Лабораторна робота 1
Концептуальне проектування баз даних

Мета роботи. Познайомити студентів з основними методиками, прийомами і технологічними інструментами для стадії концептуального проектування у процесі розробки інформаційної системи.

ЗАВДАННЯ

1. Виділіть об'єкти і характеристики проектованої інформаційної бази даних
2. Визначте основні класи сутностей
3. Визначте зв'язку між сутностями і їх типи
4. Вивчити принципи нормалізації таблиць баз даних
5. Побудуйте концептуальну модель, приведену до 3НФ

ПОРЯДОК ВИКОНАННЯ РОБОТИ

1. Вивчити за додатком матеріал роботи
2. Виконати пункти завдання
3. Відповісти на контрольні питання

ТЕОРЕТИЧНА ЧАСТИНА

Концептуальну модель системи побудуємо у формі ER — моделі (модель "сутність — зв'язок") — широко використовуваного інструменту розробки баз даних, що є формалізованим графічним представленням предметної області у вигляді прямокутників, ліній і спеціальних символів.

Перевага ER — моделі полягає в тому, що вона зрозуміла і розробникам баз даних, і замовникам, тобто майбутнім її користувачам, і тому усі можливі зміни в структурі даних узгоджуються між ними саме на рівні цієї моделі. ER — модель ніяк не пов'язана ні з якою конкретною СКБД, цей вибір здійснюється згодом. У основі ER — моделі лежать поняття "сутність"("entity") і "зв'язок" ("relation").

Можна сказати, що це — сукупність значущих об'єктів, існуючих у рамках предметної області. Наприклад, при побудові системи управління контактами видавництва і авторів доведеться оперувати

сутями "Співробітник", "Видання", а у разі створення банківської системи — сутностями "Клієнт" і "Рахунок".

Сутності прийнято іменувати іменникам в однині.

Кожна сутність включає набір атрибутів, що є іменованими властивостями сутності.

Наприклад, сутність " Співробітник" може мати атрибути:

1. Прізвище
2. Ім'я
3. Дата народження
4. Табельний номер
5.ISBN видань

Сутність — сукупність об'єктів, а кожен об'єкт цієї сукупності — екземпляр сутності. З точки зору ООП, сутність — це клас об'єктів, а кожен об'єкт цієї сукупності — екземпляр класу.

Екземпляри мають бути помітні, тобто не повинно бути двох екземплярів з однаковими наборами атрибутів. Кожна сутність має набір атрибутів, унікальний для кожного екземпляра сутності, званий ключем.

Наприклад, для сутності "автор" ключ складається з одного атрибуту "Табельний номер", оскільки точно не існує двох і більше студентів з однаковим значенням цього атрибуту.

Між сутями можуть бути встановлені зв'язки, що визначають, як сутності співвідносяться або взаємодіють між собою. Зв'язок може існувати між двома сутностями (бінарний зв'язок) або між суттю і нею ж самою (рекурсивний зв'язок). Наявність зв'язку означає, що одна з них має атрибут або набір атрибутів складовий ключ іншої сутності. Наприклад, сутність "автор" і сутність "видання" пов'язані між собою атрибутом "ISBN видань групи", який є ключем сутності "Видання" і присутній в наборі атрибутів сутності "Автор".

Отже, для побудови ER — моделі необхідно спочатку виявити сутності, характерні для заданої предметної області і визначити для кожної сутності набір атрибутів, що мають значення виходячи з поставленого завдання. Потім необхідно визначити все наявні зв'язки між сутностями, характеристики цих зв'язків. Після цього будується ER – діаграма, тобто власне графічне зображення моделі з використанням однієї з існуючих нотацій. У цій роботі ми дотримуватимемося нотації Microsoft Entity Framework.

ER — діаграмі зображуються у вигляді прямокутників, усередині яких перераховуються імена атрибутів. Атрибути підкреслюються. Ім'я сутності вказується в прямокутнику першим і відділяється від списку атрибутів рисою. Зв'язки між сутностями зображуються у вигляді ліній, що сполучають прямокутники відповідних сутностей. Для сутності, що знаходиться з боку "багато", лінія зв'язку повинна закінчуватися значком з трьох ліній, що розходяться. Необов'язковість зв'язку позначається значком, що має на кінці невелике коло.

При проектуванні БД зручно користуватися різними угодами про найменування. Microsoft використовує наступні.

Імена сутностей завжди являються іменниками в однині.

Первинний ключ завжди має вигляд Id.

Усі їх властивості розділені на дві групи:

пов'язані безпосередньо з атрибутами сутності : Id,, і т. д.

навігаційні – службові, покажчики на інші сутності.

Останні створені виходячи з аналізу зв'язків сутностей. Вони дозволяють зручно і просто запрошувати пов'язану с даного тільки зв'язку, але і їх тип. При цьому символ "зірочка" означає необмежене число елементів.

Для прикладу, для Видавництва, можна виділити наступні сутності.

1. Видання
2. Співробітник

Атрибутами сутності "Видання" будуть:

1. ISBN.
2.Название,
3. Рік видання
4.Сотрудники

Ключем є ISBN.

Сутність "Співробітник" матиме атрибути:

1. Табельний номер;
2. Прізвище;
3. Ім'я;
4. По батькові.
5. Спеціальність

6. Видання

Ключем сутності, природно, являється табельний номер співробітника.

Нагадаємо, що ключем сутності є атрибут або набір атрибутів, значення яких є унікальними для кожного екземпляра сутності. Тепер розглянемо зв'язки, наявні між сутностями.

1. Оскільки над кожним виданням можуть працювати декілька співробітників, а співробітник може працювати над декількома виданнями, то між сутностями "Співробітник" і "Видання" є зв'язок "багато до багатьох". Оскільки відразу ж після прийому на роботу співробітник може ще не бути пов'язаним роботою ні над одним виданням, а при формуванні плану видань можуть бути не призначені виконавці, то зв'язок необов'язковий з обох боків.

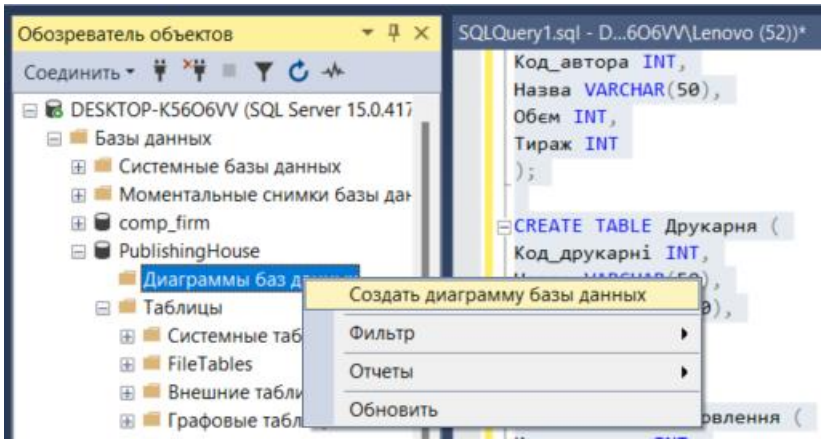
Побудову ER – діаграми проводимо в середовищі MS Management Studio.

1. *Створити базу даних «PublishigHouse».*
2. *Створити таблиці*:

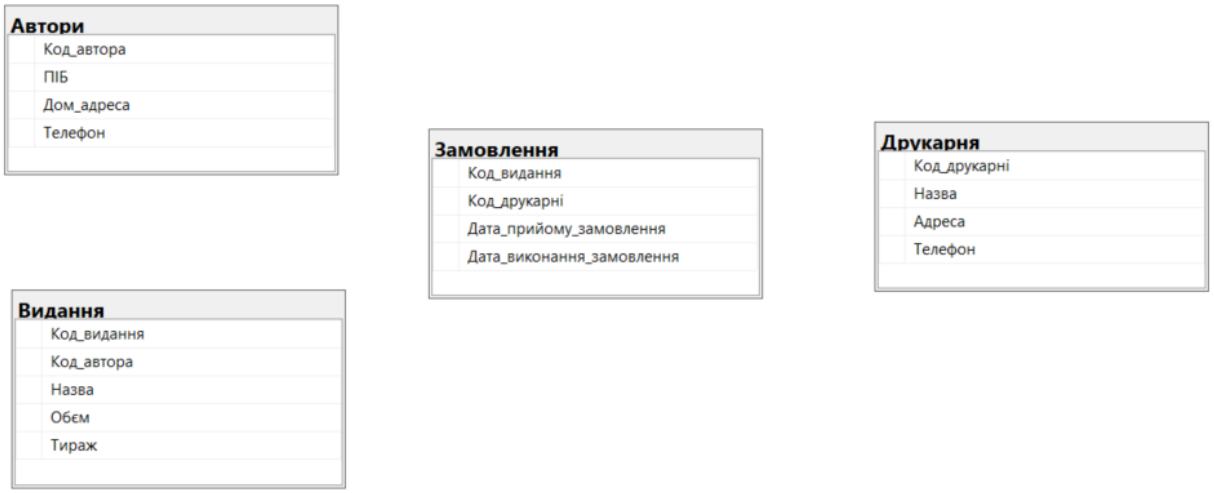
Автори (Authors) (Код_автора, ПІБ, Дом_адреса, Телефон, E-mail);
Видання (Editions) (Код_видання, Код_автора, Назва, Об’єм, Тираж);
Друкарня (Printing) (Код_друкарні, Назва, Адреса, Телефон, E-mail),
Замовлення (Opders) (Код_видання, Код_друкарні, Дата_прийому_замовлення, Дата_виконання_замовлення)

Під час створення таблиць не забуваємо вказувати тип даних.

3. *Створюємо діаграму бази даних*.



Створена діаграма повинна мати такий вигляд



4. *Збережіть діаграму*.

Примітка: При спробі зберегти таблицю після внесення змін до таблиці за допомогою конструктора SQL Server Management Studio може з'явитися таке повідомлення про помилку:

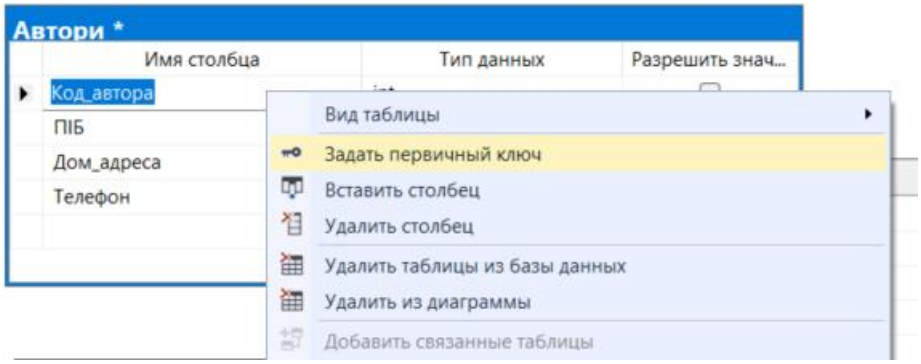
Збереження змін заборонено. Внесені зміни вимагають видалення та повторного створення наступних таблиць. Ви або внесли зміни до таблиці, яку неможливо повторно створити, або увімкнули параметр «Заборонити збереження змін, які потребують повторного створення таблиці».

Ця проблема виникає через те, що параметр Заборонити збереження змін, які вимагають повторного створення таблиці, включений за замовчуванням у SQL Server Management Studio.

Щоб змінити параметр Заборонити збереження змін, які потребують повторного створення таблиці, виконайте такі дії:

- Відкрийте середовище SQL Server Management Studio.
- У меню Сервіс клацніть Параметри.
- Натисніть Конструктори в області навігації вікна Параметри.
- Встановіть або зніміть прапорець Заборонити збереження змін, які потребують повторного створення таблиці, а потім натисніть кнопку ОК.

5. *Встановити первинні ключі*.



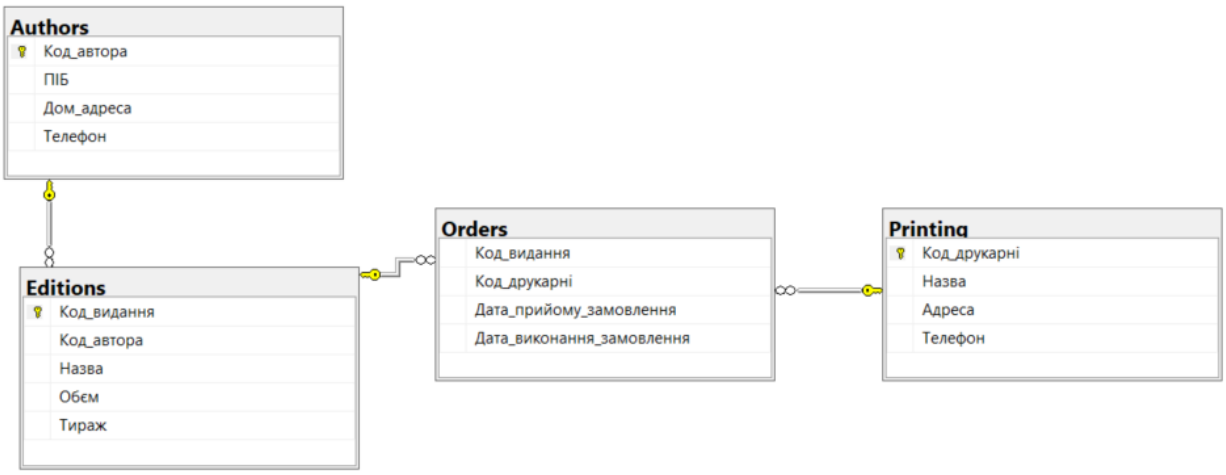
6. *Встановити зв'язок між таблицями*.

Зв'язки між стовпцями різних таблиць у конструкторі діаграм можна створювати, перетягуючи стовпці між таблицями.

Графічне створення зв'язку

- У конструкторі баз даних клацніть селектор рядків для одного або більше стовпців бази даних, які потрібно зв'язати зі стовпцем в іншій таблиці.
- Перетягніть вибрані стовпці у пов'язану таблицю.
- Відображаються два діалогові вікна: Зв'язок по зовнішньому ключу та Таблиці та стовпці, друге відображається на передньому плані.
- Ім'я зв'язку встановлюється системою у форматі FK_локальна_таблиця_таблиця_зовнішнього_ключа. Це значення можна змінити.
- Переконайтеся, що таблиця первинного ключа правильно задає таблицю.
- Сітка містить локальні стовпці та відповідні їм зовнішні стовпці. Можна додати або видалити стовпці таблиці або змінити зіставлення.
- Натисніть кнопку ОК.
- Відкривається діалогове вікно Зв'язок із зовнішнього ключа. Вибраний зв'язок відображає створений зв'язок.
- Змініть властивості зв'язку у сітці.
- Натисніть кнопку ОК, щоб створити зв'язок.

Після встановлення взаємозв’язків, діаграма має мати такий вигляд:



7. *Заповнити таблиці*. У кожній таблиці має бути не менше п'яти записів.

