МІНІСТЕРСТВО ОСВІТИ І НАУКИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»



**Лабораторна робота №2**

**З дисципліни «**Організація баз даних та знань**»**

***Виконав:*** *студент групи КН-210*

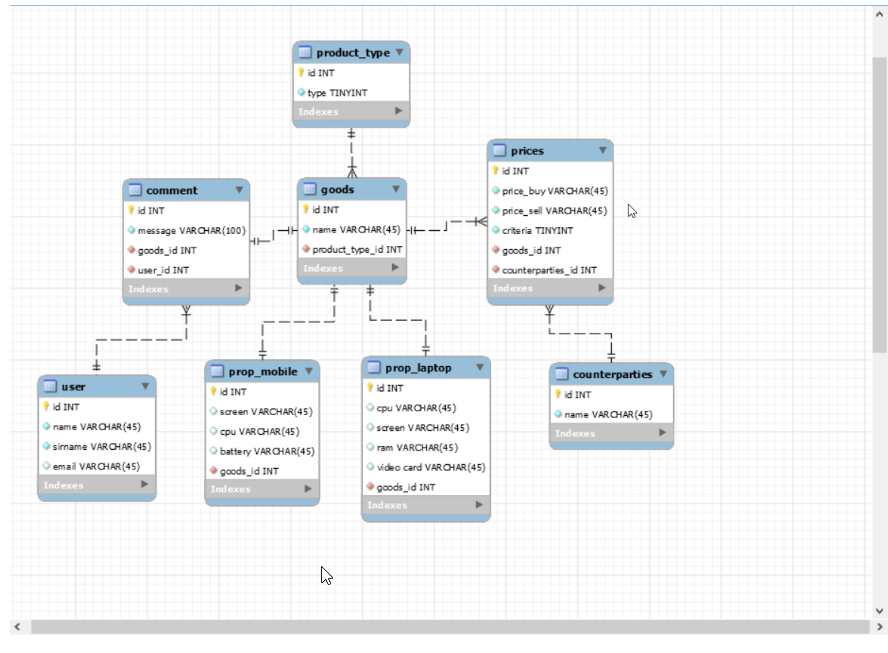
*Бурак Марко*

Львів – 2019

Мета роботи: Побудувати даталогічну модель бази даних; визначити типи, розмірності та обмеження полів; визначити обмеження таблиць; розробити SQL запити для створення спроектованих таблиць

**Хід роботи.**

Спершу потрібно визначитись з якого ресурсу буду створювати базу, для цього використав діаграму з лаб1, на якій вказані всі елементи.



Такі команди, як Create DATABASE/TABLE є основними та використовуються для побудови скелету бази, щодо наповнення, а саме полів буду використовувати такі типи даних як TINYINT, VARCHAR, INT та інші.

Для приєднання зовнішнього ключа потрібно використовувати такий синтаксис:

CONSTRAINT goods\_type FOREIGN KEY (product\_type\_id)

REFERENCES store.product\_type (id) ON DELETE NO ACTION ON UPDATE NO ACTION

Product\_type\_id – це зовнішній ключ, який був ініціалізований вище у таблиці, в свою чергу preference дає посиланн на id певної таблиці, яке буде слугувати primary key.

Ось представлено першу частину коду, у якому показано виконання таких таблиць як

Product\_type,goods,prices,counterparties,prop\_laptop,prop\_mobile.

CREATE DATABASE store;

CREATE TABLE store.product\_type (

id SMALLINT UNSIGNED NOT NULL AUTO\_INCREMENT,

typ TINYINT NOT NULL,

PRIMARY KEY(id)

);

CREATE TABLE store.goods (

id SMALLINT UNSIGNED NOT NULL AUTO\_INCREMENT,

good\_name VARCHAR(45),

product\_type\_id SMALLINT UNSIGNED NOT NULL,

PRIMARY KEY(id),

CONSTRAINT goods\_type FOREIGN KEY (product\_type\_id)

REFERENCES store.product\_type (id) ON DELETE NO ACTION ON UPDATE NO ACTION

);

CREATE TABLE store.prop\_mobile (

id SMALLINT UNSIGNED NOT NULL AUTO\_INCREMENT,

screen VARCHAR(45),

cpu VARCHAR(45),

battery VARCHAR(45),

goods\_id SMALLINT UNSIGNED NOT NULL,

PRIMARY KEY(id),

CONSTRAINT prop\_mobile\_id FOREIGN KEY (goods\_id)

REFERENCES store.goods (id) ON DELETE NO ACTION ON UPDATE NO ACTION

);

CREATE TABLE store.prop\_laptop (

id SMALLINT UNSIGNED NOT NULL AUTO\_INCREMENT,

screen VARCHAR(45),

cpu VARCHAR(45),

ram VARCHAR(45),

video\_card VARCHAR(45),

goods\_id SMALLINT UNSIGNED NOT NULL,

PRIMARY KEY(id),

CONSTRAINT prop\_laptop\_id FOREIGN KEY (goods\_id)

REFERENCES store.goods (id) ON DELETE NO ACTION ON UPDATE NO ACTION

);

CREATE TABLE store.counterparties (

id SMALLINT UNSIGNED NOT NULL AUTO\_INCREMENT,

name VARCHAR(60) NOT NULL,

PRIMARY KEY (id)

);

CREATE TABLE store.prices (

id SMALLINT UNSIGNED NOT NULL AUTO\_INCREMENT,

price\_buy VARCHAR(45) NOT NULL,

price\_sell VARCHAR(45) NOT NULL,

criteria TINYINT NOT NULL ,

goods\_id SMALLINT UNSIGNED NOT NULL,

counterparties\_id SMALLINT UNSIGNED NOT NULL,

Primary KEY (id),

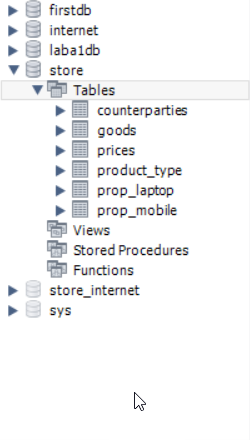
CONSTRAINT prices\_goods FOREIGN KEY(goods\_id)

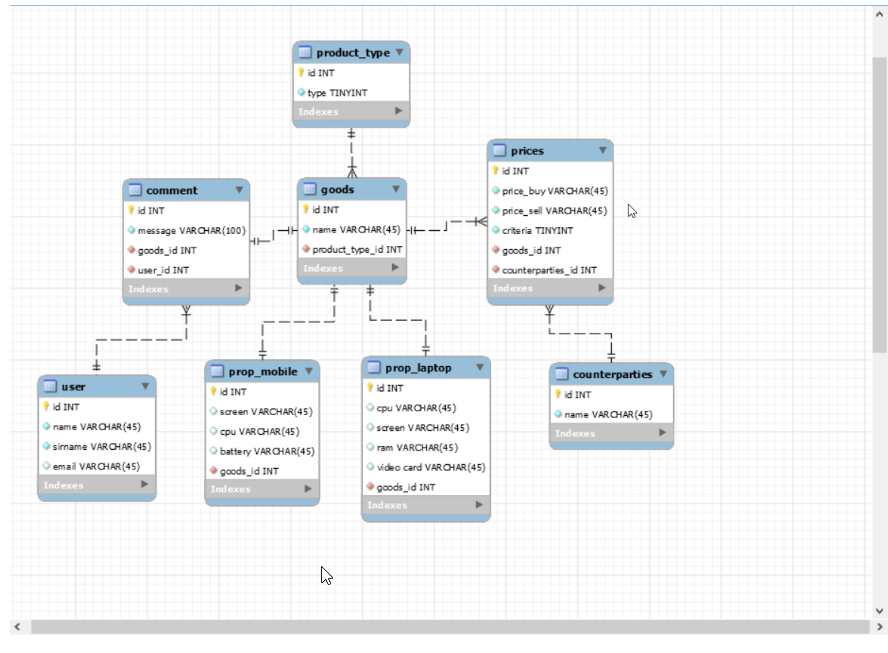
REFERENCES store.goods (id) ON DELETE NO ACTION ON UPDATE NO ACTION,

CONSTRAINT prices\_counterparties FOREIGN KEY(counterparties\_id)

REFERENCES store.counterparties (id) ON DELETE NO ACTION ON UPDATE NO ACTION

);





З цього коду і діаграми видно, що залишилось ще дві таблиці comment та user. Отже, їх я теж додав до бази.

CREATE TABLE store.users (

id SMALLINT UNSIGNED NOT NULL AUTO\_INCREMENT,

name VARCHAR(45) NOT NULL,

sirname VARCHAR(45) NOT NULL,

email VARCHAR(45),

PRIMARY KEY (id)

);

CREATE TABLE store.comments(

id SMALLINT UNSIGNED NOT NULL AUTO\_INCREMENT,

message VARCHAR(45) NOT NULL,

goods\_id SMALLINT UNSIGNED NOT NULL ,

user\_id SMALLINT UNSIGNED NOT NULL ,

Primary KEY(id),

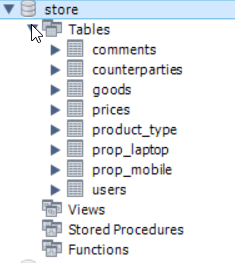
CONSTRAINT comment\_goods FOREIGN KEY (goods\_id)

REFERENCES store.goods (id) ON DELETE NO ACTION ON UPDATE NO ACTION,

CONSTRAINT comment\_user FOREIGN KEY (user\_id)

REFERENCES store.users (id) ON DELETE NO ACTION ON UPDATE NO ACTION

);



Висновок: : на цій лабораторній роботі було завершено моделювання і засобами SQL створено базу даних, що складається з восьми таблиць.