***Система керування базами даних Microsoft Access 10***

1. Поняття про бази даних і СУБД.
2. Етапи розробки баз даних.
3. Загальна характеристика СУБД Microsoft Access 10.
4. Типи та властивості полів.
5. Проектування структури баз даних.
6. Створення баз даних і таблиць.
7. творення запитів та їх застосування.
8. Створення форм та звітів.

**БД та Система управління базами даних Access**

# Коротка характеристика системи

Система управління базами даних Microsoft Access входить до складу пакета Microsoft Office. Ми будемо розглядати русифіковану версію СУБД Access. Вона дозволяє розв'язувати широке коло завдань користувачів без програмування. Це озна­чає, що Access доступна для широкого кола непрофесійних користувачів персональних комп'ютерів.

Запуск системи Access здійснюється з ОС Windows аналогічно тому, як запускається будь-яка інша прикладна програма. Після її завантаження на екрані дисплея з'являється вікно системи, на передньому плані якого зна­ходиться вікно створення чи відкриття бази даних

Зазначимо, що це вікно з'явиться тільки на початку завантаження системи. У поточному сеансі роботи з системою вам доведеться неодноразово відкривати чи створювати нові бази даних. Ці операції можна буде здійснювати, на­приклад, за допомогою меню Файл головного меню системи.

Одна з основних переваг СУБД Access полягає у тому, що вона має прості та зручні засоби обробки кількох таблиць у одній базі даних. Таблиця — це місце зберігання даних. Вона є основним об'єктом бази даних. Далі розгля­датимемо роботу з двома таблицями бази даних.

У СУБД перших поколінь для кожної таблиці створювався свій файл бази даних. Для спільного використання таблиць у цих СУБД застосовують спе­ціальні програми (утиліти). Потім для спільного використання таблиць почали застосовувати спеціальні команди, які мають виконуватись перед початком роботи з базою даних для так званого зв'язування таблиць.

Пізніше спробували зберігати у одній базі даних кілька таблиць з описом зв'язків між ними. Прикладом такої СУБД є FoxPro. Але тільки у СУБД Access з'явилися зручні та наочні засоби зв'язування таблиць, які зберігаються в одному файлі.

Зберігання кількох таблиць у одній базі даних має цілий ряд переваг. До основних з них належать:

1. Можливість розподілу даних за таблицями згідно зі змістом даних.  
   Наприклад, у першій таблиці зберігаються звітні дані вчителів, у другій - їхній тижневий розклад, а в третій — відомості про позаурочну діяльність.
2. Табличне, тобто модульне, подання даних дозволяє модернізувати кожну таблицю окремо.
3. Редагування даних в одному місці бази призводить до автоматичного їх редагування в іншому місці.

У системі Access є різні способи управління даними, а саме: система меню, панелі інструментів, контекстне меню, покажчик миші та комбінації клавіш.

СУБД Access має значну кількість спеціальних програм, які отримали назву "майстри". Так, є майстер таблиць, майстер кнопок, майстер форм і т.д. Майстри здійснюють діалог з користувачем, у процесі якого визначаються дані, необхідні для розв'язування відповідної задачі. Для зручності роботи майстри поділяються на кроки (етапи). Користувач може пропускати деякі кроки чи повертатися до попередніх.

Система Access має зручні засоби копіювання рядків, полів і фрагментів однієї таблиці до іншої. Крім того, можна отримувати повні копії таблиць чи їхніх структур. Таблицю однієї бази даних можна перенести до іншої бази даних.

Access має розвинуту систему запитів, яка дозволяє отримувати на екрані різні відомості з таблиць. Користувач може задавати умови запиту, відповідно з якими з бази вибираються визначені дані чи формується нова таблиця.

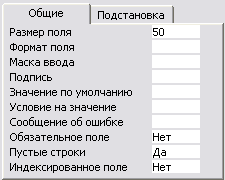
СУБД Access має засоби, які забезпечують видачу на екран даних не тільки у вигляді таблиць, але й у вигляді інших форматів (форм). За допомогою майстра форм користувач може сам визначити форму видачі даних на екран. В Access є три стандартних види форм, які можуть будуватися автоматично.

У системі Access широко використовується поняття звіт. Звіт багато в чому схожий на форму. Його використовують при виведенні документів на друку­вання. Принципова різниця між Формою та Звітом у тому, що формати звітів відповідають стандартним форматам паперу.

Система Access має потужну довідкову систему. Вона не замінює і не може замінити навчальний посібник з цієї системи, але добре його доповнює.

Якщо вам у процесі роботи буде що-небудь не зрозуміло, не хвилюйтесь — система Access y будь-який момент вам допоможе.

# Типи та властивості полів

****У системі Access як і в інших базах даних реляційного типу **рядок таблиці ототожнюється з терміном "запис", а колонка — з терміном "поле"**. Кожне поле має ім'я, тип і властивості. При створенні структури таблиці обов'язково вказують імена і типи полів. Властивості полів можуть прийматись за замовчуванням. Одне або кілька полів необхідно визначити як **ключові**. Ключовими звичайно вибирають поля, значення яких ні при яких умовах не повторюються. Ключове поле служить для налагодження зв’язків між таблицями. Якщо деяке поле в конструкторі оголошене як лічильник, то воно за замовчуванням стає ключовим.

В Access використовують такі типи полів:

**1. Текстове поле**. У текстовому полі можуть записуватись літери, цифри та інші символи. Текстові поля мають властивості, перелік яких наведено на мал.

Коротко розглянемо суть цих властивостей.

* **Размер поля**. Він має межі від 1 до 255 символів.
* **Формат поля**. За допомогою спеціальних символів можна задавати вигляд та розмір рядків, що вводяться. Для того щоб задати формат текстових полів, використовують такі кодові символи: @ — текстовий символ або пропуск; & — текстовий символ; < - перетворення символів у нижній регістр; > — пере­творення символів у верхній регістр.
* **Маска ввода.**

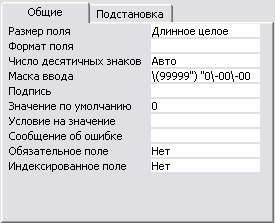
Властивість Маска введення (ІnputMask) полегшує введення даних і визначає значення, який можна ввести в елемент керування поле.

Маски введення зручно використовувати для операцій введення даних, наприклад маска введення для поля "Телефони", точно показує, у якому форматі вводити новий телефон: (\_\_\_) \_\_\_-\_\_\_\_. Щоб задати дану властивість, простіше використовувати майстер по створенню масок.

* **Подпись**. По суті це друге ім'я поля. Воно задається рідко.
* **Значение по умолчанию**. Дозволяє автоматично вставляти рядки, які часто повторюються, у всі новостворені поля.
* **Условие на значение**. Дозволяє створювати фільтр, який забезпечує введення у дане поле тільки тих значень, які відповідають заданій умові.
* **Обязательное поле**. Набуває тільки двох значень: "так" або "ні". "Так" означає, що у дане поле обов'язково мають бути введені дані, які відповідають раніше визначеним властивостям.
* **Пустые строки**. Набувають двох значень: "так" або "ні". "Так" означає, що порожні рядки дозволені. Тут над порожнім рядком починається рядок, який має пропуски.
* **Индексированное поле**. Набуває двох значень: "так" або "ні". "Так" означає, що поле індексоване.

Зазначимо, що на початку вивчення та конструювання баз даних звичайно використовують тільки властивість "розмір поля". Інші властивості, як пра­вило, приймаються за

замовчуванням. Вони використовуються після набуття деякого досвіду роботи з базами даних.

**2. Числове поле**. У числове поле можуть записуватися цілі числа і числа у нормальному вигляді, тобто з плаваючою крапкою. Числові поля мають властивості, перелік яких див. на мал. Аналіз цього переліку показує, що вони багато в чому збігаються з переліком властивостей текстових полів.

Розглянемо більш детально тільки властивість Размер поля. Вона може набирати таких значень:

* Байт — цілі числа у межах від 0 до 255.
* Целое — цілі числа від -32 768 до 32 767.
* Длинное целое — цілі числа від -2 147 483 648 до 2 147 483 647.
* С плавающей точкой (4 байт) — числа у межах від - 3.402823 Е38 до 3.402823 Е38.
* С плавающей точкой (8 байт) — числа у межех від - 1.79769313486232 Е308 до 1.79769313486232 Е308.
* Код репликации — глобально унікальний ідентифікатор (GUID) довжи­ною 16 байт.

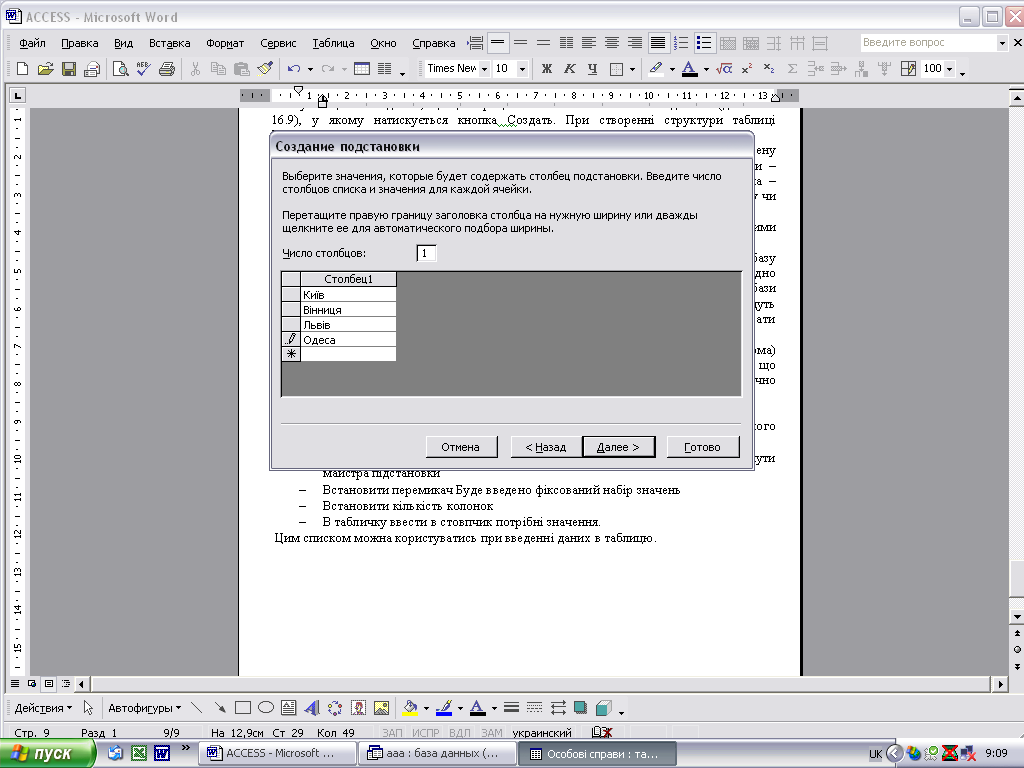
1. **Поле Грошове** містить грошові значення і числові дані (від одного до чотирьох знаків у дробовій частині), використовувані в математичних розрахунках, що проводяться з точністю до 15 знаків у цілій і до 4 знаків у дробовій частині.
2. **Поле Дата/время**. Це поле використовують для запису дат та часу. Багато із властивостей цього поля такі самі, як і текстового поля. Властивість Формат поля може набирати таких значень:

* Полный формат даты (наприклад, 12.05.97 15:20:21).
* Длинный формат даты (наприклад, 13 серпня 1997 p.).
* Средний формат даты (наприклад, 13 серп. 97 p.).
* Короткий формат даты (наприклад, 13.08.97).
* Длинный формат времени (наприклад, 15:20:21).
* Средний формат времени (наприклад, 03:21 РМ).
* Короткий формат времени (наприклад, 15:20).

Якщо властивість, Формат поля набувається за замовчуванням, то у різні записи цього поля можна вводити різну інформацію. Наприклад, в однин запис цього поля може бути записано дату в такому вигляді 02/03/97, а в інший запис поля — у такому вигляді 02-03-97.

1. **Логічне поле.** У логічному полі може записуватись одне з таких двох значень: "так" або "ні", "хибне" чи "істинне", "ввімкнено" чи "вимкнено". Логічні поля можуть використовуватися з різною метою. Але найчастіше їх використовують в анкетних даних, де є тільки дві можливі відповіді.
2. **Поле типу лічильник**. Виконують як лічильник записів, його також часто використовують як ключове поле.
3. **Поле типу Меmо.** Це довге поле. У ньому може вміщуватися текст або комбінація тексту та чисел, яка містить 64000 символів.

**7. Поле об'єкта OLE**. Це поле вміщує безпосередньо не інформацію про об'єкт, а посилання на ім'я об'єкта. Як ім'я об'єкта може бути ім'я програми, наприклад електронна таблиця Excel, редактор Word, засобами яких можуть бути створені об'єкти для впровадження їх у таблицю Access. Крім того, об'єктами можуть бути малюнки, звукозаписи та інші дані у двійковому форматі. Для введення значення в поле в режимі таблиці потрібно виконати команду Вставка – Об’єкт – Обор - вибрати потрібний об’єкт.

1. **Поле об’єкта Гіперпосилання** містить текст або комбінація тексту і чисел, збережені як текст і використовувані як адреса гіперпосилання. Щоб вставити адресу гіперпосилання в поле або в елемент керування, виберіть команду Вставка – гіперпосилання в режимі введення даних в таблицю та вкажіть файл, на який посилаєтесь..
2. **Майстер підстановок** Створює поле, що дозволяє вибрати значення з іншої таблиці або зі списку значень за допомогою списку або поля зі списком. Вибір цього параметра запускає майстер підстановок, що створює Поле підстановок. **Для створення** елементів деякого стандартного списку потрібно:

* у конструкторі таблиці у відповідному полі типа даних потрібно увімкнути майстра підстановки
* Встановити перемикач Буде введено фіксований набір значень
* Встановити кількість колонок
* В табличку ввести в стовпчик потрібні значення (див. малюнок)
* Далее - Готово

Цим списком можна користуватись при введенні даних в таблицю.

# Створення бази даних

На початковому етапі роботи з базами даних процес їх створення доцільно починати з підготовчого етапу, а саме: з побудови на папері структури таблиць. Розглянемо дві таблиці: Список та Консультації.

**Список**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| №п/п | Фио | Предмет | ДатаРожд | Стаж | Нагрузка | Оклад |
| 1 | Костенко A.M. | Математика | 12.01.70 | 3 | 18 | 140 |
| 2 | Носенко С. К. | Физика | 20.05.55 | 15 | 18 | 180 |
| 3 | Борисов H. П. | Химия | 16.04.38 | 32 | 24 | 250 |
| 4 | Кирин Т.В. | Физика | 29.11.50 | 18 | 24 | 220 |
| 5 | Стойко B.C. | Литература | 17.08.46 | 24 | 20 | 200 |
| 6 | Абрамов К.К. | Математика | 22.12.61 | 7 | 22 | 160 |

**Консультация**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № п/п | Фио | Аудитория | День | Время |
| 1 | Костенко A.M. | 215 | Вторник | 15.00-17.00 |
| 2 | Носенко С. К. | 307 | Понедельник | 16.00-18.00 |
| 3 | Борисов Н.П. | 112 | Среда | 15.00-17.00 |
| 4 | Кирин Т.В. | 307 | Пятница | 16.00-18.00 |
| 5 | Стойко B.C. | 230 | Вторник | 15.00-18.00 |
| 6 | Абрамов К.К. | 215 | Четверг | 15.00-17.00 |

Структуру таблиці Список можна подати так

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя поля | Тип поля | Размер поля | Др.свойства | Описание |
| Номер | Счетчик | Длинное целое |  |  |
| Фио | Текстовый | 14 |  |  |
| Предмет | Текстовый | 12 |  |  |
| ДатаРожд | Дата/время | Краткий формат даты |  |  |
| Стаж | Числовой | Байт |  |  |
| Нагрузка | Числовой | Байт |  |  |
| Оклад | Числовой | Целое |  |  |

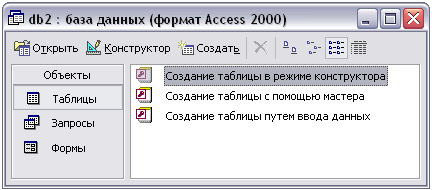
Структуру таблиці Консультация можна подати так

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя поля | Тип поля | Размер поля | Др.свойства | Описание |
| Номер2 | Счетчик | Длинное целое |  |  |
| Фио | Текстовый | 14 |  |  |
| Аудитория | Числовой | Целое |  |  |
| День | Текстовый | 12 |  |  |
| Время | Текстовый | 12 |  |  |

Перші кроки щодо створення бази даних можуть бути різними. Іі можна створювати в режимі таблиці, конструктора таблиці, майстра таблиці, імпорту таблиці. Робота користувача складається з двох етапів:створення структури таблиці за допомогою конструктора та введення даних в таблицю в режимі таблиці.

Ми зробимо так. Запускаємо систему Access і у вікні запрошення встанов­люємо перемикач у положення Новая база данных. Потім натискуємо кнопку ОК. У результаті на екрані з'являється вікно Файл новой базы данных (створення нової бази даних можна починати також і через меню Файл. Для цього потрібно відкрити це меню і вибрати команду Создать ... ). Вікно, що відкривається є стандартним вікном Windows. У ньому необхідно набрати ім'я файла (наберемо ім'я Baza2), його тип і директорію, а потім натиснути кнопку Создать. Після цього відкривається вікно База данных, зображене на малюнку

Зверніть увагу, що на екрані з'явилося головне меню системи. Воно вміщує такі меню: Файл, Правка, Вид, Вставка, Сервис і Окно. Робота у голов­ному меню СУБД Access здійснюється відповідно до загально­прийнятих пра­вил роботи у системі Windows. Багато команд головного меню системи Access дублюють ті самі операції, які можуть бути виконані іншими засобами. Оскіль­ки більшість операцій з базами даних виконуватимемо за допомогою контекстних меню, то окремо головне меню розглядати не будемо. Після прочитання всього матеріалу ви можете повернутися до головного меню і самостійно виконати у ньому багато команд.

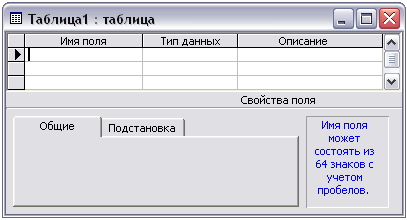
Крім головного меню, на екрані ще є панель інструментів, робота з якими також здійснюється за правилами, прийнятими для системи Windows, і особливих труднощів не викликає. Тому окремо цю панель ми розглядати не будемо, а звертатимемося до неї в міру необхідності.

У вікні База данных натискуємо кнопку Создать. Реакцією системи на цю дію буде виведення на екран вікна Новая таблица, яке зображено на мал.

У цьому вікні перераховані такі способи створення таблиць: Конструктор, Мастер, Введення даних

Режим Введення даних видає бланк (форму) абстрактної таблиці. Потім вона може приймати конкретне наповнення і вміст.

Робота Мастера таблиц заснована на використанні заготовок таблиць, з яких користувач вибирає поле для своєї нової таблиці. Звичайно, передбачувані заготовки полегшують роботу користувача, але вони не вирішують всіх життєвих випадків. Та й навряд чи будь-яка, навіть хороша, заготовка може повністю задовольнити потреби та запити всіх користувачів.

З усіх способів створення таблиць, на наш погляд, найзручнішим є спосіб (режим), який називають Конструктор. Він особливо зручний на етапі почат­кового вивчення СУБД Access. Саме цим способом ми і будемо користуватися.

У вікні Новая таблица вибираємо опцію Конструктор і натискуємо кнопку ОК. На екрані з'являється вікно табл. 1 у режимі конструктора, зображене на мал.

Заповнюємо цю таблицю тими даними, які приведені вище в таблиці Список. вище Звернімо увагу на те, що для установки розміру поля необхідно підвести курсор миші в рядок Размер поля, клацнути клавішею миші, потім установити потрібний розмір поля і натиснути клавішу Enter.

Після заповнення всієї таблиці необхідно виділити ключове поле, яким тут буде поле Номері. Для цього клавішами переміщення курсору вказівка переміщується у відповідне поле, потім курсор миші потрібно перевести на кнопку Ключевое поле і клацнути клавішею миші. При цьому поряд з іменем цього поля з'явиться зображення ключа.

У лівому верхньому куті вікна таблиці поряд з написом Таблица 1 є світлий прямокутник. Його називають значком таблиці. При фіксації курсора миші на ньому з'являється контекстне меню, у якому є команда Закрыть. Після виконання цієї команди на екрані з'явиться запит: чи зберегти структуру таблиці? Необхідно натиснути кнопку Да. Слідом за цим буде видано ще одне вікно із запитом про необхідність збереження таблиці. У ньому можна набрати конкретне ім'я таблиці і натиснути кнопку ОК чи просто натиснути кнопку ОК. В останньому випадку структура таблиці буде збережена з іменем "Таблица 1". Ми ж введемо ім'я Список.

Аналогічно розглянутому створюється і зберігається структура таблиці Консультация. Нагадаємо, що цей процес починається з вікна База данных (див. мал. 16.9), у якому натискується кнопка Создать. При створенні структури таблиці Консультация використовуються дані, наведені на мал.

Структуру будь-коли можна модифікувати (змінити). У раніше створену структуру в **режимі конструктора** **нове поле вставляють** командою Вставити – Рядки. Для **видалення поля** помічають рядок, який потрібно видалити – Правка – Видалити. Можна змінити порядок розташування полів перетягуючи їх назви вгору чи вниз.

Заповнення даними створених структур таблиць може здійснюватись різ­ними способами.

Наприклад, у вікні запрошення системи вмикається перемикач Открыть базу данных і натискується кнопка ОК. Далі у вікні відкриття файла бази даних необхідно набрати ім'я файла бази даних і натиснути клавішу Enter. На екрані з'явиться вікно бази даних, зображене на мал. У цьому вікні потрібно відкрити ту таблицю, у яку будуть вводитися дані. При цьому не має принципового значення, яку таблицю заповнювати першою.

Далі потрібно натиснути кнопку Открыть. На екрані з'явиться бланк (форма) таблиці, в який необхідно внести відповідні дані. При цьому для збереження даних, що вводяться, не потрібно застосовувати ніяких дій. Ці дані будуть збережені автоматично навіть при виході із системи.

**Стовпці таблиці можна ховати та показувати** за допомогою команди Формат – Скрыть столбцы та Формат – Отобразить столбцы.

Для **фіксації колонок** під час перегляду великої бази даних користуються командою Формат – Закрепить столбцы, щоб вивільнити колонки використовують команду Формат – Освободить столбцы.

**Назву Поля можна змінювати і в режимі таблиці**. Для цього потрібно помітити потрібну колонку, виконати команду Формат – Переименовать столбец.

Для того, щоб **в режимі таблиці добавити нові полі** існує команда Вставка – Столбец

Для **видалення поля** Правка – Удалить столбец та підтвердити видалення. Слід бути обережним, при видаленні поля видаляються також всі дані, які були в нього введені.

Для того, щоб **видалити запис** користуються командоб Правка – Удалить запись.

**Зміна розміру рядка або колонки**

Відкрийте БД режимі таблиці, виконаєте наступнії дії:

**Зміна розміру стовпця**

* Щоб задати конкретну ширину стовпця, встановіть покажчик на праву границю стовпця, як показано на малюнку, і перетягніть границю стовпця до потрібної ширини.
* Для того щоб змінити ширину стовпця по розміру даних, двічі клацніть праву границю області заголовка стовпця.

**Зміна розмірів рядків**

* встановіть покажчик між будь-якими двома областями виділення рядка в лівій частині об'єкта в режимі таблиці, і перетаскуйте покажчик до потрібної висоти рядків.

**Примітка.** Зміну ширини стовпців не можна скасувати за допомогою команди Скасувати з меню Виправлення. Для того щоб скасувати результати цієї операції, закрийте об'єкт у режимі таблиці і натисніть кнопку Нет для усіх у відповідь на запрошення зберегти зміна макета. При цьому будуть також скасовані всі інші зміни макета.

**Зміна шрифту, його стилю, розміру і кольору в режимі таблиці**

* Відкрийте БД в режимі таблиці.
* У меню Формат виберіть команду Шрифт.
* Виберіть потрібні параметри.

**Зміна стилю лінії сітки і кольори тла в режимі таблиці**

* Відкрийте БД в режимі таблиці.
* У меню Формат виберіть команду Режим таблиці.
* Виберіть потрібні параметри.

**Сортування записів у формах і таблицях**

* Виберіть поле, що потрібно використовувати для сортування записів.
* Натисніть кнопку Сортування по зростанню або Сортування по убуванню .

# Форми.

Є два основних способи відображення з баз даних для візуального огляду% у вигляді таблиці та вигляді форми.

Форма подібна до бланка чи карточки. Прикладами форми є анкета, сторінки з паспорта. Один примірник форми містить дані тільки про один об’єкт.

Звичайно бази даних створюють спочатку у вигляді таблиці а потім відображають у вигляді форми.

**Форма** – це об’єкт бази даних, призначений для введення і відображення інформації, яка обов’язково містить елементи (поля) до яких користувач вводить дані.

Перехід до форм, окрім ліпшої візуалізації даних, дає низку додаткових можливостей. На формі можна зручно розташувати:

* поля типу OLE з картинками, фотографіями тощо;
* елементи керування (кнопки, перемикачі тощо);
* надписи (заголовки форм, рубрик); можна розрисувати форму чи задати фоновий рисунок-заставку, зокрема, сутінки, глобус, хмари тощо;
* обчислювальні поля (це також елементи керування) для відображення результатів обчислень, виконаних на базі наявних полів;
* закладки (багатосторінкові форми, де поля групують за змістом на різних закладках);
* підпорядковані форми тощо.

Більшість форм є приєднаними до одній або декільком таблицям і запитам з бази даних. Джерелом записів форми є поля в базових таблицях і запитах. Форма не повинна включати всі поля з кожної таблиці або запиту, на основі яких вона створюється.

Форма одержує дані з базового джерела записів. Інші, що виводяться у формі зведення, такі як заголовок, дата і номери сторінок, зберігаються в макеті форми.

Графічні елементи, такі як лінії і прямокутники, зберігаються в макеті форми.

Дані надходять з полів у базовому джерелі записів.

Результати розрахунків надходять з виражень, що зберігаються в макеті форми.

Текст, що пояснює, зберігається в макеті форми.

# Способи створення форм.

Форму можна створити трьома різними способами.

**1. За допомогою автоформы на основі таблиці або запиту**. За допомогою автоформ можна створювати форми, у яких виводяться всі поля і записи базової таблиці або запиту. Якщо обране джерело записів має зв'язані таблиці або запити, то у формі також будуть присутні всі поля і записи цих джерел записів. Для створення слід виконати наступні дії:

* У вікні бази даних виберіть закладку **Форми** в списку **Об'єкти**.
* Натисніть кнопку **Створити** на панелі інструментів вікна бази даних.
* У діалоговому вікні **Нова** **форма** виберіть **майстра**.

**Автоформа: у стовпець**. Кожне поле розташовується на окремому рядку; підпис знаходиться ліворуч від поля.

**Автоформа: стрічкова**. Поля, що утворять один запис, розташовані в одному рядку; їхні підписи виводяться один раз у верхній частині форми.

**Автоформа: таблична.** Поля записів розташовані у форматі таблиці, де кожного запису відповідає один рядок, а кожному полю один стовпець. Імена полів служать заголовками стовпців.

**Автоформа: зведена таблиця**. Форма відкривається в режимі зведеної таблиці. Є можливість додавати поля шляхом їхнього перетаскування зі списку полів у різні області макета.

**Автоформа: зведена діаграма**. Форма відкривається в режимі зведеної діаграми. Є можливість додавати поля шляхом їхнього перетаскування зі списку полів у різні області макета.

* Виберіть таблицю або запит, що містять дані, на основі яких створюється форма.
* Натисніть кнопку OK.

**Можна також створити автоформу в стовпець на основі відкритого джерела записів або на основі джерела записів, виділеного у вікні бази даних.** Виберіть команду Автоформа в меню Вставка або клацніть стрілці поруч із

**2. За допомогою майстра на основі однієї або декількох таблиць або запитів.** Майстер задає докладні питання про джерела записів, полях, макеті, необхідних форматах і створює форму на підставі отриманих відповідей.

* У вікні бази даних виберіть **Форми** в списку **Об'єкти**.
* Натисніть кнопку **Створити** на панелі інструментів вікна бази даних.
* У діалоговому вікні **Нова форма** виберіть потрібного майстра. Опис майстра з'являється в лівій частині діалогового вікна.
* Виберіть **ім'я таблиці** або іншого джерела записів, що містить дані, на яких повинна бути заснована форма.
* Натисніть кнопку OK.
* Додержуйтеся інструкцій майстра.
* Вибрати поля, які повинні бути в формі кнопкою >(якщо вибираються всі поля, то слід використати кнопку >>)
* Виберіть зовнішній вигляд форми
* Виберіть стиль форми
* Готово

Створену форму можна змінити в режимі конструктора, зведеної таблиці або зведеної діаграми.

**3. Вручну в режимі конструктора.**

Спочатку створюється базова форма, що потім змінюється відповідно до вимог у режимі конструктора.

* У вікні бази даних у списку **Об'єкти** виберіть **Форми** .
* На панелі інструментів вікна бази даних натисніть кнопку **Створити**.
* У діалоговому вікні **Нова форма** виберіть рядок **Конструктор**.
* Виберіть **ім'я таблиці** або іншого джерела записів, що містить дані, на яких потрібно заснувати форму. Якщо форма не буде містити дані (наприклад, якщо потрібно створити кнопкову форму для відкриття інших форм або звітів або якщо потрібно створити користувальницьке діалогове вікно), не вибирайте нічого з цього списку.
* Натисніть кнопку OK.

Форма буде відкрита в режимі конструктора.

# Елементи керування.

Зв'язок між формою і її джерелом записів створюється за допомогою графічних об'єктів, що називають елементами керування. Найбільше часто використовуваним для виведення і введення даних типом елементів керування є поле

Розглянемо можливості конструк­тора форм. Для конструювання форми використовують *панельелементів керування* з кнопками (дивись мал.)

Щоб вставити елемент керування, його треба вибрати на панелі і клацнути на формі у точці вставляння.

За допомогою кнопки Запуск майстра можна в діалоговому режимі вставити у форму інші елементи: кнопки, перемикачі тощо.

Типи елементів керування в Mіcrosoft Access

У Mіcrosoft Access використовуються наступні типи елементів керування, доступні через панель елементів у режимі конструктора форм, звітів і сторінок доступу до даних: поле, напис, група перемикачів, перемикач, прапорець, список, кнопка, набір вкладок, малюнок, лінія, прямокутник і додаткові елементи Actіve. Можна також уставити зведений список Mіcrosoft Offіce, діаграму Offіce або електронну таблицю Offіce у форму, звіт або на сторінку доступу до даних.

Для форм і звітів існують наступні додаткові елементи керування: вимикач, поле зі списком, приєднана рамка об'єкта, вільна рамка об'єкта, набір вкладок, розривши сторінки. У форму або звіт можна також додавати підлеглі форми і звіти. Сторінки доступу до даних також включають наступні компоненти: список, що розкривається, гіперпосилання, що біжить рядок і зведений список, електронна таблиця і діаграма.

Елементи керування являють собою об’єкти у формах, звітах або на сторінках доступу до даних, використовувані для відображення даних, виконання дій або як прикраси. Наприклад, можна використовувати поле у формі, звіті або на сторінці доступу до даних для відображення даних, кнопку у формі для відкриття іншої форми або звіту, лінію або прямокутник для поділу й угруповання елементів керування, що робить більш зрозумілим їхнє призначення.

**Поля**

Поля використовуються у формах, звітах або на сторінках доступу до даних для відображення даних із джерела записів. Поле такого типу називають приєднаним, тому що воно зв'язано з даними в поле. Крім того, існують вільні поля. Наприклад, можна створити вільне поле для відображення результатів обчислень або для прийому даних, що вводяться користувачем. Уміст вільного поля ніде не зберігається.

**Надписи**

****Написи використовуються у формах, звітах або на сторінках доступу до даних для відображення описового тексту, наприклад, заголовків, підписів або коротких інструкцій. Напису не відображають значення з полів або виразів; вони завжди є вільними і не змінюються при переході від одного запису до іншої.

Напис може бути приєднаний до іншого елементоа керування (такий напис називають підписом). Наприклад, поле створюється з приєднаним написом, що містить підпис цього полючи. Цей напис відображається як заголовок стовпця у формі в режимі таблиці. Напис, створений за допомогою інструмента Напис, розміщається окремо і не приєднується ні до якого елемента керування. Такі написи використовуються для відображення різних відомостей, наприклад, заголовків форми або звіту, а також для висновку тексту, що пояснює. Написиу, не приєднані до елементів керування, не відображаються в режимі таблиці.

**Списки, поля зі списком і списки, що розкриваються**

У багатьох випадках зручніше вибрати значення зі списку, чим уводити конкретне значення по пам'яті. Крім того, вибір зі списку дозволяє бути упевненим, що введене значення є припустимим. Якщо у формі або на сторінці доступу до даних досить місця для постійного відображення списку, можна використовувати список. Якщо потрібно створити елемент керування, що відображає список, але вимагає менше місця, використовуйте поле зі списком при роботі з формою або список, що розкривається, при роботі зі сторінкою доступу до даних.

**Списки.** Список містить рядки з даними. У формі список може містити один або кілька стовпців, що можуть бути озаглавленими.

**Поля зі списком.** Поле зі списком є комбінацією двох елементів, поля і списку, що розкривається, і займає менше місця. Користувач може ввести значення в поле або вибрати значення зі списку. У поле зі списком список складається з рядків з даними. Рядка містять один або кілька стовпців, із заголовками або без заголовків.

**Списки, що розкриваються.** На сторінці доступу до даних замість списку можна використовувати список, що розкривається. Список, що розкривається, на сторінці доступу до даних виглядає, як поле зі списком у формі. Як і поле зі списком, що розкривається список показує тільки один запис доти, поки його вміст не буде розгорнуто; однак у список, що розкривається, не можна вводити нові значення. Список, що розкривається, складається з рядків даних. Рядки можуть складати тільки один стовпець, відображуваний без заголовка.

**Кнопки**

Кнопки є зручним засобом виконання дій. Коли користувач клацає кнопку, не тільки виконується відповідна дія, але і на екрані видно, як кнопка натискається і відпускається. На кнопку у формі можна помістити текст або малюнок; на кнопку на сторінці доступу до даних можна помістити тільки малюнок.

Створення кнопок

Є можливість створити кнопку самостійно або за допомогою майстра. Майстер прискорює процес розробки кнопки, автоматично виконуючи всі основні дії. Майстер виводить запрошення ввести необхідні зведення і створює кнопку на основі відповідей користувача. За допомогою майстра можна створювати більш 30 різних типів кнопок. Кнопки створюють для виконання наступних дій:

* набір телефонного номера;
* запуск запиту або макросу;
* запуск або закриття додатка;
* зміна або застосування фільтра;
* печатка або відправлення звіту поштою ;
* печатка поточного запису;
* відновлення даних у формі;
* пошук конкретного запису.

Для того, щоб запустити майстра, потрібно спочатку натиснути кнопку Запуск майстра на панелі інструментів керування, а потім вибрати елемент Кнопка і намалювати кнопку в формі.

**Прапорці**

У формі, звіті або на сторінці доступу до даних прапорець може бути використаний як окремий елемент керування, у якому відображаються значення Так чи Ні з логічного поля з базової таблиці. Крім того, прапорці можна включати в групи параметрів для відображення набору обираних значень.

**Перемикачі**

Перемикач можна використовувати у формах, звітах або на сторінках доступу до даних як окремий елемент керування для відображення логічного (Так/Ні) значення з базового джерела записів.

**Групи перемикачів**

Групи перемикачів можна використовувати у формах, звітах або на сторінках доступу до даних для відображення обмеженого набору альтернатив. Для вибору значення в групі перемикачів досить його клацнути. У кожен момент часу в групі може бути обраний тільки один параметр. Для відображення великої кількості параметрів зручніше використовувати не групу, а список, поле зі списком або список, що розкривається.

**Набори вкладок у формах**

Елемент керування "Набір вкладок" використовується для представлення декількох сторінок даних в одному наборі. Це особливо зручно при роботі з багатьма елементами керування, що можуть бути розподілені на двох або більш категорій.

# Зв’язки між таблицями.

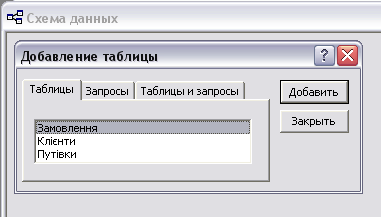
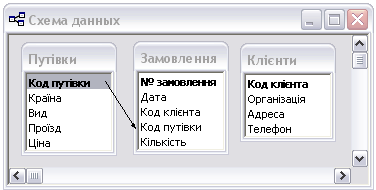
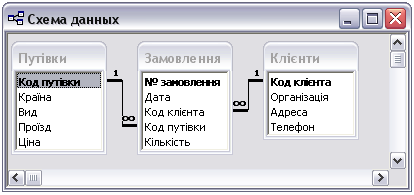
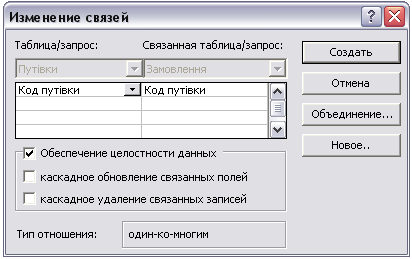
Між декількома таблицями можуть бути налагоджені зв’язки. Власне зв’язані таблиці і бази даних, що їх містять, називають реляційними. Є такі типи зв’язків: одини до одного, один до багатьох, багато до одного і багато до багатьох.

Найпоширенішим є відношення один до багатьох. Яке означає, що одному запису одної таблиці відповідають кілька записів в іншій таблиці. **Наприклад,** є дві бази даних: Путівки і Клієнти. Отже путівку з кодом 2 можна продати декільком клієнтам. Сторона ОДИН (путівки) називається головною таблицею, Сторона Багато (Клієнти) – зв’язаною талицею.

Тип відношень один до одного трапляється рідше. Наприклад, якщо розглядати таблиці Банківські дані, де містяться банківські реквізити клієнтів і таблицю Клієнти, то між полями Рахунок і Код клієнта існувало б відношення один до одного.

**Встановлення зв’язків.**

Отже, між полями таблиць можуть існувати певні відношення. Однак Access про них нічого не відомо доти, поки ми не визначимо зв’язки за допомогою спеціальної процедури. Розглянемо встановлення зв’язків на прикладі „Путівки”, „Клієнти”, Замовлення”

* Починаючи зв'язування таблиць, переконайтеся, що всі таблиці і форми закриті. Потім переключіться у вікно бази даних, натиснувши клавішу F11.
* Виберіть Сервис - Схема даних, або на кнопці  на панелі інструментів вікна Access відкриється однойменне діалогове вікно. Водночас з вікном  
  Схема даних з'явить­ся діалог Добавление таблиці Однак цей діалог може і не з'явитися, якщо раніше були створені зв'язки між таблицями (вікно Схема даних не по-рожнє). У цьому разі потрібно очистити вікно від зв'язків, клац­нувши по кнопці  Очис­тить макет. Потім натисніть кнопку  Доба­вить таблицу на панелі ін-струментів. За замов-чанням в буде відкрита вкладка Таблицы.
* Виділіть у списку таблицю «Замовлення» і натисніть кнопку Добавить. У вікні Схема Данных з'явиться список полів таблиці Замовлення. Аналогічно виведіть списки полів таб-лиць Клієнти і Путівки у вікно Схема данных, після чого закрийте діалог Добав-ление таблицы.
* Для зв'язування полів «Код путівки» оберіть це поле у головній таблиці «Путівки» і перетягніть його ми-шею до зв'язаної таблиці Замовлення. Зазначимо, що нап-рямок перетягуван-ня поля завжди по-винен бути «від головної таблиці до зв'язаної».
* У діалозі Связи вста-новіть перемикач Обеспечение целостно-сти данных. Це дозволить уникнути деяких помилок при створенні й експлуатації бази даних.
* Клацніть по кнопці Создать, і встановлений зв'язок буде відображений у вікні Схема данных. Зв'язок показаний лінією, позначеною цифрою 1 і символом нескінченності ∞, що означає відношення «один до багатьох».
* Аналогічно створіть зв'язок між полями «Код клієнта» таблиць «Клієнти» і «Замовлення». Ви одержите схему зв'язку, показану на мал.

**Макет зв'язків.** Зручним для вас способом (напри-клад, щоб лінії зв'язків не перетиналися) розмістіть спис-ки полів у вікні Схема данных. Списки можна перетягувати, захопивши мишею заголовок списку. Розташування списків у вікні називається ***макетом зв'язків****.* Після цього можете закрити вікно Схема данных. Програма виведе на екран запит щодо того, чи потрібно зберігати макет зв'язків. Клацніть по кнопці Да. Якщо ви відповісте Нет, то збережуться лише створені зв'язки, а не компонування списків.

Зверніть увагу, що в розглянутих випадках ми зв'язували ключове поле (позначене напівжирним шрифтом у списку полів) головної таблиці з відповідним йому полем зв'язаної таблиці. Поле зв'язаної таблиці називають полем зов­нішнього ключа.

Для видалення будь-якого зв'язку не потрібно знову створювати макет зв'яз­ків. Виділіть зв'язок клацанням миші й натисніть клавішу Delete. Після появи запиту на видалення зв'язку клацніть по кнопці Да.

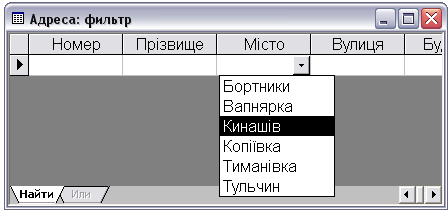
# Використання фільтрів для пошуку даних.

Під час роботи з базою даних важливу роль відіграє пошук та аналіз інформації, яка задовольняє певним умовам (критеріям). Програма Access дозволяє створювати фільтри, за допомогою яких можна ховати непотрібні записи таблиць. У загальному випадку ***фільтрація даних***— це відображення на екрані лише тих записів таблиці, які задовольняють певним умовам. Фі­льтри не є окремими об'єктами бази даних, а лише складовими таблиць.

**Умови (критерії) пошуку**

Умови записують у вигляді: **<логічний оператор> <вираз>.**

*Логічними операторами* можуть бути :



= - дорівнює

< - менше ніж

> - більше ніж

<= - менше або дорівнює

>= - більше або дорівнює

**<>** - не дорівнює

Between and - в діапазоні між

Not - не

Or - виконується хоча б одна умова

And - виконуються усі умови

Like - подібний до.

Часто в умовах використовують символи шаблонів:

\* - заміняє будь-яку кількість символів,

? заміняє один символ.

Логічний оператор = часто опускається.

**Типи фільтрів**

Програма Access використовує три види фільтрації:

* фільтрацію за зразком, тобто відбір даних, які співпадають з виділеним фрагментом, дозволяє вибирати записи, які задовольняють одночасно декільком умовам, накладеним на різні поля;
* просту фільтрацію - пошук інформації, яка задовольняє хоча б одній з заданих умов;
* фільтрацію за допомогою розширеного фільтра - відбір інформації, яка відповідає складним критеріям пошуку.

**Створення та застосування фільтрів**

Для створення фільтра потрібно відобразити таблицю в режимі таблиці.

Перед створенням будь-якого фільтра, потрібно пересвідчитись у то­му, що на дані таблиці не накладено жодних умов, тобто що всі філь­три знищено (при цьому кнопка Применение фильтра ****панелі ін­струментів буде неактивна).

Створити фільтр можна у вікні конструктора простого чи розширено­го фільтрів, або ж шляхом виділення фрагменту поля, за зразком якого необхідно здійснити пошук.

# Простий фільтр

Вікно конструктора простого фільтру відобразиться після виконання команди пункту меню **Записи** *=>* **фільтр** => **Изменить фильтр** або натискання кнопки **Изменить фильтр.** Після цього відкриється вікно конструктора звичайного фільтра.

На екрані відобра-зиться рядок таблиці, кожна комірка якого містить список значень відповідного поля, з яких можна вибрати умову для фільтру. У нижній частині вікна розміщені ярли­ки Найти та Или. Вони вико-ристовуються для ство-рення складних умов, які об'єднані логічним оператором **АБО.**

Для створення критерію відбору даних потрібно вибрати зі списку поля відповідне значення або ж ввести з клавіатури потрібну умову.

Як зазначалося вище, звичайний фільтр дозволяє здійснювати фільт­рування за складними умовами. Умови, зв'язані логічним оператором **І**, заносяться в один рядок конструктора, а умови, зв'язані оператором **АБО,** заносять в новий рядок конструктора, який відобразиться на екрані після натискання ярлика Или. Після формування усіх потрібних умов, на­тискають кнопку **Применение фильтра** - на екрані відобразяться відібрані записи.

Для відображення зняття фільтру вибирають кнопку **Удалить фильтр**(це та сама кнопка, що й применить фильтр, але при застосованому фільтрі вона міняє свою назву на Удалить фільтр)або виконати команду пункту меню **Записи = Фильтр** => **Удалить фильтр.**

*Перед початком створенням нового фільтра, необхідно зни­щити попередній, очистивши всю інформацію у вікні конс­труктора фільтра, інакше результатом фільтрування буде накладання двох або більше фільтрів, тобто виконання сукуп­ності умов, зв 'язаних між собою логічним І.*

# Фільтр за виділеним зразком

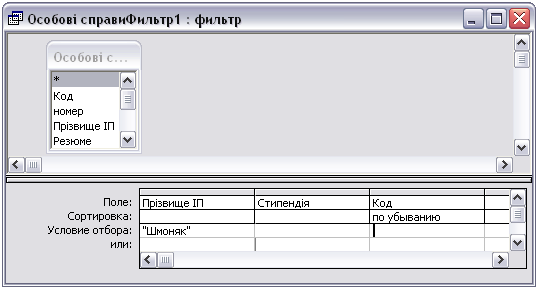
Для того, щоб відобразити записи, в яких значення поля співпадають із виділеним фрагментом, використовують фільтрування за зразком. Для цього виділяють потрібний фрагмент поля та виконують команду пункту меню **Записи** => **Фильтр** або натискають кнопку **Фильтр по выделенному .**

Щоб знову відобразити усі записи, натискають кнопку **Удалить фильтр.**

Умови, які формуються при створенні фільтру за виділеним фрагме­нтом, можна переглянути у вікні простого фільтру. Для цього до­статньо натиснути на кнопку **Изменить фильтр** - на екрані відо­бразиться вікно конструктора простого фільтру з занесеною умовою. Якщо при фільтруванні зразком був фрагмент поля, то умова містить оператор **Like** (подібний на) та символи шаблону \*, який вказує на те, що на його місці може бути будь-яка кількість довільних символів. При створенні умов можна також вико­ристовувати символ ?, який замінює один дові­льний символ.

# Розширений фільтр

*Створення розширеного фільтра -* узагальнений спосіб вибору по­трібної інформації шляхом фільтрування даних.

Розширений фільтр проектується у вікні конструкторі розширеного фільтра, яке відкривається після виконання команди меню **Записи => Фильтр => Расширенный фильтр.**

При цьому у верхній частині вікна конструктора з'явиться вікно таб­лиці зі списком усіх її полів.

Вікно конструктора роз­ши­реного фільтру складається з двох частин: верхньої, в якій розміщено вкладене вікно таб­лиці, з даними якої працює ко­ристувач в даний момент (міс­тить назву таблиці та список усіх її полів), та нижньої - *бланка фільтра,* де формують критерій відбору та впорядкування.

*У вікні конструктора розширеного фільтра можна побачи­ти всі критерії відбору даних, які встановлювались при ство­ренні звичайного фільтра чи фільтра за зразком.* Для створення критерію фільтрації, потрібно у поле бланку *Поле* вве­сти назву поля таблиці, на яке накладатиметься умова. Вказану дію можна зробити одним із способів:

* двічі клацнувши по полю у вікні таблиці, яке знаходиться у верх­ній частині вікна;
* виділивши поле (чи декілька полів за допомогою клавіші **[Ctrl])** у вікні таблиці та перетягнувши його, зачепивши йо­го курсором миші;
* вибравши потрібне поле зі списку, який відкриється у комірці ря­дка *Поле.*

Для створення складного критерію у бланк фільтра вносять назви усіх тих полів, на які накладаються умови.

На перетині рядка *Условие отбора* та відповідного поля потрібно ввести критерій. Можна одночасно передбачити сортування від фільтрова­них записів, задавши відповідну опцію у рядку *Сортировка.*

# Запити

У СУБД Ассеss для пошуку та відбору інформації зазвичай вико­ристовують запити - об'єкти, які окрім пошуку потрібних даних, дозволя­ють змінювати їх та аналізувати. Часто вони використовуються як джерела записів для створення форм та звітів.

**Запит** – це засіб відшукання записів, різних маніпуляцій з даними, зокрема, перетворення таблиць і створення на їх основі нових таблиць.

Порівняно з фільтрами, запити мають набагато більші можливості. Окрім того, фільтр - це частина таблиці, **запит - окремий об'єкт**, тому во­ни працюють з даними таблиці, не залежно від того, чи були внесені якісь зміни у інформацію, чи ні. Фільтр лише ховає непотрібні записи, а запит дозволяє відображати лише певні поля таблиці.

Результатом виконання запитів зазвичай є динамічний (тимчасовий) масив даних. У такій таблиці можна додавати, змінювати та знищувати записи, проте такі зміни не відображаються на даних основних таблиць. Сам запит можна зберегти як окремий об'єкт, але дані, отримані при його виконанні, не зберігаються (якщо не задана спеціальна опція для збереження динамічної таблиці).

Запити за своїм призначенням та результатами відбору поділяються на:

* запити на вибірку;
* запити на змінювання (доповнення, знищення, оновлення, створен­ня нової таблиці);
* параметричні;
* перехресні.

***Запит*** *на вибірку* дозволяє вибрати з бази даних інформацію, яка ці­кавить користувача в даний момент.

Іноді при створенні запитів необхідно використовувати поля, які на­ лежать відразу декільком таблицям бази даних. Такі запити називають **реляційними**. їх можна використовувати для створення віртуальних таблиць,  
тобто набору даних, що належать різним таблицям, але виглядають як одна таблиця.

За допомогою **запитів на змінювання**користувач може змінювати дані в таблиці, додавати, знищувати групу записів, створювати нові табли­ці. Відміна цих запитів від запитів на вибірку полягає в тому, що вони не просто фільтрують дані, а виконують певні дії над відфільтрованими да­ними. Запити на змінювання найчастіше використовуються для знищення, поновлення чи архівування записів, а також для виконання обчислень).

Використовуючи **запит на оновлення***,* користувач може змінити гру­пу записів, яка відібрана на основі певних критеріїв. Ці запити корисні у випадку внесення змін у певний набір даних.

До запитів на змінювання належать **запити на доповнення**однієї таб­лиці записами іншої. Вони можуть працювати як з таблицями однієї бази даних, так і декількох. Таблиці при цьому не обов'язково мають ідентичну структуру, але типи полів повинні бути однаковими.

**Запити на знищення**ліквідують в таблиці всі записи, які відповіда­ють певній умові. Такі запити також часто використовуються для архіву­вання даних.

**Запит на створення нової таблиці**дозволяє з динамічної таблиці створити нову таблицю бази даних. Часто такі запити використовуються для архівування або створення резервних копій таблиць.

Якщо потрібно об'єднати дані у форматі рядків-стовпців (тобто у ви­гляді двомірної таблиці), створюють **перехресні запити.**Такі запити ком­пактно відображають дані та об'єднують однорідну інформацію. Таблиці, в яких відображаються результати виконання перехресного запиту, зручні для створення діаграм та графіків.

**Робота із запитами**

Запит можна відкрити в режимі таблиці (запустити запит на вико­нання) і в режимі конструктора.

Перебуваючи у вікні бази даних на вкладці Запроси, **запустити за­пит на виконання**можна:

* виділивши потрібний запит, натиснути кнопку **Открьіть** панелі інструментів вікна бази даних;
* двічі клацнувши курсором миші по піктограмі запиту.

**Відкрити запит в режимі конструктора**можна, виділивши потріб­ний запит на вкладці Запросы, натиснути кнопку **Конструктор** панелі інструментів вікна бази даних. Перейти з режиму конструктора в режим таб­лиці можна за допомогою кнопки **Запуск** панелі інструментів конструктора запитів або елемента спи­ску кнопки **Вид.**

**Створення запитів**

Перед створенням запиту користувач повинен чітко усвідомити, який аналіз йому потрібно провести із записами таблиці. Залежно від потреби, визначається тип створюваного запиту, а також структура таблиці, яка утвориться в результаті виконання запиту. Програма Mirosoft Access 2000 дозволяє автоматично створювати запити за допомогою спеціальних про­грам - майстрів створення запитів. Створений майстром запит можна ви­дозмінити, відкривши його в режимі конструктора (режим конструктора дозволяє повністю створювати будь-який запит, не звертаючись до послуг майстра, проте це потребує певних умінь від користувача).

Створення запиту розпочинається з вибору таблиці (або декількох таб­лиць), на основі якої буде створюватись запит. Залежно від завдання, яке по­винен виконувати запит, користувач з полів таблиці створює структуру результуючої динамічної таблиці. Після цього створюються умови (критерії"), які будуть виконувати фільтрування, сортування, заміну чи обчислення у за­писах таблиці.

# Прості запити

**Створення простого запиту** можливо здійснити за допомогою майстра або в режимі конструктора. Під час ознайомлення з технологією запитів ми використовуємо майстер простих запитів. Дотримуючись його інструкцій ви можете обрати потрібну таблицю і поля даних, переглянути результати відбору на екрані. Розглянемо бази фірми Подорож. Наприклад нас цікавить інформація про організації, які замовляли путівкаи та про їх кількість. Такий запит повинен бути зроблений на основі баз даних Замовлення та клієнти. (див. малюнок в розділі макет зв’язків)

1. Відкрийте вікно БД та перейдіть на вкла-дку Запроси, натисніть кнопку Создать. У вікні, що відкриється виберіть кнопку Простой запрос. ОК
2. Виберіть табли-цю Замовлення в розгор-таючому списку Табли-цы/Запросы
3. Виберіть в списку Доступні поля поле Код клієнта.
4. Оберіть у верхньому списку Таблицы/запросы опцію «Таблица: Клієнти»і перемістіть позицію «Організація» із лівого списку до правого.
5. Знову встановіть у списку опцію «Таблица: Замовлення» і перемістіть з лівого списку до правого поле «Кількість». Потім клацніть по кнопці Далее.
6. У наступному діалозі залиште установленим перемикач подробньїй (вьівод каждого поля каждой записи) і клацніть по кнопці Готово.
7. У вікні Access з'явиться запит, що містить дані трьох зазначених полів із двох зв’язаних таблиць.

# Перехресний запит

Ефективним засобом аналізу даних є перехресний запит. Цей вид запиту до­зволяє згрупувати дані рядків або стовпців і вивести підсумкові значення до окре­мого стовпця.

Розглянемо приклад простого запиту, показаного на мал. В ньому клієнти Технікум, Телеком і Школа представлені кількома рядками. Зручніше було б мати таблицю запиту, в якій дані замовлень кожного клієнта бу­ли б підсумовані, а загальна кількість замовлень умістилася б до окремого стовп­ця. Найпростіше це зробити за допомогою перехресного запиту.

* Відкрийте вікно бази даних, перейдіть на вкладку Запросы і клацніть по кнопці Создать.
* У діалозі Новый запрос зазначте опцію Перекрестный запрос і натисніть кнопку ОК.
* Відкриється перше діалогове вікно майстра перехресних запитів. Клацніть по переми-качу запросы і виділіть у списку існуючих запитів позицію «За-мовлення: Запрос». На-тисніть кнопку Далее.
* Наступний діалог при-значений для задання заголовків рядків у май-бут­ньому запиті. У спис-ку **Доступные поля** оберіть позицію «Орган-ізація», натисніть кноп-ку >, і зазначена позиція переміститься до списку **Выбранные поля.** Клацніть по кнопці **Далее.**
* Третій діалог майстра перехресних запитів служить для задання заголовків стовпців. У списку **Доступные поля** виділіть позицію «Код клієнта» і натисніть кнопку **Далее.**
* У діалозі потрібно задати ті обчис-лення, які виконуватимуться в комір-ках на перетині рядків і стовпців. Оберіть опцію Sum, яка у нашому прикладі встановить підсумовування за значеннями стовпця «Кількість». Клацніть по кнопці **Далее.**
* В останньому діалозі майстра введіть ім'я запиту, наприклад «Сума замовлень». Натисніть кнопку **Готово** й отримайте таблицю запиту, показану на малюнку.
* У стовпчику підсумкових значень показано загальну кількість путівок, замовлених кожною організацією. Стовпці праворуч мають заголовки кодів клієнтів і містять значення, на основі яких сформований стовпець підсумків.

# Використання виразів у запитах

При формуванні запиту ви можете задати обробку даних, наприклад знайти суму або середнє значення для будь-якого поля. Для цього використовуються *вирази,* за якими виконуються обчислення, а результати обчислень заносяться в окреме поле. У виразах можна вживати знаки арифметичних операцій +, -, \*, /, оператори порівняння =, <, >, <=, >=, а також імена полів, які взяті у квадратні дужки. Наприклад, вираз [Ціна]\*1,25 означає, що вміст поля **Цена** збільшується у 1,25 разу.

Надалі ми використовуватимемо вирази при заданні умов відбору і під час формування поля, що обчислюється в запиті.

**Умови відбору**

Запит, сформований згідно з вказівками пункту «Створення запиту», містить всі замовлення путівок. Якщо ж вас цікавлять лише великі замовлення (кількість путівок перевищує певне число), краще сформувати запит із заданням умови відбору у такий спосіб:

* У вікні бази даних перейдіть на вкладку Запросы і відкрийте **«Замовлення:** Запрос». У вигляді конструктора.
* У діалозі (див мал.) наведено схему даних для розглянутих таблиць, нижче - бланк запиту. Клацніть по комірці, розташованій на перехресті рядка **Условие отбора** і стовпця «Кількість». Введіть із клавіатури вираз «>3» і натисніть Enter

Клацніть по кнопці **Вид,** і ви побачите таблицю із записами, що задоволь­няють умові відбору. Після введення умови відбору кількість записів у запиті зменшилася до трьох

# Розрахунки в запиті

Проілюструємо виконання розрахунків на прикладі запиту, сформованого на основі таблиць «Замовлення» і «Путівки». Нас цікавитиме сума кожного замо­влення, що обчислюється як добуток ціни путівки та кількості путівок: [Ціна]\*[Кількість]. Виконується подібний запит таким чином.

* Спочатку сформуйте простий запит, показаний на мал. По­трібно використовувати поля таблиць «Замовлення» і «Путівки».
* Клацніть правою кнопкою миші в зоні поля з виразом і оберіть у кон­текстному меню команду **Свойства.** У діалозі **Свойства поля** задайте формат поля **С разделителем (два** десяткові знаки після коми) і назву поля «Сума».
* Натисніть кнопку **Вид** і пе-рейдіть у **Режим таблиці. Ви** отримаєте запит в останньому стовпці якого буде зазначена сума кожного замовлення

Отже, на основі таблиць бази даних ви отримали запит, у якому було виведено обчислюване поле - сума всіх зроблених замовлень на путівки. Роз­рахунки виконуються безпосередньо при виведенні запиту. Результати обчи­слень у таблицях не зберігаються. Тому результати запиту завжди представ­ляють поточний вміст бази даних.

# Редагування запитів

Редагувати запит можна лише в режимі конструктора. До зміни стру­ктури запиту належать:

* зміна критеріїв;
* зміна параметрів сортування записів;
* зміна опції для відображення чи ховання полів;
* додавання, знищення чи зміна обчислювальних виразів;
* додавання, знищення чи перенесення полів.

Для того, щоб *додати* у *запит поле,* перетягують його зі списку полів вікна таблиці, яке знаходиться у верхній частині конструктора запиту, у відповідну колонку бланку конструктора запиту. При цьому поле, яке там знаходилося, а також усі поля правіше від нього, переміщаються вправо.

Для *вилучення поля* зі структури запиту, виділяють його та натиска­ють клавішу *Del* або виконують команду **Правка** => **Удалить.**

Для переміщення поля на інше місце використовують технологію пе­ретягування об'єктів. Після переміщення інші поля також можуть зміститися.

# Створення звіту

Для представлення даних бази в звичайному вигляді потрібно створити *звіт.* Для цього в Access передбачено декілька засобів. Ми розглянемо майстер звітів, конструктор і майстер діаграм.

Найпростіше скласти звіт за допомогою майстра звітів. Складемо звіт про клієнтів фірми «Подорож» і надрукуємо його.

* Відкрийте базу даних натисканням клавіші. Перейдіть на вкладку Отчеты
* Клацніть по кнопці Создать й у діалозі Новый отчет оберіть позицію Мастер отчетов. У розкривному списку вкажіть таблицю «Клієнти», на основі якої буде створюватися звіт. Клацніть по кнопці ОК.
* У діалозі Создание отчетов перемістіть усі поля зі списку Доступньїе поля до списку Вьібранньїе поля, після чого натисніть кнопку Далее.
* Наступний діалог призна­чений для задання рівнів групування для полів звіту. Можете прийняти установки за умовчанням. Натисніть кнопку Далее.
* Дотримуйтесь вказівок майстра, задаючи в на­ ступних діалогах порядок сортування полів, вид макета, стиль оформлення звіту, а також ім'я звіту. На завершення клацніть по кнопці Готово.
* Ви отримаєте звіт, виведений у вікно перегляду