

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНА МЕТАЛУРГІЙНА АКАДЕМІЯ УКРАЇНИ**

Г. Г. Швачич, В. П. Барвінов, Г. І. Рижанкова В. В. Кузьменко

**ОСНОВИ ТЕХНОЛОГІЇ ПРОЕКТУВАННЯ ТА СТВОРЕННЯ
ІНФОРМАЦІЙНИХ СИСТЕМ**

**ДОВІДКОВЕ КЕРІВНИЦТВО ПО ПРОЕКТУВАННЮ ТА СТВОРЕННЮ
ІНФОРМАЦІЙНИХ СИСТЕМ З ВПРАВАМИ**

Затверджено на засіданні Вченої ради академії
як навчальний посібник

Дніпропетровськ НМетАУ 2006

УДК 004(075.8)

Швачич Г. Г., Барвінок В. П., Рижанкова Г. І., Кузьменко В. В. Основи технології проектування та створення інформаційних систем: Навчальний посібник. – Дніпропетровськ: НМетАУ, 2006. – 47с.

Викладені основні поняття стосовно проектування та створення інформаційних систем в середовищі MS Access. Призначений для студентів усіх спеціальностей які проходять елективну підготовку.

Іл. 6. Табл.1 Бібліогр.: 9 найм.

Відповідальний за випуск Г. Г. Швачич, канд. техн. наук, проф.

Рецензенти: Д. Г. Зеленцов, канд. техн. наук, доц. (Український Державний хіміко-технологічний університет)

Б. І. Мороз доктор техн. наук, проф. (Академія митної служби України)

© Національна металургійна академія
України, 2006

ТЕОРІЯ СТВОРЕННЯ ІНФОРМАЦІЙНИХ СИСТЕМ

ТЕМА1 МОДЕЛЮВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

Етапи моделювання інформаційної системи (ІС) включають:

1. Створення концептуальної моделі конкретної предметної області.
2. Створення логічної моделі даних.
3. Створення фізичної моделі даних, котра представляє собою базу даних.

Проектування бази даних починається з окреслення предметної області, яка цікавить дослідника, і побудови концептуальної моделі, що відбиває взаємозв'язки між реальними об'єктами й процесами в цій предметній області.

Інформація про предметну область може включати відомості про персонал, заробітну плату, про навчальні дисципліни, результати сесії, фінансові угоди, історії хвороб і т.д.

Концептуальна модель потім перетворюється в модель даних. Версія концептуальної моделі, котрою може бути забезпечена конкретна СУБД, називається логічною моделлю.

Логічна модель відбиває логічні зв'язки між елементами даних, поза залежністю від їх змісту в середовищі зберігання. Логічна модель відображається у фізичну, з урахуванням типів даних, які підтримуються конкретною СУБД (FoxPro, MS Access, Oracle і т.д.).

Фізична модель визначає розміщення даних, методи доступу й техніку індексування. Вона прив'язана до конкретики реалізації й називається внутрішньою моделлю системи – тобто базою даних.

Аналіз предметної області й створення концептуальної моделі

Розглянемо предметну область, пов'язану з контролем навчання. Сформулюємо постановку завдання. Приведемо спрощену схему взаємозв'язків компонентів цього процесу.

Як приклад предметної області розглянемо роботу деканату, який формує процес організації занять по ряду дисциплін, а потім організує контроль успішності студентів.

При аналізі предметної області необхідно:

1. Виявити найбільш важливі для даної предметної області компоненти. У нашому випадку це будуть об'єкти:
 - викладачі;
 - студенти;
 - дисципліни;
 - оцінна відомість.
2. Одержати список основних процесів навчання й контролю. Для нашого прикладу в такий список можна включити наступні пункти:
 - викладання;
 - вивчення студентами дисциплін;
 - контроль навчання.
3. Вирішити проблему телекомунікацій й організації спільної роботи персоналу. У розглянутому прикладі допустимо, що всі операції виконуються в межах одного будинку, а організація спільного використання даних заснована на можливостях локальної мережі й сервера БД.
4. Визначити організаційну структуру підрозділу.
5. Визначити періодичність здійснюваних процесів контролю, що дозволить правильно сформулювати процеси обробки даних у майбутній прикладній програмі. Прийmemo таку тимчасову послідовність виконуваних процесів: підсумковий контроль – один раз наприкінці семестру.
6. Визначити мотивацію деканату:
 - забезпечення умов для успішної діяльності персоналу;
 - одержання студентами необхідного рівня знань;
 - підвищення якості навчання.

Формалізуємо опис поставленого завдання.

Відповіді на шість пунктів дозволяють підійти до побудови концептуальної моделі, яка представлена у вигляді взаємозв'язків між компонентами й процесами. У практиці проектування інформаційних систем такі схеми одержали назву ER-діаграм (Entity-relationship diagram (ERD) – діаграма «Сутність-зв'язок»).

Перелік уводить таку інформацію:

- прізвище, ім'я, по батькові студента;

- номер групи;
- дата народження;
- оцінки;
- назва дисципліни;
- прізвище, ім'я, по батькові викладача;
- дата народження;
- посада;
- телефон;
- заробітна плата.

ТЕМА 2 СТВОРЕННЯ ІНФОРМАЦІЙНО-ЛОГІЧНОЇ МОДЕЛІ

Визначення об'єктів (сутностей)

Для створення логічної моделі проаналізуємо перелік уведеної інформації:

1. Які нам знадобляться об'єкти.
2. Як розподіляться атрибути цих об'єктів.

Об'єктом називається елемент інформаційної системи. У реляційній теорії баз даних об'єкт називається сутністю.

Об'єкт може бути реальним (наприклад, людина, який-небудь предмет або населений пункт) і абстрактним (наприклад, подія або досліджуваний студентами курс). Кожен об'єкт має певний набір властивостей, але в базі даних інформаційної системи запам'ятовуються тільки ті властивості, які використовуються для видачі звітів.

Виходячи з поставленого завдання, виділимо наступні сутності:

- Співробітники.
- Студенти.
- Дисципліни.
- Оцінки.

Об'єкти в реляційній моделі даних представляються відносинами, які відображені у вигляді таблиць. У стовпцях таблиць розміщуються властивості об'єктів.

Атрибут – це інформаційне відображення властивості об'єкта. Кожен об'єкт характеризується рядом основних атрибутів. Кожен атрибут у моделі повинен мати унікальне ім'я – ідентифікатор. Атрибут часто називають елементом даних, полем даних або просто полем.

У рядках таблиць розташовані записи. Запис даних – це сукупність значень елементів даних. Кожен запис однієї таблиці складається з однакового числа полів. Конкретне поле кожного запису однієї таблиці може містити дані тільки одного типу.

Інформацію про деяку предметну область можна представити за допомогою ряду об'єктів, кожний з яких описується декількома елементами даних.

Між атрибутами сутностей існує зв'язок – свого роду функціональна залежність, що дозволяє бачити багатокomпонентну інформаційну систему як єдине ціле. Для відбиття й організації таких зв'язків між таблицями використовується поняття ключового елемента.

Якщо відоме значення, яке приймає ключовий елемент, ми можемо ідентифікувати значення, що приймають інші елементи даних цього ж об'єкта. Атрибут (або група атрибутів), які єдиним образом ідентифікують кожен рядок у таблиці, називається первинним ключем.

Альтернативний (вторинний) ключ – це атрибут (або група атрибутів), незбійний з первинним ключем й унікально ідентифікуючий екземпляр об'єкта.

Наприклад, для об'єкта «Студент», що має атрибути «Код студента», «Прізвище», «Ім'я» й «По батькові», група атрибутів «Прізвище», «Ім'я», «По батькові» може бути альтернативним ключем стосовно атрибута «Код студента» (у припущенні, що в групі не вчать повні тезки).

Для підтримки цілісності даних використовується механізм зовнішніх ключів. Зміст цього механізму полягає в тому, що деякому атрибуту (або групі атрибутів) одного відношення призначається посилання на первинний ключ іншого відношення. Тим самим закріплюються зв'язки підпорядкованості між цими відносинами. При цьому відношення, на первинний ключ якого посилається зовнішній ключ іншого відношення, називається master-відношенням, або головним відношенням; а відношення, від якого виходить посилання, називається detail-відношенням, або підлеглим відношенням. Після призначення такого посилання СУБД має можливість ав-

томатично відслідковувати питання «не порушення» зв'язків між відносинами. СУБД згенерує помилку:

- При спробі вставити в підлеглу таблицю запис, для зовнішнього ключа якої не існує відповідності в головній таблиці (наприклад, там немає ще запису з таким первинним ключем).
- При спробі видалити з головної таблиці запис, на первинний ключ якої є хоча б одне посилання з підлеглої таблиці.
- При спробі змінити первинний ключ запису головної таблиці, на яку є хоча б одне посилання з підлеглої таблиці.

Підтримка несуперечності функціональних залежностей між сутностями називається посилальною цілісністю.

Посилальна цілісність – це забезпечення відповідності значення зовнішнього ключа екземпляра дочірньої сутності значенням первинного ключа в батьківській сутності. Посилальна цілісність може контролюватися при всіх операціях, що змінює дані.

Визначення взаємозв'язків між сутностями

Розрізняють взаємозв'язку типу «один до одного», «один до багатьох» й «багато до багатьох».

Взаємозв'язок «один до одного» (між двома типами об'єктів)

Зв'язок «один до одного» являє собою найпростіший вид зв'язку даних, коли первинний ключ таблиці є в той же час зовнішнім ключем. Такий зв'язок зручно встановлювати тоді, коли не вигідно тримати різні по розміру (або за іншими критеріями) дані в одній таблиці. Наприклад, можна виділити дані з докладним описом про вироби в окрему таблицю із установами зв'язку «один до одного».

Припустимо, що один предмет читає тільки один викладач. У цьому випадку між об'єктами «Викладач» і «Дисципліна» встановлюється взаємозв'язок «один до одного». Якщо один предмет читає три викладача, то встановлюється зв'язок один - до - багатьох.

Студент вивчає кілька дисциплін, і кожна дисципліна вивчається безліччю студентів; це породжує зв'язок багато - до - багатьох.

Взаємозв'язок «один до багатьох» (між двома типами об'єктів)

Зв'язок «один до багатьох» у більшості випадків відбиває реальний взаємозв'язок сутностей у предметній області. Він реалізується парою «зовнішній ключ – первинний ключ», тобто коли визначений зовнішній ключ посилається на первинний ключ іншої таблиці.

Саме цей зв'язок описує широко розповсюджений механізм класифікаторів. Нехай є довідкова таблиця, що містить назви, імена і якісь коди, причому, первинним ключем є код. Таблицю, що збирає інформацію, назовемо інформаційною таблицею. Вона визначається зовнішнім ключем, котрий посилається на первинний ключ класифікатора. Після цього в таблицю заноситься не назва класифікатора, а код. Така система стає стійкою від зміни назви в класифікаторах. Є способи швидкої «підміни» у відображуваній таблиці кодів на їхні назви.

У цьому випадку одного запису даних першого об'єкта (його часто називають батьківським або основним) буде відповідати кілька записів другого об'єкта (дочірнього або підлеглого). Взаємозв'язок «один до багатьох» дуже поширено при розробці реляційних баз даних. Як батьківський об'єкт часто виступає довідник, а в дочірньому зберігаються унікальні ключі для доступу до записів довідника.

У нашому прикладі як такий довідник можна представити об'єкт «Дисципліни», у якому зберігаються відомості про всі дисципліни. При звертанні до запису для певної дисципліни, нам доступний список всіх викладачів, які читають цей предмет, і відомості, про які зберігаються в об'єкті «Викладачі».

Взаємозв'язок «багато до багатьох»

Зв'язок «багато до багатьох» у явному виді в реляційних базах даних не підтримується. Однак є ряд способів непрямой реалізації такого зв'язку. Один з найпоширеніших способів полягає у введенні додаткової таблиці, рядки якої складаються із зовнішніх ключів, що посилаються на первинні ключі двох таблиць.

У реляційних СУБД для контролю цілісності даних з можливістю каскадного відновлення й видалення даних необхідно створити допоміжний

об'єкт зв'язку, що складається із ключових реквізитів об'єктів, що зв'язують, і який може бути доповнений описовими реквізитами. У нашому випадку таким новим об'єктом для зв'язку служить об'єкт Оцінки, реквізитами якого є код студента, код дисципліни й оцінки. Кожен студент має оцінки по декількох дисциплінах, тому зв'язок між об'єктами Студенти й Оцінки буде Один-до-багатьох. Кожну дисципліну здає безліч студентів, тому зв'язок між об'єктами Дисципліни й Оцінки також буде Один-до-багатьох. У результаті отримуємо наступну інформаційно-логічну модель бази даних

ТЕМА 3 ПРИВЕДЕННЯ МОДЕЛІ ДО НЕОБХІДНОГО РІВНЯ НОРМАЛЬНОЇ ФОРМИ

Приведення моделі до необхідного рівня нормальної форми є основою побудови оптимальної структури таблиць і зв'язків у реляційній базі даних.

Введення нормалізації при розробці інформаційної моделі забезпечує її мінімальний обсяг і максимальну швидкодію. Нормалізація дозволяє розбити цілісний інформаційний об'єкт на більш дрібні логічні одиниці.

Перший етап нормалізації полягає, по-перше, в утворенні двовимірної таблиці, яка містить всі необхідні атрибути інформаційної моделі, по-друге – у виділенні ключових атрибутів.

Приведення об'єкта бази даних (таблиці) до другої нормальної форми полягає в розбивці однієї таблиці з зайвою інформацією на ряд таблиць, у яких всі атрибути мають повну функціональну залежність тільки від ключа власної таблиці. При приведенні моделі до другої нормальної форми виключається дублювання даних.

Відношення в інформаційному об'єкті задане в третій нормальній формі, якщо кожен атрибут цього відношення не є первинним і не залежить транзитивно від кожного можливого ключа цього відношення.

Транзитивна залежність виявляє дублювання даних в одному відношенні. Якщо А, В і С – три атрибути одного відношення й С залежить від В, і С від А, то говорять, що С транзитивно залежить від А.

Сформулюємо основні правила, якими потрібно користуватися при проектуванні бази даних:

- Виключайте повторювані групи – для кожного набору зв'язаних атрибутів створіть окрему таблицю й забезпечте її первинним ключем. Виконання цього правила автоматично приведе до другої нормальної форми.
- Якщо в списку дисциплін указати прізвища викладачів й у написанні прізвищ виникне помилка, то для СУБД це вже інші викладачі. Необхідно вести список викладачів в окремій таблиці, а в списку дисциплін використати тільки привласнені їм унікальні ідентифікатори.
- Виключайте зайві дані: якщо атрибут залежить тільки від частини складеного ключа, перемістіть атрибут в окрему таблицю. Це правило допомагає уникнути втрати одних даних при видаленні якихось інших. Скрізь, де можливе використання ідентифікаторів замість опису, виносьте в окрему таблицю список ідентифікаторів з поясненнями до них.
- Виключайте стовпці, які не залежать від ключа: якщо атрибути не вносять свою лепту в опис ключа, перемістіть їх в окрему таблицю.

Зміни в моделі пов'язані із введенням штучних атрибутів, які у вигляді кодів беруть участь у відносинах замість природних атрибутів. До необхідності введення в модель штучних атрибутів приходимо в процесі нормалізації.

Аналіз розроблювальної моделі показує, що вона відповідає необхідному рівню нормальної форми.

Перехід до фізичної моделі

Фізична модель є проектом з таблиць, створених на основі логічної схеми даних з урахуванням їх типів для обраної СУБД. Тип даних адекватний поняттю типів даних у мовах програмування й характеризує кожен елемент запису. Кожна таблиця проекту для фізичної моделі повинна включати:

- Імена полів.
- Характеристику (призначення) полів.
- Тип даних.
- Розмір.

Ці характеристики, як обов'язкові, задаються для кожного поля таблиць СУБД Access у режимі конструктор.

ТЕМА 4 БАЗИ ДАНИХ В MS ACCESS

Інструментальні компоненти

Вікно бази даних – це контейнер, що містить всі об'єкти, які формують базу даних: таблиці, запити, форми, звіти, макроси, модулі.

Відомості про кожне джерело зберігаються в окремій таблиці. При роботі з декількома таблицями між ними встановлюються зв'язки.

Для перегляду, введення або зміни даних прямо в таблиці застосовуються форми. Форма дозволяє відображати дані, використовуючи стандартний або користувальницький макет.

Для відбору й пошуку даних, що задовольняють певним умовам, створюється запит. Запит дозволяє оновити або видалити кілька записів, виконати вбудовані або спеціальні обчислення.

Для аналізу й роздруківки даних використовується звіт. Він дозволяє групувати дані й обчислювати підсумки.

Для виконання різних дій із цими об'єктами можна використати кнопки **Створити**, **Відкрити**, **Конструктор**, які перебувають із правої сторони вікна бази даних:

- Створити – створюється новий об'єкт відображуваного типу.
- Відкрити – відкривається виділений об'єкт для перегляду й введення даних.
- Конструктор – дозволяє змінити структуру виділеного об'єкта.

Особливості вікна Access

- Маса операцій виконується з вікна бази даних; система використовує меню набагато рідше, ніж інші додатки Microsoft Office.
- Закладки: *Звіти*, *Макроси*, *Модулі* мають замість кнопки **Відкрити** інші кнопки, але кнопки **Створити** й **Конструктор** у них ті ж самі.
- Після відкриття вікна для відображення об'єкта можна перемкнутися між режимами відображення, використовуючи інструмент або меню **Вид**.

Загальні відомості про таблиці

Таблиці – головні об'єкти в Access. Кожна таблиця зберігає відомості по конкретному питанню.

Кожен об'єкт БД повинен бути спочатку сконструйований, а потім заповнений. До використання таблиці необхідно створити її структуру, указавши список полів й їх тип. Це можна зробити в режимі **Конструктор**, який слід включити, використовуючи інструмент **Вид** у лівій частині панелі інструментів.

Настроювання таблиці в режимі конструктор включають:

- Додавання, переміщення й перейменування поля.
- Визначення типу даних.
- Установку властивостей полів.
- Визначення ключового поля.

Робота з таблицею в режимі таблиці включає:

- Редагування, перевірку орфографії, печатку.
- Додавання й видалення записів.
- Сортування, відбір і пошук записів.
- Зміну макета таблиці (розмірів рядків і полів, кольори тла, сітки, символів і т.д.).
- Вставку, видалення й перейменування стовпців.

Використання Запиту.

Запити дозволяють формувати вибірки з таблиць, визначаючи поля й записи, виведені на екран, поєднувати кілька таблиць у базах даних, модифікувати дані в таблицях. Найпростішим і важливим типом запиту є запит на вибірку, що дозволяє вказати:

- Які поля, яких таблиць й у якому порядку повинні відобразитися, і тим самим, визначити структуру майбутніх записів.
- Ввести критерій для відбору відображуваних записів.
- Визначити порядок сортування цих записів.

Як і вікно «Таблиця», вікно «Запит» може відобразитися як у режимі таблиці, так й у режимі конструктора. Панель інструментів у вікні «Запит» включає також кнопку **Режим SQL**, що дозволяє відобразити лежачий в основі запиту код програми.

Для кожного поля можуть бути задані умови відбору, що дозволяють визначити, які записи із загального набору запиту будуть відобразитися.

Можна задати опцію вибору для відображуваних у запиті полів, які будуть виведені на екран при перегляді запиту у вигляді таблиці. Запити вибірки можна використати замість таблиць при створенні форм і звітів.

Використання форми

Форми дозволяють управляти процесом відображення даних на екрані. Формат таблиць дозволяє бачити безліч записів одночасно, але для довгих записів відображення всієї інформації звичайно неможливо.

Форма, як правило, відображає один запис одночасно. У формі можна розташувати додаткові інформаційні елементи, що роблять її привабливою й функціональною – наприклад, фотографії, картинки, елементи керування у вигляді кнопок і т.д.

Форми використовуються для різних цілей:

- Для уведення даних у таблицю.
- Для відкриття інших форм і звітів (кнопкова форма).
- Для вибору, попереднього перегляду й печатки звіту (спеціальне вікно діалогу).

Інструмент **Вид**, розташований у лівій частині панелі інструментів, дозволяє задати один із трьох режимів функціонування: режим форми, таблиці, конструктора.

Вікно бази даних у режимі форми представляється макетом, що містить:

- Текст, що пояснює, для полів.
- Графічні елементи (лінії, кнопки).
- Обчислені значення.
- Поля, дані для яких беруться з полів базової таблиці.

Для створення зв'язку між формою й записами таблиці, що є джерелом даних для форми, використовуються графічні елементи, називані елементами керування. Найчастіше для відображення й уведення даних використовуються поля.

Звіти

Звіти призначені спеціально для печатки даних. Тому при відображенні звіту замість кнопки вікна бази даних **Відкрити** використовується кнопка **Перегляд**. Джерелом даних для звіту може служити базова табли-

ця, запит або інструкції на SQL. Для створення зв'язку між звітом і його вихідними даними застосовуються елементи керування: поля (утримуючі імена або числа), напису для заголовків, декоративні лінії для графічного оформлення звіту. Вікно звіту в режимі конструктора дуже схоже на вікно форми в цьому ж режимі.

Дані звіту не редагуються, тому що це робиться в таблицях, запитах або формах. Вікно попереднього перегляду відображає дані так, як вони будуть виглядати при печатці.

Звіти конструюються з використанням «Майстра звітів». Робота типового майстра являє собою серію кроків, постачених інструкціями й елементами вибору, що дозволяють задати об'єктам певні характеристики й оцінити візуально в малих вікнах зроблений вибір. Кожен крок представляється черговим вікном діалогу, що містить навігаційні кнопки – **Назад**, **Далі (Готове)**, **Скасування**.

ТЕМА 5 СТВОРЕННЯ БАЗИ ДАНИХ В MS ACCESS

Створити базу можна двома різними способами:

- Створити порожню базу даних, а потім таблиці й інші необхідні об'єкти.
- Використати один із шаблонів бази даних, і, скориставшись майстром баз даних, зробити його тим, що відповідають пропонованим вимогам.

Якщо вибрати **Створити** базу даних з меню **Файл**, виникне вікно діалогу «Створення» із двома закладками: *Загальні* й *Бази даних*. Закладка *Загальні* має значок "Нова база даних"; закладка *Бази даних* дозволяє вибрати один із шаблонів, розроблених для спеціальних цілей й утримуючого особливого виду таблиці, форми й звіти (Основні фонди, Замовлення, Особисте майно й т.д.).

При створенні порожньої бази даних Access виводить вікно «Файл нової бази даних» для визначення папки й імені нової бази даних. Задавши папку й ім'я файлу, клацнемо на кнопці **Створити**. Access автоматично додасть до імені файлу розширення *.mdb*. З'явиться вікно з новою, незаповненою ба-

зою даних. Якщо вибрати шаблон, Access відобразить вікно Майстра баз даних, що допоможе пристосувати цей шаблон для цілей користувача.

Створення таблиці

Створене вікно нової порожньої бази даних має за замовчуванням у якості обраної закладки *Таблиці*. Якщо клацнути на кнопці **Створити**, Access відобразить вікно діалогу «Нова таблиця» з вибором ряду режимів створення нової таблиці: **Режим таблиці, Конструктор, Майстер таблиць, Імпорт таблиць, Зв'язок з таблицями**.

Режим Таблиці створює нову таблицю й відображає її в режимі таблиці, так що можна відразу ж вводити дані. Знову створена таблиця має 20 полів з іменами «Поле 1»... «Поле 20». Можна перейменувати поля, двічі клацнувши по заголовках стовпців і відредагувавши їхні імена. Потім увести в таблицю дані. При першому збереженні таблиці Access визначить тип даних у кожному полі, базуючись на уведеній інформації. Можна потім змінити опис цих полів.

Іноді буває зручно створювати таблицю в *Режимі таблиці*, якщо треба швидко ввести дані. Але надалі можуть виникнути проблеми. Краще розробити структуру таблиці в режимі Конструктора до уведення даних, щоб бути впевненим у складі полів і типі даних для кожного поля.

Майстер таблиць

Дозволяє здійснити вибір з певної кількості вже готових таблиць, після чого можна зробити необхідні налаштування. Кожна з таблиць призначена для конкретного виду даних.

Перший крок майстра дозволяє зробити вибір зразка таблиці й визначити, які поля будуть входити в створювану таблицю.

Другий крок майстра дозволяє ввести ім'я таблиці й вибрати спосіб створення ключового поля. Третій крок – крок створення самої таблиці. Можна відобразити її в режимі конструктора й модифікувати її, або ж відразу ввести в неї дані.

Вікно таблиці в режимі конструктора

При конструюванні таблиці задаються імена полів, а також тип даних для кожного із цих полів; для деяких типів полів необхідно задати розмір. Такий процес конструювання таблиці називають визначенням структури або розробкою таблиці.

Вікно таблиці в режимі конструктора містить три стовпчики: «Ім'я поля», «Тип даних», «Опис». У режимі конструктора кожне поле таблиці відображається у вигляді ряду (рядка). Переміщатися між стовпцями «Ім'я поля», «Тип даних», «Опис» можна за допомогою миші, клавішею горизонтального переміщення курсору, клавіші Tab.

Ім'я поля вводиться в стовпці «Ім'я поля»; якщо перемістити курсор у стовпець «Тип даних», Access розгорне поле, що розкривається, списку зі списком можливих типів даних (необхідно вибрати тип даних для кожного поля таблиці). Використання стовпця «Опис» не обов'язково; якщо ввести деяку інформацію в цей стовпець, то вона відобразиться як рядок довідки в рядку стану, якщо поле, що має опис, виділити в таблиці або формі. Можна в стовпець «Опис» уводити довідкову інформацію, корисну під час уведення даних.

Нижня секція вікна використовується для завдання властивостей зазначеного поля.

Панель у лівій нижній частині вікна містить дві вкладки: *Загальні* (Розмір поля, Формат поля, Маска уведення, Початкове значення й т.д.) і *Підстановка* (Тип елемента керування, Тип джерела рядків, Джерело рядків і т.д.).

Панель у правій нижній частині вікна використовується для висновку короткої контекстно залежної довідки по обраній характеристиці властивості поля.

Типи даних у структурі таблиці

Для визначення структури таблиці необхідно, щонайменше, увести ім'я й тип даних для всіх полів.

Типи даних:

- Текстовий (до 255 символів).
- МЕМО (до 65000 символів). Поле з даними типу МЕМО має змінну довжину, і розмір його не задається.

- Числовий (тип чисел і точність обчислень залежать від заданого розміру числового поля – довге ціле, ціле, із плаваючою крапкою, байт й ін.).
- Дата/час (залежно від формату, привласненого полю, у поле вводяться календарні дані або значення часу).
- Грошовий (використається в грошових розрахунках; обчислювальна точність – 4 знаки після коми; можуть використатися з позначенням грошової одиниці або без).
- Лічильник (послідовний набір натуральних чисел; уведений лічильником числа не можна змінити).
- Логічний (Yes/No). Визначається одним із двох значень залежно від формату, заданого полю в панелі *Властивості поля*.
- Поле об'єкта OLE. Містить дані з інших додатків.
- Гіперпосилання (містить адресу об'єкта, документа або Web-сторінки);
- майстер підстановок (створює поле, за допомогою якого можуть бути обрані значення зі списку).

За замовчуванням довжина даних текстового поля становить 50 символів.

Для цілочисленних полів найчастіше вибирають як розмір *Довге ціле*; тип *Із плаваючою крапкою* – для речовинних змінних або случаючи, коли обчислення повинні бути точними.

Якщо достатня точність – 4 знаки, то можна використати тип *Грошовий*, навіть якщо значення не є грошовими сумами (розрахунок для цього типу йде з фіксованої коми, а не із плаваючої, і тому виробляється швидше). Можна використати список, що розкривається, *властивості Формат* поля для визначення, яким образом на екрані будуть представлені ці поля - зі знаком грошової одиниці або без нього.

Поля первинного ключа

Ключове поле однозначно ідентифікує кожен запис у таблиці. Найпоширенішим первинним ключем є послідовно пронумероване поле. Не варто використати поля з конкретною інформацією як первинні ключі: якщо дані по такому полю змінюються, то створюються проблеми з висновком даних і можна втратити частина інформації. Але у випадку, коли дані

являють собою унікальний код (наприклад, Номер страхування), таке поле може бути використано як первинний ключ.

Щоб створити ключове поле, необхідно:

- Виділити поле.
- Вибрати опцію *Ключове поле* в меню **Виправлення** або клацнути на інструменті **Ключове поле**.

Access може сам створити поле первинного ключа, якщо користувач не створив його самостійно. При закритті вікна «Таблиця» Access запитує підтвердження на створення поля первинного ключа. Якщо в таблиці є поле з типом даних *Лічильник*, то Access призначить його як первинний ключ; якщо немає – Access додасть у таблицю нове поле з типом даних *Лічильник* і привласнить йому ім'я «Код».

Робота в режимі таблиці й конструктора

Після створення таблиці її ім'я з'являється у вікні бази даних, що дозволяє їй відкрити в одному з режимів – таблиці або конструктора. Для цього необхідно:

- Виділити у вікні бази даних піктограму таблиці й клацнути на кнопці **Відкрити** або двічі клацнути на імені таблиці – вона відкриється в режимі таблиці.
- Виділити піктограму таблиці й клацнути по кнопці **Конструктор** – таблиця відкриється в режимі конструктора.

Після відкриття таблиці, в лівій частині панелі інструментів з'явиться кнопка **Вид**, що дозволяє перемикати режими відображення й функціонування.

Модифікація структури таблиці

Режим таблиці

Можливі деякі прості зміни в конструкції таблиці:

- Додати.
- Видалити.
- Перейменувати поля.
- Перемістити.
- Змінити розмір (висоту рядка, ширину стовпця).

Для виконання більшості із цих дій зручно використати меню швидкого доступу.

Якщо клацнути на заголовку стовпця правою кнопкою миші, стануть доступними команди: **Ширина стовпця**, **Додати стовпець**, **Видалити стовпець**, **Перейменувати стовпець**, **Стовпець підстановок**.

Команда **Додати стовпець** вставляє новий стовпець ліворуч від того, меню якого використається; за замовчуванням йому буде привласнене ім'я «Поле 1» текстового типу (поле можна перейменувати, а властивість поля змінити в режимі конструктора).

Ширину стовпця можна змінити, перетаскуючи праву границю заголовка або використовуючи команду **Ширина стовпця**.

Для переміщення стовпця необхідно його виділити (щигликом лівої кнопки миші по заголовку), захопити курсором заголовок і перемістити стовпець на потрібне місце (індикатором нового положення служить чорна вертикальна риса за замінним стовпцем у напрямку переміщення).

Якщо клацнути правою кнопкою миші на поле маркування записів, то рядок буде виділено, і стануть доступні команди: **Вирізати**, **Вставити** (копію), **Висота рядка**.

Режим «Конструктор»

Якщо клацнути правою кнопкою миші на поле маркування рядків, то рядок буде виділено й стануть доступні команди: **Додати рядка**, **Видалити рядка**. Перемістити рядок можна, захопивши його в поле маркування лівою кнопкою миші й перетаскуючи в потрібному напрямку; індикатором місця вставки для переміщеного рядка служить чорна горизонтальна смуга.

Висоту рядків можна змінити, переміщаючи границю поділу маркованих рядків.

Втрата даних

Якщо змінено конструкцію таблиці так, що це може привести до втрати даних, Access попередить про це при спробі зберегти зміни. Наприклад:

- Видалення поля, що містить дані.

- Зменшення розміру поля, що містить дані (текст може скоротитися й втратити урізану частину, числові дані округляться із втратою точності).
- Зміна типу даних (дані, які не можуть бути збережені в новому типі даних, губляться).

Перед модифікацією структури бажано створити резервну копію таблиці. Для цього можна використати наступну послідовність дій:

- Виділити таблицю у вікні бази даних.
- Виконати команди: **Виправлення** → **Копіювати**, **Виправлення** → **Вставити**. З'явиться вікно «Вставка таблиці», що запитує нове ім'я й параметри вставки.

Керування відображенням даних

Панель *Властивості поля* (режим Конструктора) має вікно діалогу із двома закладками: *Загальні* й *Підстановка*, що дозволяють задати ряд властивостей поля, і вікно коментарів, наприклад:

- *Підпис* (Caption); властивість дозволяє вказати напис, що буде відображатися як найменування поля в таблиці й авто формі;
- *Умова на значення* (уведення умови вимагає використання Побудувача виражень; у простому випадку, якщо для поля необхідно встановити, наприклад, тільки межу нижньої величини, можна ввести умову: >5). Уведення даних, не відповідних умові, викличе заперечення на етапі збереження даних.

Необхідно врахувати:

- Характеристика *Розмір поля* застосовується до полів з текстовим типом даних, числовим і типом *Лічильник*, але в кожному випадку працює по-різному.
- Характеристика *Число десяткових знаків* ставиться тільки до полів із чисельним і грошовим типом даних.
- Характеристика *Формат поля* працює по-різному для полів з різним типом даних (формати тексту, чисел і дат, безумовно, різні).
- Закладка *Підстановка* панелі *Властивості поля* дозволяє відповідним настроюванням створити списки, що розкриваються, для вибору з них значень, що замінюють код, при введенні даних у режимі таблиці.

Характеристики текстових полів

Розмір поля: визначає максимальне число символів, які можуть бути уведені в це поле (1 – 255, 50 за замовчуванням). Якщо необхідно ввести більше 255 символів, використовується поле МЕМО.

Access не додає при заповненні пробіли в кінець полів з текстом. Тому не слід зменшувати розміри полів для економії дискового простору.

Формат текстового поля

Формат поля визначає тільки спосіб подання даних і не є контролем уведення (для контролю необхідно використати маску уведення).

Для форматування текстових полів використовуються символи:

- @ потрібен символ;
- & не потрібен символ;
- перетворення в заголовні символи;
- < перетворення в прописні символи.

Інші символи використовуються буквально й уставляються в дані. Наприклад, для уведення тексту з поділяючими дефісами можна використати, наприклад, шаблон @@-@@-@@@ і вводити дані підряд, опираючись на шаблон.

Поля «МЕМО»

Не мають характеристики *Розмір поля* до 64000 байт інформації, але Access використає стільки простору на диску, скільки займають уведені дані.

Поля МЕМО звичайно використовують для анотацій.

Характеристики числових полів

Розмір числового поля визначає, який вид чисел може втримуватися в полі, а також діапазон значень цих чисел. Завжди варто задавати розмір поля для даних цього типу. Як правило, для числових полів установлюють характеристику *Формат поля*, де вказують число десяткових знаків після коми.

Розмір поля: байт, ціле, довге ціле, із плаваючою крапкою (4 байт), із плаваючою крапкою (8 байт), код реплікації (створює всесвітньо прийняте ідентифікаційне число, використовуване як складний тип даних).

За замовчуванням дані числових полів відображаються на екрані так, як їх увів користувач. Але можна привласнити полям один зі стандартних форматів, обраних зі списку *Формат поля*:

- *Основний* (вибір за замовчуванням);
- *Грошовий* (включає символ валюти після числа; негативні числа відображаються в круглих дужках; кожні три цифри цілої частини розділяє роздільник, певний настроюванням у Мовах і Стандартах);
- *Фіксований* (у дробовій частині – два знаки після коми, якщо характеристики числа десяткових знаків не змінені);
- *Процентний* (відображає значення у вигляді відсотків);
- *Експонентний* (наукова система позначень).
- *Число десяткових знаків*: – список цієї характеристики поля включає опцію Auto, що дозволяє відобразити число знаків після коми, зазначене в характеристиці *Формат*. Список також дає можливість вибрати число знаків після коми від 1 до 15, перевизначивши установку характеристики поля *Формат*.

Тип даних «Дата/Час»

Можливий вибір стандартних форматів з вікна списку характеристики поля *Формат*:

- Для дати: Повний формат дати – Дата (короткий формат) Час (довгий формат); Довгий формат (6 вересня 2004); середній (06-сент-2004); короткий (06.09.04).
- Для часу: Довгий (2:45:26); Середній (2:45); Короткий (14:45).

Тип даних «Грошовий»

У цьому полі зберігаються дані з фіксованою десятковою крапкою й форматуються так само, як і числові. Розмір поля із цим типом даних не задають.

Тип даних «Лічильник»

За замовчуванням характеристика Розмір поля встановлюється як Довге ціле, а Нові значення – як «Послідовні» (можна вибрати для цієї характеристики значення «Випадкові»).

Логічний тип даних

Може мати три формати: Так/Ні, Істина/Неправда, Вкл/Выкл. За замовчуванням це поле з'являється в таблиці у вигляді вимикача стану (прапорця).

Майстер підстановок

Поле підстановок зручне, якщо уведення даних у це поле здійснюється з деякого фіксованого набору значень, наприклад, Дисципліна.

Якщо створено поле *Майстром підстановок*, можна при уведенні даних у це поле вибирати значення зі списку, що розкривається (його називають стовпцем підстановок).

Майстер підстановок можна викликати, виділивши його як тип даного поля (це не змінить тип даного поля: він відновиться після кінця роботи Майстри).

Перший крок роботи Майстра вимагає вказати спосіб створення списку:

- Використати уведення фіксованих значень.
- Значення з таблиці (запиту).

Наступні кроки залежать від першого вибору. Якщо обрано режим уведення в список, то другий крок вимагає вказати, яку кількість стовпців необхідно відобразити, і ввести значення в ці стовпці.

Наступний крок – вказати в секції вибору «Доступні поля», який стовпець містить значення для уведення в таблицю (у наведеній таблиці – стовець 1).

Останній крок – задати підпис, використовуваний в описі списку, що розкривається, генерованого Майстром (кращим варіантом звичайно є ім'я поля, використовуване за замовчуванням).

Після запуску *Майстра підстановок* поля закладки *Підстановка* заповнюються автоматично. Характеристика *Обмежитися списком* дає можливість вибрати: задовольнятися тільки значеннями зі списку або можна вводити замість них інші значення (Так/Немає).

Логічні поля

У полях логічного типу Access відображає вимикачі. Включити (виключити) такий вимикач можна клацнувши на ньому мишею або нажавши пробіл, коли покажчик перебуває в цьому полі.

Поля «МЕМО»

Якщо текст, котрий вводять у поле МЕМО, містить ряд абзаців, то для одержання нового рядка необхідно натиснути Ctrl/Enter (натискання Enter переводить курсор у нове поле). Клавiші вертикального переміщення курсору дозволяють переміщатися по абзацах, а горизонтального – усередині абзаців. З полями МЕМО важко працювати в режимі таблиці, якщо текст складається з ряду абзаців; у цьому випадку варто створити форму й використати деякі спеціальні прийоми.

Тип даних Гіперпосилання.

У полях із цим типом даних Access зберігає адреси Web-сторінок і документів, що перебувають у мережі або на власному комп'ютері. Дані в цьому полі відображаються блакитними кольорами й підкреслюються. Для редагування такого поля необхідно:

- Клавiшею Tab потрапити в це поле.
- Натиснути F2, увiвши в поле текстовий курсор (з кожної сторони адреси виникнуть символи # - ознака редагування); відредагувати.
- Якщо перемістити курсор в інше поле - символи # зникнуть.

Поля об'єктів OLE.

Як правило, у полях цього типу зберігається графічна інформація, наприклад, фотографії співробітників.

Робота з даними

Редактор Access (відмінності від редагування в інших додатках):

- Клавiші переміщення курсору використовуються для швидкого переміщення між полями.

- За замовчуванням дані в полях завжди виділяються при використанні клавіатури для переміщення по таблиці.
- Для приміщення текстового курсору в підсвічений (при використанні клавіатури) текст поля, необхідно натиснути F2, а потім стрілками горизонтального переміщення переміщатися по тексту.
- Помістити курсор у потрібне поле можна, клацнувши лівою кнопкою миші в цьому полі; подвійний щиглик виділяє дані.

Сортування даних

Для зручності перегляду можна сортувати записи в таблиці в певній послідовності, вибравши поля, використовувані для сортування. Наприклад, у таблиці *Викладачі* запису можна відсортувати в порядку зростання стажу викладачів, використовуючи кнопки сортування на панелі інструментів (або команди меню **Запису, Сортування, Сортування по зростанню (Сортування по убутанню)**). Для вибору поля досить помістити курсор у будь-який його запис. У режимі таблиці можна виділити відразу два або кілька сусідніх стовпців, а потім виконати по них сортування. За замовчуванням в Access сортування записів починається із крайнього лівого виділеного стовпця. При цьому запису таблиці будуть відсортовані спочатку по крайньому лівому виділеному стовпцю, потім (для однакових значень у першому стовпці) – по другому й т.д. Якщо необхідно відновити порядок відображення записів, використайте команду **Запису, Видалити фільтр**.

Примітка. Сучасні СУБД, такі, як Access, ніколи не сортують таблиці фізично, як це робилося раніше. Засоби сортування даних (а також фільтрації, пошуку й заміни) реалізовані в Access як автоматично створювані запити. Записи таблиці завжди розташовуються у файлі бази даних у тім порядку, у якому вони були додані в таблицю.

Відбір даних за допомогою фільтра

Фільтр включає набір умов, застосовуваних для відбору підмножини записів. В Access існують фільтри чотирьох типів: фільтр по виділеному фрагменті, звичайний фільтр, розширений фільтр і фільтр по уведенню.

Фільтр по виділеному фрагменту – це спосіб швидкого відбору записів

по виділеному зразку. Наприклад, вам потрібно переглянути в таблиці запису тільки про доцентів. Виділіть слово *Доцент* у кожному із записів таблиці *Викладачі*. Клацніть по кнопці **Фільтр по виділеному** (або виконаєте команду **Запису, Фільтр, Фільтр по виділеному**), і Access вибере тільки ті записи, для яких значення в стовпці «Посада» дорівнює *Доцент*. Зверніть увагу: у рядку стану вікна таблиці присутнє слово *Фільтр*, а кнопка **Застосувати фільтр** (третя кнопка фільтрування, на якій зображена лійка) затінена; це означає, що використовується фільтр. При відключенні цієї кнопки всі фільтри будуть зняті. Установки фільтра не пропадуть: він просто буде відключений.

Фільтр по виділеному може збирати разом критерії вибору при кожному використанні кнопки **Фільтр по виділеному**. Нехай виділені тільки записи про всіх доцентів. Якщо потім помістити курсор у стовпець «Код дисципліни» і виділити код дисципліни *Інформатика* (якщо така дисципліна присутня у вашій таблиці), а потім клацнути по кнопці **Фільтр по виділеному**, з'являться тільки записи про доцентів, які викладають інформатику. Якщо ви хочете переглянути значення, які не задовольняють цьому критерію, наприклад, переглянути всіх викладачів, крім доцентів, що читають інформатику, клацніть правою кнопкою миші (курсор повинен перебувати усередині таблиці) і виберіть команду **Виключити виділене**. Будуть обрані весь записи, крім недавно обраних (інверсний вибір).

Фільтрування даних в Access виробляється також за допомогою кнопки **Змінити фільтр** (команда меню **Запису, Фільтр, Змінити фільтр**). Після натискання цієї кнопки від таблиці залишається один запис. Кожне поле стає полем зі списком (коли в ньому перебуває курсор), у якому можна вибрати зі списку кожне з неповторюваних значень для даного поля. Після щиглика по кнопці **Застосувати фільтр** будуть обрані записи, що відповідають зміненому фільтру. Ще більш складні умови фільтрації можна задати командою меню **Запису, Фільтр, Розширений фільтр**.

Введення й перегляд даних за допомогою форми

Форми забезпечують гнучкий спосіб введення, редагування, перегляду й видалення даних і фактично є шаблонами, що управляють відображенням інформації. Форма дозволяє відображати одночасно всі поля одного або декіль-

кох записів. Оптимально побудована форма може вміщати кілька десятків полів на одному екрані, а якщо полів набагато більше, те для кожного запису можна створити багато строкову форму. Можна створити меню для виклику інших форм, таблиць, запитів або звітів. У формі кожне поле можна розмістити в точно заданому місці, вибрати для нього кольори або заливання й додати елементи керування текстом для ефективного уведення даних.

При введенні даних можна не тільки поміщати обчислювані поля у форму, але й додавати розширені можливості перевірки коректності уведення даних, а також додавати елементи керування (наприклад, перемикачі, прапорці, що розкривають списки). Лінії, рамки, кольори й фонові зображення поліпшують зовнішній вигляд даних, полегшують сприйняття форми й підвищують продуктивність роботи. На додаток до цього OLE-об'єкти (такі, як малюнки й графіки) можна побачити тільки у формі або у звіті.

Створити форму можна декількома способами. Якщо ви в режимі бази даних відкриєте вкладку **Форми** й клацнете по кнопці **Створити**, то відкриється вікно, у якому зазначені способи створення форми.

Конструктор дозволить вам створити форму самостійно, але це для починаючих користувачів досить складно. *Майстер форм* дає можливість автоматично створити форму на основі обраних полів. Access у режимі діалогу з'ясовує в користувача, яку форму він хоче одержати, і створює її автоматично. Якщо вас у цій формі що-небудь не задовольняє, ви можете виправити це в режимі конструктора.

Автоформи є окремими випадками майстра форм, тобто вони автоматично створюють задані види форм практично без участі користувача. Це може бути зручно, коли базова таблиця одна, містить небагато полів і вам потрібно швидко створити просту форму. *Діаграма* створює форму з убудованою діаграмою, а *Зведена таблиця* – зі зведеною таблицею Excel.

Вікно вибору варіантів побудови форми

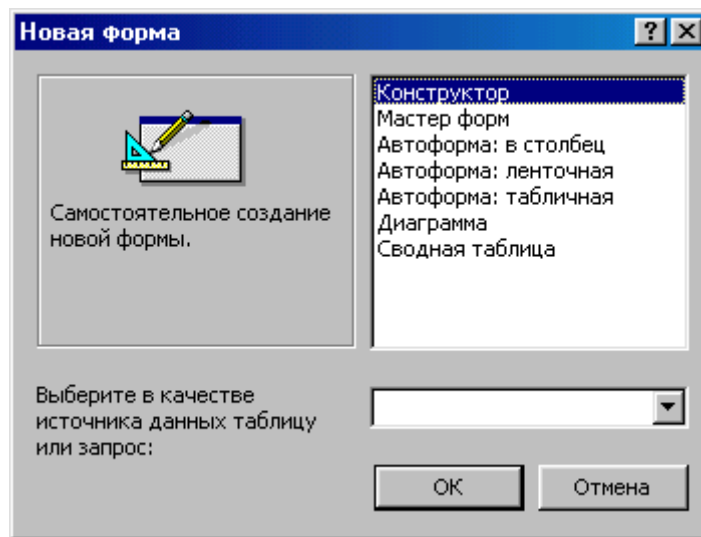


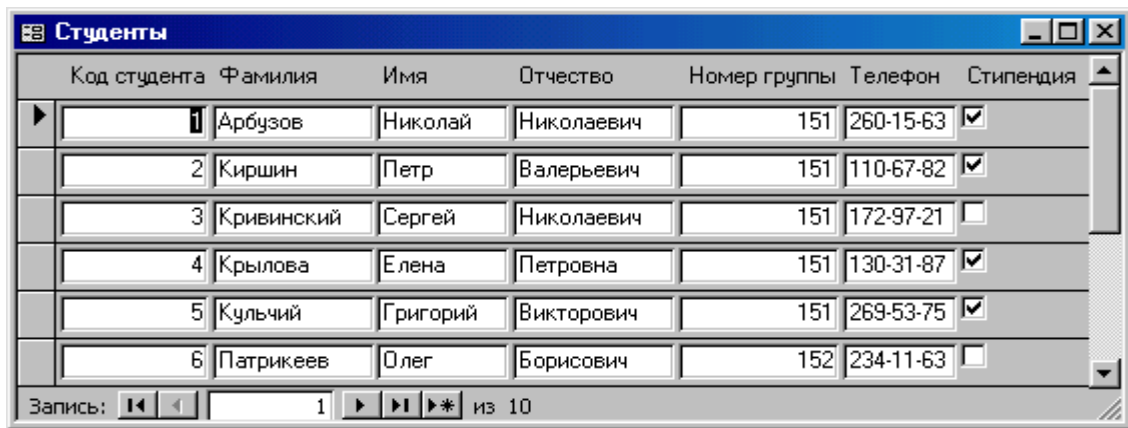
Рис.1

Простим керуванім способом створення форм є «Майстер форм». У цьому режимі ви можна вибрати поля таблиці для відображення у формі, стиль і кольори оформлення тла й осередків, а також вид форми. «Майстер форм» пропонує чотири види форми подання даних: в один стовпець, стрічкова, таблична й вирівняна.

Форма в один стовпець

Рис.2

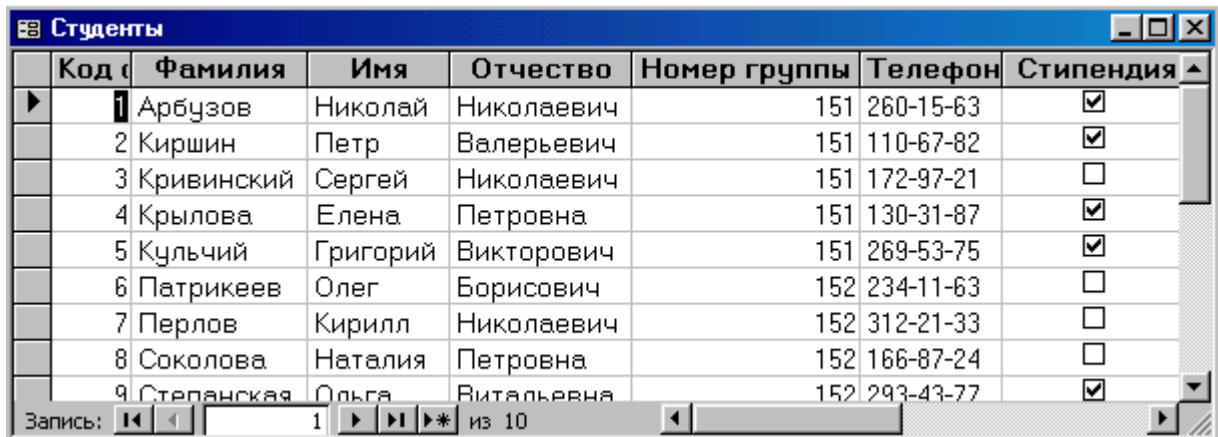
Стрічкова форма



Код студента	Фамилия	Имя	Отчество	Номер группы	Телефон	Стипендия
1	Арбузов	Николай	Николаевич	151	260-15-63	<input checked="" type="checkbox"/>
2	Кишин	Петр	Валерьевич	151	110-67-82	<input checked="" type="checkbox"/>
3	Кривинский	Сергей	Николаевич	151	172-97-21	<input type="checkbox"/>
4	Крылова	Елена	Петровна	151	130-31-87	<input checked="" type="checkbox"/>
5	Кульчий	Григорий	Викторович	151	269-53-75	<input checked="" type="checkbox"/>
6	Патрикеев	Олег	Борисович	152	234-11-63	<input type="checkbox"/>

Рис.3

Таблична форма



Код	Фамилия	Имя	Отчество	Номер группы	Телефон	Стипендия
1	Арбузов	Николай	Николаевич	151	260-15-63	<input checked="" type="checkbox"/>
2	Кишин	Петр	Валерьевич	151	110-67-82	<input checked="" type="checkbox"/>
3	Кривинский	Сергей	Николаевич	151	172-97-21	<input type="checkbox"/>
4	Крылова	Елена	Петровна	151	130-31-87	<input checked="" type="checkbox"/>
5	Кульчий	Григорий	Викторович	151	269-53-75	<input checked="" type="checkbox"/>
6	Патрикеев	Олег	Борисович	152	234-11-63	<input type="checkbox"/>
7	Перлов	Кирилл	Николаевич	152	312-21-33	<input type="checkbox"/>
8	Соколова	Наталья	Петровна	152	166-87-24	<input type="checkbox"/>
9	Степанская	Ольга	Витальевна	152	293-43-77	<input checked="" type="checkbox"/>

Рис.4

Access дає можливість використати більшість стандартних елементів керування Windows, які створюють звичний інтерфейс при уведенні даних. Лінії, рамки, кольори й навіть спеціальні ефекти (наприклад, тіні). Стрічкова форма й таблична форма схожі одна на одну, але таблична форма фактично повторює вид уже знайомій вам таблиці й у ній обмежені можливості редагування.

Якщо форма розробляється на основі декількох таблиць або запитів, то Access запропонує вам створити або підлеглі, або зв'язані форми в стрічковому або в табличному виді.

Переміщення у формі ідентично переміщенню в рядках і стовпцях. Переміщатися за формою можна, просто клацаючи по потрібному полю й вносячи зміни або доповнення в дані.

ВПРАВИ ТА ТЕХНОЛОГІЯ ЇХ ВИКОНАННЯ

ВПРАВА №1

ОДНОТАБЛИЧНА БАЗА ДАНИХ

Розглянемо процес створення однотобличної бази даних на основі таблиці «Викладачі».

Незалежно від методу, який застосовано для створення таблиці, завжди є можливість використати режим «конструктор» для зміни макета таблиці. Наприклад, для додавання нових полів, установки значень за замовчуванням або для створення масок вводу даних

ЗАВДАННЯ 1

Створення бази даних

1. Створіть нову базу даних.
2. Створіть таблицю бази даних.
3. Визначте поля таблиці відповідно до таблиці даних «Викладачі».
4. Збережіть створену таблицю.

Таблиця 1 – Структура даних таблиці «Викладачі»

Ім'я поля	Тип даних	Розмір поля
Код викладача	Числовий	Ціле
Прізвище	Текстовий	15
Ім'я	Текстовий	15
По батькові	Текстовий	15
Дата народження	Дата/час	Короткий
Посада	Текстовий	9
Дисципліна	Текстовий	11
Телефон	Текстовий	9
Зарплата	Грошовий	

ТЕХНОЛОГІЯ ВИКОНАННЯ ВПРАВИ

1. Для створення нової бази даних:
 - завантажте Access, у вікні, що з'явилося, виберіть пункт «**Нова база даних**»;
 - у вікні «**Файл нової бази даних**» задайте ім'я вашої бази (пункт **Ім'я фай-**

лу) і виберіть папку (пункт **Папка**), де ваша база даних буде перебувати. За замовчуванням Access пропонує вам ім'я бази db1, а тип файлу – Бази даних Access (mdb). Ім'я задайте Викладачі, а тип файлу залишіть колишнім, тому що інші типи файлів потрібні в спеціальних випадках;

- клацніть по кнопці **Створити**.

2. Для створення таблиці бази даних:

- у вікні бази даних виберіть вкладку «Таблиці», а потім клацніть по кнопці **Створити**;
- у вікні «Нова таблиця» виберіть пункт «конструктор» і клацніть по кнопці **ОК**. У результаті пророблених операцій відкриється вікно таблиці в режимі конструктора, у якому варто визначити поля таблиці (рис.5).

3. Для визначення полів таблиці:

- введіть у рядок стовпця «Ім'я поля» ім'я першого поля «**Код викладача**»;
- у рядку стовпця «Тип даних» клацніть по кнопці списку й виберіть тип даних *Числовий*; у рядку Індексоване поле поставте *Так*. Інші поля вкладки **Загальні** залишіть такими, як пропонує Access.

Макет таблиці в режимі «Конструктор»

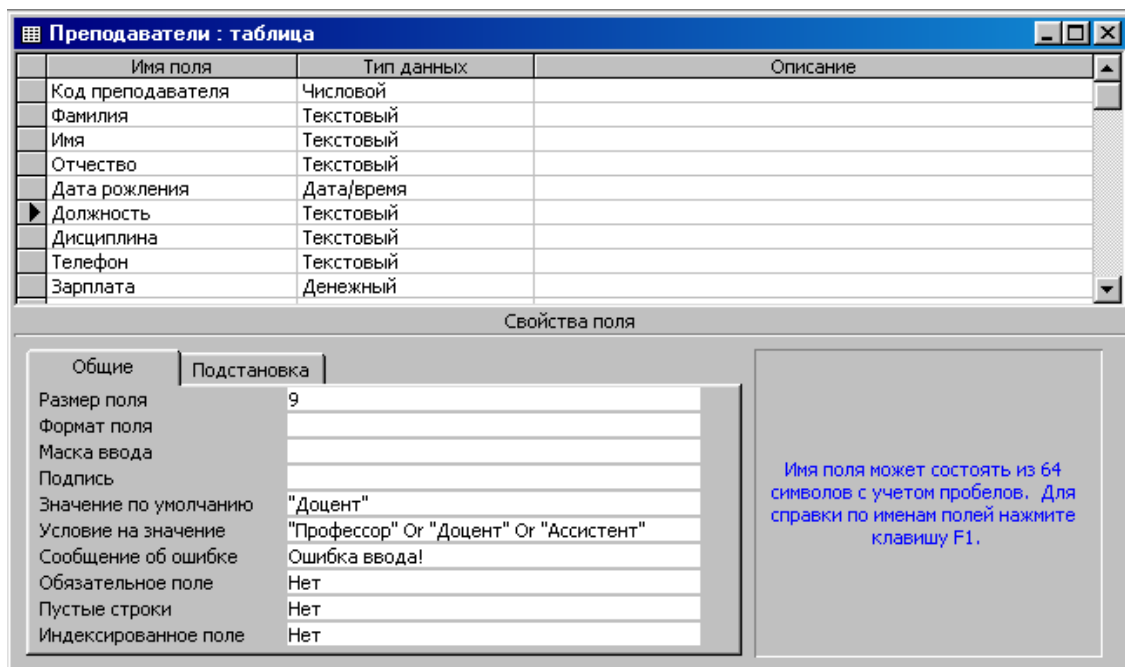


Рис.5

Примітка. Заповнення рядка стовпця «Опис» необов'язково й звичайно використовується для внесення додаткових відомостей про поле.

Увага! Радимо змінити дані в пункті **Розмір поля**, а інші пункти залишити за замовчуванням (їхні функції розглянемо далі).

Наприклад, для текстового типу даних Access пропонує за замовчуванням довжину 50 символів. Але навряд чи поле «**Прізвище**» буде містити більше 15 символів. Не бійтеся помилитися – надалі можна скорегувати довжину поля. Для числового типу Access пропонує *Довге ціле*, але ваші дані можуть бути або невеликі цілі числа (у діапазоні від -32768 до 32767) – тоді треба вибрати *Ціле*, або дробові числа – тоді треба вибрати **Із плаваючою крапкою**.

Для вибору необхідного параметра треба клацнути по полю, а потім нажати кнопку, що з'явилася, списку й вибрати необхідні дані. У результаті таблиця буде мати більше компактний вид, а обсяг бази даних зменшиться.

4. Для збереження таблиці:

- виберіть пункт меню **Файл, Зберегти**;
- у діалоговому вікні «**Збереження**» уведіть ім'я таблиці «*Викладачі*»;
- клацніть по кнопці **ОК**.

ЗАВДАННЯ 2

Заповнення бази даних

1. Уведіть обмеження на дані, що вводяться у поле «**Посада**»; повинні вводитися тільки слова *Професор, Доцент* або *Асистент*.
2. Задайте текст повідомлення про помилку, що буде з'являтися на екрані при введенні неправильних даних у поле «**Посада**».
3. Задайте значення за замовчуванням для поля «**Посада**» у вигляді слова **Доцент**.
4. Введіть обмеження на дані в поле «**Код викладача**»; ці дані не повинні повторюватися.
5. Заповніть таблицю даними і перевірте реакцію системи на введення неправильних даних у поле «**Посада**».
6. Змініть ширину кожного поля таблиці відповідно до ширини даних.
7. Зробіть заміну даних: змініть заробітну плату асистентові Сергеевій з 450 грн. на 470 грн.

8. Зробіть сортування даних у поле «**Рік народження**» по убиванні.
10. Зробіть фільтрацію даних по полях «**Посада**» й «**Дисципліна**».
11. Перегляньте створену таблицю, як вона буде виглядати на аркуші паперу при печатці.

ТЕХНОЛОГІЯ ВИКОНАННЯ ВПРАВИ

1. Для завдання умови на значення для даних, що вводять:
 - увійдіть у режим *Конструктор* для проєктованої таблиці. Якщо ви перебуваєте у вікні бази даних, то виберіть вкладку *Таблиці* й клацніть по кнопці **Конструктор**. Якщо ви перебуваєте в режимі таблиці, то клацніть по кнопці **Режим конструктора** на панелі інструментів або виконаєте команду **Вид, Конструктор**;
 - у верхній частині вікна клацніть по полю «**Посада**»;
 - у нижній частині вікна клацніть по рядку параметра *Умова на значення*;
 - клацніть по кнопці **Вибір** для визначення умов на значення за допомогою режиму **побудова виразів**.
 - у вікні, що з'явилося, напишіть *слово* Професор, потім клацніть по **кнопці Or** (ця кнопка виконує функцію **АБО**), *напишіть* Доцент, знову клацніть по цій же кнопці, *напишіть* Асистент і клацніть по **кнопці ОК**. У такий спосіб ви ввели умову, при якому в поле «Посада» можуть уводитися тільки зазначені значення.
2. У рядку *Повідомлення про помилку* введіть пропозицію «Такої посади не існує, правильно введіть дані».
3. У рядку *Значення за замовчуванням* уведіть слово «Доцент».
4. Уведіть дані в таблицю. Спробуйте в поле «Посада» будь-якого запису ввести слово Лаборант. Подивитися, що вийшло. На екрані повинне з'явитися повідомлення: «Такої посади не існує, правильно введіть дані». Уведіть правильне слово.
5. Для зміни ширини кожного поля таблиці відповідно до ширини даних зробіть подвійного щиглика в заголовку стовпця.
6. Для пошуку в таблиці викладача Миронова:
 - переведіть курсор у перший рядок поля «Прізвище»;
 - виконаєте команду **Виправлення, Знайти**;

- у рядку, що з'явився уведіть Миронов;
 - у рядку параметра *Перегляд* повинне бути слово *Всі* (мається на увазі шукати по всіх записах);
 - у рядку параметра *Збіг* виберіть зі списку *З будь-якою частиною поля*;
 - у рядку параметра *Тільки в поточному полі* встановите прапорець (повинна стояти «галочка»);
 - клацніть по кнопці **Знайти**. Курсор перейде на другий запис і виділить слово Миронов;
 - клацніть по кнопці **Знайти далі**. Курсор перейде на сьомий запис і також виділить слово Міронов;
 - клацніть по кнопці **Закрити** для виходу з режиму пошуку.
7. Для заміни заробітної плати асистентові Сергеевій з 450 грн. на 470 грн.;
- переведіть курсор у перший рядок поля «Зарплата»;
 - виконаєте команду Виправлення, Замінити;
 - у вікні, що з'явилося, у рядку Зразок уведіть 450;
 - у рядку **Замінити на введіть 470**. Зверніть увагу на інші опції. Необхідно вести пошук по всіх записах даного поля;
 - клацніть по кнопці **Знайти далі**. Курсор перейде на четвертий запис, але тут не потрібно міняти дані, тому знову клацніть по кнопці **Знайти далі**. Курсор перейде на дев'ятий запис;
 - клацніть по кнопці **Замінити**. Дані будуть змінені;
 - клацніть по кнопці **Закрити**.

Примітка. Щоб замінити відразу всі дані, треба скористатися кнопкою **Замінити все**.

8. Для сортування даних у полі «Рік народження» по убубанні:
- клацніть по будь-якому записі поля «Рік народження»;
 - клацніть по кнопці **Сортування** на панелі керування або виконаєте команду **Запису, Сортування, Сортування по убубанню**. Всі дані в таблиці будуть відсортовані відповідно до убубання значень у поле «Рік народження».

9. Для фільтрації даних по полях «Посада» й «Дисципліна»:

- клацніть по записі Доцент поля «Посада»;
- клацніть по кнопці **Фільтр по виділеному** або виконаєте команду **Запису, Фільтр, Фільтр по виділеному**. У таблиці залишаться тільки записи про викладачів-доцентів;
- клацніть по записі Інформатика поля «Дисципліна»;
- клацніть по кнопці **Фільтр по виділеному** або виконаєте команду **Запису, Фільтр, Фільтр по виділеному**. У таблиці залишаться тільки записи про викладачів-доцентів кафедри інформатики;
- для скасування фільтрації клацніть по кнопці **Видалити фільтр** на панелі інструментів або виконаєте команду **Запису, Видалити фільтр**. У таблиці з'являться всі дані.

10. Для перегляду створеної таблиці:

- клацніть по кнопці **Перегляд** або виконаєте команду **Файл, Попередній перегляд**. Ви побачите таблицю як би на аркуші паперу;
- закрийте вікно перегляду.

Примітка. Якщо ви захочете змінити поля або орієнтацію таблиці на аркуші паперу, виконаєте команду **Файл, Параметри сторінки**. У вікні, що відкрилося, можете змінювати зазначені параметри.

ЗАВДАННЯ 3

Перегляд і редагування даних за допомогою форми

1. За допомогою Майстра форм створіть форму «Склад викладачів» (тип - форма в один стовпець).
2. Знайдіть запис про доцента Гришина, перебуваючи в режимі форми.
3. Змініте зарплату асистентові Корсун з 470 грн. на 490 грн.
4. Зробіть сортування даних у поле «Прізвище» по убаванню.
5. Зробіть фільтрацію даних по полю «Посада».
6. Змініте назву поля «Дисципліна» на «Дисципліна яка викладається».
7. Переглянете форму з погляду того, як вона буде виглядати на аркуші паперу.

ТЕХНОЛОГІЯ ВИКОНАННЯ ВПРАВИ

1. Для створення форми «Склад викладачів»:

- відкрийте вкладку *Форми* у вікні бази даних; клацніть по кнопці **Створити**;
- у вікні, що з'явилося, виберіть (підведіть курсор миші й клацніть лівою кнопкою) пункт **Майстер** форм;
- клацніть по значку списку в нижній частині вікна;
- виберіть зі списку, що з'явився, таблицю Викладачі;
- клацніть по кнопці **ОК**;
- у вікні, що з'явилося, виберіть поля, які будуть присутні у формі. У даному прикладі бути присутнім будуть всі поля, тому клацніть по кнопці **Всі поля**;
- клацніть по кнопці **Далі**;
- у вікні, що з'явилося, уже обране вид Форма в один стовпець, тому клацніть по **кнопці Далі** у вікні, що з'явилося, виберіть стиль оформлення.
- у вікні, що з'явилося, задайте ім'я форми, набравши на клавіатурі *параметр Склад* викладачів. Інші параметри у вікні залишіть без змін;
- клацніть по кнопці **Готово**. Перед вами відкриється форма в один стовпець. Стовпець ліворуч – це назви полів, стовпець праворуч – дані першого запису (у нижній частині вікна в рядку параметра Запис коштує цифра «1»).

2. Для пошуку викладача Миронова:

- переведіть курсор у перший рядок поля «Прізвище»;
- виконаєте команду **Виправлення, Знайти**;
- у вікні, що з'явилося, у *рядку* Зразок уведіть прізвище Миронов;
- у рядку параметра *Перегляд* повинне бути слово *Всі* (мається на увазі шукати по всіх записах);
- у рядку параметра *Збіг* виберіть зі списку параметр *Із будь-якою частиною поля*;
- у рядку параметра *Тільки в поточному полі* встановите прапорець;
- клацніть по кнопці **Знайти**. Курсор перейде на другий запис і виділить слово Миронов;

- клацніть по кнопці **Знайти далі**. Курсор перейде на сьомий запис і також виділить слово Миронов;
- клацніть по кнопці **Закрити** для виходу з режиму пошуку.

3. Для заміни зарплати асистентові Корсун з 470 грн. на 490 грн.

- переведіть курсор у перший рядок поля «Зарплата»;
- виконаєте команду **Виправлення, Замінити**;
- у вікні, що з'явилося, у рядку *параметра* «Зразок» уведіть 470;
- у рядку *параметра* «Замінити» на введіть 490. Зверніть увагу на інші опції – вам треба вести пошук по всіх записах даного поля;
- клацніть по кнопці **Знайти далі**. Курсор перейде на четвертий запис, але тут не потрібно міняти дані, тому знову клацніть по кнопці **Знайти далі**. Курсор перейде на дев'ятий запис – це те, що нам треба;
- клацніть по кнопці **Замінити**. Дані будуть змінені;
- клацніть по кнопці **Закрити**.

4. Для сортування даних у поле «Рік народження» по убубанню:

- клацніть по будь-якому записі поля «Рік народження»;
- клацніть по кнопці **Сортування** на панелі керування або виконаєте команду **Запису, Сортування, Сортування по убубанню**. Всі дані в таблиці будуть відсортовані відповідно до убубання значень.

5. Для фільтрації даних по полю «Посада»:

- клацніть по записі *Доцент* поля «Посада»;
- клацніть по кнопці **Фільтр по виділеному** або виконаєте команду **Запису, Фільтр, Фільтр по виділеному**. У формі залишаться тільки записи про викладачів – доцентах;
- клацніть по записі Інформатика поля «Дисципліна»;
- клацніть по кнопці **Фільтр по виділеному** або виконаєте команду **Запису, Фільтр, Фільтр по виділеному**. У формі залишаться тільки записи про викладачів - доцентах кафедри інформатики;
- для скасування фільтра клацніть по кнопці **Видалити фільтр** на панелі інструментів або виконаєте команду **Запису, Видалити фільтр**.

6. Змініте назву поля «Дисципліна» на «Дисципліна яка викладається».

- перейдіть у режим конструктор, клацнувши по кнопці **Режим конструктор**

ра на панелі інструментів або виконавши команду **Вид, Конструктор**;

- клацніть правою кнопкою миші на підписі поля «Дисципліна» (вона ліворуч, а поле праворуч із ім'ям Дисципліна – це осередок для даних, властивості яких ми не будемо міняти). У меню, що з'явився, виберіть пункт Властивості;
- на вкладці Макет клацніть у поле з ім'ям Підпис (тобто там, де перебуває слово Дисципліна);
- зітріть слово «Дисципліна» й уведіть «Дисципліна яка викладається»;
- для перегляду результату перейдіть у режим форми, виконавши команду **Вид, Режим форми**.

7. Для перегляду створеної форми:

- клацніть по кнопці **Попередній перегляд**.

ВПРАВА №2

ФОРМУВАННЯ ЗАПИТІВ І ЗВІТІВ ДЛЯ ОДНОТАБЛИЧНОЇ БАЗИ ДАНИХ

Формування запитів на вибірку

Запити є потужним засобом обробки даних, які збережені у таблицях Access. За допомогою запитів можна переглядати, аналізувати й змінювати дані в кількох таблицях. Запити дозволяють обчислювати підсумкові значення й виводити їх у компактному форматі, подібному до формату електронної таблиці, а також виконувати обчислення над групами записів.

Запити можна створювати самостійно й за допомогою майстрів. Майстри запитів автоматично виконують основні дії залежно від відповідей користувача на поставлені питання. Самостійно розробити запити можна в режимі конструктора.

В Access можна створювати наступні типи запитів:

- Запит на вибірку.
- Запит з параметрами.
- Перехресний запит.
- Запит на зміну (запит на видалення, відновлення й додавання записів, на створення таблиці).
- Запити SQL (запити на об'єднання, запити до сервера, підлеглі запити).

Запит на вибірку використовується найбільш часто. При його виконанні дані, що задовольняють умовам добору, вибираються з однієї або з декількох таблиць і виводяться в певному порядку. Наприклад, можна вивести на екран дані про прізвища доцентів, стаж яких більше 15 років (на основі таблиці Викладачі).

Запит на вибірку можна також використати, щоб згрупувати записи для обчислення сум, середніх значень, перерахувань й інших дій. Наприклад, використовуючи запит на вибірку, можна одержати дані про середній стаж доцентів і професорів (на основі таблиці Викладачі).

Запит з параметрами – це запит, при виконанні якого в його діалоговому вікні користувачеві видається запрошення ввести дані, на основі яких буде виконуватися запит. Наприклад, часто потрібні дані про те, які дисципліни ведуть викладачі. Щоб не створювати окремі запити по кожному викладачі, можна створити один запит з параметрами, де в якості параметра буде використатися прізвище викладача. При кожному виклику цього запиту буде запропоновано ввести прізвище викладача, а потім на екран будуть виведені всі поля, які ви вказали в запиті, наприклад прізвище, ім'я, по батькові викладача й дисципліна, яку він читає.

Для створення нового запиту треба у вікні бази даних вибрати вкладку **Запити** й клацнути по кнопці **Створити**. Відкриється вікно «**Новий запит**», у якому ви повинні вибрати один з пунктів: простий запит, перехресний запит, записи що повторюються, записи без підлеглих (рис.6).

Вікно вибору варіантів побудови запитів

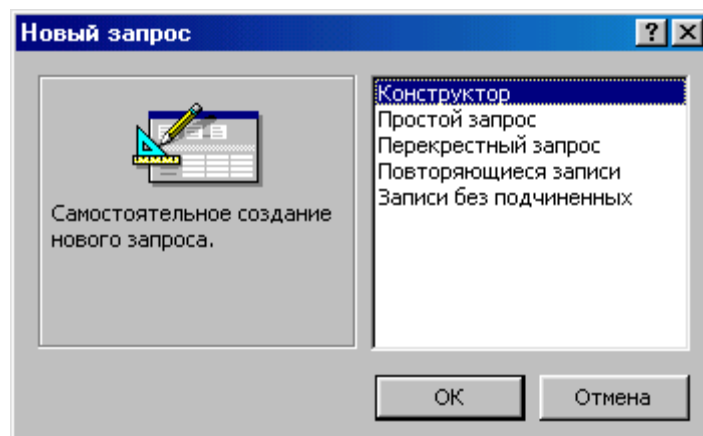


Рис.6

Конструктор дозволить самостійно створити будь-який тип запиту, але цей режим рекомендується користувачам, що вже мають деякий досвід створення запитів.

Простий запит дозволить створити за допомогою **Майстра** запит на вибірку з певних полів таблиць або інших запитів. (Це найкращий спосіб створення запиту для починаючих користувачів.).

При виборі пункту **Повторювані записи** буде створений запит на пошук повторюваних записів у простій таблиці або в запиті.

При виборі пункту **Запис без підлеглих** створюється запит на пошук записів, яким не відповідає жоден запис у підлеглий таблиці. Такий запит використовується для багато табличних баз даних.

При виконанні запиту на вибірку Access витягає записи з таблиць і формує результуючий набір даних. Він виглядає, як таблиця. Результуючий набір даних є динамічним (або віртуальним) набором записів і не зберігається в базі даних.

Після закриття запиту результуючий набір даних цього запиту припиняє своє існування.

При збереженні запиту залишається тільки структура запиту – перелік таблиць, список полів, порядок сортування, обмеження на записи, тип запиту й т.д. При збереженні в базі даних запит, у порівнянні з результуючим набором даних, має ряд переваг:

- на фізичному носії інформації потрібен менший обсяг простору
- запит може використати оновлені версії будь-яких записів, змінених із часу останнього запуску запиту.

При кожнім виконанні запит звертається до базових таблиць і знову створює результуючий набір даних. Оскільки результуючий набір даних не зберігається, запит автоматично відображає будь-які зміни, що трапилися в базових таблицях з моменту останнього запуску цього запиту.

Для збереження запиту варто виконати наступні дії. Виконайте команду **Файл, Зберегти** або клацніть по кнопці **Зберегти** на панелі інструментів. Якщо ви вперше зберігаєте запит, введіть нове ім'я запиту в діалоговому вікні «**Збереження**».

Формування звітів

Звіт – це гнучкий й ефективний засіб для організації перегляду й роздрукування підсумкової інформації. У звіті можна одержати результати складних розрахунків, статистичних порівнянь, а також помістити в нього малюнки й діаграми.

Користувач має можливість розробити звіт самостійно або створити звіт за допомогою *майстра*. *Майстер по розробці звітів* виконує всю рутинну роботу й дозволяє швидко розробити звіт. Після виклику Майстра виводяться діалогові вікна із запрошенням ввести необхідні дані, і звіт створюється на підставі відповідей користувача. *Майстер* необхідний навіть для досвідчених користувачів, тому що дозволяє швидко розробити макет створюваного звіту. Після використання Майстра звітів можна перемкнутися в режим конструктора й внести зміни в стандартний макет.

При роботі з *майстром* залежно від того, який звіт ви хочете створити, Access пропонує вам різні варіанти макетів звіту. Наприклад, якщо ви створюєте простий звіт без угруповання даних, то вам пропонується три варіанти макета: у стовпець, табличний і вирівняний. При цьому в невеликому вікні представляється вид цих макетів. Якщо ви задаєте рівні угруповання (тобто ознаки, по яких треба згрупувати дані, наприклад за посадою), то вам пропонуються шість видів макетів.

Основне розходження між звітами й формами полягає в їх призначенні. Якщо форми задумані переважно для вводу даних, то звіти – для перегляду даних (на екрані або на папері). У звітах обчислюють поля, що формуються на основі загальної групи записів.

Для створення звіту треба відкрити вкладку «*Звіти*» й клацнути по кнопці **Створити**. Відкриється вікно «Новий звіт», у якому наведені шість пунктів меню, тобто шість способів створення звіту: майстер звітів, автозвіт у стовпець, стрічковий автозвіт, майстер діаграм і поштові наклейки.

Конструктор дозволяє самостійно створити звіт, але це не просто навіть для досвідченого користувача. Майстер звітів автоматично створить звіт на основі обраних полів таблиць (запитів) і макетів звітів.

Автозвіт у стовпець й Стрічковий автозвіт – самі найпростіші способи створення звітів. Досить указати тільки ім'я таблиці або запити, на

основі якого буде створений звіт, а інше зробить майстер звітів.

Майстер діаграм допоможе створити звіт у вигляді діаграми.

Поштові наклейки створять звіт, відформатований для печатки поштових наклейок.

ЗАВДАННЯ 1

Формування запитів на вибірку

1. На основі таблиці Викладачі створіть простий запит на вибірку, у якому повинні відображатися прізвища, імена, по батькові викладачів й їхня посада.
2. Дані запиту відсортуйте по посадах.
3. Збережіть запит.
4. Створіть запит на вибірку з параметром, у якому повинні відображатися прізвища, імена, по батькові викладачів як параметр задайте прізвище викладача й виконаєте цей запит для викладача Гришина.

ТЕХНОЛОГІЯ ВИКОНАННЯ ВПРАВИ

1. Для створення простого запиту:

- у вікні бази даних відкрийте вкладку **Запити**;
- у вікні, що відкрилося, клацніть по кнопці **Створити**;
- з пунктів, що з'явилися, вікна **«Новий запит»** виберіть Простий запит і клацніть по **кнопці ОК**;
- у вікні, що з'явилося, у рядку Таблиці/Запити виберіть таблицю Викладачі (якщо інших таблиць або запитів не було створено, вона буде одна в списку, що відкривається);
- у вікні «Доступні поля» переведіть виділення на параметр Прізвище;
- щигликом по кнопці переведіть поле Прізвище у вікно «Обрані поля»;
- аналогічно у вікно «Обрані поля» переведіть поля «Ім'я», «По батькові», «Посада» (порядок важливий – у такому порядку дані й будуть виводитися);
- клацніть по кнопці **Далі**;

- у рядку параметра *Задайте ім'я запиту* введіть нове ім'я *Посади викладачів*;
- клацніть по кнопці **Готово**. На екрані з'явиться таблиця з результатами запиту.

2. Для сортування даних:

- клацніть у будь-якому рядку поля **«Посада»**;
- відсортуйте дані по убубанню. Для цього клацніть по кнопці **Сортування по убубанню** на панелі інструментів або виконаєте команду **Запису, Сортування, Сортування по убубанню**.

3. Для збереження запиту:

- клацніть по кнопці **Зберегти** або виконаєте команду **Файл, Зберегти**;
- закрийте вікно запиту.

4. Для створення запиту на вибірку з параметром:

- створіть запит на вибірку з наступних полів таблиці *Викладачі*: «Прізвище», «Ім'я», «По батькові».
- клацніть по кнопці **Готово**. На екрані з'явиться таблиця з результатами запиту;
- перейдіть у режимі конструктора, клацнувши по кнопці **Режим конструктора** (або виконавши команду **Вид, Конструктор**);
- у рядку параметра **Умови відбору** для поля «Прізвище» уведіть фразу (дужки теж уводити): [Уведіть прізвище викладача];
- виконаєте запит, клацнувши по кнопці **Запуск запиту** на панелі інструментів або виконавши команду **Запит, Запуск**;
- у вікні, що з'явилося, уведіть прізвище Гришин і клацніть по **кнопці ОК**. На екрані з'явиться таблиця з даними про викладача Гришине – його ім'я, по батькові;
- збережете запит;
- закрийте вікно запиту.

ЗАВДАННЯ 2

На основі таблиці «Викладачі» створіть звіт із групуванням даних по посадах.

ТЕХНОЛОГІЯ ВИКОНАННЯ ВПРАВИ

Для створення звіту:

- відкрийте вкладку *Звіти* й клацніть по кнопці **Створити**;
- у вікні, що відкрилося, виберіть пункт Майстер звітів;
- клацніть по значку списку, що розкривається, у нижній частині вікна;
- виберіть зі списку, що з'явився, *таблицю* «Викладачі»;
- клацніть по кнопці **ОК**. У вікні, що з'явилося, виберіть поля, які будуть присутні у звіті. У даному звіті будуть присутні всі поля таблиці.
- клацніть по кнопці **Далі**;
- у вікні, що з'явилося, є присутнім перелік полів. Потрібно вказати, по яких полях буде виконане угруповання. Переведіть у праве вікно поле «Посада». У такий спосіб ви задаєте угруповання даних за посадою;
- клацніть по кнопці **Далі**;
- сортування у звіті виконувати не будемо, тому у вікні Сортування клацніть по кнопці **Далі**;
- у вікні, що з'явилося, виберіть стиль оформлення звіту;
- клацніть по кнопці **Далі**;
- у вікні, що з'явилося, уведіть назву *звіту* «Викладачі»;
- клацніть по кнопці **Готово**. На екрані з'явиться сформований звіт;
- переглянете, а потім закрийте звіт.

Література

1. Баженов В. А., Венгерський П. С., Горлач В. М., Левченко О. М., та ін. Інформатика. Комп'ютерна техніка. Комп'ютерні технології: Підручник. – К.: Каравела, 2003. – 464с.
2. Береза А. М. Основи створення інформаційних систем: Навч. Посібник. – К.:КНЕУ,1998.–140с.
3. Дейт К.Дж. Введение в системы баз данных – К.; М.; СПб.: “Вильямс”, 1999. – 848с.
4. Економічна інформатика. Підручник для вузів. Під ред. проф. В.В. Евдокимова. – СПб.: Питер, 1997. – 592с.
5. Куправа Т. А. Самоучитель Access 97/2000. – С.Пб: Наука и Техника, 2001. – 144с.
6. Попель Г., Гоулстмайн Б. Информационная технология – миллионные прибыли: Пер. с англ. / Науч. Ред. И авт. Предисл. В. В. Симанов. – М.: Экономика, 1990. – 238с.
7. Праг К.Н., Ирвин М.Р. Біблія користувача Access для Windows 98 – К.: Діалектика, 1999. – 576с.
8. Хансен Г., Хансен Дж. Базы даних: розробка і керування – М.: БІНОМ, 1999. – 704с.
9. Microsoft Access 2000: наглядно и конкретно / перевод с английского. – М.: Издательский отдел «Русская редакция» ТОО “Channel Trading Ltd.”, 2000. – 256с.

Зміст

ТЕОРІЯ СТВОРЕННЯ ІНФОРМАЦІЙНИХ СИСТЕМ	3
ТЕМА1 МОДЕЛЮВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ	3
ТЕМА 2 СТВОРЕННЯ ІНФОРМАЦІЙНО-ЛОГІЧНОЇ МОДЕЛІ	5
ТЕМА 3 ПРИВЕДЕННЯ МОДЕЛІ ДО НЕОБХІДНОГО РІВНЯ НОРМАЛЬНОЇ ФОРМИ	9
ТЕМА 4 ТЕМА 4 БАЗИ ДАНИХ В MS ACCESS	11
ТЕМА 5 СТВОРЕННЯ БАЗИ ДАНИХ В MS ACCESS	14
ВПРАВИ ТА ТЕХНОЛОГІЯ ЇХ ВИКОНАННЯ	30
ЛІТЕРАТУРА	45

Навчальне видання

Швачич Геннадій Григорович
Барвінов Віталій Петрович
Рижанкова Галина Іванівна
Кузьменко Вячеслав Віталійович

**Основи технології проектування та створення
інформаційних систем**

Довідкове керівництво по проектуванню та створенню
інформаційних систем з вправами
Навчальний посібник

Тем. План 2006, поз.

Підписано до друку 00. 00.06. Формат 60x84 ^{1/16}. Папір друк. Друк плоский.
Облік.-вид. арк. 2,2 Умов.-друк. арк. 2,1. Тираж 100 пр. Замовлення №

Національна металургійна академія України
49600, Дніпропетровськ – 5, пр. Гагаріна, 4

Редакційно – видавничий відділ НМетАУ