## Тема 4. Системи управління базами даних

***Короткі теоретичні відомості***

БАЗА ДАНИХ (БД) – це поіменована сукупність структурованих даних, що відносяться до визначеної предметної області.

*Система управління базами даних –* це комплекс програмних і мовних засобів, призначених для створення баз даних, оновлення та організації пошуку в них необхідної інформації.

**ОСНОВНІ МОЖЛИВОСТІ СУБД**

*Введення даних, поновлення, поповнення і розширення БД*

*Висока надійність зберігання інформації*

*Засоби захисту інформації в БД*

*Виведення повної і вірогідної інформації на запит*

*Типи БЗ (за способом організації даних):*

* *Реляційні БД* – це БД, дані в яких подані у вигляді двомірних таблиць, за допомогою яких можна описати предмет
* В *ієрархічній БД* всі елементи розташовуються у послідовності від вищого до нижчого.
* В *мережній БД* до вертикальних ієрархічних зв’язків додаються горизонтальні. Тому мережна БД є більш гнучкою у здійсненні пошуку потрібних даних у область.

MICROSOFT ACCESS – це система управління реляційними БД, призначена для роботи в середовищі Windows. Працюючи з Access, користувач може розв'язувати наступні завдання:

* розбивати дані на логічно пов'язані між собою частини;
* вводити, змінювати, видаляти необхідні дані;
* знаходити підмножини даних за заданими умовами;
* створювати форми і звіти;
* автоматизувати виконання стандартних завдань;
* графічно встановлювати зв'язки між даними;
* вставляти малюнки у форми і звіти;
* створювати власні, готові до роботи з БД форми, які вміщують меню, діалогові вікна і командні кнопки.

*Запис БД* – це рядок таблиці, конкретна реалізація (значення) поняття предметної області.

*Поле БД* – це стовпець у таблиці даних, властивість (атрибут) даного поняття предметної області.

Для кожного поля задається тип даних, що можуть знаходитися в ньому. Записи в таблиці відрізняються значеннями своїх полів.

*Ключове поле* – це стовпець таблиці, що дозволяє встановити зв’язок із  записами в іншій таблиці. Значення ключового поля однозначно визначає кожен запис у таблиці. Первинний ключ не допускає значень Null і завжди повинен мати унікальний індекс.

Існує три типи ключів:

*Поле лічильника* (Тип даних "Лічильник"). Тип даних поля в базі даних, у якому для кожного запису, що додається в таблицю, у полі автоматично заноситься унікальне числове значення.

*Простий ключ*. Якщо поле містить унікальні значення, такі як коди або інвентарні номери, то це поле можна визначити як первинний ключ. У якості ключа можна визначити будь-яке поле, що містить дані, якщо це поле не містить повторювані значення або значення Null.

*Складений ключ*. У випадках, коли неможливо гарантувати унікальність значень кожного поля, існує можливість створити ключ, що складається з декількох полів.

Зв’язок установлюється між двома полями (стовпцями) двох таблиць. Існують зв’язки з відношенням "один-до-одного", "один-до-багатьох", "багато до одного" та "багато-до-багатьох":

* "один-до-одного", кожному запису з однієї таблиці відповідає один запис в іншій таблиці (створюється в тому випадку, коли обоє поля, що зв’язуються, є ключовими);
* "один-до-багатьох", кожному запису з однієї таблиці відповідає кілька записів  іншій таблиці (створюється в тому випадку, коли тільки одне з полів є полем первинного ключа);
* "багато до одного" – кілька записів однієї таблиці пов’язуються з одним записом до іншої таблиці;
* "багато до багатьох" – дає змогу встановити відношення між кількома записами однієї таблиці та кількома записами іншої.

*Створення бази даних*

*Перший спосіб* дозволяє на основі вибраного зразка (шаблону) створити файл нової бази даних з таблицями, формами та звітами, при цьому користувач не втручається у процес налаштування майбутньої бази даних (присвоєння властивостей полям, проектування форм чи звітів).

*Другий спосіб* дозволяє користувачеві самому визначити структуру бази даних та кожного окремого об'єкта, використовуючи або шаблони окремих об'єктів, або майстри проектування, або режим конструктора.

*Об**'єкти бази даних*

|  |  |
| --- | --- |
| *Об'єкт* | *Опис* |
| *Таблиця* | Містить дані у вигляді двомірної таблиці. Таблиці – основа БД, від них залежать всі інші об’єкти |
| *Запит* | Вибирає дані з таблиці на основі заданої умови |
| *Форма* | Вибирає дані з таблиці або запиту на підставі описаного користувачем формату. Форми дозволяють переглядати, редагувати і друкувати дані.  У формі можна відображати дані, взяті з декількох таблиць або запитів |
| *Звіт* | Служить для відображення підсумкових даних з таблиць та запитів у зручному для перегляду та друку вигляді, а також для проведення аналізу даних |
| *Макроси* | Автоматизує стандартні дії на підставі вибраних користувачем команд і подій |
| *Програмний модуль* | Активізує складні операції, які не можна описати з допомогою макрокоманд. Програмні модулі – це процедури написані мовою програмування Access Basic |
| *Сторінка*  *(Веб-сторінка)* | Перетворює будь-який об’єкт БД у веб-сторінку і дозволяє іншим користувачам переглядати, вводити та аналізувати дані в зовнішній або внутрішній мережі. |

*Проектування бази даних*

Проектування БД для ефективної обробки даних у конкретній предметній області – досить складний процес. Тільки в самих простих випадках достатньо однієї таблиці для опису предметної області. Звичайно вимагається декілька взаємопов’язаних таблиць, тобто значення поля однієї таблиці використовується в іншій таблиці.

**ОСНОВНІ** **ПРАВИЛА, ЯКИХ СЛІД ДОТРИМУВАТИСЯ ПРИ РОЗРОБЦІ БД**

Визначити структуру майбутньої бази даних, визначити джерело даних, продумати завдання, що слід вирішити за допомогою створюваної БД

визначивши дані й завдання, поділити їх на групи – вони стануть таблицями

визначити поля для кожної таблиці

виділити ключові поля (загальні для всіх таблиць)

продумати оформлення форм і звітів

визначити умови вибору для запитів

*Створення таблиць*

У вікні БД можна за допомогою вкладки "Створити" групи "Таблиці" створити таблицю одним із таких способів:

|  |  |
| --- | --- |
| *Спосіб* | *Опис* |
| *Режим таблиці* | Виводить бланк (форму) абстрактної таблиці куди можна безпосередньо вводити дані |
| *Конструктор* | У режимі конструктора можна безпосередньо вказати параметри всіх елементів структури таблиці |

Незалежно від способу створення таблиці режим конструктора дозволяє в будь-який момент змінити структуру таблиці, наприклад, додати нові поля, вказати значення за умовчанням або створити маски

введення.

*Додавання та редагування даних*

Додавання записів в існуючу таблицю здійснюється в режимі таблиці. Щоб відкрити таблицю в цьому режимі, необхідно двічі клацнути мишкою на імені потрібної таблиці у вікні БД або скористатися командою "Відкрити" з контекстного меню.

Для редагування даних необхідно активізувати потрібне поле таблиці і ввести або відредагувати дані. Під час редагування ліворуч від запису з'явиться зображення олівця як ознака того, що запис редагується. Як тільки користувач залишає рядок, зміни записуються в таблицю.

*Зміна структури таблиці*

Структуру таблиці можна змінювати лише до наповнення її даними. Якщо в таблицю вже введено дані, то зміни в структурі можуть зумовити втрату даних або змінити структуру таблиці буде неможливо.

|  |  |
| --- | --- |
| *Дія* | *Способи виконання* |
| *Зміна назви поля* | Виділити стару назву, знищити її та ввести нову |
| *Зміна типу поля* | Потрібно помістити курсор у комірку, яка знаходиться в стовпці "Тип даних" у рядку з назвою поля та вибрати зі спадаючого списку елемент відповідного типу даних. |
| *Зміна властивостей поля* | Для зміни параметра поля потрібно помістити курсор у комірку з назвою властивості та вибрати відповідне значення зі спадаючого списку чи ввести його з клавіатури. |
| *Вставка пропущеного поля* | Виділити поле, перед яким слід вставити нове поле, клацнувши мишкою на кнопці ліворуч від назви поля та виконати одну із команд: скористатися кнопкою "Вставити рядки" вкладки "Робота з таблицями – Конструктор" або скористатися командою "Вставити рядки" з контекстного меню. |
| *Видалення поля* | Щоб видалити поле, спочатку потрібно його виділити, потім виконати команду "Видалити рядок" вкладки "Робота з таблицями – Конструктор" або скористатися клавішею "Delete" |
| *Зміна послідовності полів в таблиці* | Виділити поле, яке необхідно перемістити, і, тримаючи натиснутою ліву кнопку миші (вказівник миші при цьому набуває форми стрілки з стрілки з пунктирним прямокутником), переміщають його так, щоб контрастна лінія опинилася між тими полями, де повинно знаходитись переміщене поле. Дія закінчується відпусканням кнопки миші. |
| *Копіювання полів* | Щоб скопіювати поле, спочатку потрібно його виділити, потім виконати команду "Копіювати" з контекстного меню та встановивши курсор у рядок проекту таблиці, активізувати команду "Вставити" з контекстного меню |
| *Переключення між*  *режимом конструктора і режимом таблиці* | Кнопка  вкладки "Робота з таблицями – Конструктор" (в режимі таблиці) |
| Кнопка  вкладки "Робота з таблицями – Конструктор" (в режимі конструктору) |
| *Створення або зміна*  *ключового поля* | Кнопка  вкладки "Робота з таблицями - Конструктор" (в режимі конструктору) або скористатися командою "Ключове поле" з контекстного меню. |
| *Створення зв’язків між*  *таблицями* | * Закрити всі таблиці * Скористатися кнопкою  групи "Відношення" вкладки "Робота з базами даних" * Добавити всі таблиці в поле "Схема даних", скористатися кнопкою "Додати" з діалогового вікна "Додавання таблиць", попередньо виділивши їх. * Вибрати поле в одній таблиці й перетягти його за допомогою миші на відповідне поле в другій таблиці або скористатися кнопкою  закладки "Робота зі зв’язками – Конструктор" * Активізувати всі параметри в діалоговому вікні "Зміна зв'язків" |

*Переваги використання ключових полів*

* *Швидкість роботи.* Використовуючи ключ, Access створює індекси, що дозволяють прискорити роботу запитів і ряду інших функцій.
* *Упорядкування.* Access автоматично сортує і відображає записи БД в порядку зростання або зменшення ключового поля.
* *Відсутність збігів.* Access не дозволяє користувачу вводити дані з таким само ключем, як і в існуючого запису.
* *Зв’язки.* Access організує зв’язки між таблицями, використовуючи загальний ключ.

**Увага!** *Зв’язані поля не обов’язково повинні мати однакові імена, однак вони повинні мати однакові типи даних і мати вміст одного типу.*

*Сортування та фільтрація даних*

**Увага!** *Сортування та фільтрація даних виконується в режимі таблиці.*

|  |  |
| --- | --- |
| *Дія* | *Способи виконання* |
| *Сортування даних у полі* | Для впорядкування записів за даними одного поля курсор поміщаємо у це поле та використовуємо відповідну команду  або  з групи "Сортування та фільтр" вкладки "Головна". |
| *Накладення фільтра на записи згідно з виділеним фрагментом* | Для цього виділяють потрібний фрагмент поля та виконують команду  з групи "Сортування та фільтр" вкладки "Головна". |
| *Застосування фільтру* | Для цього поміщаємо курсор у поле, по якому буде відбуватися пошук даних, та виконуємо команду  з групи "Сортування та фільтр" вкладки "Головна", вказавши умову та значення відбору. |
| *Відміна фільтру (відновлення вмісту таблиці)* | Для відновлення вмісту таблиці потрібно виконати команду  з групи "Сортування та фільтр" вкладки "Головна". |

*Умови (критерії) пошуку*

|  |  |
| --- | --- |
| *Логічні оператори* | *Опис* |
| *=* | Дорівнює |
| *<* | Менше ніж |
| *>* | Більше ніж |
| *<=* | Менше або дорівнює |
| *>=* | Більше або дорівнює |
| *<>* | Не дорівнює |
| *Between and* | В діапазоні між |
| *Not* | Не |
| *Or* | Виконується одна із умов |
| *And* | Виконуються всі умови |
| *Like* | Подібний до |

**Увага!** *Часто в умовах використовують символи шаблонів \* (замінює будь-яку кількість символів) та ? (замінює один символ)*

*Запити*

ЗАПИТИ – це умови, за допомогою яких можна скласти набір необхідних полів (у тому числі і з декількох таблиць) і задати такі значення полів, за якими вибираються записи для обробки, тобто задати фільтр відбору інформації.

|  |  |
| --- | --- |
| *Тип запиту* | *Опис* |
| *Запит на вибірку* | Дані, що задовольнять умови відбору, вибираються з однієї або декількох таблиць і виводяться в певному порядку. |
| *Запит з параметрами* | При створенні запиту необхідно у рядку "Умова відбору" бланку конструктору ввести не конкретне значення параметра, а текст у квадратних дужках (наприклад, [Введіть значення параметру]) |
| *Перехресні запити* | Відображають результати статистичних розрахунків, виконаних за даними з одного поля. Ці результати групуються за двома наборами даних у форматі перехресної таблиці, створюючи номери рядків і заголовки стовпців |
| *Запит на змінювання (доповнення, знищення, оновлення, створення нової таблиці)* | Дозволяють в одній операції внести зміни в багато записів. |
| *Запит SQL* | Запитом SQL називають запит, що створюється за допомогою інструкції SQL (мови структурних запитів). Прикладами запитів SQL є запит на об’єднання, запит до серверу, управляючий запит та ін. |

**Увага!** *Під час створення запитів над даними полів таблиці можна здійснювати різного роду операції (математичні, логічні тощо); змінювати значення у полі для всіх записів одночасно, тобто здійснювати перерахунок для всіх записів по визначеному полю; проводити аналіз та обчислення лише над групами записів. Для цього створюють обчислювальні поля, тобто поля які містять результати обчислення.*

Створити *обчислювальне поле* можна у вікні бланку конструктору, ввівши відповідний вираз (формулу, за якою треба обчислити значення; причому у виразі поля беруться у квадратні дужки) у першу вільну комірку рядка "Поле". Після цього з'явиться нове поле з назвою "Вираз", яку можна змінити.

Деякі поради щодо проектування запитів:

* визначити, яка таблиця (таблиці) містить необхідну інформацію;
* визначити зв’язки таблиць (чи мають ці таблиці відповідні ключі);
* визначити тип запиту, що буде створюватися;
* виділити умови і критерії полів, які повинні задовольняти записи;
* продумати необхідні обчислення;
* визначити порядок сортування;
* визначити ім’я, під яким буде збережений запит.

*Форми*

ФОРМИ СУБД – це засіб відображення інформації з БД із використанням елементів управління для зручної і наочної роботи з даними – введення, редагування, відображення. Форми складають інтерфейс БД.

ЕЛЕМЕНТИ УПРАВЛІННЯ СУБД – це графічні елементи (поле, кнопка, малюнок), які розміщуються у формі або звіті для відображення даних або для прикрашання.

Принципи використання форм

* форми можуть відображати тільки один запис за один раз, звичайно у простому форматі;
* форми можуть відображати як поля, що можна редагувати, так і незмінні поля;
* форми можна розробити схожими на звичні паперові бланки;
* форми дозволяють змінити порядок дотримання полів, щоб полегшити введення даних і зробити його більш акуратним;
* форми можуть містити дані з декількох таблиць;
* форми можуть містити графіки і діаграми;
* форми дозволяють автоматизувати задачі введення і створювати меню користувача.

**Увага!** *Для побудови форм* *зручніше скористатися Майстром для швидкого створення макету з набором стандартних елементів управління форми. Після цього можна доопрацювати його в режимі конструктора форми.*

*Звіти*

ЗВІТ– це засіб створення і друку підсумкових документів за наявною у БД інформацією. Звіт може бути створений на основі таблиць і запитів.

Складові частини звіту:

* *Заголовок*. Інформація на початку першої сторінки (назва).
* *Верхній колонтитул*. Інформація на початку кожної сторінки заголовки стовпців таблиць).
* *Область даних*. Відображення даних із таблиць або запитів.
* *Примітка групи*. Інформація в кінці групи даних (підсумок за групою).
* *Нижній колонтитул*. Інформація в кінці кожної сторінки (номер сторінки).
* *Область приміток звіту*. Інформація в кінці останньої сторінки (підсумкові обчислення по всіх записах звіту).