РЕФЕРАТ

**на тему:**

**“Створення двохвимірних таблиць”**

(на прикладі Access та Word)

**Створення двохвимірних таблиць в Access**

Кожну інформаційну область за своєю структурою можна віднести до деякого типу. Найпоширені типи – ієрархічні, мережні, реляційні. Якщо дані уявити у графічному вигляді як пов′язані між собою **двовимірні таблиці**, то одержану сукупність даних можна назвати реляційною (від англійського слова relation - відношення). Відповідно з типом даних називають і системи управління базами даних. Найбільш поширеними сьогодні є реляційні СУБД, прикладом яких є ORACLE, MS SQL Server, MS FoxPro, MS Access тощо. Переваги реляційних СУБД - це легкість використання та ефективність реалізації. У даному навчальному посібнику розглядаються СУБД реляційного типу, а саме MS Access. Треба відмітити, що будь-яку інформаційну структуру можна визначити через модель іншої структури. Але саме СУБД реляційного типу дозволяє це зробити з найменшими втратами.

У подальшому рядки двовимірних таблиць будемо називати записами, стовпчики – полями. У термінах реляційних СУБД подібні таблиці називаються відношеннями, їх записи – кортежі відношень, поля – атрибути відношень. Тепер зрозуміло, чому використовується термін реляційна СУБД (relation - відношення). Записи відрізняються своїм номером, а поля – своїм ім′ям. Основні умови щодо змісту таблиць такі:

* однакові записи забороняються;
* всі записи повинні мати однакову кількість полів;
* значення полів атомарні, тобто таблиця не може мати своїми компонентами інші таблиці.

Елементами даних MS Access є сталі, змінні пам′яті та поля таблиць. Елементи даних та функції над ними складають вираз, які надалі будемо позначати словом Exp, додаючи, якщо це потрібно, ще один з символів C, M, N, D, L – початкові букви англійських назв типів даних (наприклад, ExpN). Типи даних розподіляються на базові і такі, що визначені користувачем. Деякі базові типи даних наведені у Табл. 1.

Табл. 1 - Базові типи даних MS Access

| Тип даних | Позначення | Назва в MS Access |
| --- | --- | --- |
| Рядок (Character) | C | текст |
| Текст ( Memo ) | M | текст Memo |
| Числовий (Numeric) | N | числовий |
| Дата (Date) | D | дата/час |
| Логічний (Logical) | L | логічний |

Приклади сталих: “010201” – рядок (ExpC), 010201 – число (ExpN), 01.02.01 – дата (ExpD). Типи полів таблиць визначаються під час визначеня таблиці або модіфікаціїї її структури. Треба пам′ятати, що тип даних рядок – це множина послідовностей символів. Кількість символів такої послідовності називається довжиною рядка і не може перевищувати 255. Тип даних рядок використовується для позначення кодів, назв, прізвищ тощо. Існує операція конкатенації даних типу рядок. Позначається ця операція знаком + . Наприклад, значення зміних A і B є відповідно ‘Micro’ і ‘soft’, то значенням виразу A+B , буде рядок ‘Microsoft’. Тип даних текст (Memo) використовується для даних, значення яких – символи, а їх кількість може сягати за 255. Числовий тип даних використовується для даних, що характеризують кількісну їх властивісь, наприклад, вага, стаж, заробітна плата тощо. Тип даних дата очевидний, він використовується для обробки дат. Логічний тип даних використовується в виразах, що характеризують ситауцію, причому значення таких виразів або істина або хибність. Наприклад, вираз 10>100 логічний, а значення такого виразу – хибність. Для даних логічного типу існують декілька операцій: OR - логічне додавання, AND – логічне множення, NOT – логічне заперечення. Наприклад, значення виразу NOT (10>100) – істина.

### 

### Проектування структури таблиць

Процес визначення таблиці починається на стадії її проектування. Саме на цій стадії розробник бази даних має зробити аналіз інформації, яку необхідно обробляти. Для визначеності розглянемо інформацію про кадри підприємства, яка складається з особового складу, відомостей про переміщення тощо. Цю інформацію у реляційних СУБД необхідно представити сукупністю двовимірних таблиць. Нехай, для прикладу, база даних, що створюється, називається Кадри та складається з двох таблиць: Особи та Переміщення . Визначемо таблицю Особи . Визначимо, що в цій таблиці розміщується деяка інформація з першої сторінки відомої картки особи, а саме:

1. табельний номер (номер картки особи)
2. прізвище
3. ім′я
4. по-батькові
5. дата народження
6. стать
7. освіта

Кожний описаний вище пункт картки особи – це майбутнє поле таблиці! Визначемо тепер назву, тип даних, розмір та необхідну множину значень цих полів та зведемо все у Табл. 2:

Табл. 2 - Поля таблиці осіб

| Пункт картки особи | Назва поля | Тип даних | Розмір, байт | Множина значень |
| --- | --- | --- | --- | --- |
| 1 | ТабНомер | C | 4 |  |
| 2 | Прізвище | C | 16 |  |
| 3 | Ім′я | C | 12 |  |
| 4 | ПоБатькові | C | 16 |  |
| 5 | ДатаНародження | D |  |  |
| 6 | Стать | C | 1 | ‘Ч’;’Ж’ |
| 7 | Освіта | C | 16 | ‘середня’;’вища’; ‘середня-технічна’ |

Як бачимо, деякі клітинки цієї таблички не заповнені. Так, розмір поля з типом D визначається автоматично, а можливі значення даних пунктів 1)-5) картки особи тут очевидно не потрібні. На цьому процес проектування таблиці Особи призупиняється, щоб визначитись з так званими первинними ключами.

### Індекси та ключі

Для зв’язування таблиць у реляційних СУБД використовують ключі. Ключем називають сукупність полів таблиці, значення яких ідентифікують її записи. Первинним ключем називають такий ключ, який однозначно ідентифікує записи таблиці. Іншими словами, значення первинного ключа є унікальним для даної таблиці. Часто первинний ключ складається з одного поля. Наприклад, первинним ключем таблиці Особи має бути поле ТабНомер, і аж ніяк не Прізвище, оскільки записів з однаковим табельним номером не повинно бути. Водночас можна очікувати на записи, де є однакові прізвища.

Впорядкування записів таблиці за значенням ключа називають індексацією, а отриманий образ таблиці – індексом. Взагалі фізично таблиці та індекси можуть зберігатися у різних файлах (як, наприклад, у СУБД MS FoxPro). У MS Access всі об′єкти бази даних (у тому числі і індекси) зберігаються в одному файлі.

Для уникнення непотрібних повторювань даних, яке може бути причиною помилок при введені даних, і нераціонального використання дискового простору комп′ютера проектування таблиць завершується їх нормалізацією – процесом зменшення надлишкової інформації [3]. Так, згідно з принципами нормалізації таблицю Особи слід редагувати, замінивши в ній поле Освіта на КодОсвіти (тип даних С, розмір 1), а крім цього ввести ще таблицю Освіта з двома полями КодОсвіти та Освіта, які співпадають з відповідними полями у модифікованій та немодифікованій таблиці Особи . Таблицю Освіта будемо ще називати таблиця-довідник Освіта .

Взагалі таблиці створюються так, щоб задовольнити трьом нормальним формам:

Перша нормальна форма

1. Таблиця не має записів, що повторюються.
2. У таблиці відсутні групи полів, що повторюються.
3. Рядки повинні бути не впорядковані.
4. Стовпчики повинні бути не впорядковані.

Зрозуміло, щоб задовольнити умові 1, кожна таблиця повинна мати первинний ключ.

Друга нормальна форма

1. Таблиця задовольняє умовам першої нормальної форми.
2. Будь-яке неіндексне поле однозначно ідентифікується повним набором індексних полів.

Очевидно, що таблиці, які мають один первинний ключ автоматично задовольняють умові 2.

Третя нормальна форма

1. Таблиця задовольняє умовам другої нормальної форми.
2. Жодне з неіндексних полів таблиці не ідентифікується за допомогою іншого неіндексного поля.

Процес нормалізації таблиць, як правило, супроводжується створенням окремих додаткових таблиць, які пов′язані між собою з використанням введених первинних ключів. Повертаючись до таблиці Особи, відмітимо, що її первинним ключем буде поле ТабНомер. Очевидно первинним ключем таблиці Освіта буде поле КодОсвіти. Тепер настає черга побудови таблиць:

Табл. 3 - Таблиця Особи

| Номер поля | Назва поля | Тип даних | Розмір, байт | Множина значень |
| --- | --- | --- | --- | --- |
| 🗝1 | ТабНомер | C | 4 |  |
| 2 | Прізвище | C | 16 |  |
| 3 | Ім′я | C | 12 |  |
| 4 | ПоБатькові | C | 16 |  |
| 5 | ДатаНародження | D |  |  |
| 6 | Стать | C | 1 | ‘Ч’;’Ж’ |
| 7 | КодОсвіти | C | 1 | Значення поля КодОсвіти з таблиці Освіта |

Табл. 4 - Таблиця Освіта

| Номер поля | Назва поля | Тип даних | Розмір, байт | Множина значень |
| --- | --- | --- | --- | --- |
| 🗝1 | КодОсвіти | C | 1 |  |
| 2 | Освіта | C | 16 |  |

Тут знак 🗝 означає первинний ключ.

### Побудова та редагування структури таблиць

Для визначення таблиці треба у вікні бази даних (активізація вікна бази даних - F11) натиснути кнопку “Создать” та вибрати той чи інший режим, наприклад, режим конструктор. В цому режимі кожний рядок в верхній частини вікна визначає поле таблиці. Три стовпчики необхідні для визначення поточного поля: назва, тип даних, опис. Тут опис – для необов′язкових пояснень. Перед рядком є можливість поставити ознаку первинного ключа за допомогою значка з малюнком у вигляді золотого ключика, що розташований на панелі інструментів. Розмір поля необхідно поставити у частині вікна з назвою “Свойства поля”. Щоб забезпечити підстановку значень (в нашому прикладі значення полів Стать та КодОсвіти таблиці Особи) необхідно в стовпчику “Тип данных” знову звернутися до вибору можливих типів і вибрати “Мастер подстановок”. Після очевидного діалогу буде забезпечено необхідний вибір значень при формування записів таблиці. Слід відмітити, що характер двох підстановок в даному прикладі різний. Для поля Стать -це підстановка фіксованих значень “М” або “Ж”, а для поля КодОсвіти - це підстановка даних поля КодОсвіти таблиці Освіта . Майстер підстановок забезпечує вибір даних серед елементів фіксованих значень або серед значень поля таблиці-довідника, причому для надійності вибору цей процес, як правило, посилюється візуалізацією інших полів таблиці-довідника (в даному випадку при виборі значень поля КодОсвіти візуалізується ще й поле Освіта). Визначення підстановок за рахунок значень полів таблиць-довідників автоматично веде до так званого зв′язування таблиць з відношенням “один-до-багатьох”.

Якщо первинний ключ користувач не визначив, то MS Access допоможе при необхідности це зробити введенням додаткового поля, значення якого – порядковий номер запису.

Для редагування структури таблиці необхідно натиснути кнопку “Конструктор” вікна база даних, а далі діяти за схемою, що наведена вище.

### 

### Зв’язування таблиць

Процес зв′язування таблиць може розпочинатись ще на етапі визначення підстановок. Завершує цей процес робота з визначення схеми даних. Ця робота розпочинається з натискання на значок “Схема данных”, що розташований на панелі інструментів. В результаті відкривається вікно з такою же назвою “Схема данных”, де вже існують графічні образи зв′заних при підстановці таблиць. Решта таблиць може бути додана при необхідності. Зв′язки між таблицями можна редагувати, видаляти. Для редагування або видалення існуючого зв′язку необхідно правою кнопкою миші клацнути по лінії зв′язку (графічне зображення зв′язку), а далі дотримуватись очевидних дій меню та діалогу. Так, дія “Удалить” означає видалення зв′язку (це можна забезпечити і натисканням клавіатурної клавіши DELETE), а дія “Изменить связь” – редагування зв′язку. Причому в останній дії передбачено діалог, який містить питання щодо цілісності даних. Якщо забезпечити цілісність (вибір значком “галочка”), то отримаємо зв′язок “один-до-багатьох”. Якщо між таблицями попередньо ніякі зв′язки не встановлені, то встановити іх можна за допомогою лівої кнопки миші: необхідно не відпускаючи кнопки миші з′єднати нею відповідні поля. В результаті відкриється вікно “Связи”, де необхідно вибрати “Обеспечение целостности данных”.

**Створення двохвимірних таблиць у Word**

Для уставлення в документ таблиці необхідно установити курсор у місці, де повинна починатися таблиця та в меню **Таблица** вибрати пункт **Добавить,**  потім **Таблица.** У діалоговому вікні, що з'явилося, слід увести число стовпців і рядків, і натиснути **ОK**. Для уставлення таблиці також можна натиснути кнопку . У вікні, що розкрилося, розтягнути виділення на необхідне число вічок і натиснути клавішу миші. Усі команди для роботи з таблицями знаходяться у меню **Таблица**.



Пересування по таблиці здійснюється за допомогою покажчика миші або клавішами: **🡩, 🡫, 🡨, 🡪,** **Tab** (на вічко праворуч), **Shift+Tab** (на вічко ліворуч). Кожне вічко таблиці розглядається як абзац, і дані у вічках форматуються як абзаци тексту. Для роботи з таблицями зручно користуватися панеллю інструментів Таблицы и границы, яку можна вивести на екран за допомогою кнопки .



При наведенні покажчика миші на верхню лінію таблиці, він перетворюється в чорну стрілку. Якщо в цей момент натиснути мишею, то виділиться один стовпець. Пересуваючи мишею чорну стрільцю, можна виділити відразу декілька стовпців. Рядки таблиці виділяються як рядки звичайного тексту. Для виділення декількох суміжних вічок необхідно натиснути мишею в одне вічко фрагменту й розтягнути виділення на інші.

Коли курсор уведення знаходиться у таблиці, на координатних лінійках з’являються позначки меж стовпців  і рядків . При пересуванні цих позначок змінюються розміри відповідних стовпців і рядків. За допомогою прихованого переліку  можна вибрати тип вирівнювання тексту у вічках. Кнопка слугує для зміни напрямку розташування тексту у виділених вічках.



Щоб об'єднати декілька вічок в одне, слід виділити їх і викликати команду **Объединить ячейки** меню **Таблица** або натиснути кнопку на панелі Таблицы и границы. Для розділення вічка на декілька вічок слід вибрати пункт **Разбить ячейки** меню **Таблица** або натиснути кнопку .



Для додавання елементів таблиці (рядків, стовпців, вічок) необхідно виділити елементи, на місці яких необхідно уставити нові та в меню **Таблица** вибрати команду **Добавить**, потім – потрібний пункт (Столбцы слева, Столбцы справа, Строки выше, Строки ниже, Ячейки). Для видалення елементів таблиці слід виділити їх та у меню **Таблица** вибрати пункт **Удалить,** потім – потрібний пункт (**Таблица,** Столбцы, Строки, Ячейки).

За замовчанням лінії сітки таблиці мають товщину 0,5 пт. Змінити товщини та вигляду ліній сітки можна декількома способами.

І спосіб:

* виділити вічка, обрамлення яких потрібно змінити;
* у прихованому переліку Тип линии на панелі Таблицы и границы вибрати тип лінії;



* у прихованому переліку Толщина линии – товщину лінії;



* якщо натиснути кнопку , з’явиться палітра кольорів, в якій можна вибрати колір обрамлення;



* відкрити прихований перелік і вибрати вигляд обрамлення.



ІІ спосіб:

* у прихованих переліках панелі Таблицы и границы вибрати тип, товщину і колір лінії;
* натиснути кнопку ;



* покажчиком миші, який матиме вигляд олівця, вказати початок лінії та розтягнути до її кінця
* після натискання кнопки покажчиком миші можна стирати лінії обрамлення.



Щоб залити вічка кольором необхідно виділити їх і у прихованому переліку вибрати колір.



# Список літератури

1. Вейскас Д. Эффективная работа с Microsoft Access 7.0 для Windows 95/ Перев. с англ. – СПб.: Питер, 1997. – 848 с.
2. Крамм Р. Системы управления базами данных dBASEII для персональных компьютеров. – М.: Финансы и статистика, 1988. – 283 с.
3. Каратыгин С.А., Тихонов А.Ф., Тихонова Л.Н. Работа в Visual FoxPro на примерах: М: БИНОМ, 1995. – 512 с.
4. .В.Попов, Є.С.Вакал, О.В.Обвінцев, В.В.Личман. Розробка учбових баз даних у MS Access. – Київ, 2001. – 33 с.