

Дата: 23.01.2023

Група: 23

Предмет: Мова SQL та бази даних

УРОК 4

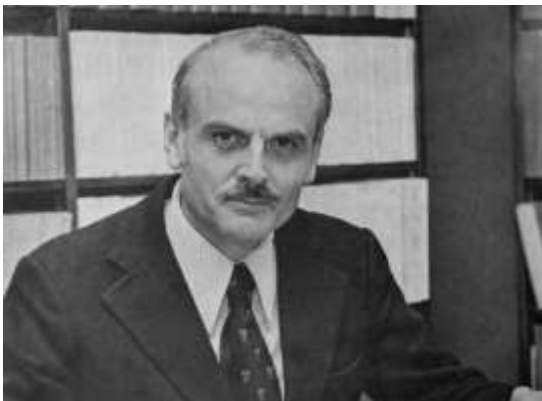
Тема: «Суть баз даних і джерел їх формування»

Мета:

- Опанувати поняття «реляційної бази даних», її переваги та недоліки
- Сформувати в учнів уяву про основні інструменти MS ACCESS
- Вивчити характеристики об'єкти MS ACCESS
- Виховати інформаційно освічену і компетентну особистість, зацікавленість до обраної професії

1. Відомості про реляційну модель бази даних

Реляційна модель даних — логічна модель даних. Вперше була запропонована британським ученим співробітником компанії IBM Едгаром Франком Коддом (E. F. Codd) в 1970 році в статті «A Relational Model of Data for Large Shared Data Banks» (російський переклад статті, в якій вона вперше описана, опублікований в журналі «СУБД» N 1 за 1995 р.). В даний час ця модель є фактичним стандартом, на який орієнтуються практично всі сучасні комерційні СУБД.



Франк Кодд

У реляційній моделі досягається набагато більш високий рівень абстракції даних, ніж в ієрархічній або мережевий. У згаданій статті Е. Ф. Кодда стверджується, що «реляційна модель надає засоби опису даних на основі тільки їх природної структури, тобто без потреби введення якої додаткової структури для цілей машинного представлення». Іншими словами, подання даних не залежить від способу їх фізичної організації. Це забезпечується за рахунок використання математичної теорії відносин (сама назва «реляційна» походить від англійського relation— «відношення»).

До складу реляційної моделі даних зазвичай включають теорію нормалізації. Крістофер Дейт визначив три складові частини реляційної моделі даних:

- структурна

- маніпуляційна
- цілісна

Структурна частина моделі визначає, що єдиною структурою даних є нормалізоване n-арное ставлення. Відносини зручно представляти у формі таблиць, де кожен рядок є кортеж, а кожен стовпець — атрибут, визначений на деякому домені. Даний неформальний підхід до поняття відносини дає більш звичну для розробників і користувачів форму представлення, де реляційна база даних являє собою кінцевий набір таблиць.

Маніпуляційна частина моделі визначає два фундаментальних механізми маніпулювання даними — реляційна алгебра і реляційне числення. Основною функцією маніпуляційної частини реляційної моделі є забезпечення заходів реляційності будь-якої конкретної мови реляційних БД: мова називається реляційною, якщо вона має не меншу виразність і потужність, ніж реляційна алгебра або реляційне числення.

Цілісна частина моделі визначає вимоги цілісності сутностей і цілісності посилань. Перша вимога полягає в тому, що будь-який кортеж будь-якого відношення відмітний від будь-якого іншого кортежу цього відношення, тобто іншими словами, будь-яке відношення має володіти первинним ключем. Вимога цілісності по посиланнях, або вимога зовнішнього ключа полягає в тому, що для кожного значення зовнішнього ключа, що з'являється в посиланному відношенні, у відношенні, на яку веде посилання, повинен знайтися кортеж з таким же значенням первинного ключа, або значення зовнішнього ключа повинно бути невизначеним (тобто ні на що не вказувати).

Можна провести аналогію між елементами реляційної моделі даних і елементами моделі «сутність-зв'язок». Реляційні відносини відповідають наборам сутностей, а кортежі — сутностям. Тому, також як і в моделі «сутність-зв'язок» стовпці в таблиці, що представляє реляційне відношення, називають атрибутами.

Кожен атрибут визначений на домені, тому домен можна розглядати як безліч допустимих значень даного атрибуту. Кілька атрибутів одних відносин і навіть атрибути різних відносин можуть бути визначені на одному і тому ж домені.

Іменоване безліч пар «ім'я атрибуту — ім'я домену» називається схемою відношення. Потужність цієї множини — називають ступенем чи «арністю» відносини. Набір іменованих схем відносин представляє із себе схему бази даних.

Атрибут, значення якого однозначно ідентифікує кортежі, називається ключовим (або просто ключем). У нашому випадку ключем є атрибут «Табельний номер», оскільки його значення унікально для кожного працівника підприємства. Якщо кортежі ідентифікуються тільки зчепленням значень декількох атрибутів, то говорять, що відношення має складовий ключ. Ставлення може містити кілька ключів. Завжди один із ключів оголошується первинним, його значення не можуть обновлятися. Всі інші ключі відносини називаються можливими ключами.

На відміну від ієрархічної і мережної моделей даних в реляційній відсутнє поняття групових відносин. Для відображення асоціацій між кортежами різних відносин використовується дублювання їх ключів.

У реляційній моделі даних об'єкти і взаємозв'язки між ними представляються за допомогою таблиць. Взаємозв'язки також подаються як об'єкти. Кожна таблиця представляє один об'єкт і складається з рядків і стовпців. Таблиця повинна мати первинний ключ (ключовий елемент) — поле чи комбінацію полів, що єдиним способом ідентифікують кожний рядок у таблиці (рис. 3.4).

Назва «реляційна» (relational) пов'язана з тим, що кожен запис у таблиці даних містить інформацію, яка стосується (related) якогось конкретного об'єкта. Крім того, зв'язані між собою (тобто такі, що знаходяться в певних відношеннях — relations) дані навіть різних типів в моделі можуть розглядатися як одне ціле.

Таблиця має такі властивості:

- кожний елемент таблиці являє собою один елемент даних;
- повторювані групи відсутні;
- усі стовпці в таблиці однорідні; це означає, що елементи стовпця мають однакову природу;
- стовпцям присвоєні унікальні імена;
- у таблиці немає двох однакових рядків.

Порядок розміщення рядків і стовпців у таблиці довільний; таблиця такого типу називається відношенням. У сучасній практиці для рядка використовується термін «запис», а для стовпця термін «поле».

Переваги реляційної моделі:

- простота і доступність для розуміння користувачем. Єдиною використовуваною інформаційною конструкцією є «таблиця»;
- строгі правила проектування, які базуються на математичному апараті;
- повна незалежність даних. Зміни в прикладній програмі при зміні реляційної БД мінімальні;
- для організації запитів і написання прикладного ПЗ немає необхідності знати конкретну організацію БД у зовнішній пам'яті.

Недоліки реляційної моделі:

- далеко не завжди предметна область може бути представлена у вигляді «таблиць»;
- в результаті логічного проектування з'являється безліч «таблиць». Це призводить до труднощів розуміння структури даних;
- БД займає відносно багато зовнішньої пам'яті;
- відносно низька швидкість доступу до даних.

2. Основні відомості про СКБД Access

До основних об'єктів бази даних Access належать:

1. Таблиці – призначені для впорядкованого зберігання даних.
2. Запити – призначені для пошуку, виведення даних, виконання обчислень.
3. Форми – призначені для зручного перегляду, змін та додавання даних в таблицях.
4. Звіти – використовуються для аналізу даних та їх виводу на друк.

5. Макроси – використовуються для виконання набору макрокоманд, які часто повторюються та здійснюють обробку даних.
6. Модулі – призначені для опису інструкцій та процедур на мові VBA.

Основним об'єктом бази даних є таблиця, яка складається із записів (рядків) та полів (стовпців). На перетині запису та поля існує комірка, в якій містяться дані.

Кожне поле таблиці має унікальне ім'я, яке не може містити більше 64 символів. В кожному полі містяться дані одного типу.

Типи даних

Тип	Опис
Текстовий	Використовується для зберігання символічних чи числових даних, які не потребують обчислень. У властивості «Розмір поля» задається максимальна кількість символів, які можуть бути введені в дане поле. По замовчужанню розмір містить 50 знаків. Максимальна кількість символів, які можуть міститись в текстовому полі – 255.
Поле МЕМО	Призначене для введення текстової інформації, яка перевищує 255 символів. Може містити до 65536 символів.
Числовий	Призначений для зберігання числових даних, які використовуються для математичних розрахунків. На вкладках «Загальні» та «Підстановка» можна встановити певні властивості числового поля, серед яких «Розмір поля», «Формат поля», «Число десяткових знаків».
Дата / час	Використовується для представлення дати та часу. Вибір конкретного формату дати та часу встановлюється у властивостях «Формат дати».
Грошовий	Призначений для зберігання даних, точність представлення яких коливається в межах від 1 до 4 знаків після коми. Ціла частина може містити до 15 десяткових знаків.
Лічильник	Призначений для автоматичного введення унікальних послідовних (збільшуються на 1) або випадкових чисел в якості номера нового запису. Номер, який присвоюється запису, не може бути видалений чи змінений. Поля з цим типом даних використовуються в ролі ключових полів таблиці.
Логічний	Призначений для зберігання одного із двох значень, які інтерпретуються як «Так / Ні», «Правда / Неправда», «Вкл / Викл»
Поле об'єкта OLE	Містить дані, які створені в інших програмах, котрі використовують протокол OLE. Це можуть бути, наприклад, документи Word, електронні таблиці Excel, малюнки, звукові та відео записи, тощо. Об'єкти OLE зв'язуються з базою даних Access. Сортувати, групувати та індексувати поля об'єктів OLE не можна.
Гіперпосилання	Спеціальний тип, призначений для зберігання гіперпосилань.
Майстер підстановок	Призначений для автоматичного визначення поля. З його допомогою створюється поле з списком, з якого можна вибирати дані, які містяться в іншій таблиці або в наборі постійних значень.

3. Створення бази даних

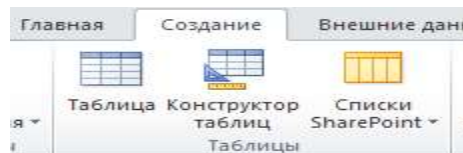
1. Запустіть Microsoft Access 2010.



2. Натисніть на кнопку «Створити».

3. Задайте ім'я нової бази даних – «Записна книжка» (розширення – accdb).

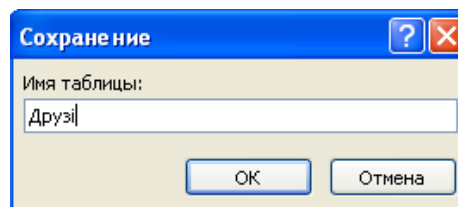
4. На вкладці «Створення» в панелі інструментів «Таблиці» виберіть кнопку «Конструктор таблиць».



5. Введіть імена полів та вкажіть типи даних, до яких вони належать.

Имя поля	Тип данных
№ п/п	Счетчик
Прізвище	Текстовый
Ім'я	Текстовый
По батькові	Текстовый
Адреса	Текстовый
Індекс	Числовой
Телефон	Текстовый
Захоплення	Текстовый
Ел пошта	Гиперссылка

6. Вийдіть з режиму «Конструктора», зберігши перед тим таблицю під іменем «Друзі»; ключові поля не задавайте.



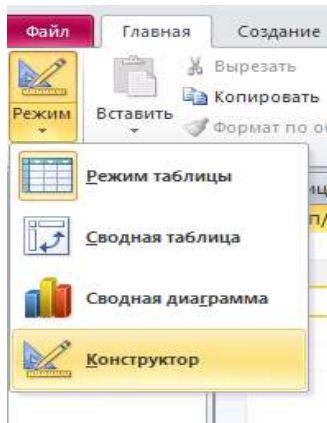
7. Відкрийте таблицю «Друзі» та заповніть в ній 10 рядків за своїм власним бажанням.

№ п/п	Прізвище	Ім'я	По батькові	Адреса	Індекс	Телефон	Захоплення	Ел пошта
1	Іванов	Іван	Іванович	Устилузька 56	44700	2-34-56	народні танці	ivan@ukr.net
2	Петров	Петро	Петрович	Луцька 213	44700	3-45-67	бальні танці	petro@ukr.net

8. Додайте поля «Дата народження» та «Місце народження». Для цього:

- 1) Встановіть курсор на поле, перед яким потрібно вставити новий стовпець.
- 2) Виконайте команду: вкладка «Режим таблиці» панель інструментів «Поля та стовпці» «Вставити».
- 3) Двічі клікнути на Поле1 і перейменувати його на «Дата народження» та Поле2 – на «Місце народження».

9. Перейдіть в режим Конструктора командою: вкладка «Головна», «Режим», «Конструктор».



10. Для поля «Дата народження» встановіть тип даних «Дата / Час», у властивостях поля виберіть «Короткий формат дати»

Дата народження	Дата/время
Місце народження	Текстовый
Адреса	Текстовый
Індекс	Числовой
Телефон	Текстовый
Захоплення	Текстовый
Ел пошта	Гиперссылка

Свойства поля

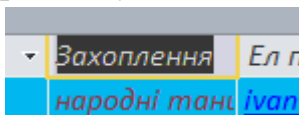
Общие	Подстановка
Формат поля	Краткий формат даты
Маска ввода	Полный формат даты 19.06.2007 17:34:23
Подпись	Длинный формат даты 19 червня 2007 р.
Значение по умолчанию	Средний формат даты 19-Чер-2007
Условие на значение	Краткий формат даты 19.06.2007
Сообщение об ошибке	Длинный формат времени 17:34:23
Обязательное поле	Средний формат времени 5:34
Индексированное поле	Краткий формат времени 17:34
Режим IME	Нет контроля

11. Відформатуйте таблицю таким чином:

- ✓ Колір фону – голубий,
- ✓ Колір тексту – темно-червоний, розмір – 12 пт, накреслення – курсив.

№ п/п	Прізвище	Ім'я	По батько	Дати
1	Іванов	Іван	Іванович	
2	Петров	Петро	Петрович	
(№)				

12. Переіменуйте поле «Захоплення» в «Хоббі».



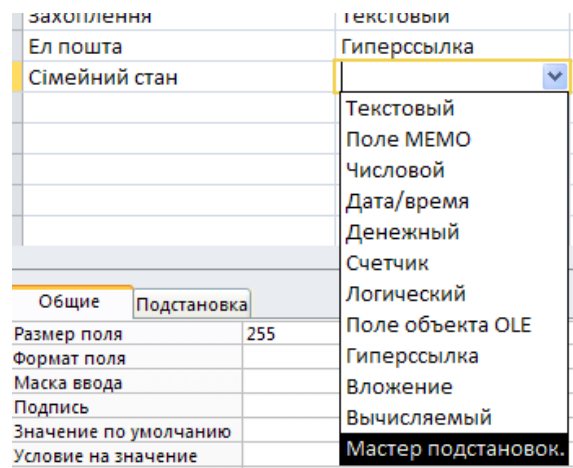
13. Видаліть запис під номером 8.

14. Змініть розмір комірок так, щоб було видно усі дані. Для цього достатньо двічі клікнути лівою клавішею миші на границі полів.

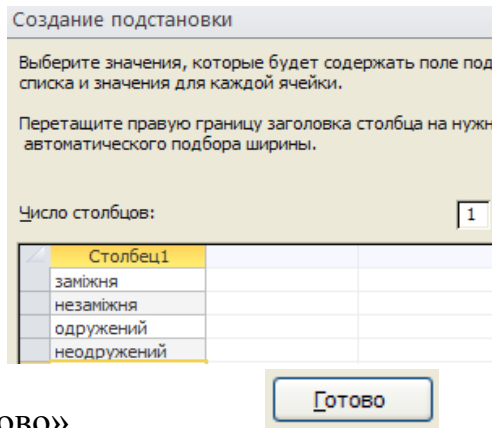
15. Розмістіть поля в такому порядку: «№ п/п», «Прізвище», «ім'я», «По батькові», «Телефон», «Дата народження», «Хоббі», «Адреса», «Індекс», «Ел.пошта».

16. Заповніть порожні комірки таблиці.

17. В режимі Конструктора додайте поле «Сімейний стан», в якому буде міститись фіксований набір значень – заміжня, незаміжня, одружений, неодружений. Для створення списку, що розкривається, будемо використовувати «Майстер підстановок»:

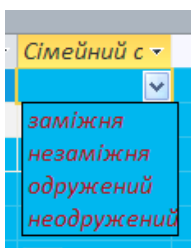


- ✓ Встановіть тип даних «Майстер підстановок»:
- ✓ В діалоговому вікні виберіть рядок «Буде введений фіксований набір значень» та натисніть кнопку «Далі»
- ✓ Число стовпців – 1
- ✓ Введіть дані списку – заміжня, незаміжня, одружений, неодружений.

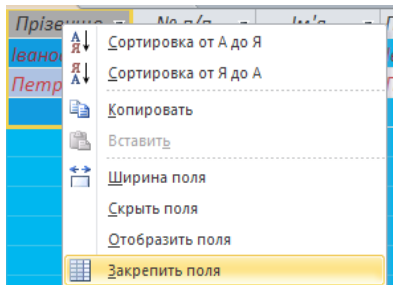


- ✓ Натисніть кнопку «Готово».

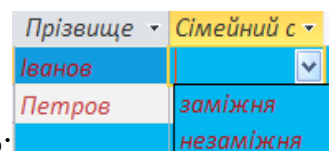
18. За допомогою розкриваючого списку заповніть новий стовпець.



Оскільки таблиця вийшла широка, то при заповненні даного стовпця виникають певні незручності: не видно прізвища людини, для якої заповнюється поле «Сімейний стан». Щоб прізвище було постійно видно при заповненні таблиці, необхідно використати команду «Закріпити поля» з контекстного меню поля «Прізвище».



Одержимо:



Питання для самоперевірки:

1. Назвіть особливості реляційної моделі даних
2. Назвіть переваги та недоліки реляційної бази

3. Назвіть правила організації даних таблиць реляційної моделі
4. Який об'єкт в базі даних є основним?
5. Що називається полем та записом в БД?
6. Які типи даних ви знаєте?
7. Як можна перейменувати поле?
8. Як можна створити поле з списком, який розкривається?
9. З яким розширенням зберігається файл в БД Microsoft Access?

Домашнє завдання:

1. Створіть базу даних «*Оптовий склад*», що складається з чотирьох таблиць з такою структурою:
 - Склад* – код товару, кількість, дата поступлення.
 - Товари* – код товару (ключове поле), назва товару, термін зберігання.
 - Заявки* – код заявки (ключове поле), назва організації, код товару, необхідна кількість.
 - Відпуск товару* – код заявки (ключове поле), код товару, відпущена кількість, дата відпущення товару.
2. Встановіть зв'язки між таблицями.

2) читать підручник:

Руденко В., Потієнко В. «Інформатика 11» (рік видання 2019) §1.3, §2