**Лабораторна робота №1**

**Тема:** Концептуальне моделювання БД

**Мета:** набуття практичних навичок щодо побудови концептуальних моделей баз даних із використанням діаграмних технологій (модель сутність-зв‘язок).

Варіант 4

*Хід роботи*

1. Моє завдання:

*«****"Ведення бази даних обліку сировини і матеріалів на складі"***

*Підприємство взуттєвої промисловості укладає договори з постачальниками на постачання сировини і матеріалів для нормальної роботи підприємства. У договорах указується скільки сировини або матеріалів повинні поставити кожний постачальник за місяць. Одну і ту ж сировину або матеріал можуть постачати одночасно декілька постачальників.*

*При надходженні сировини і матеріалів на склад оформлюють ПРИБУТКОВІ НАКЛАДНІ в яких вказують: постачальник, дата надходження, вид сировини (матеріалів), кількість.*

*Сировина і матеріали відпускається цехам для виконання виробничої програми. При відвантаженні сировини і матеріалів у цеху оформлюється ВИДАТКОВА НАКЛАДНА, у якій вказують: цех, вид сировини (матеріалів), дата відвантаження, кількість.*

*Інформаційна система, що проектується, повинна вирішувати такі задачі:*

* *Визначати загальну кількість кожного виду сировини (матеріалів), що повинна надійти відповідно до договорів за місяць на склад.*
* *Визначити залишки сировини і матеріалів на поточний день місяця, що повинні надійти від постачальників до кінця місяця.*
* *Визначити кількість сировини і матеріалів, що надійшло в цех на дане число.*
* *Відобразити кількість сировини і матеріалів, що надійшло в цех за вказаний період. Списки згрупувати за типами сировини.»*

1. Складена діаграма Чена зображена на рисунку 1.1. Сутностями виступають:

* Постачальник
* Договір
* Сировина
* Видаткова накладна
* Прибуткова накладна
* Цех

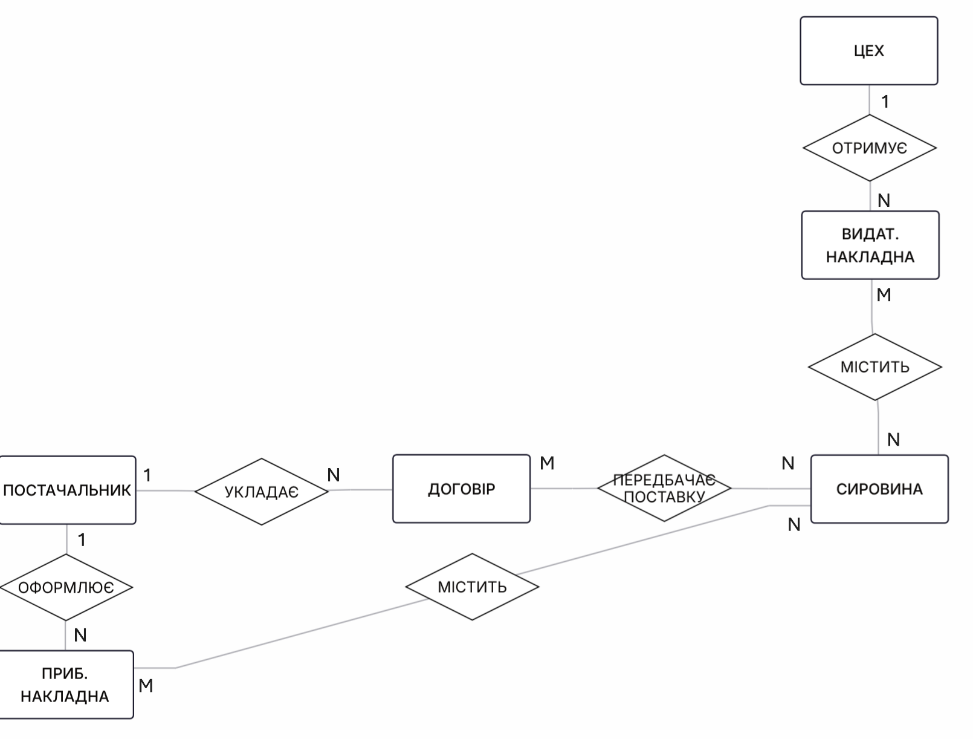


Рисунок 1.1 – ER-діаграма Чена

1. Діаграма Баркера зображена на рисунку 1.2.

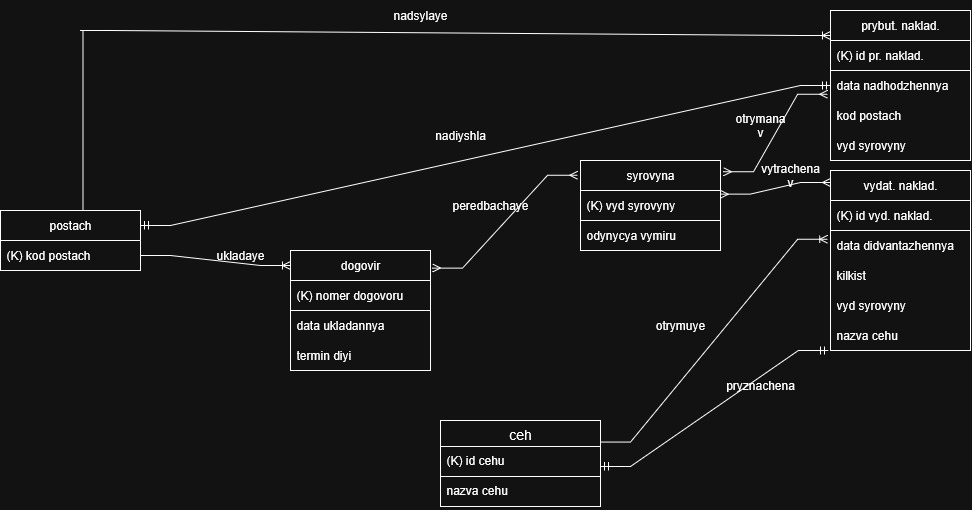


Рисунок 1.2 – Діаграма Баркера

1. Був створений SQL-скрипт для моєї ER-діаграми, адаптований під MySQL.

CREATE DATABASE IF NOT EXISTS WarehouseDB;

USE WarehouseDB;

CREATE TABLE POSTACH (

    kod\_postach INT AUTO\_INCREMENT PRIMARY KEY,

    nazva\_postach VARCHAR(255) NOT NULL,

    kontakt\_info VARCHAR(255)

);

CREATE TABLE SYROVYNA (

    vyd\_syrovyny INT AUTO\_INCREMENT PRIMARY KEY,

    nazva\_syrovyny VARCHAR(255) NOT NULL UNIQUE,

    odynytsya\_vymiru VARCHAR(20) NOT NULL

);

CREATE TABLE CEKH (

    id\_cehu INT AUTO\_INCREMENT PRIMARY KEY,

    nazva\_cehu VARCHAR(100) NOT NULL UNIQUE

);

CREATE TABLE DOGOVIR (

    nomer\_dogovoru INT AUTO\_INCREMENT PRIMARY KEY,

    kod\_postach INT NOT NULL,

    vyd\_syrovyny INT NOT NULL,

    data\_ukladannya DATE NOT NULL,

    termin\_diyi DATE NOT NULL,

    kilkist DECIMAL(10,2) NOT NULL CHECK (kilkist > 0),

    CONSTRAINT FK\_DOGOVIR\_POSTACH FOREIGN KEY (kod\_postach)

        REFERENCES POSTACH(kod\_postach)

        ON UPDATE CASCADE ON DELETE RESTRICT,

    CONSTRAINT FK\_DOGOVIR\_SYROVYNA FOREIGN KEY (vyd\_syrovyny)

        REFERENCES SYROVYNA(vyd\_syrovyny)

        ON UPDATE CASCADE ON DELETE RESTRICT

);

CREATE TABLE PRYBUT\_NAKLAD (

    id\_pr\_naklad INT AUTO\_INCREMENT PRIMARY KEY,

    kod\_postach INT NOT NULL,

    vyd\_syrovyny INT NOT NULL,

    data\_nadkhodzhennya DATE NOT NULL,

    kilkist\_otrymana DECIMAL(10,2) NOT NULL CHECK (kilkist\_otrymana > 0),

    CONSTRAINT FK\_PR\_NAKLAD\_POSTACH FOREIGN KEY (kod\_postach)

        REFERENCES POSTACH(kod\_postach)

        ON UPDATE CASCADE ON DELETE RESTRICT,

    CONSTRAINT FK\_PR\_NAKLAD\_SYROVYNA FOREIGN KEY (vyd\_syrovyny)

        REFERENCES SYROVYNA(vyd\_syrovyny)

        ON UPDATE CASCADE ON DELETE RESTRICT

);

CREATE TABLE VYDAT\_NAKLAD (

    id\_vyd\_naklad INT AUTO\_INCREMENT PRIMARY KEY,

    id\_cehu INT NOT NULL,

    vyd\_syrovyny INT NOT NULL,

    data\_vidvantazhennya DATE NOT NULL,

    kilkist DECIMAL(10,2) NOT NULL CHECK (kilkist > 0),

    CONSTRAINT FK\_VYD\_NAKLAD\_CEKH FOREIGN KEY (id\_cehu)

        REFERENCES CEKH(id\_cehu)

        ON UPDATE CASCADE ON DELETE RESTRICT,

    CONSTRAINT FK\_VYD\_NAKLAD\_SYROVYNA FOREIGN KEY (vyd\_syrovyny)

        REFERENCES SYROVYNA(vyd\_syrovyny)

        ON UPDATE CASCADE ON DELETE RESTRICT

);

1. Було створено 6 таблиць, спираючись на скрипт.

Таблиця 1. Постачальник

|  |  |  |  |
| --- | --- | --- | --- |
| **1. Структура таблиці POSTACH** | | | |
| **Field Name** | **Data Type** | **PK/FK** | **Description** |
| kod\_postach | INT AUTO\_INCREMENT | "+/-" |  |
| nazva\_postach | VARCHAR(255) | "-/-" |  |
| kontakt\_info | VARCHAR(255) | "-/-" |  |

Таблиця 2. Сировина

|  |  |  |  |
| --- | --- | --- | --- |
| **2. Структура таблиці SYROVYNA** | | | |
| **Field Name** | **Data Type** | **PK/FK** | **Description** |
| vyd\_syrovyny | INT AUTO\_INCREMENT | "+/-" |  |
| nazva\_syrovyny | VARCHAR(255) | "-/-" |  |
| odynytsya\_vymiru | VARCHAR(20) | "-/-" |  |

Таблиця 3. Цех

|  |  |  |  |
| --- | --- | --- | --- |
| **3. Структура таблиці CEKH** | | | |
| **Field Name** | **Data Type** | **PK/FK** | **Description** |
| id\_cehu | INT AUTO\_INCREMENT | "+/-" |  |
| nazva\_cehu | VARCHAR(100) | "-/-" |  |

Таблиця 4. Договір

|  |  |  |  |
| --- | --- | --- | --- |
| **4. Структура таблиці DOGOVIR** | | | |
| **Field Name** | **Data Type** | **PK/FK** | **Description** |
| nomer\_dogovoru | INT AUTO\_INCREMENT | "+/-" |  |
| kod\_postach | INT | "-/+" |  |
| data\_ukladannya | DATE | "-/-" |  |
| vyd\_syrovyny | INT | "-/+" |  |
| termin\_diyi | DATE | "-/-" |  |
| kilkist | DECIMAL(10,2) |  |  |

Таблиця 5. Прибуткова накладна

|  |  |  |  |
| --- | --- | --- | --- |
| **5. Структура таблиці PRYBUT\_NAKLAD** | | | |
| **Field Name** | **Data Type** | **PK/FK** | **Description** |
| id\_pr\_naklad | INT AUTO\_INCREMENT | "+/-" |  |
| kod\_postach | INT | "-/+" |  |
| vyd\_syrovyny | INT | "-/+" |  |
| data\_nadkhodzhennya | DATE | "-/-" |  |
| kilkist\_otrymana | DECIMAL(10,2) | "-/-" |  |

Таблиця 6. Видаткова накладна

|  |  |  |  |
| --- | --- | --- | --- |
| **6. Структура таблиці VYDAT\_NAKLAD** | | | |
| **Field Name** | **Data Type** | **PK/FK** | **Description** |
| id\_vyd\_naklad | INT AUTO\_INCREMENT | "+/-" |  |
| id\_cehu | INT | "-/+" |  |
| vyd\_syrovyny | INT | "-/+" |  |
| data\_vidvantazhennya | DATE | "-/-" |  |
| kilkist | DECIMAL(10,2) | "-/-" |  |

*Висновок*: у ході лабараторної роботи я набув навички побудови концептуальних моделей баз даних із використанням діаграмних технологій Чена та Баркера.