таблиці. Рекомендується залишити цей рядок порожнім, а необхідне сортування виконувати у вікні запиту за допомогою інструментів панелі **Стандартная**.

У рядку **Вывод на экран** позначки у віконцях опцій забезпечує виведення на екран у вікні запиту відповідного стовпця.

У рядку **Условие выбора** друкують умови відбору певних записів для вікна запиту. Для кожного поля можна ввести свою умову відбору - просту або складну. Рекомендується на цьому етапі роботи залишити цей рядок порожнім.

**8.2.6.** Завершується створення запиту натисканням кнопки **Закреть** у заголовку вікна **Запрос1 : запрос на выборку** - виводиться вікно із запрошенням ввести назву запиту; вона повинна складатися із назв таблиць, за змістом яких утворений запит, і мати цифру **2** (рис. 8.4).

**8.2.7.** Аналогічно п.п. 8.2.3 - 8.2.5. створити 2-й та 3-й запити за варіантом користувача. Звернути увагу на те, що у цих випадках в області схеми мають бути присутні дві відповідні таблиці і лінії зв'язку їх параметрів, подібні до зв'язків **Схемы данных**.

**8.2.8.** Повторити створення зазначених вище запитів і присвоїти їм у назвах цифру **3** (рис. 8.4). Ця дія пояснюється тим, що надалі запити



із цифрою **2** залишаться незмінними (для порівняння); у запити із цифрою **3** будуть введені умови вибору.

**8.2.9.** На рис. 8.5 відображений стан вікна відкритого запиту **Інформація 2**. Відкриття запитів для перегляду виконується у первинному вікні бази даних зазначенням об'єкта **Запросы** і подвійним натисканням ЛК миші на назві запиту.

Після складання усіх шести запитів слід виконати сортування рядків за порядковими номерами.

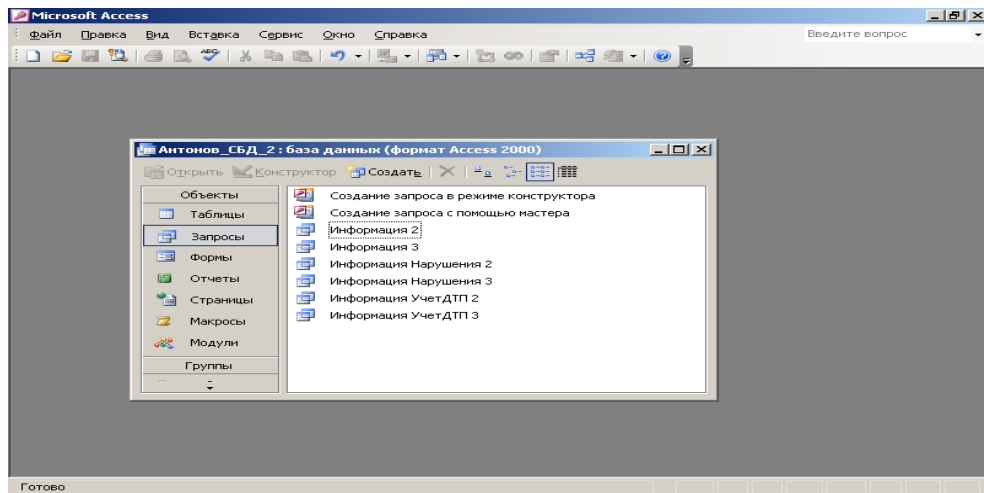


Рис. 8.4. Склад об'єкту **Запросы** у вікні бази даних.

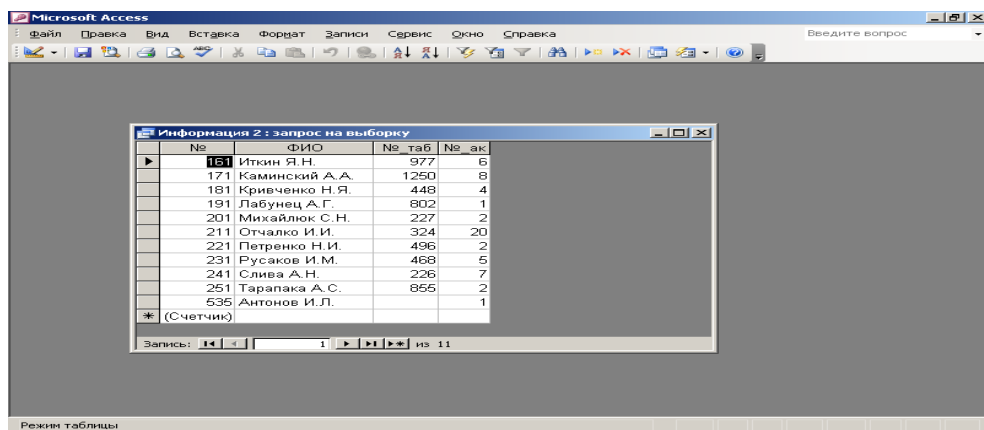


Рис. 8.5. Стан вікна запиту **Інформація 2**.

**Завдання 8.3.** Засвоїти методи створення запитів із умовами вибору.

**8.3.1.** Поняття **запит** можна вважати як деяку команду на вибір, перегляд, зміну, створення або вилучення даних. Якщо база даних добре продумана, то користувачі, що звертаються до бази даних, можуть не мати доступу до таблиць, які є надзвичайно цінним об'єктом; ніхто,

окрім автора або адміністратора бази даних, не має виконувати зміни у таблицях. Тому усі необхідні запити адміністратор повинен приготувати заздалегідь; для одержання результату користувач використовує запити. Найбільш використовуваними є прості запити, які створюються із урахуванням певних умов - умов вибору

**8.3.2.** Вивчити відомості про правила складання умов вибору, наведені у табл. 8.1 і 8.2.

Таблиця 8.1. Приклади шаблонів запису умов для пошуку

Символи, що використовуються	Зразок введення	Результат пошуку у Access
Знак (?) питання; шаблон для будь-якого символу	За?ор	Проміжок; Затор
Зірочка (*); шаблон для будь-якої групи символів	Д*нь	День; Добридень; Довга тіль
Зірочка (*); шаблон для будь-якої групи початкових символів	**й	128й; Останній
Знак фунта (#); шаблон для будь-якої цифри.	#-#-й	5-й; 8-й
Знак (!) оклику після лівої дужки; служить для пошуку символу, який не входить у зазначений набір символів.	Іванов[!ЫМ]	Іванову [але не Івановим]

Таблиця 8.2. Приклади умовних виразів

Умовні вирази	Виведення записів
Харків	Що мають значення Харків
Not Харків	Що не мають значення Харків
In (Харків, Чортків)	Що мають значення Харків, Чортків
<М	Від букви А до букви Л
>=М	Від букви М до букви Я
100	Що мають числове значення 100
<=20	Чисел менше або рівних 20
Date()	Що мають значення поточної дати
>=01.01.94	Що мають значення дати після 01.01.94
Between 01.01.93 AND 31.12.93	Що мають значення роки 1993
*.02.*	Що мають значення місяця Лютий
Null	Що містять у полі порожнє значення
Is Not Null	Що мають не порожнє значення у полі

Припустимо, що необхідно з'ясувати, хто із водіїв працює в автоколонах № 1 і № 2 (запит **Інформація 2**, рис. 8.5). Для цього у вікні бази даних позначається об'єкт **Запросы**, зазначається назва запиту **Інформація 3** і натискається кнопка **Конструктор**. У вікні, що відкрилось, у полі **№\_ак** у рядку **Условие отбора:** вводиться цифра **1** (умова вибору), а у рядку **или** вводиться цифра **2** (умова вибору); натиснути кнопку **Закреть** у заголовку вікна. Далі програма у діалоговому вікні виводить запит про необхідність збереження виконаних змін - відповісти ствердно; виконується повернення у первинне вікно бази даних.

Відкрити запит **Інформація 3** і переконатись у тому, що умова вибору виконана без помилок; закрити запит.

### 8.3. Контрольні запитання

1. Що є об'єктом **Запрос** у СУБД Access?
2. Які типи запитів існують у СУБД Access?
3. Які існують способи створення запитів у СУБД Access?
4. Яким чином проектуються запити у СУБД Access?
5. Для яких об'єктів проектуються запити?
6. Для чого використовується інструмент **Конструктор** при проектуванні запитів?
7. Як використовуються зв'язки між таблицями бази даних при проектуванні та використанні запитів у СУБД Access?
8. Чим є і для чого використовуються умови вибору у базі даних?
9. Які типи умов вибору характерні для СУБД Access?

## Тема 9. ДОСЛІДЖЕННЯ ІНФОРМАЦІЙНИХ МОЖЛИВОСТЕЙ БАЗИ ДАНИХ АТП

**Завдання 9.1.** Засвоїти основи створення звітів у базі даних.

**9.1.1.** Звіти призначені виключно для форматowanego виведення інформації на друкуючий пристрій, тому для орієнтації розташування даних на друкарській сторінці необхідно повідомити програму про тип принтера. В операційній системі Windows має бути встановлений драйвер принтера, інакше на екран виведеться повідомлення про необхідність присутності драйвера принтера і наступні дії із створення звіту припиняться. Для створення звіту потрібно у первинному вікні бази даних позначити об'єкт **Звіти** і виконати одну із двох команд:

**Создание отчета в режиме конструктора**  
або

**Создание отчета с помощью мастера**

Перевага надається другій команді завдяки тому, що вона сприяє стеженню програмою за діями користувача і порадам для наступних дій. Як основу для звіту використовуються таблиці або запити.

**9.1.2.** Виконати команду **Создание отчета с помощью мастера** - виводиться вікно **Создание отчетов** (рис. 9.1).

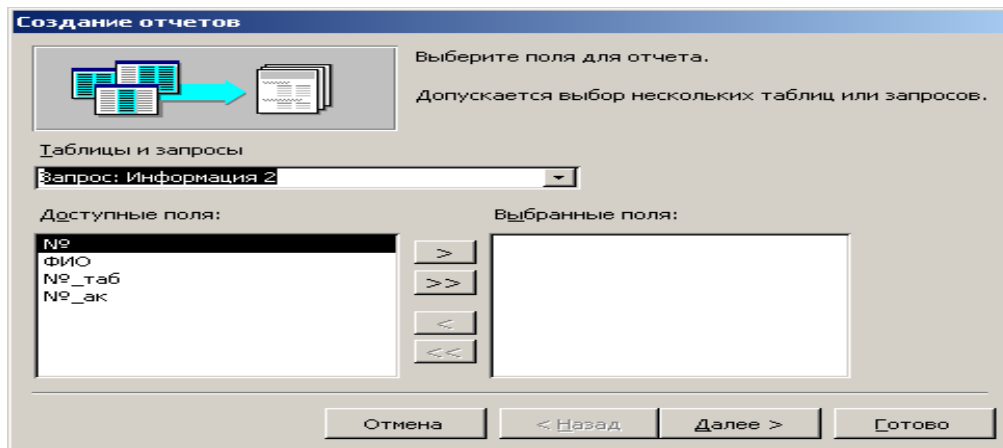


Рис. 9.1. Вікно вибору об'єктів і полів для створення звіту.

Зазначити у вікні **Таблицы и запросы** запит **Информация 2**. Для переведення переліку полів із області **Доступные поля** в область **Выбранные поля** натиснути кнопку [**>>**].

**9.1.3.** Для продовження роботи натиснути кнопку **Далее** - відкривається наступне вікно (рис. 9.2), де пропонується дослідити можливості зазначення рівнів угруповання.

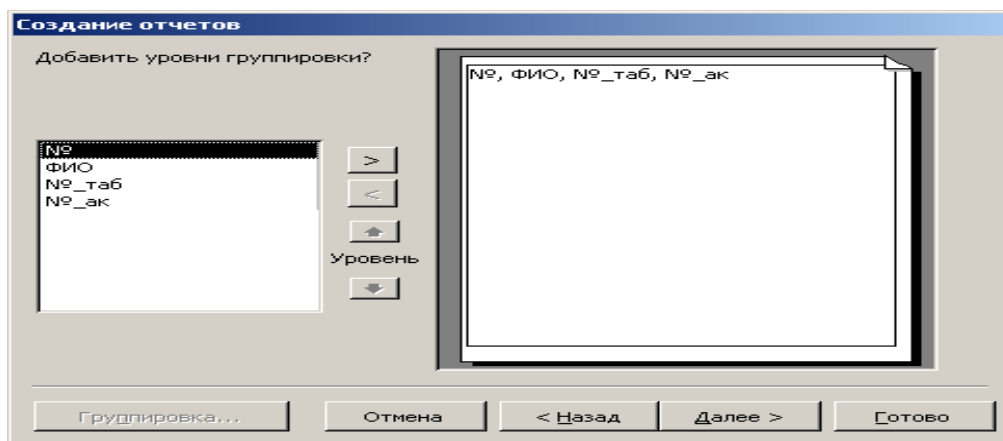


Рис. 9.2. Вікно угруповання записів звіту.

Якщо треба вивести усі дані поля **№\_таб**, то першим слід розташувати саме його (надати перевагу). Ця дія виконується позначенням поля **№\_таб** і наступним натисканням кнопки [**>**]; інші поля групуються усередині цього поля. Після дослідження можливостей цього вікна нати-

снути кнопку [**<**] для повернення вікна до попереднього стану, тому що зазначена операція завданням не передбачена.

**9.1.4.** Натиснути кнопку **Далее** - у вікні **Мастер** пропонує виконати сортування записів у звіті (рис. 9.3).

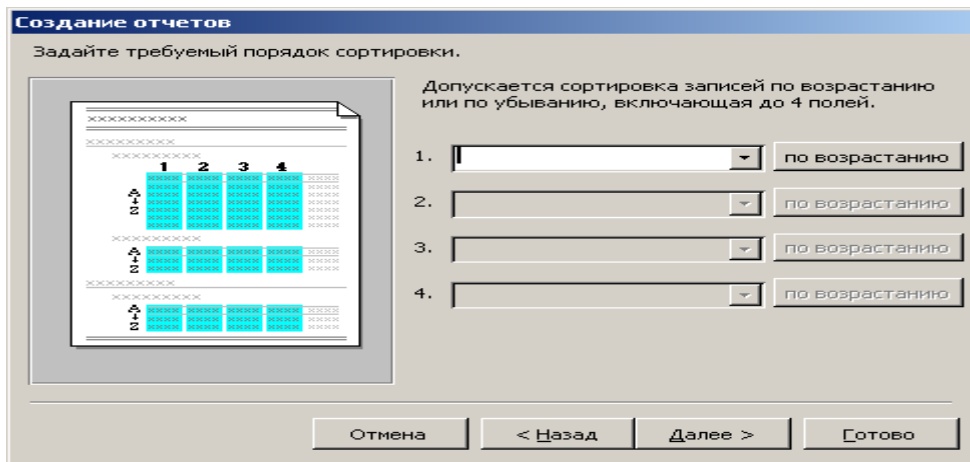


Рис. 9.3. Вікно сортування записів звіту.

Прийняти до уваги, що нові записи у таблиці вводяться в останніх рядках і тому у таблицях, запитах і у звітах записи можуть бути неупорядковані. У цьому вікні передбачена можливість сортування записів для звіту. При виконанні попередніх завдань записи були упорядковані і тому позначення у цьому вікні надмірне.

**9.1.5.** Натискання кнопки **Далее** викликає на екран чергове вікно, в якому користувач **самостійно**, залежно від застосування документа, зазначає вид макету і орієнтацію сторінки (рис. 9.4). Рекомендується в області **Макет** зазначити опцію "табличный", в області **Ориентация** зазначити опцію "книжная" і позначити опцію "Настроить ширину полей для размещения на одной странице".

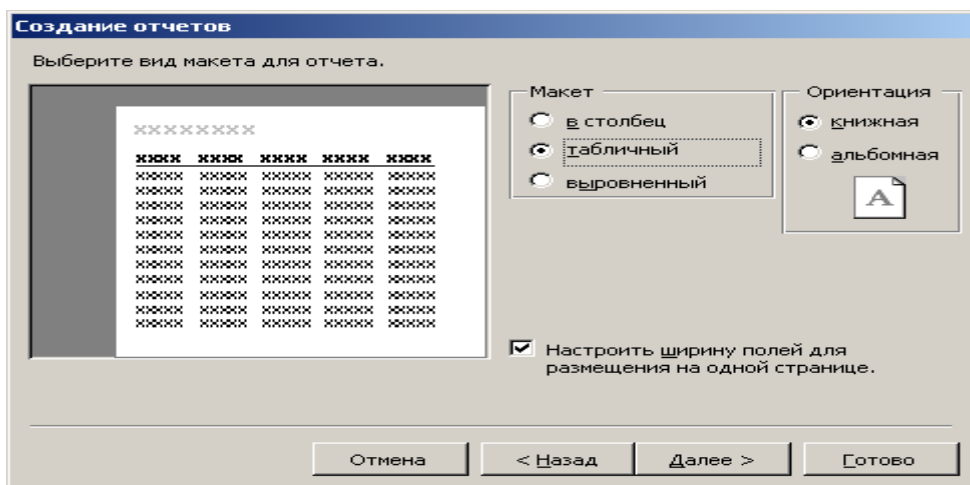


Рис. 9.4. Вікно вибору виду макету звіту.

**9.1.6.** Після чергового натискання **Далее** виводиться вікно для зазначення стилю звіту (рис. 9.5). Цей вибір користувач виконує залежно від призначення звіту.

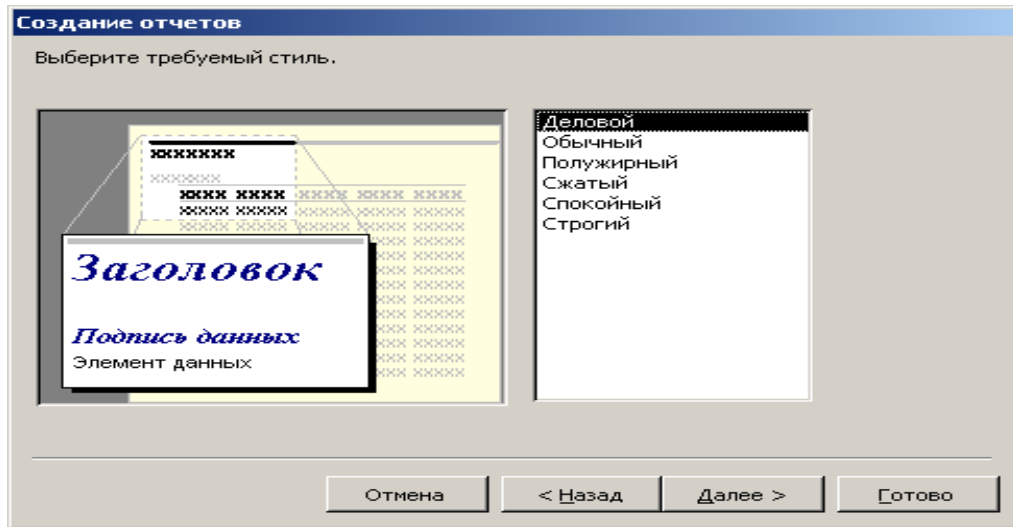


Рис. 9.5. Вікно вибору стилю звіту.

**9.1.7.** Натиснути кнопку **Далее** - на екран виводиться вікно для введення назви звіту. Рекомендується призначити назву звіту відповідно до назви запиту, за яким створений звіт, але із додаванням чергової цифри, наприклад, **Информация 4** (рис 9.6).

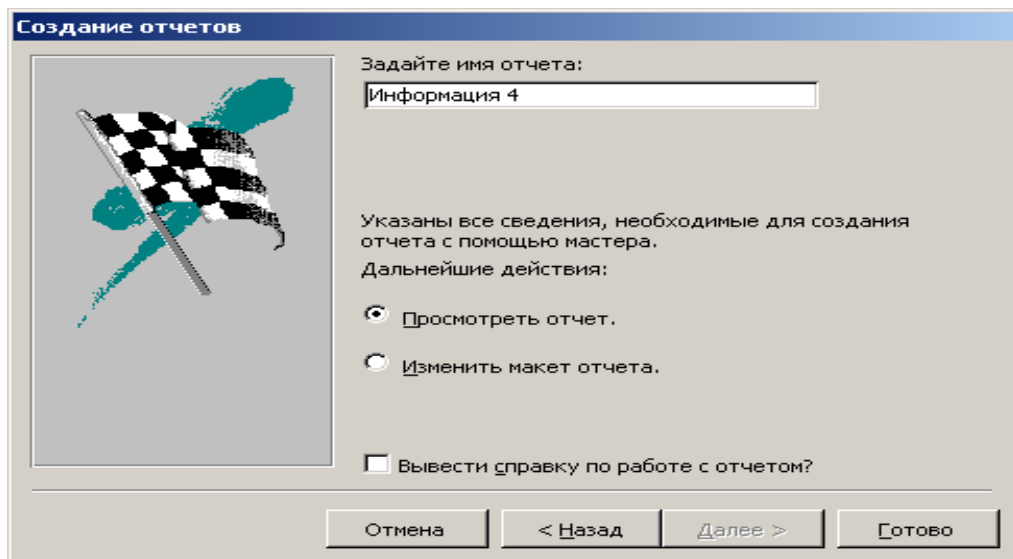


Рис. 9.6. Вікно введення назви звіту.

**9.1.8.** Після призначення назви звіту натиснути кнопку **Готово** - на екран виводиться перша сторінка звіту **Информация 4** (рис. 9.7). Кнопка **Готово** була присутня в усіх попередніх вікнах. Натискання цієї кнопки раніше не перешкоджає збереженню звіту, але звіт матиме певну

кількість недоробок, повернення буде неможливим. Крім того, при передчасному запам'ятовуванні звіту буде призначено назву таблиці або запиту, які були для нього джерелами даних (назва звіту автоматично виноситься на першу сторінку як заголовок). Користувачам пропонується самостійно дослідити призначення кнопок панелі інструментів і вибрати масштаб відображення.

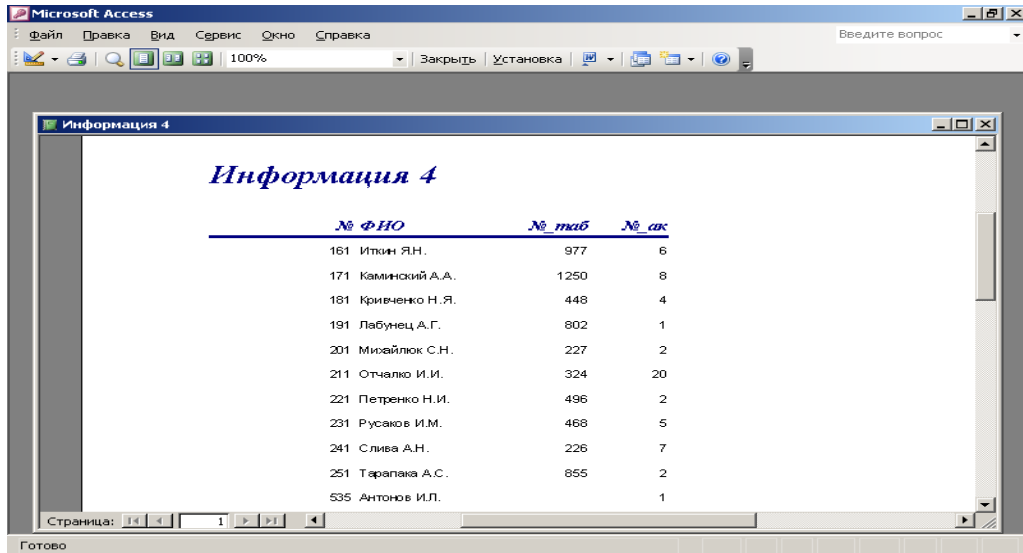


Рис. 9.7. Вікно із сторінкою звіту.

### Завдання 9.2. Дослідити звіт у режимі Конструктор.

Для ознайомлення із структурою звіту відкрити його у режимі Конструктора (рис. 9.8).

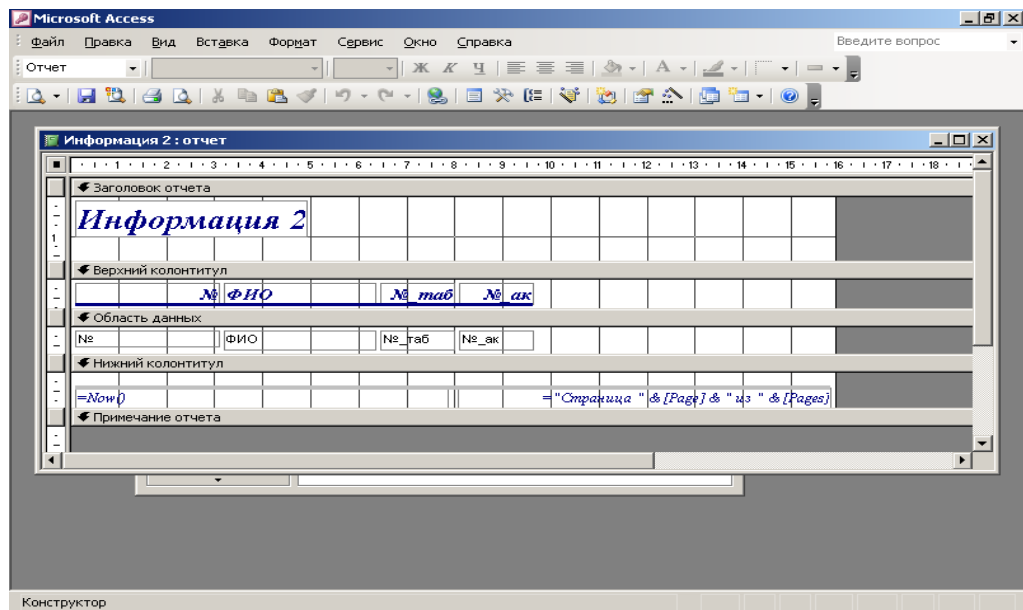


Рис. 9.8. Звіт у режимі Конструктор.

Структура звіту утримує наступні п'ять розділів:

1) **Заголовок** - потрібний для друкування загального заголовку звіту;  
2) **Верхний та Нижний колонтитулы** - використовуються для друкування підзаголовків і номерів сторінок;

3) **Область данных** - містить елементи управління, пов'язані із змістом полів таблиць бази; у ці елементи управління виводяться дані із таблиць для друкування на принтері.

4) у першому елементі управління виводиться **текущая дата**. Для цього використовується вбудована в Access функція **Now( )**. Вона повертає поточну дату і розміщує її у полі, а звіт відтворює її при друкуванні.

5) у другому елементі управління виводиться **номер страницы** і загальна кількість сторінок. Для їх визначення використані вбудовані функції **Page( )** і **Pages( )**. Текст, обмежений лапками, відтворюється відповідно до запису, а оператор **&** застосовується для поєднання тексту у лапках і значень, які відтворюються функціями.

Натиснути кнопку **Закреть** у рядку заголовка сторінки звіту - виконується повернення до первинного вікна бази даних.

**Завдання 9.3.** Скласти звіт для усіх запитів бази даних СБД.

Звіти **Информация 4, Информация 5, Информация Нарушения 4, Информация Нарушения 5, Информация Учет ДТП 4, Информация Учет ДТП 5** створюються аналогічно п.п. 9.2.2 - 9.2.8.

Після завершення створення звітів необхідно повернутись до первинного вікна бази даних.

Переконайтесь, що на екрані знаходиться відображення первинного вікна бази даних і по черзі переглянути стан усіх звітів. Для цього треба позначити назву звіту і натиснути на панелі інструментів вікна бази даних кнопку **Просмотр** - відкривається сторінка зазначеного звіту. Після перегляду закрити сторінку натисканням кнопки **Закреть**.

**Завдання 9.4.** Створити у базі даних кнопку форму.

**9.4.1.** У первинному вікні бази даних позначити об'єкт **Формы** і виконати команду **Создание формы в режиме конструктора** - виводиться порожня форма за назвою **Форма1: форма** і додаткова панель елементів. Якщо додаткова панель відсутня, то слід у вікні **Конструктор** натиснути ПК миші і у спливаючому списку команд виконати команду **Панель элементов**.

За допомогою маркерів розтягнути форму покажчиком миші по ширині до **13** см і по висоті до **9** см

**9.4.2.** На панелі елементів натиснути один раз ЛК миші на кнопці **Аа (Надпись)** і перевести покажчик миші в область даних вікна **Конструктор** - покажчик змінить своє зображення із стрілки на символ **[+<sup>A</sup>]**.



Перемістити покажчик у першу ліворуч клітину першого рядка і натиснути ЛК миші - виводиться тоненька смужка текстового поля. Починаючи від зазначеної позиції, надрукувати у першому рядку назву заголовку форми **База даних СБД ФТС гр. ТД21** (а далі - прізвище користувача, наприклад, **Іванов**). Розташувати текстовий курсор після літер **СБД** і натиснути [**Shift**] + [**Enter**] для того, щоб розділити напис на два рядки (рис. 9.9).

Натиснути [**Enter**] - у результаті на лініях поля виводяться маркери, фон поля зміниться на сірий колір, а під панеллю головних команд середовища Access виводиться панель інструментів **Форматирования**.

Зазначити для надпису розмір шрифту **14**, а для вирівнювання - **По центру**. Для виправлення ширини поля надпису виконати команду **Формат→Размер→По размеру данных**.

Кольори фону і шрифту зазначити самостійно за допомогою інструментів **Цвет заливки** та **Цвет шрифта** панелі **Форматирования**. Користуючись мишею, розтягнути надпис на усю ширину поля, як це зображено на рис. 9.9.

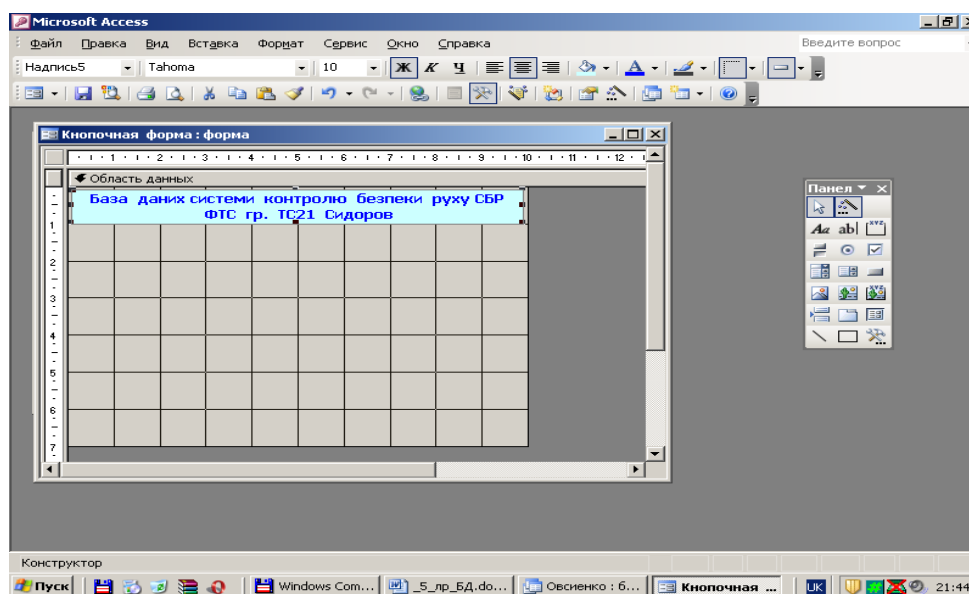


Рис. 9.9. Вікно проектування кнопкової форми у режимі **Конструктор**.

**9.4.3.** На панелі елементів зазначити елемент **Кнопка**, перевести покажчик миші у вікно **Конструктор** і натиснути ЛК миші в області даних на місці розташування кнопки; це може бути будь-яке місце у полі, згодом кнопку можна буде перемістити.

У відповідь виводиться діалогове вікно **Создание кнопок** (рис. 9.10).

У області **Категории:** позначити **Работа с формой**, а області **Действия:** позначити **Открыть форму** і натиснути кнопку **Далее** - відкривається чергове вікно **Создание кнопок** (рис. 9.11).

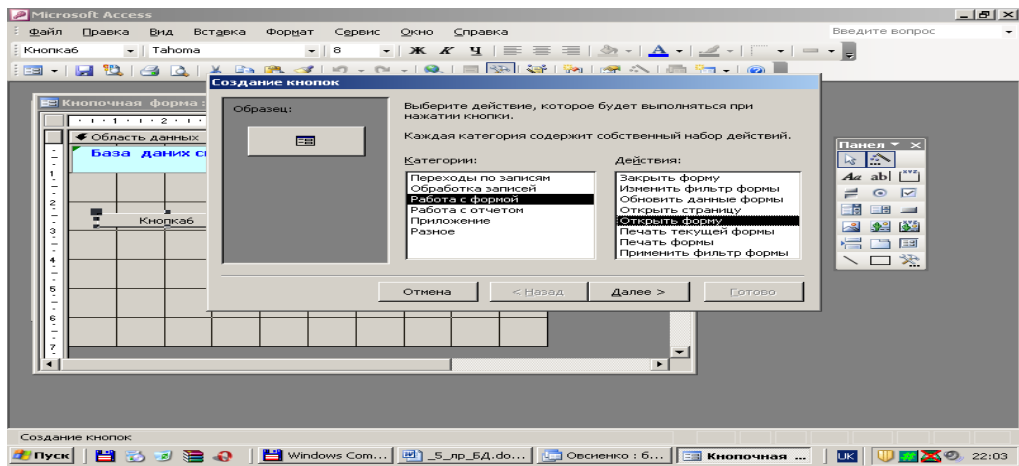


Рис. 9.10. Вікно зазначення категорій та дій при створенні кнопок.

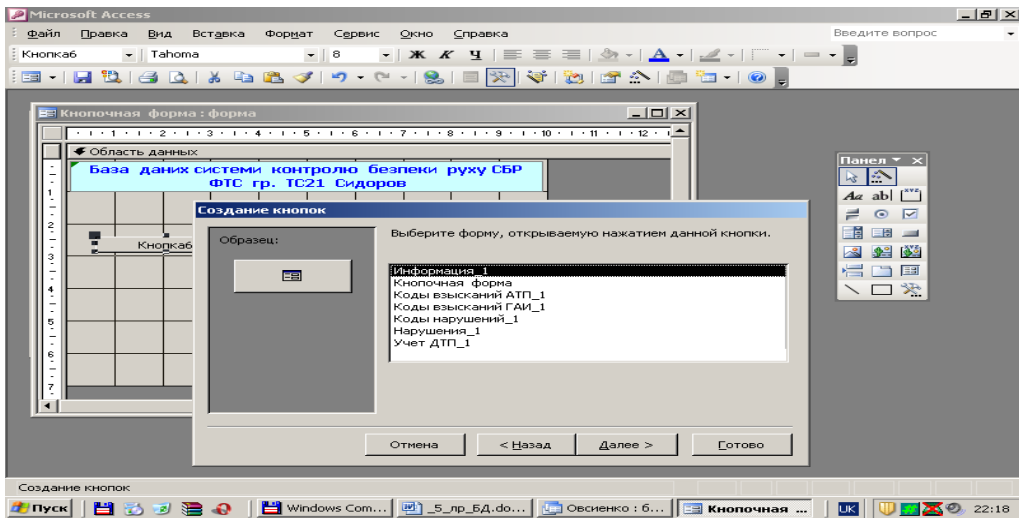


Рис. 9.11. Вікно призначення форми для відкриття за допомогою даної кнопки.

**9.4.4.** Зазначити форму **Информация\_1**, яка відкриватиметься цією кнопкою, і натиснути кнопку **Далее**.

У черговому вікні натиснути кнопку **Далее**. У наступному вікні позначити перемикач **Текст**; вилучити текст **Открыть форму**, ввести назву форми **Информация\_1** (рис. 9.12) і натиснути кнопку **Далее**.

**9.4.5.** У останньому вікні призначається назва кнопки (рис. 9.12). Для цього слід вилучити назву, яку по замовчужанню виводить програма, і надрукувати назву, відповідну формі, для відкриття якої створюється кнопка, а саме - **Информация\_1**.

Завершується проектування натисканням кнопки **Готово** - програма повертається до вікна **Конструктор**, в якому відображена створена кнопка. Розміри кнопки та її розташування можна змінювати за допомогою перетягування маркерів мишею (рис. 9.13), орієнтуючись на лінії сітки.

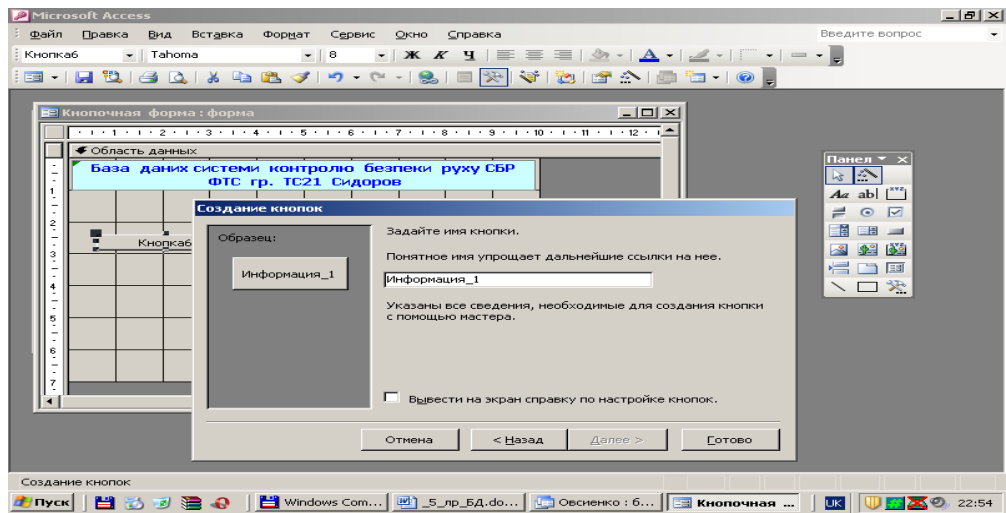


Рис. 9.12. Вікно зазначення назви кнопки для відкриття форми.

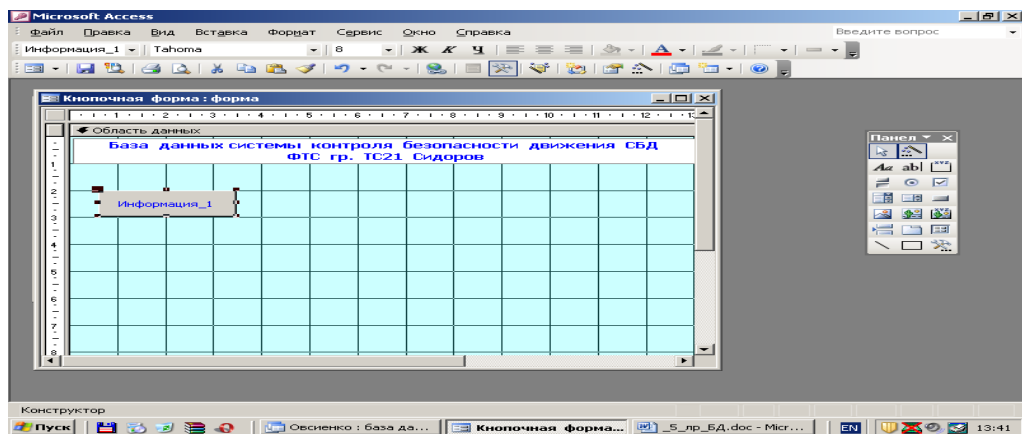


Рис. 9.13. Вікно конструктора із створеною кнопкою.

9.4.6. Аналогічно п.п. 9.4.3 - 9.4.5 створити кнопки для усіх форм СБД і рівномірно розташувати їх у межах поля сітки (рис. 9.14).

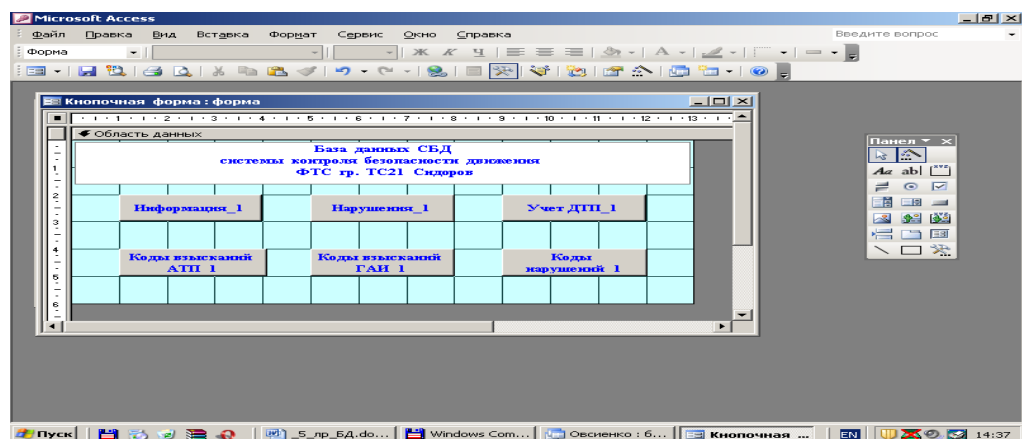


Рис. 9.14. Вікно конструктора із кнопками управління.

Після закінчення створення кнопок повернутись у вікно **Кнопочная форма** (рис. 9.15) за допомогою кнопки **Закреть** у заголовку вікна.

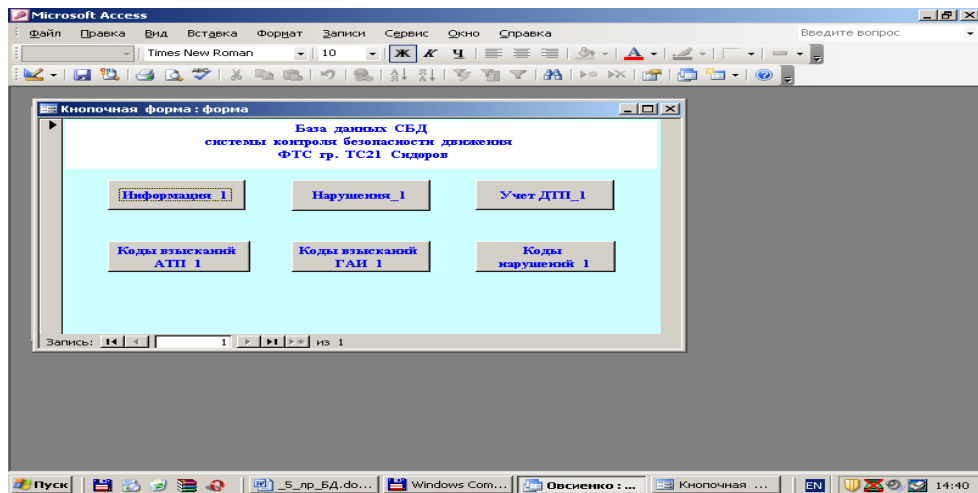


Рис. 9.15. Кнопочная форма для відкриття форм.

За замовчуванням у першій із створених кнопок назва буде оточена крапковою рамкою, що свідчить про активність кнопки.

При натисканні іншої кнопки позначення активності переноситься на назву другої кнопки.

**9.4.7.** Для того, щоб форма із кнопками управління автоматично виводилась на екран при звертанні до бази даних СБД слід у вікні **Кнопочная форма** виконати команду **Сервис→Параметры запуска**. У полі **Вывод формы/страницы** натиснути кнопку списку, позначити **Кнопочная форма** і натиснути кнопку **ОК**. Закрити форму натисканням кнопки **Закреть** у заголовку вікна. Якщо програма виведе форму із питанням про збереження змін у формі, відповісти ствердно.

**9.4.8.** Закрити вікно середовища Access і знову виконати файл бази даних - спочатку відкриється вікно бази даних а через 1-2 секунди на екран виводиться кнопочная форма.

Перевірити дію кнопок управління, а потім закрити вікно середовища Access за допомогою кнопки **Закреть** у рядку заголовка.

## 9.5. Контрольні запитання

1. Що є об'єктом **Отчет** у СУБД Access?
2. Які типи звітів існують у СУБД Access?
3. Які існують способи створення звітів у СУБД Access?
4. Яким чином проектується звіти у режимі Конструктор?
5. Що являє собою **Кнопочная форма**?
6. Яким чином проектується **Кнопочная форма**?

## Список літератури

1. Симбірська Л. М., Симбірський Г. Д., Левтеров А. І. Інформаційні системи і технології: Навчально-методичний посібник для самостійного вивчення. – Харків: ХНАДУ, 2011. – 101 с.
2. Прокофьев В. А. Информационные технологии на транспорте: Учебное пособие / Прокофьев В. А. - СПб: ГМА им. адм. С. О. Макарова, 2006. - 127 с.
3. Лапкина И. А. Информационные системы на транспорте: Учебное пособие / Лапкина И. А., С. П. Онищенко - Одесса: Феникс, 2006. - 196 с.
4. Мак-Федрис Пол Формы, отчеты и запросы Microsoft Access 2003: Пер. с англ. / Мак-Федрис Пол - М.: "Вильямс", 2005. - 416 с.
5. Система управления базами данных Microsoft Access для самостоятельного изучения: Учебное пособие / Составители Н. В. Баловсяк, И. А. Григоришин, Л. В. Кулибаба - К.: Дакор, КНТ, 2006. - 156 с.
6. Автоматизированная система обработки информации и управления на автомобильном транспорте: Учебник / Николаев А. Б, Алексахин С. В., Кузнецов И. А., Строганов Ю. В. - М.: "Академия", 2003. - 224 с.
7. Ярмуш О. В. Інформатика і комп'ютерна техніка: Навчальний посібник / Ярмуш О. В., Редько Г. Г. - К.: Вища освіта. 2006.
8. Бройдо В. П. Вычислительные системы, сети и телекоммуникации: Учебник для ВНЗ. 2-е издание / Бройдо В. П. - СПб.: Питер, 2006. - 360 с.

## Зміст

Вступ.....	3
Тема 1. Дослідження оглядача Internet Explorer (IE).....	4
Тема 2. Аналіз обміну інформацією у комп'ютерних мережах.....	14
Тема 3. Дослідження інформаційних можливостей веб-документів.....	26
Тема 4. Дослідження методів прогнозування транспортних перевезень.....	37
Тема 5. Аналітичне і структурне проектування інформаційної системи бази даних АТП.....	43
Тема 6. Інформаційне наповнення бази даних АТП.....	58
Тема 7. Моделювання інформаційної системи бази даних АТП.....	65
Тема 8. Дослідження інформаційних можливостей бази даних АТП.....	73
Тема 9. Аналіз вихідної інформації бази даних АТП за допомогою звітів.....	79
Список літератури.....	89

Навчальне видання

СИМБІРСЬКА Леся Минівна  
СИМБІРСЬКИЙ Геннадій Дмитрович  
ЛЕВТЕРОВ Андрій Іванович

## ІНФОРМАЦІЙНІ СИСТЕМИ І ТЕХНОЛОГІЇ

*Навчально-методичний посібник  
(лабораторний практикум)*

Відповідальний за випуск О. Я. Ніконов

В авторській редакції

Комп'ютерна верстка М. В. Дурова

Дизайн обкладинки

План 2011 р, поз. 24.

Підп. друку р. Формат 60×84 1/16. Папір офсетний.  
Гарнітура Times New Roman Суг. Віддруковано на ризографі.  
Ум. друк. арк. . Обл.-вид. арк. .  
Зам. . Наклад пр. Ціна договірна.

### **ВИДАВНИЦТВО**

**Харківського національного автомобільно-дорожнього університету**

**Видавництво ХНАДУ, 61002, Харків-МСП, вул. Петровського, 25.  
Тел. /факс: (057)700-38-72; 707-37-03, e-mail: rio@khadi.kharkov.ua**

*Свідоцтво Державного комітету інформаційної політики, телебачення  
та радіомовлення України про внесення суб'єкта видавничої справи  
до Державного реєстру видавців, виготівників і розповсюджувачів  
видавничої продукції, серія ДК № 897 від 17.04 2002 р.*