

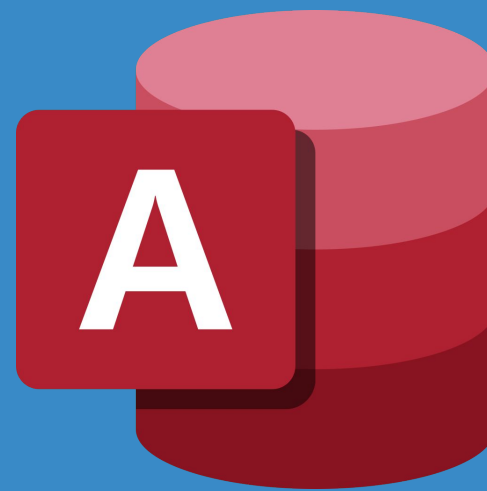


ІНФОРМАТИКА

Ключі й зовнішні ключі. Зв'язки між записами і таблицями

За навчальною програмою 2018 року

10



Урок 18



Пригадай

- *структуру електронної таблиці;*
- *типи даних клітинки електронної таблиці в середовищі табличного процесора.*

Ти дізнаєшся

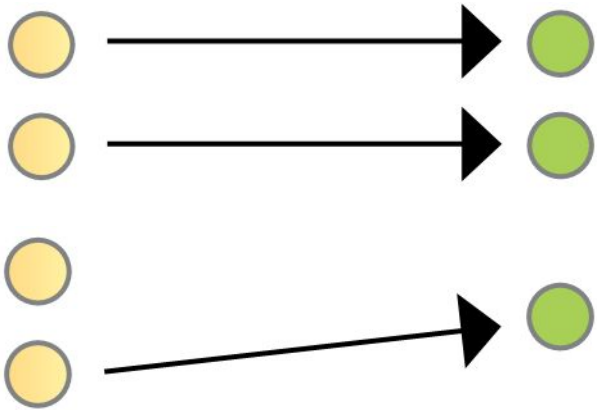
- *зв'язки яких типів можуть існувати між сутностями;*
- *як побудувати модель даних предметної області;*
- *чим відрізняються таблиці баз даних і таблиці табличного процесора;*
- *як змінити структуру таблиці бази даних;*
- *як налаштувати параметри зв'язків між таблицями.*

Зв'язки яких типів можуть існувати між сутностями?

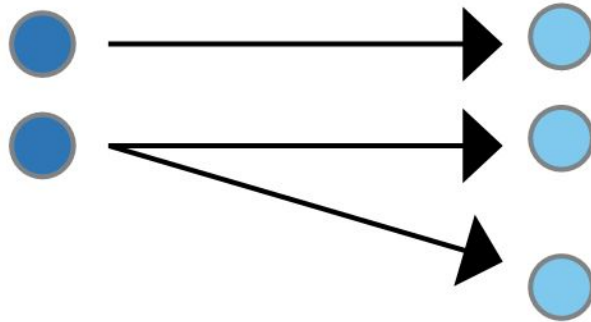


Між двома сутностями, наприклад, А і Б, можливі чотири типи зв'язків:

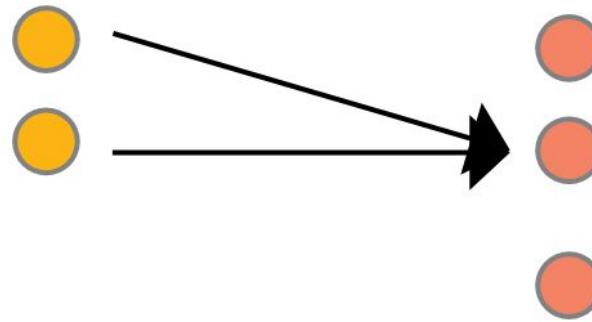
один-до-одного



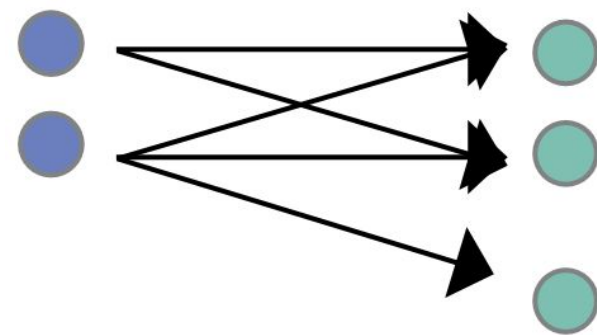
один-до-багатьох



багато-до-одного



багато-до-багатьох



Зв'язки яких типів можуть існувати між сутностями?



«**один-до-одного**» (1:1) означає, що в кожний момент часу кожному екземпляру сутності А відповідає 1 або 0 екземплярів сутності Б;



1

Код	Прізвище	Ім'я
1	Іванов	Кузьма
2	Петров	Василь
...		

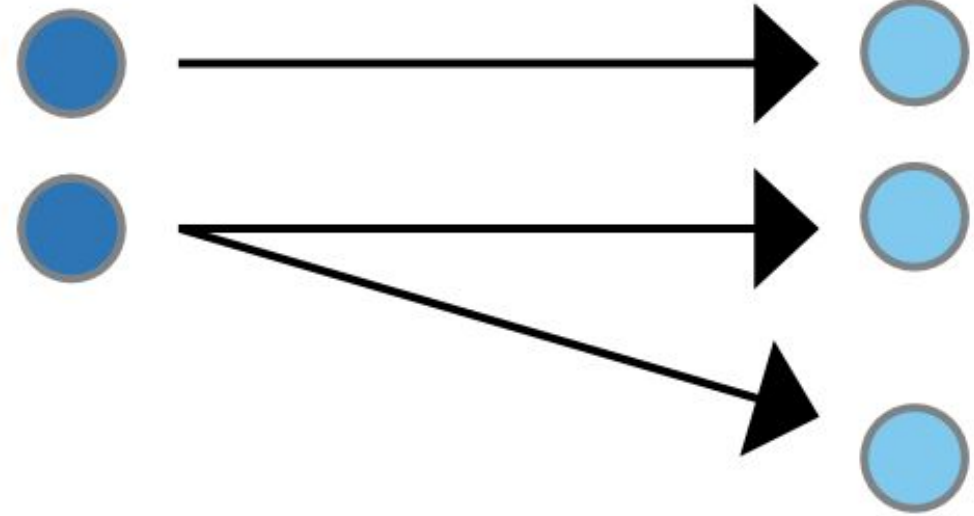
1

Код	Рік народження	Адреса
1	1992	Шевченка, б.20, кв. 6
2	1993	Франка вул., б. 30, кв. 18
...		

Зв'язки яких типів можуть існувати між сутностями?



«один-до-багатьох»
(1:N) — одному
екземпляру сутності А
відповідає 0, 1 або
більше екземплярів
сутності Б;



товари

Код	Назва
1	Монітор
2	Вінчестер
...	

Б

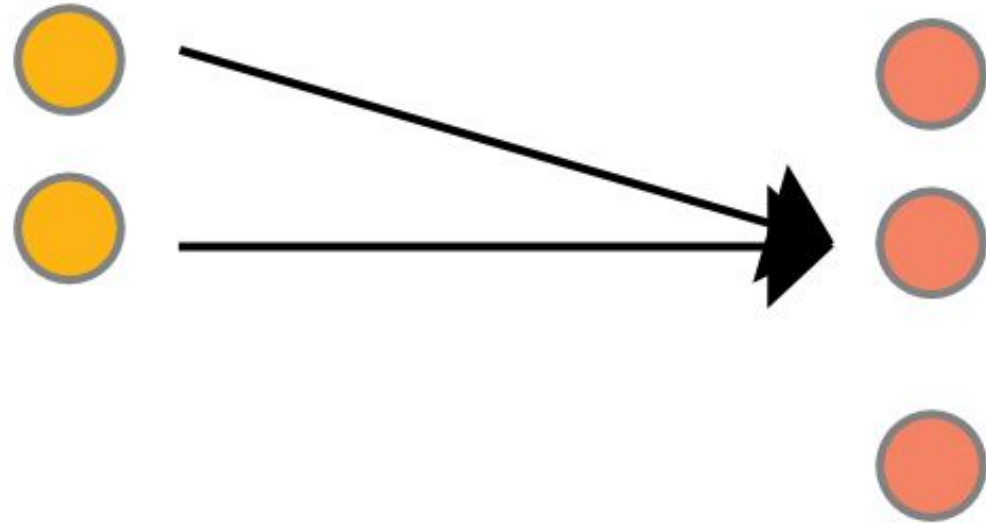
Код	Код товару	Ціна
123	1	2 999
345	1	2 499
...		

прайс-лист

Зв'язки яких типів можуть існувати між сутностями?



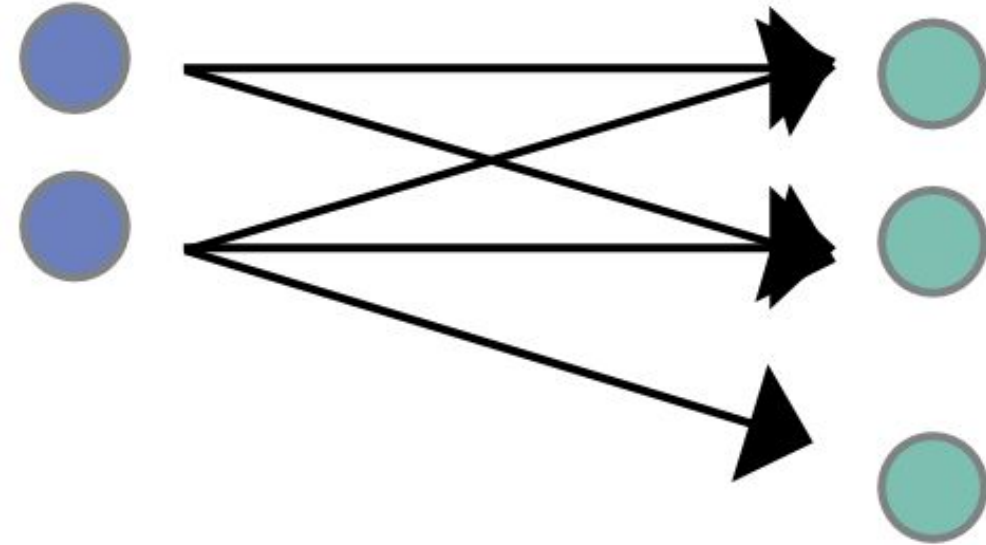
«багато-до-одного»
(N:1) – обернений
випадок до
попереднього;



Зв'язки яких типів можуть існувати між сутностями?



«багато-до-багатьох» (N:M) — одному екземпляру сутності А відповідає 0, 1 або більше екземплярів сутності Б, і навпаки, одному екземпляру сутності Б відповідає 0, 1 або більше екземплярів сутності А.



вчителі

Код	Прізвище
1	Іванов
2	Петров
...	

Б

Б

Код	Назва
1	Історія
2	Географія
3	Біологія
...	

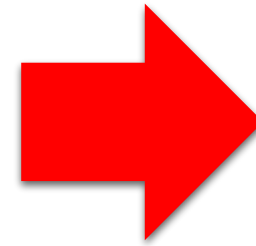
предмети

Зв'язки яких типів можуть існувати між сутностями?



Наведемо декілька прикладів.

Учень отримав у шкільній бібліотеці підручник з алгебри для 10 класу. Зв'язок між читацьким формуляром учня та книгою обліку підручників — «ОДИН-ДО-ОДНОГО**», оскільки один учень користується підручником з таким інвентарним номером.**

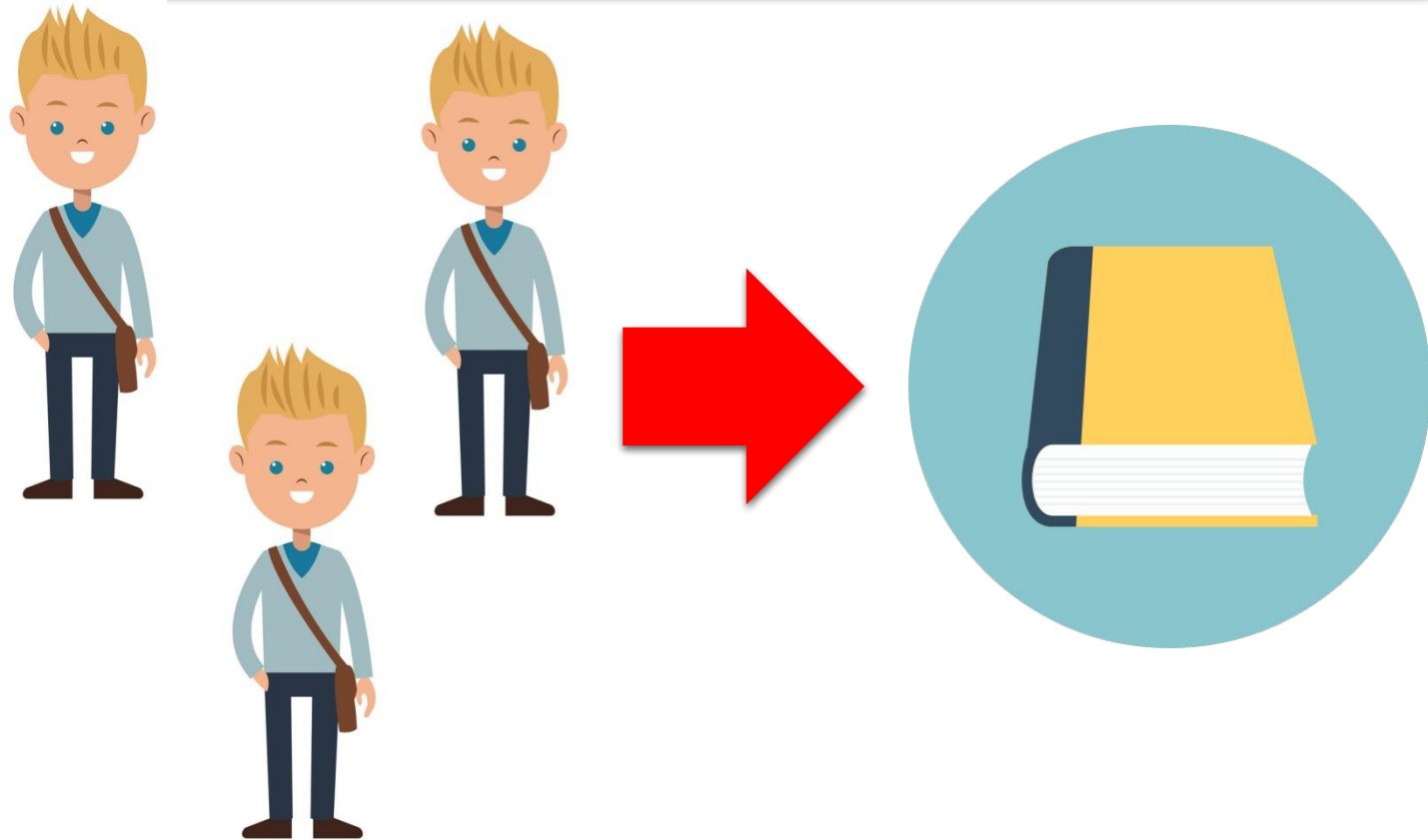


Зв'язки яких типів можуть існувати між сутностями?



На уроках української літератури учні вивчають творчість сучасного українського поета.

У бібліотеці твори цього автора видають на короткий проміжок часу. Тому в багатьох читацьких формулярах є запис про одну й ту саму книжку (маємо зв'язок «багато-до-одного»).



Зв'язки яких типів можуть існувати між сутностями?



У декількох читацьких формулярах є записи щодо користування бібліотечним фондом за різними рубриками: підручники, художня література, періодичні видання (зв'язок «багато-до-багатьох»).

В одному читацькому формулярі записано дані про всі підручники, якими користується читач протягом навчального року (зв'язок «один-до-багатьох»).



Зв'язки яких типів можуть існувати між сутностями?



Звичайний зв'язок у реляційних базах даних — це «**один-до-багатьох**».

Якщо ви виявили зв'язок між вашими таблицями «**один-до-одного**», то швидше за все потрібно переглянути структуру. Можливо те, що ви виділили як дві окремі сутності, насправді є однією, і їх потрібно об'єднати в одну таблицю.

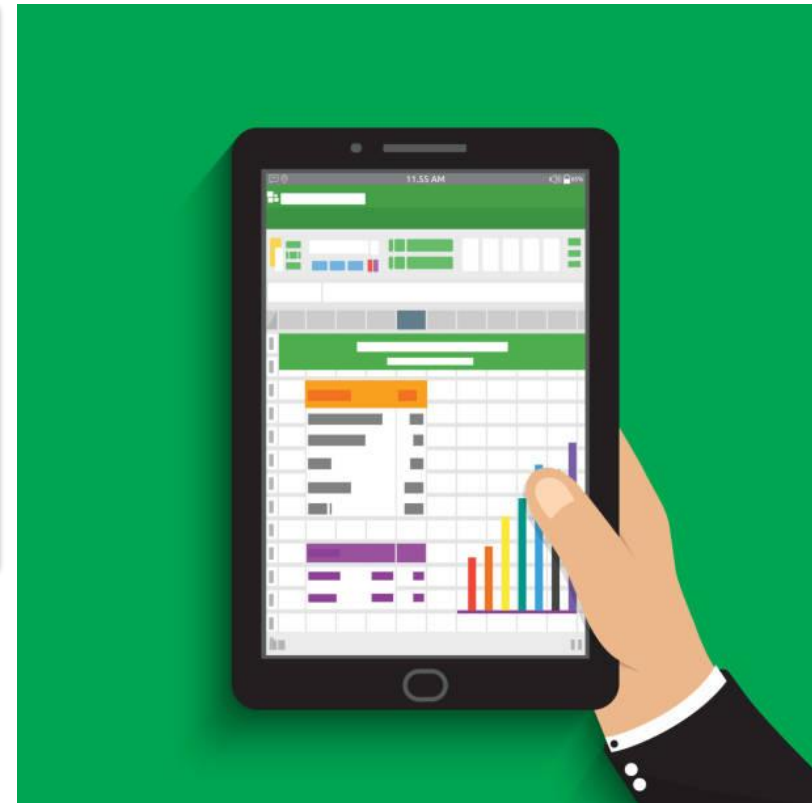


Зв'язки яких типів можуть існувати між сутностями?



При зв'язку «багато-до-багатьох» необхідно також переглянути структуру та пов'язати дві сутності таким чином, щоб зв'язок між ними був «**один-до-багатьох**».

Якщо серед атрибутів не виявлено потрібної пари для зв'язку, то в цьому разі створюють так звану зв'язуючу таблицю, яка має зв'язки з кожною із сутностей як «**один-до-багатьох**».





Процес побудови моделі даних предметної області належить до абстрактного моделювання і, як правило, поділяється на кілька етапів, що базуються на системному підході. Ці етапи розглянемо на прикладі.

*Побудуємо модель бази даних **Бібліотека**.*





1. Визначення мети створення бази даних.

На першому етапі побудови моделі необхідно визначити:

- ✓ **мету створення бази даних**
- ✓ **основні її функції**
- ✓ **набір даних, які будуть зберігатися**



Тобто визначити основний зміст таблиць бази даних і дані, що зберігатимуться в полях таблиць.



Під час роботи з базою даних Бібліотека користувач може шукати відповіді на такі запитання:

 Чи є в бібліотеці конкретна книжка?

 Чи є книжка в наявності?

 Книжки яких авторів даної тематики є в бібліотеці?

 Які книжки даної тематики видано в заданий період?

 Які видавництва публікують книжки даної тематики?

 Яке видавництво видало дану книжку?



Таким чином, можна так сформулювати мету створення цієї бази даних:

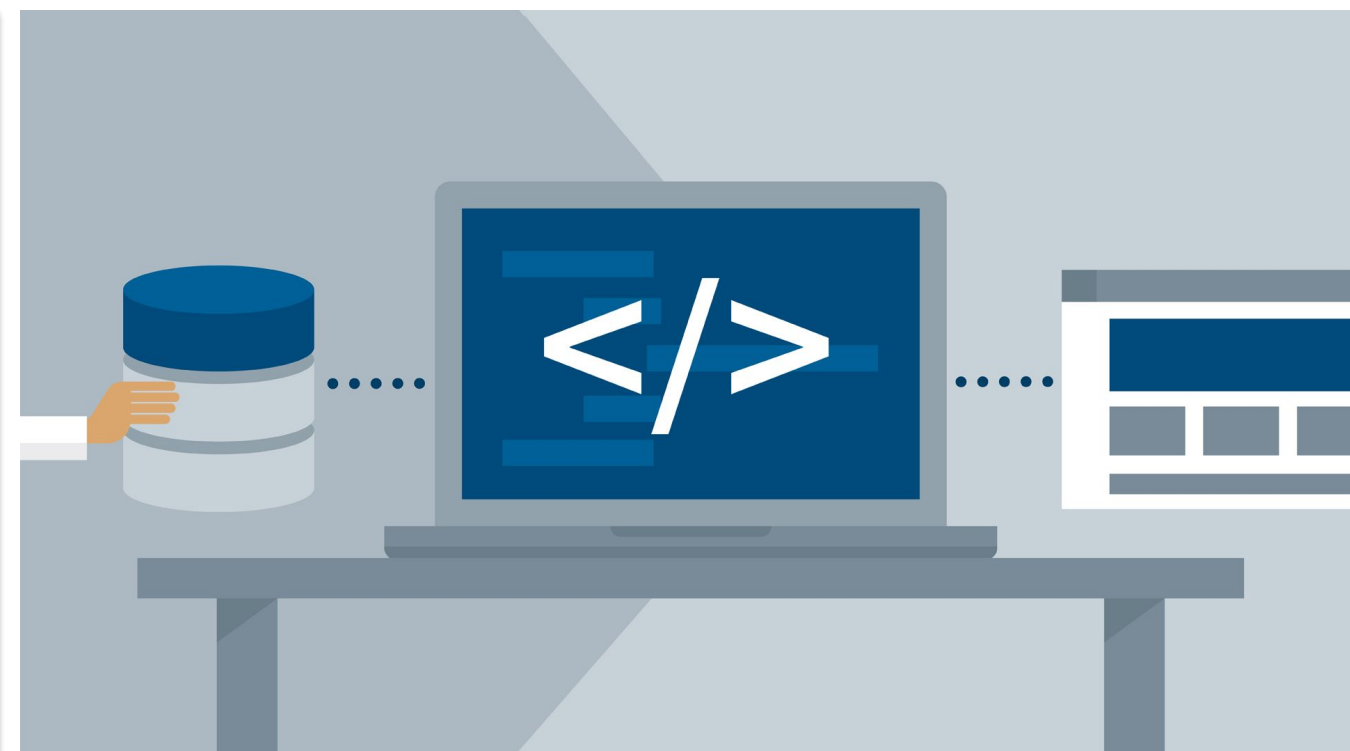
зберігати дані про книжки, авторів та видавництва для подальшого задоволення запитів користувачів.





2. Розробка таблиць, з яких складатиметься база даних.

Одним з найскладніших етапів у процесі побудови моделі є розробка таблиць, оскільки очікувані результати не завжди дають повне уявлення про структуру таблиць.



Як побудувати модель даних предметної області?



Наприклад, відносно бази даних Бібліотека можна виділити три об'єкти (сутності), характеристики яких можна зберігати в таблицях, а саме: книжка, автор, видавництво. Причому, з огляду на перший етап, для кожного об'єкта можна виділити такі атрибути:

КНИЖКА	<i>назва книжки, автор, кількість сторінок, видавництво, рік видання, ціна, наявність у сховищі, тематика, ключові слова</i>
АВТОР	<i>прізвище, ім'я, по батькові, жанр творчості</i>
ВИДАВНИЦТВО	<i>назва, адреса, спеціалізація, сайт</i>

Як побудувати модель даних предметної області?



Для опису кожного виділеного об'єкта доцільно побудувати таблицю, до якої внести імена атрибутів об'єктів і типи значень цих атрибутів, а також обмеження на їх значення.

Під час проектування таблиць спочатку краще розробити їх структуру на папері, при цьому доцільно користуватися такими основними правилами.

Дані в таблиці не повинні дублюватися, також не має бути повторень між таблицями.

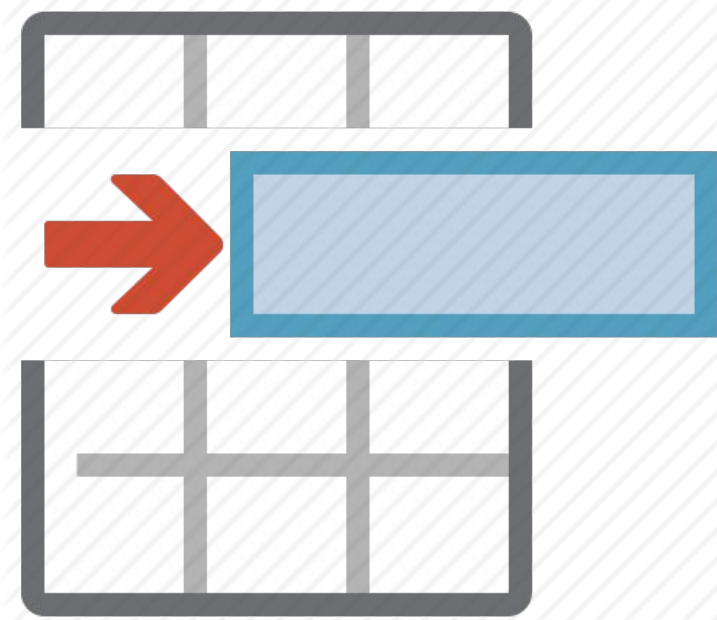


Як побудувати модель даних предметної області?



Кожна таблиця має містити дані лише з однієї теми. Дані з кожної теми опрацьовуються значно легше, якщо вони містяться в незалежних одна від одної таблицях.

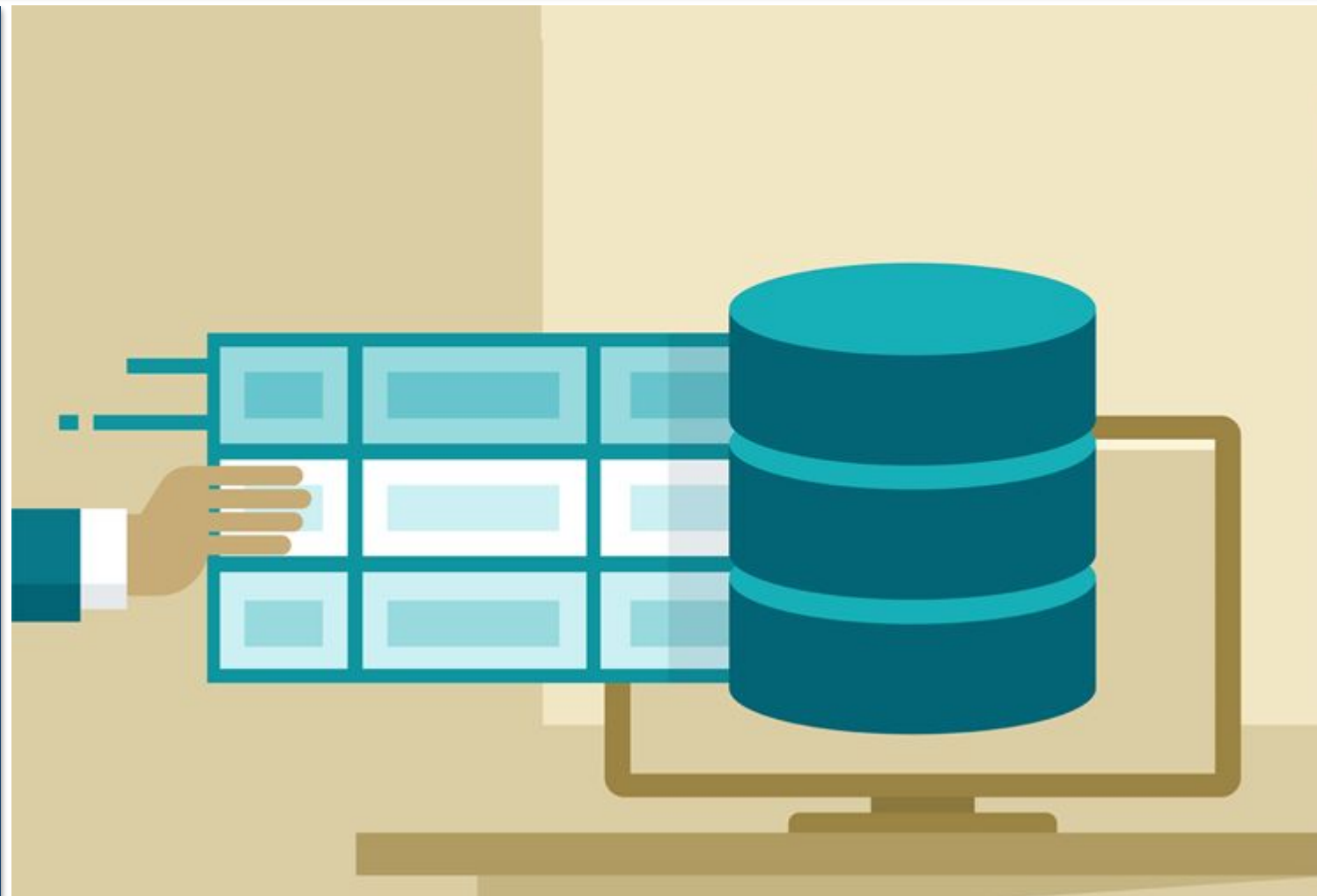
Наприклад, адреса видавництва й відомості про книжки зберігаються в різних таблицях — з тим, щоб при видаленні відомостей про книжку дані стосовно видавництва залишалися в базі даних, а при закритті видавництва відомості про книжки, що видані ним раніше, не були також видалені.





3. Визначення полів таблиці.

Кожна таблиця містить дані окремої теми, а кожне поле в таблиці містить окреме значення. Наприклад, у таблиці з даними про авторів можуть міститися поля з прізвищем, адресою та номером телефону.



Як побудувати модель даних предметної області?



Під час розроблення полів для кожної таблиці необхідно пам'ятати:

□ кожне поле має відповідати темі таблиці;

□ не рекомендується включати до таблиці дані, які є результатом обчислення значень виразів;

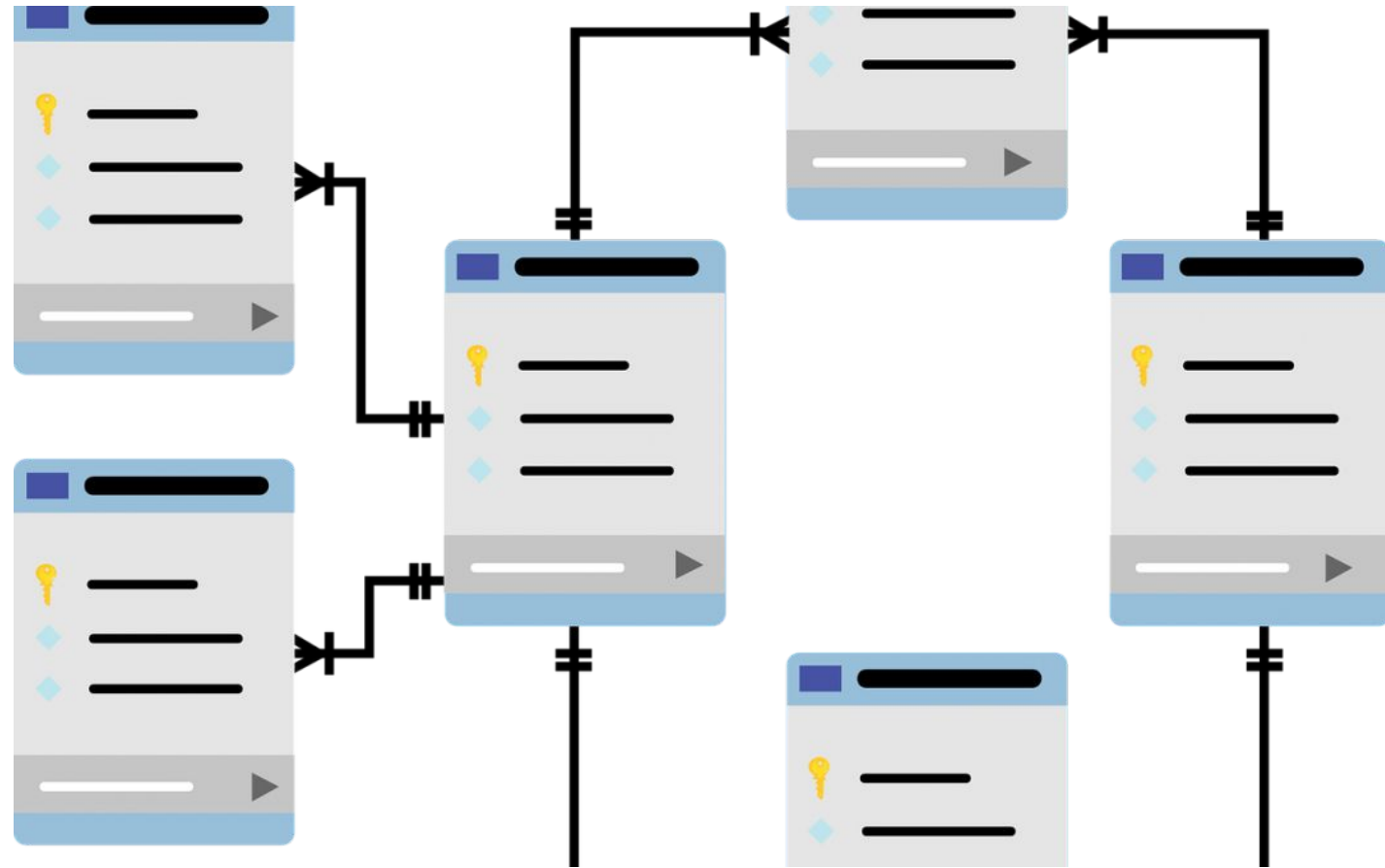
□ у таблиці слід розміщувати всі необхідні дані;

□ дані потрібно ділити на найменші логічні одиниці (наприклад, поля Ім'я і Прізвище, а не загальне поле ПІБ, поля Місто, Вулиця, Будинок, Квартира, а не загальне поле Адреса).



4. Визначення ключа таблиці.

*Кожна таблиця має містити поле або набір полів, які задаватимуть унікальне значення кожному запису в таблиці, за яким можна точно визначити потрібний запис. Таке поле або набір полів називають **ОСНОВНИМ КЛЮЧЕМ.***





5. Визначення зв'язків між таблицями.

Після розподілу даних у таблицях і визначення ключових полів необхідно вибрати схему для зв'язування даних у різних таблицях. Для цього слід визначити зв'язки між таблицями. Наприклад,

Автор — Книжка

Видавництво — Книжка

зв'язок 1:N

зв'язок 1:N



6. Оновлення структури бази даних.

Після опису таблиць, полів і зв'язків необхідно ще раз переглянути структуру бази даних і виявити можливі недоліки. Крім того, потрібно виключити з таблиць усі можливі повторення даних.



Чим відрізняються таблиці баз даних і таблиці табличного процесора?



На відміну від таблиць табличного процесора, таблиці баз даних мають чітко визначену структуру, яка формується в режимі конструктора таблиць.



Структура таблиці — це набір поіменованих полів, у яких описують властивості об'єктів



Чим відрізняються таблиці баз даних і таблиці табличного процесора?

Розділ 3
§ 16



У режимі конструктора не передбачено введення, редагування та перегляд даних таблиці, його використовують лише для створення та зміни структури таблиці.

Робота з таблицями База даних : База даних- C:\Users\ProMa\Desktop\H

Файл Конструктор

Створення Зовнішні дані Знання бази даних Довідка Office Tab

Конструктор Скажіть, що потрібно зробити

Подання Ключове поле Конструктор Перевірка правил Знання

Вставити рядки Видалити рядки Змінити підстановку

Аркуш властивостей Індекси

Відображення або приховання

Створити макроси даних Перейменувати або видалити макрос

Події для полів, записів і таблиць

Зв'язки Залежності об'єкта зв'язки

База даних

Таблиця1

Ім'я поля	Тип даних	Опис
Ідентифікатор	Автонумерація	
Номер	Короткий текст	

Чим відрізняються таблиці баз даних і таблиці табличного процесора?



Створення структури таблиці як об'єкта бази даних передбачає такі дії.

- 1. Визначення параметрів (атрибутів), що характеризуватимуть властивості об'єкта і значення яких будуть занесені до полів конкретної таблиці бази даних.**
- 2. Надання унікального імені кожному полю таблиці, яке будується за певними правилами.**
- 3. Визначення формату подання даних у кожному полі, тобто типу даних та необхідних обмежень.**
- 4. Визначення ключового поля чи сукупності ключових полів.**

Чим відрізняються таблиці баз даних і таблиці табличного процесора?

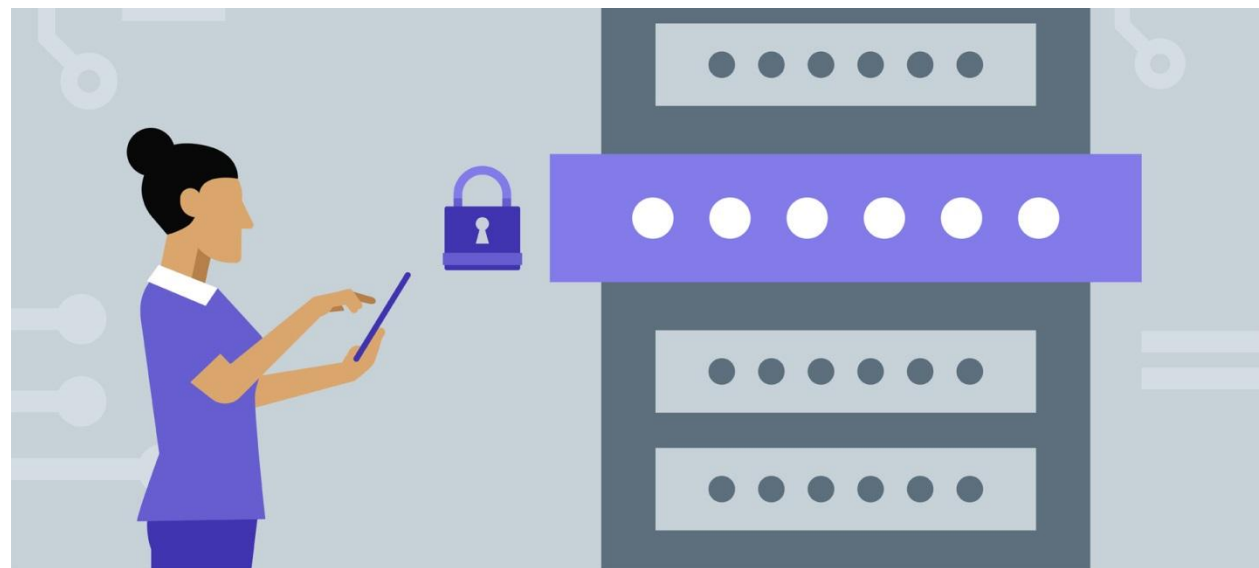


Існують ключі двох типів:

первинні

вторинні (зовнішні)

*Якщо для таблиці визначені ключові поля, то **MS Access** запобігає дублюванню або введенню порожніх значень у ці поля.*



Чим відрізняються таблиці баз даних і таблиці табличного процесора?



Первинний ключ — це одне або кілька полів (стовпців), комбінація значень яких однозначно визначає кожний запис у таблиці.

Первинний ключ не допускає значень Null і завжди повинен мати унікальний індекс.

Первинний ключ використовується для зв'язування таблиці із зовнішніми ключами в інших таблицях.





Зовнішній (вторинний) ключ — це одне або кілька полів (стовпців) у таблиці, що містять посилання на поле або поля первинного ключа в іншій таблиці. Зовнішній ключ визначає спосіб об'єднання таблиць.



Чим відрізняються таблиці баз даних і таблиці табличного процесора?



З двох логічно пов'язаних таблиць:

одну називають таблицею первинного ключа, або головною таблицею

а іншу – таблицею вторинного (зовнішнього) ключа, або підпорядкованою таблицею

СКБД дають змогу зіставити споріднені записи з обох таблиць і спільно вивести їх у:

формі

звіті

запиті

Чим відрізняються таблиці баз даних і таблиці табличного процесора?



Існує три типи первинних ключів:

Ключові поля лічильника (лічильник)

Поле лічильника (тип даних «автозаповнення»). Для кожного запису цього поля таблиці автоматично заноситься унікальне числове значення.

Простий ключ

Якщо поле містить унікальні значення, такі як коди чи інвентарні номери, то це поле можна визначити як первинний ключ. Ключем можуть бути всі поля, що містять дані, якщо вони не містять 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.

Чим відрізняються таблиці баз даних і таблиці табличного процесора?



Зауважте, що:

в полі первинного ключа

в полі вторинного або зовнішнього ключа

мають бути тільки унікальні значення в кожному рядку таблиці, тобто збіг не допускається

збіг значень у рядках таблиці допускається

Якщо виникають труднощі з вибором потрібного типу первинного ключа, то як ключ доцільно вибрати поле лічильника.

Чим відрізняються таблиці баз даних і таблиці табличного процесора?









*Для створення первинного ключа слід відкрити таблицю в режимі конструктора, виділити поле, яке використовується як первинний ключ, і натиснути кнопку **Ключове поле**, що міститься в розділі **Знаряддя** вкладки **Конструктор**.*

Скріншот інтерфейсу Microsoft Access, що демонструє вкладку **Конструктор** у режимі редагування таблиці **МАГАЗИНИ**. У групі **Знаряддя** (Tools) кнопка **Ключове поле** (Primary Key) виділена червоною рамкою та червоною стрілкою.

Ім'я поля	Тип даних	Опис (необов'язково)
Номер магазину	Число	Первинний ключ



Залежно від потреб користувачів, можна змінювати структуру таблиць бази даних шляхом:

-  зміни порядку розташування полів;
-  видалення (додавання) нового поля;
-  зміни назв полів;
-  зміни властивостей полів;
-  зміни типів даних;
-  додавання чи зміни ключового поля.





Вікно конструктора таблиць

Список типів даних

Імена полів

Опис

Властивості поля

Як змінити структуру таблиці бази даних?



Під час створення або редагування імен полів потрібно дотримуватися таких правил:

- назви полів у межах однієї таблиці не повинні повторюватися (тобто мають бути унікальними);**
- імена полів не мають перевищувати 64 символи;**
- бажано не використовувати як назви полів вбудовані функції (наприклад, Name — ім'я);**
- назва поля не має починатися з пропуску;**

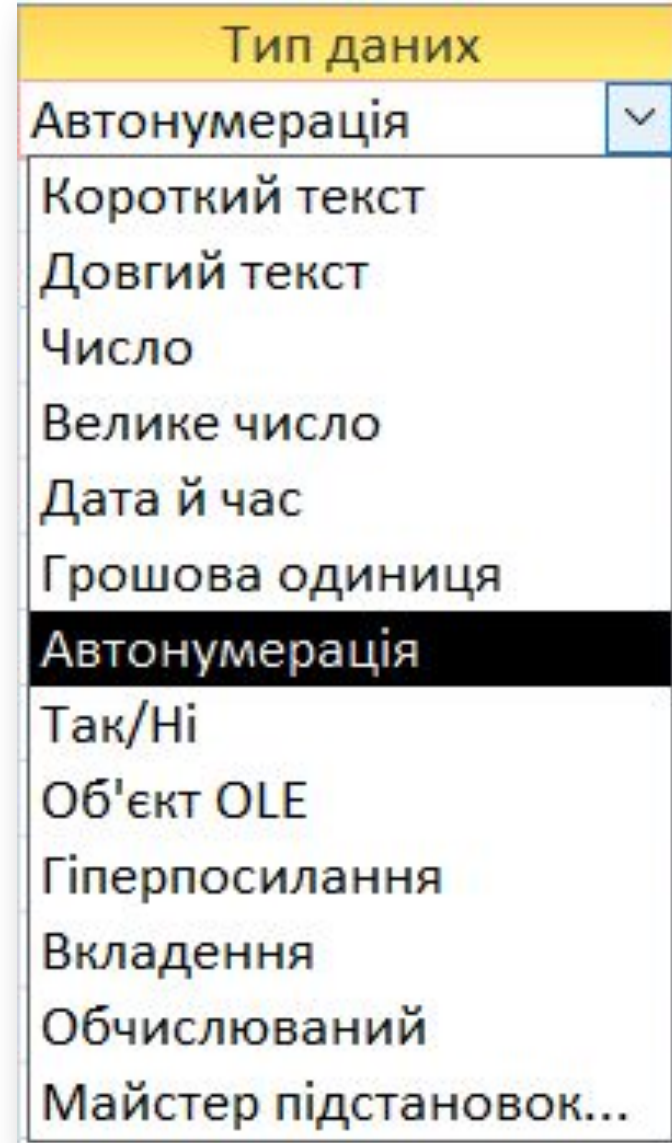
Як змінити структуру таблиці бази даних?



Тип даних обирається зі списку типів, що підтримуються системою керування базами даних.

Найбільш вживаними є типи:

- текстовий,**
- числовий,**
- дата/час,**
- автозаповнення (лічильник).**



Як змінити структуру таблиці бази даних?



Необхідні обмеження на введення даних можна задати (чи змінити), якщо перейти до вкладки Властивості поля/Загальні:

розмір поля

задає довжину значення, яке буде зберігатися в цьому полі;

формат поля

визначає, як буде відображатися збережене в таблиці значення;

маска введення

допомагає користувачеві вводити дані в заданому форматі;

підпис

визначає, як буде підписуватися поле під час перегляду даних.

Як налаштувати параметри зв'язків між таблицями?



Для організації подальшого пошуку та відбору даних з логічно пов'язаних таблиць бази даних спочатку потрібно встановити зв'язки між створеними таблицями. У більшості випадків зв'язують ключове поле однієї таблиці (первинний ключ) з відповідним йому полем в іншій таблиці (зовнішній ключ).

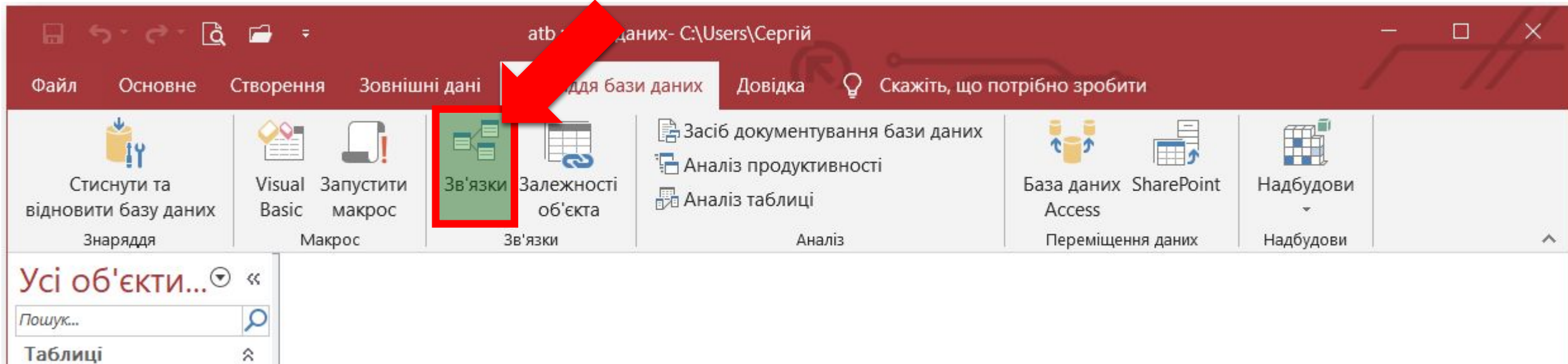
У зв'язаних полів не обов'язково мають бути однакові імена, але в них мають зберігатися дані одного типу, і вони повинні мати однакові розміри для збереження вмісту.





Розглянемо порядок створення зв'язку в таблицях МАГАЗИНИ і КАДРИ.

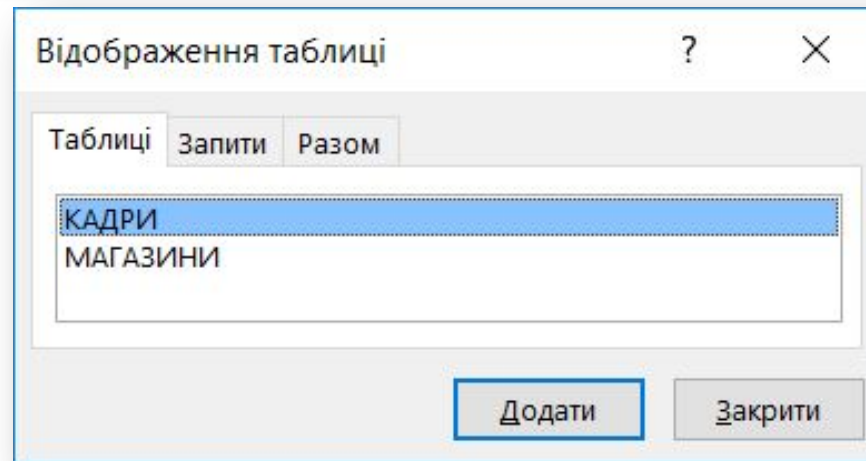
Завантажимо БД *atb* і у вікні, що відкриється, активуємо вкладку *Знаряддя бази даних*. Далі натиснемо кнопку *Зв'язки*.





Продовження...

1. Відкриється вікно **Відображення таблиці**.



2. У цьому вікні вибираємо таблиці, які необхідно зв'язати (у цьому випадку обидві таблиці), і натискаємо кнопку **Додати**. На екрані з'являться ці таблиці з іменами їх полів.



Продовження...

В. Установлюємо курсор на первинному ключі таблиці **МАГАЗИНИ**, натискаємо кнопку миші, не відпускаючи її, переміщуємо курсор у поле зовнішнього ключа й відпускаємо кнопку. У результаті відкриється вікно **Редагування зв'язків**.

Редагування зв'язків

Таблиця/запит: Пов'язана таблиця/запит:

МАГАЗИНИ КАДРИ

Номер магазину	Номер магазину

З**а**безпечення цілісності даних

Каскадне оновлення пов'язаних полів

Каскадне видалення пов'язаних полів

Тип зв'язку: Не визначено

Створити

Скасувати

Тип об'єднання...

Нове...



Продовження...

4. Умикаємо прапорець **Забезпечення цілісності даних.**

Після цього стануть доступними прапорці **Каскадне оновлення пов'язаних полів** і **Каскадне видалення пов'язаних полів**. Умикаємо прапорець **Каскадне видалення пов'язаних полів**.

Редагування зв'язків

Таблиця/запит: МАГАЗИНИ Пов'язана таблиця/запит: КАДРИ

Номер магазину	Номер магазину

Забезпечення цілісності даних

Каскадне оновлення пов'язаних полів

Каскадне видалення пов'язаних полів

Тип зв'язку: Не визначено

Створити

Скасувати

Тип об'єднання...

Нове...



Продовження...

4. Якщо прапорець **Каскадне видалення пов'язаних полів** вимкнаний, то в таблиці можна додавати нові записи, змінювати ключові поля й видаляти пов'язані записи без попередження про порушення цілісності.

Сутність **каскадного оновлення пов'язаних полів** полягає в тому, що за будь-якої зміни первинного ключа в головній таблиці автоматично оновиться значення відповідного поля в усіх зв'язаних таблицях.



Продовження...

4. Сутність каскадного видалення пов'язаних полів полягає в тому, що під час видалення будь-якого запису з головної таблиці автоматично видаляються зв'язані записи в пов'язаній таблиці.

Таким чином, каскадне оновлення та каскадне видалення роботи з БД і сприяють підвищенню надійності її функціонування.

Редагування зв'язків

Таблиця/запит: МАГАЗИНИ Пов'язана таблиця/запит: КАДРИ

Номер магазину Номер магазину

Забезпечення цілісності даних

Каскадне оновлення пов'язаних полів

Каскадне видалення пов'язаних полів

Тип зв'язку: Не визначено

Створити

Скасувати

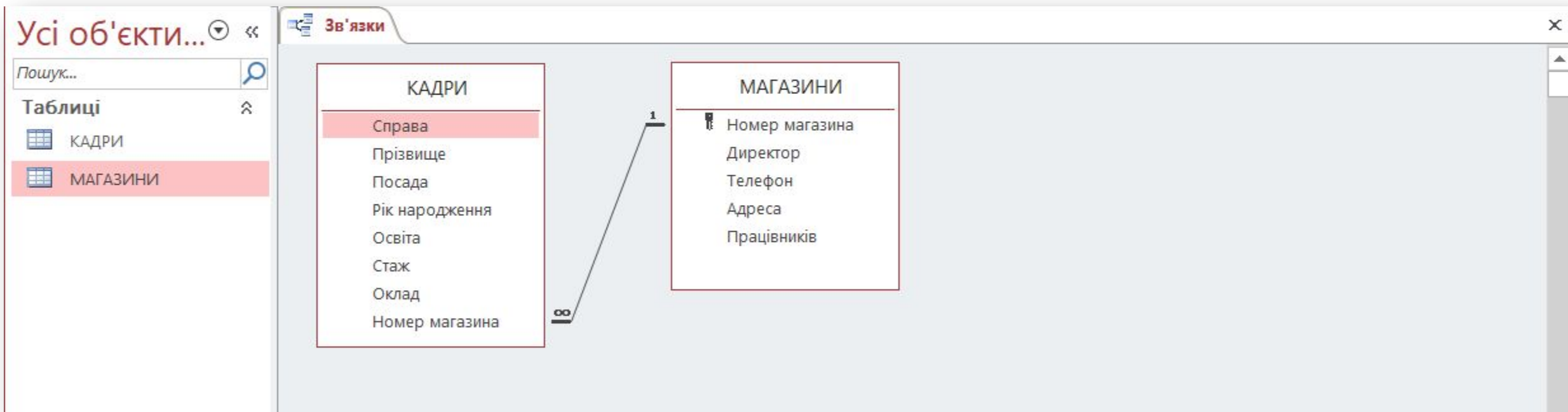
Тип об'єднання...

Нове...



Продовження...

5. У вікні **Редагування зв'язків** слід натиснути кнопку **Створити**, у результаті чого у вікні **Зв'язки** з'явиться лінія зв'язку між певними полями таблиць.



Як налаштувати параметри зв'язків між таблицями?



Для перегляду всіх зв'язків між таблицями бази даних необхідно на відкритій вкладці **Знаряддя бази даних** натиснути кнопку **Зв'язки**, активізувати вкладку **Конструктор** і в розділі **Зв'язок** натиснути кнопку **Усі зв'язки**.

atb : База даних- C:\Users\Сергій

Знаряддя для зв'язків

Файл Основне Створення Зовнішні дані Знаряддя бази даних Довідка Конструктор Скажіть, що потрібно зробити

Очистити макет

Змінити зв'язки

Звіт про зв'язки

Відобразити таблицю

Приховати таблицю

Прямі зв'язки

Усі зв'язки

Закрити

Знаряддя

Зв'язок

Усі об'єкти...

Пошук...

Таблиці

КАДРИ

КАДРИ

Справа

Прізвище

МАГАЗИНИ

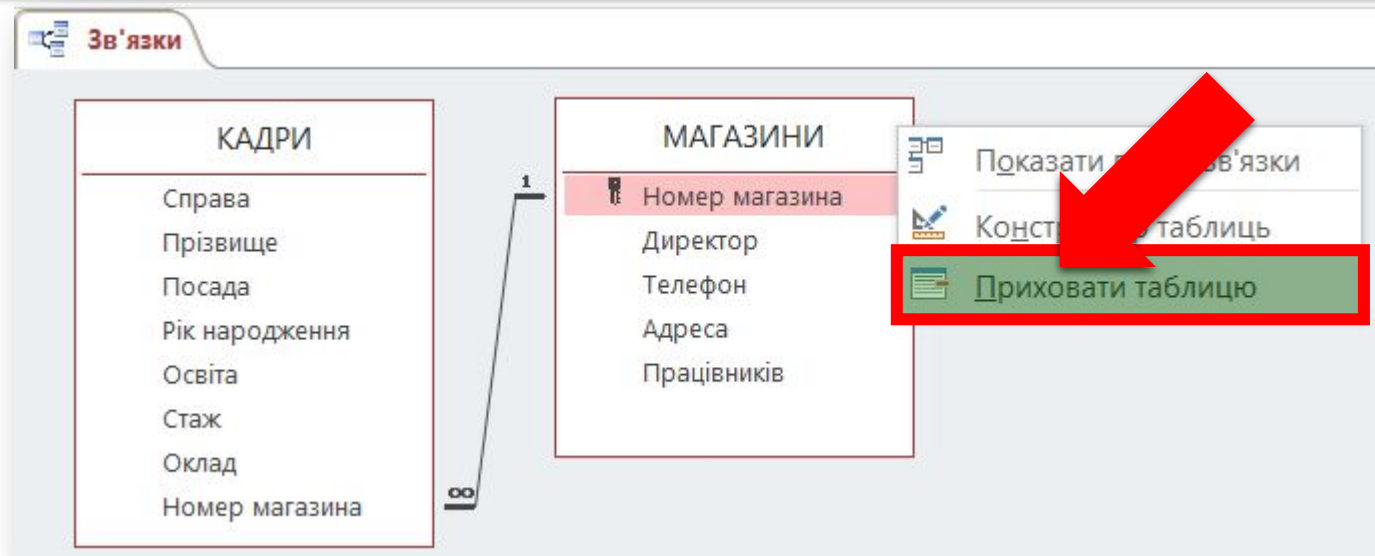
Номер магазину

Директор

Як налаштувати параметри зв'язків між таблицями?



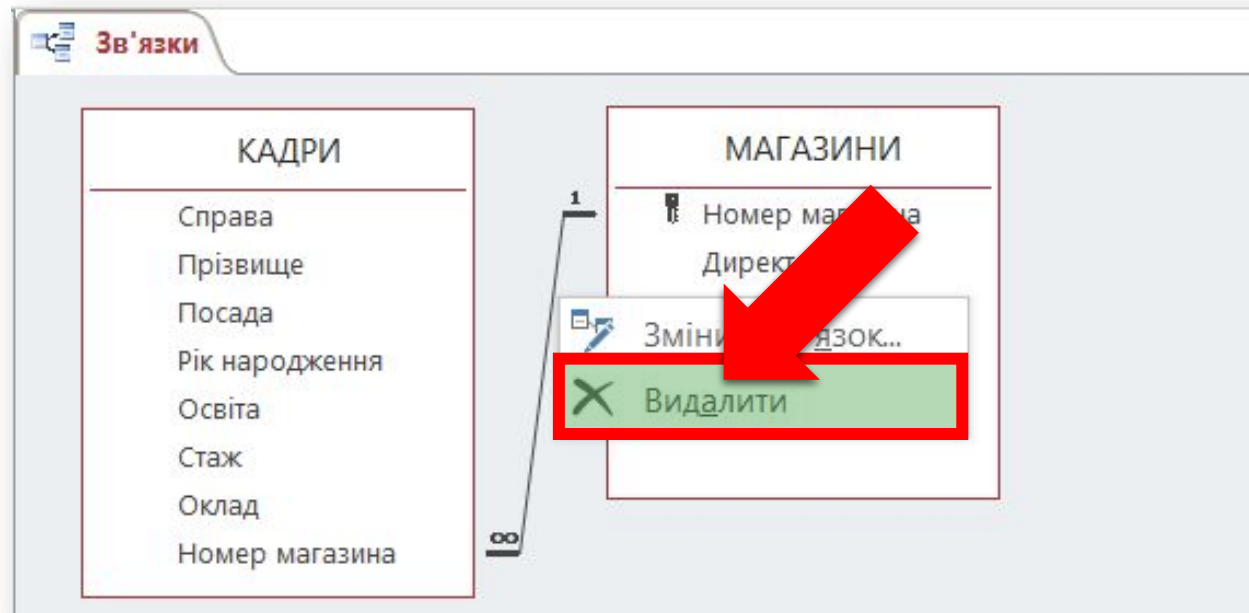
Якщо схема складна, можна приховати частину таблиць і зв'язків, видаливши їх із вікна **Зв'язки**, для чого необхідно виділити певну таблицю й натиснути клавішу **Del**. У цьому разі зв'язки й таблиці видаляються лише з вікна, фізично вони залишаються, тому їх можна відновити в будь-який час.



Як налаштувати параметри зв'язків між таблицями?



Для скасування зв'язку між таблицями необхідно встановити курсор на лінії зв'язку, натиснути кнопку миші, а потім — клавішу **Del**. Можна також відкрити контекстне меню лінії зв'язку й виконати команду **Видалити**.



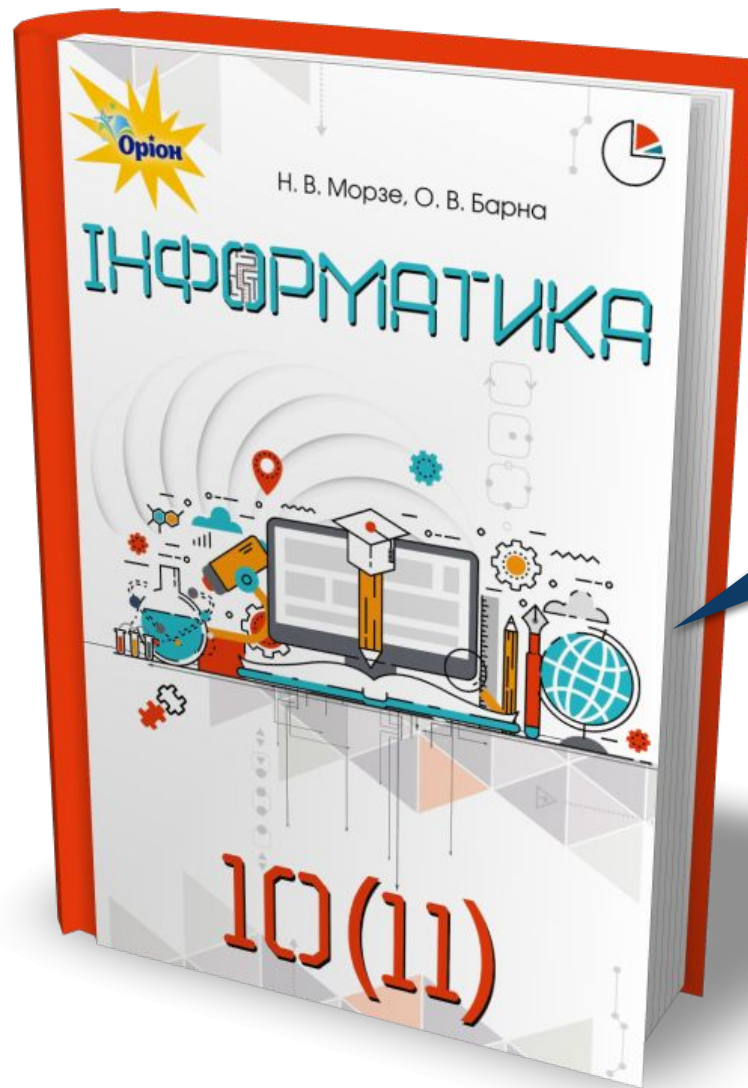


- 1. Як ви розумієте поняття моделі «сутність—зв'язок»?**
- 2. За яких умов таблицю можна вважати аналогом відношення?**
- 3. З якою метою встановлюються зв'язки між таблицями?**
- 4. Яких правил побудови моделі даних предметної області слід дотримуватись і чому?**
- 5. Яке поле називають ключовим?**



Розділ 3
§ 16

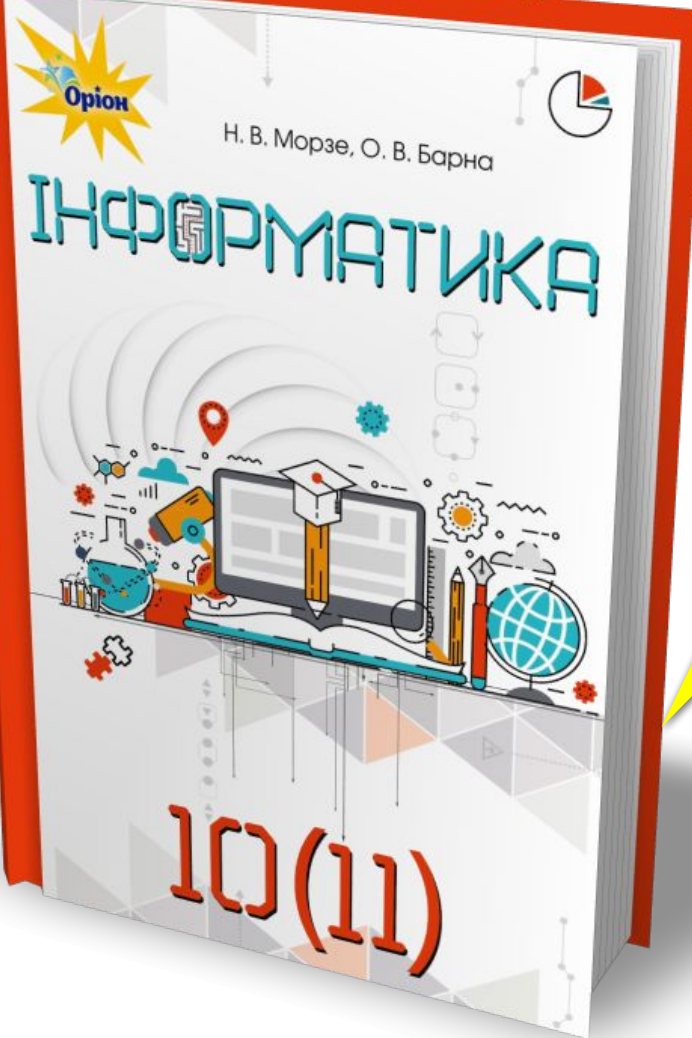
Домашнє завдання



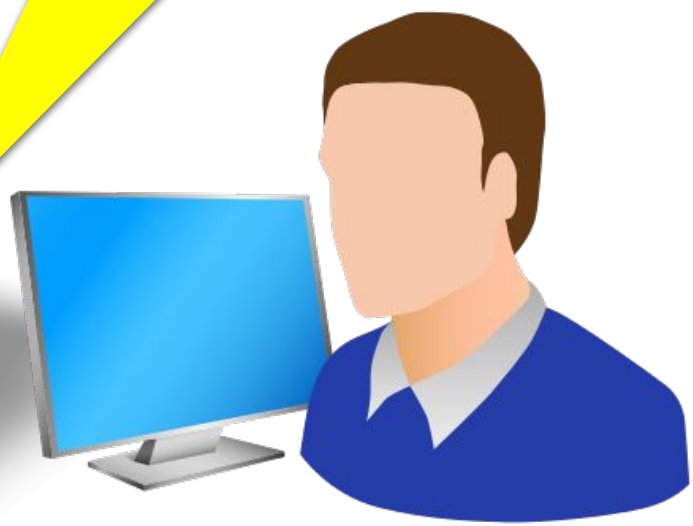
Проаналізувати
§ 16, ст. 158-166

Працюємо за комп'ютером

Розділ 3
§ 16



**Сторінка
164**





ІНФОРМАТИКА

Дякую за увагу!



За навчальною програмою 2018 року



Урок 18