

Діаграми можна відображати в різних редакторах, наприклад, **MS Visio** - програмне забезпечення для створення діаграм і схем (безкоштовний для наших студентів) (*Синтаксис Chen's notation та Crow's foot*) **Draw.io** (попередня назва), <https://app.diagrams.net/> - безкоштовний онлайн-інструмент для створення та редагування діаграм, що дає змогу користувачам розробляти візуальні моделі та схеми у веб-браузері (*Синтаксис Chen's notation та Crow's foot*) **PlantUML** - мова розмітки для створення діаграм. Вона дає змогу описувати діаграми в текстовому форматі, який потім можна перетворити на графічні зображення. <https://plantuml.com/> - представлено можливості цієї мови та приклади, як саме її можна використовувати (*Синтаксис Crow's foot*) Можна ще **Lucidchart** - хмарна платформа для створення діаграм і візуалізації даних, у безкоштовній версії 3 схеми. (*Є підписка для університетів, ще не розбиралася*)

Однак ці редактори, на відміну від case засобу Erwin, тільки відображають уже готові рішення. Erwin призначений для супроводу та підтримки проектування ER схеми з урахуванням особливостей створення моделі БД.

Якщо є проблеми зі встановленням Erwin 7.3, можна використовувати наступні варіанти для отримання ER діаграм для лабораторних робіт і курсового (з невеликою втратою в балах).

ER діаграми в синтаксисе Чена або Crow's foot

1. **DBDiagram.io** -засіб відтворення діаграми за кодом (синтаксис Воронья лапка) <https://dbdiagram.io/home>
2. **PlantUML**

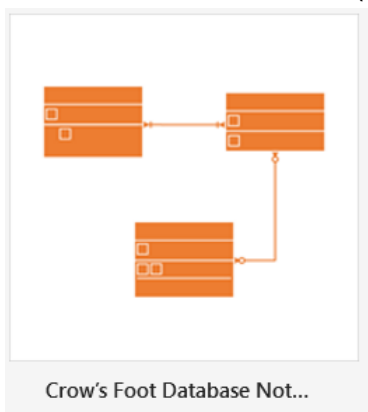
Схема на фізичному рівні

3. **MS SQL Server Management Studio** за DDL скриптом SQL створити БД і вивести діаграму
4. **MySQL Workbench** (для MySQL)

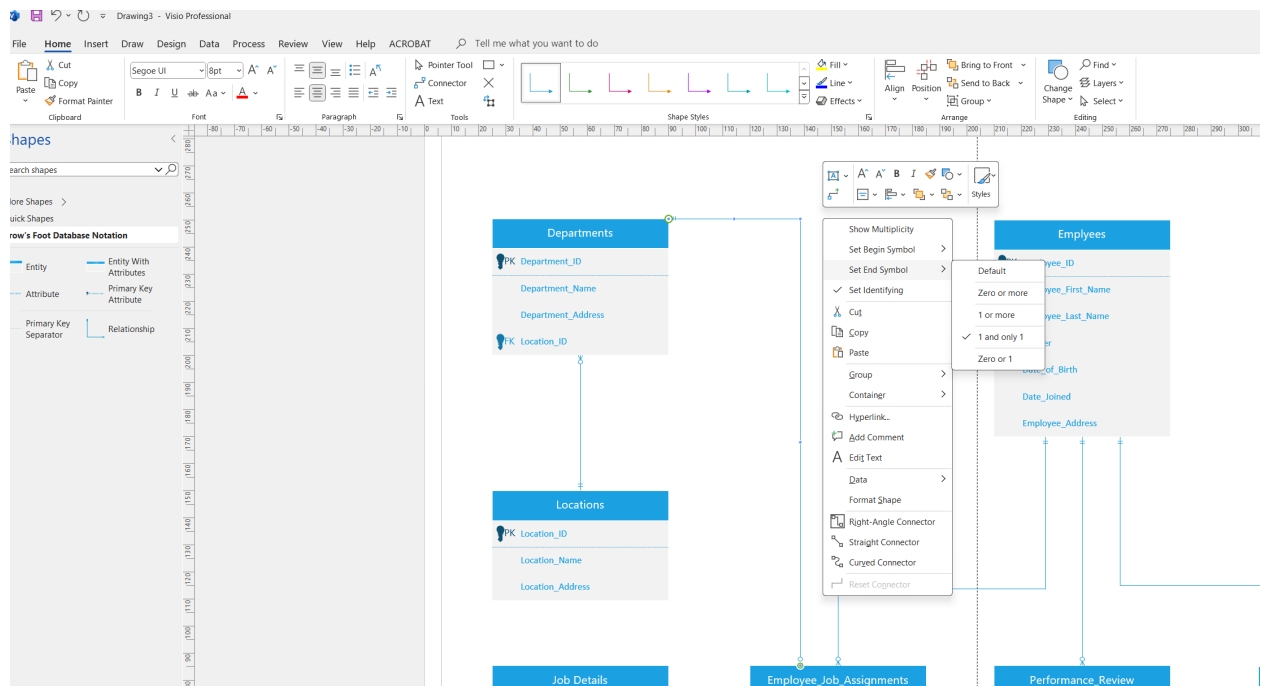
Далі детальніше про MS Visio, Draw.io, PlantUML

MS Visio ()

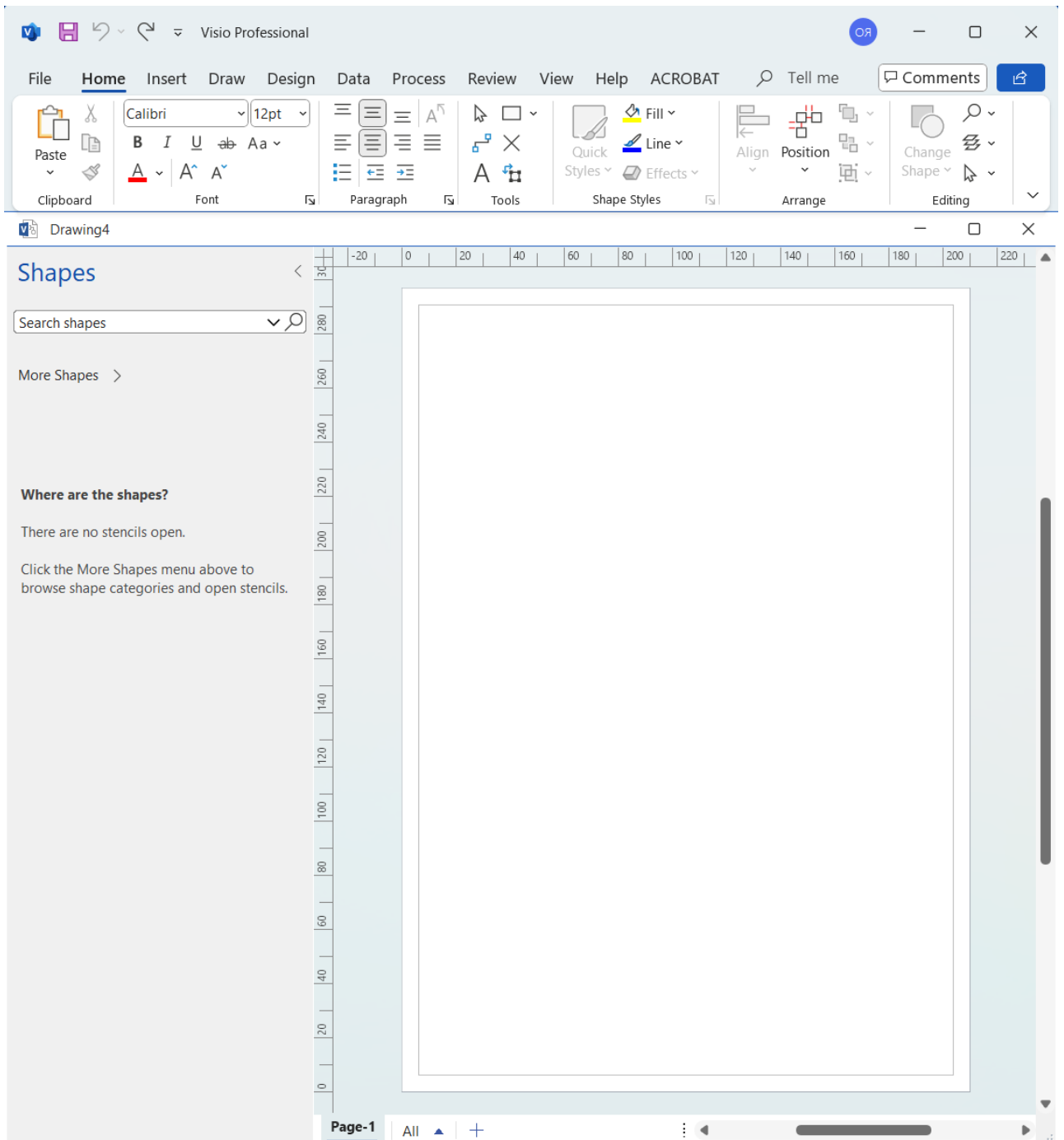
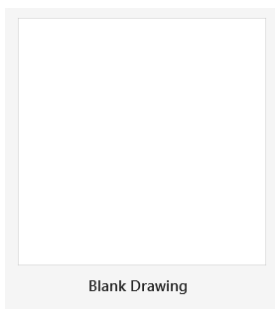
- 1) У синтаксисі Crow's foot (вороняча лапка)



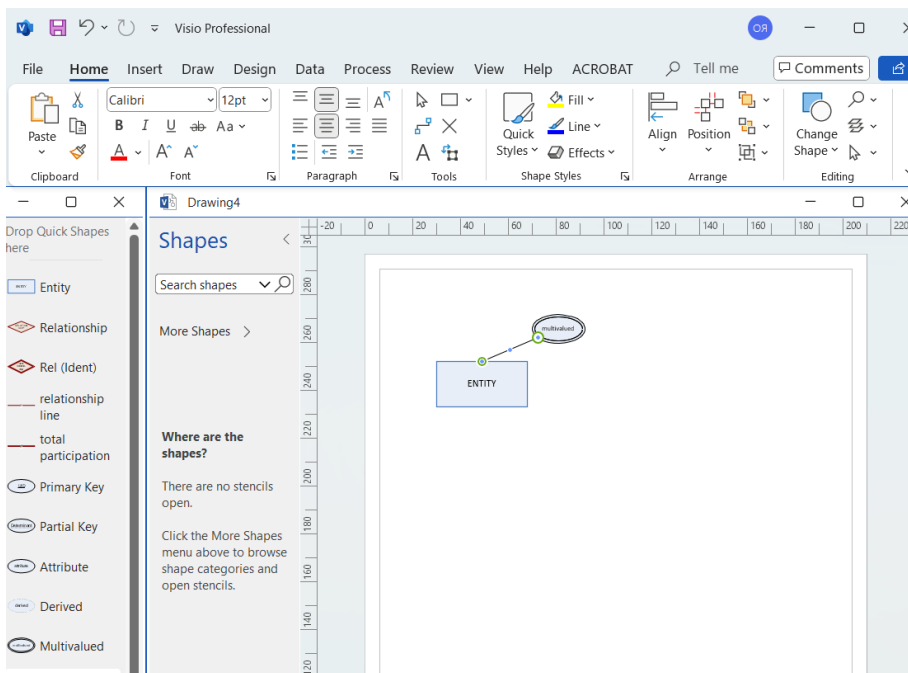
Керування зв'язками (кардинальністю та обов'язковістю участі) за допомогою контекстного меню



2) У системі Чена
Вибираєте



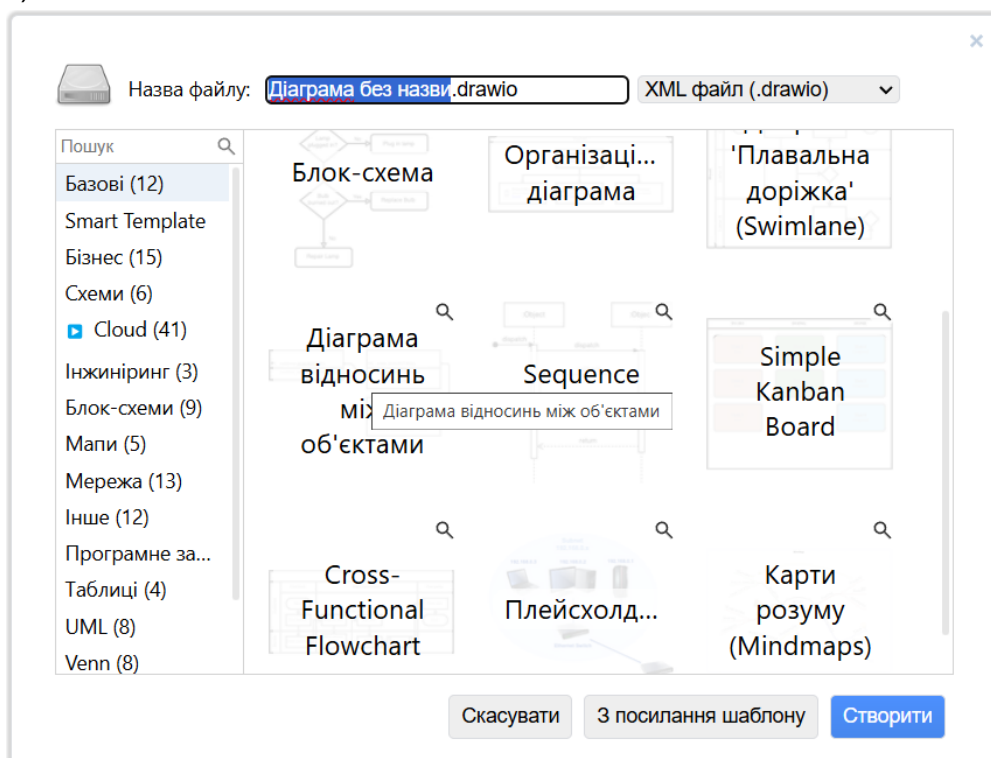
Потім підключіть бібліотеку Чена (завантажити), просто перетягніть файл із бібліотекою в область Shapes.



Примітка: є можливість відобразити класи/підкласи

(Draw.io) app.diagrams.net

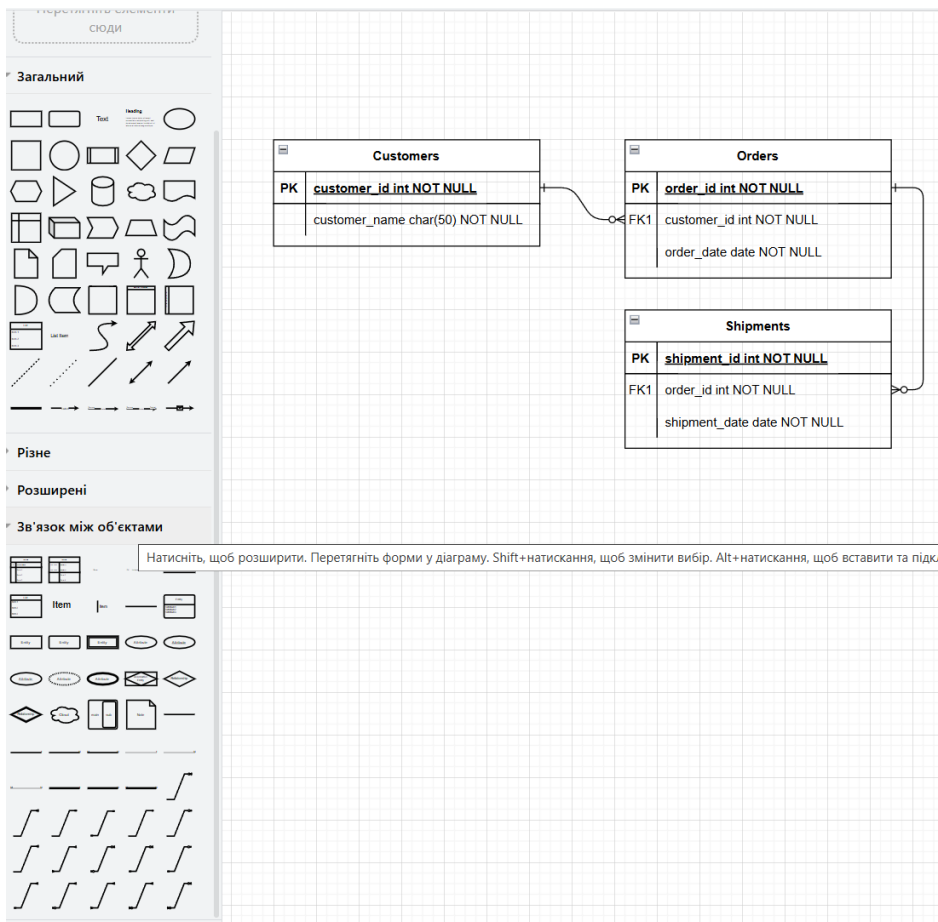
1)



2 За замовчуванням діаграма буде в синтаксисі Crow's foot (вороняча лапка)

3 Для доступу до синтаксису Чена натисніть

Примітка: немає можливості відобразити класи/підкласи



PlantUML

<https://plantuml.com/>

На <https://plantuml.com/> обираємо Entity Relationship (ER) diagram або відразу <https://plantuml.com/er-diagram>

plantuml.com

Sequence
Use Case
Class
Activity
Component
State
Object
Deployment
Timing
Regex
Network
Wireframe

Home
What's New ?
Getting Started
Online Server
Running
F.A.Q.
Download
Forum
Theme
Preprocessing
Standard Library
Hitchhiker's Guide
PDF Guide

Getting Started

PlantUML is a highly versatile tool that facilitates the rapid and straightforward creation of a wide array of diagrams.

Utilizing a simple and intuitive language, users can effortlessly draft various types of diagrams. For a detailed exploration of the [Reference Guide](#).

If you are new to PlantUML, we recommend starting with our [quick start page](#) to get up and running quickly. Should you have can be seamlessly integrated with a [variety of other tools](#) to enhance your workflow.

Supported UML Diagrams

With PlantUML, you can create well-structured UML diagrams, including but not limited to:

- [Sequence diagram](#)
- [Usecase diagram](#)
- [Class diagram](#)
- [Object diagram](#)
- [Activity diagram](#) (*Find the [legacy syntax](#) [here](#)*)
- [Component diagram](#)
- [Deployment diagram](#)
- [State diagram](#)
- [Timing diagram](#)

Supported Non-UML Diagrams

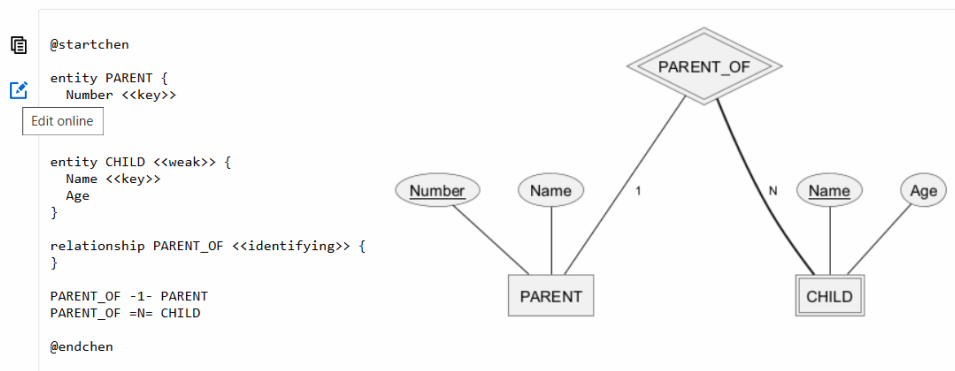
Beyond traditional UML diagrams, PlantUML also supports the creation of various other diagram types, such as:

- [JSON data](#)
- [YAML data](#)
- [EBNF diagram](#)
- [Regex diagram](#)
- [Network diagram \(nwdiag\)](#)
- [UI mockups \(salt\)](#)
- [Archimate diagram](#)
- [Specification and Description Language \(SDL\)](#)
- [Ditaa diagram](#)
- [Gantt diagram](#)
- [Chronology diagram](#)
- [MindMap diagram](#)
- [WBS diagram](#)
- [Mathematics with AsciiMath or JLaTeXMath notation](#)
- [Information Engineering \(IE\) diagram](#)
- [Entity Relationship \(ER\) diagram](#)

За замовчуванням перейдете в Entity Relationship Diagrams (Chen's notation)
Далі вивчаєте приклади.
Щоб створити щось своє, натисніть на будь-який приклад і перейдіть у вікно редагування

Identifying relationships

A *weak* entity does not have a key attribute that uniquely identifies each instance of that entity. Instead, it is identified by the combination of a *partial key* on the weak entity its entity, which it is related to via an *identifying relationship*. A weak entity must have total participation in its identifying relationship.



```
@startchen
entity PARENT {
  number <key>
  name
}
entity CHILD <weak> {
  name <key>
  Age
}
relationship PARENT_OF <identifying> {
}
PARENT_OF 1.. PARENT
PARENT_OF ..n CHILD
endchen
```

<https://www.plantuml.com/plantuml/png/S6kT2ag35dgpYf1tqhcOfi0W6T7R2r6v69G21D7KcJ02b6cJ3C321Lc4d8uafcaIt5a6t2bocC1-1154gAf0q2AQ3H4BvYDPR6SRFASabYfT7buh8304PNqPuPSE0Q2SVYcQubR10waQJ42x6WY1XDY7v700>

[Submit](#) [Pure Javascript](#) [PNG](#) [SVG](#) [ASCII Art](#)

```
graph TD
    PARENT -- "1" -- PARENT_OF -- "n" -- CHILD
    PARENT --- Number
    PARENT --- Name
    CHILD --- Name
    CHILD --- Age
```

У вікні редагування ви можете писати код і по Submit одразу бачити результат, щоб зберегти натисніть PNG і "Зберегти картинку як"

Примітка:

- 1) є можливість відобразити класи/підкласи
- 2) На жаль, немає можливості керувати розташуванням об'єктів як у графічних редакторах

Якщо бажаєте працювати у синтаксисе Crow's foot, натисніть [Information Engineering diagrams](#).

Entity Relationship Diagrams (Chen's notation)

This page is for Chen's [Entity Relationship](#) notation, which is commonly used in teaching. See also [Information Engineering diagrams](#).

Entity Relationship (ER) diagrams are used to model databases at a conceptual level by describing entities, their attributes, and the relationships between them. PlantUML also supports subclasses and union types. This extended notation is sometimes referred to as Enhanced Entity Relationship (EER) or Extended

[Ref. GH-945 and GH-1718]

Minimal Example

87 10 11 12 13