МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІСЬКА ПОЛІТЕХНІКА»

Кафедра ІСМ



**Звіт**

До лабораторної роботи №2

З дисципліни:

«Системи управління баз даних»

На тему:

**«Створення таблиць бази даних засобами SQL**»

Виконав:

Студент групи КН-38

Стахів М.А.

Прийняла:

Асистент каф. ІСМ

Заяць М.М.

**Львів - 2016**

**Мета роботи:**

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

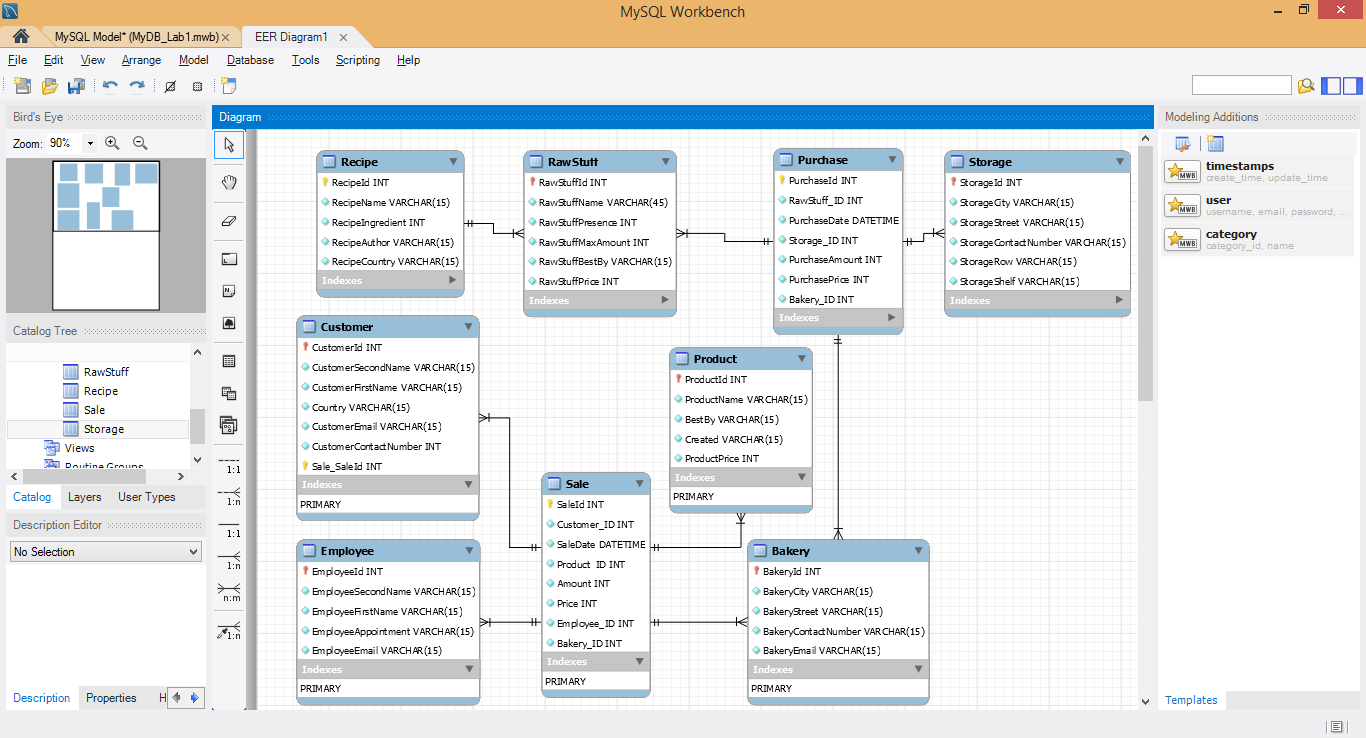
**Короткі теоретичні відомості:**

SQL (англ. Structured query language — мова структурованих запитів) — декларативна мова програмування для взаємодії користувача з базами даних, що застосовується для формування запитів, оновлення і керування реляційними БД, створення схеми бази даних і її модифікації, системи контролю за доступом до бази даних. Сам по собі SQL не є ні системою керування базами даних, ні окремим програмним продуктом. Не будучи мовою програмування в тому розумінні, як C або Pascal, SQL може формувати інтерактивні запити або, бувши вбудованою в прикладні програми, виступати в якості інструкцій для керування даними. Стандарт SQL, крім того, вміщує функції для визначення зміни, перевірки і захисту даних.

SQL — це діалогова мова програмування для здійснення запиту і внесення змін до бази даних, а також управління базами даних. Багато баз даних підтримує SQL з розширеннями до стандартної мови. Ядро SQL формує командна мова, яка дозволяє здійснювати пошук, вставку, оновлення, і вилучення даних, використовуючи систему управління і адміністративні функції. SQL також включає CLI (Call Level Interface) для доступу і управління базами даних дистанційно.

**Хід роботи:**

Даталогічна модель вимагає визначення конкретних полів бази даних, їхніх типів, обмежень на значення, тощо. Нижче зображено даталогічну модель проектованої бази даних.



Створимо нову базу даних, виконавши такі команди:

**CREATE DATABASE** mydb CHARACTER SET utf8 COLLATE DEFAULT;

**CREATE TABLE** IF NOT EXISTS `mydb`.`Sale` (

`SaleId` INT NOT NULL AUTO\_INCREMENT,

`Customer\_ID` INT NOT NULL,

`SaleDate` DATETIME NOT NULL,

`Product\_ID` INT NOT NULL,

`Amount` INT NOT NULL,

`Price` INT NOT NULL,

`Employee\_ID` INT NOT NULL,

`Bakery\_ID` INT NOT NULL,

**PRIMARY KEY** (`SaleId`)

);

**CREATE TABLE** IF NOT EXISTS `mydb`.`Employee` (

`EmployeeId` INT NOT NULL AUTO\_INCREMENT,

`EmployeeSecondName` VARCHAR(15) NOT NULL,

`EmployeeFirstName` VARCHAR(15) NOT NULL,

`EmployeeAppointment` VARCHAR(15) NOT NULL,

`EmployeeEmail` VARCHAR(15) NOT NULL,

**PRIMARY KEY** (`EmployeeId`),

**CONSTRAINT** `fk\_Employee\_Sale1` **FOREIGN KEY** (`EmployeeId`)

**REFERENCES** `mydb`.`Sale` (`Employee\_ID`) **ON DELETE** NO ACTION **ON UPDATE** NO ACTION

);

**CREATE TABLE** IF NOT EXISTS `mydb`.`Customer` (

`CustomerId` INT NOT NULL AUTO\_INCREMENT,

`CustomerSecondName` VARCHAR(15) NOT NULL,

`CustomerFirstName` VARCHAR(15) NOT NULL,

`Country` VARCHAR(15) NOT NULL,

`CustomerEmail` VARCHAR(15) NOT NULL,

`CustomerContactNumber` INT NOT NULL,

`Sale\_SaleId` INT NOT NULL,

**PRIMARY KEY** (`CustomerId`, `Sale\_SaleId`),

**CONSTRAINT** `fk\_Customer\_Sale` **FOREIGN KEY** (`CustomerId`)

**REFERENCES** `mydb`.`Sale` (`Customer\_ID`) **ON DELETE** NO ACTION **ON UPDATE** NO ACTION

);

**CREATE TABLE** IF NOT EXISTS `mydb`.`Product` (

`ProductId` INT NOT NULL AUTO\_INCREMENT,

`ProductName` VARCHAR(15) NOT NULL,

`BestBy` VARCHAR(15) NOT NULL,

`Created` VARCHAR(15) NOT NULL,

`ProductPrice` INT NOT NULL,

**PRIMARY KEY** (`ProductId`),

**CONSTRAINT** `fk\_Product\_Sale1` **FOREIGN KEY** (`ProductId`)

**REFERENCES** `mydb`.`Sale` (`Product\_ID`) **ON DELETE** NO ACTION **ON UPDATE** NO ACTION

);

**CREATE TABLE** IF NOT EXISTS `mydb`.`Purchase` (

`PurchaseId` INT NOT NULL AUTO\_INCREMENT,

`RawStuff\_ID` INT NOT NULL,

`PurchaseDate` DATETIME NOT NULL,

`Storage\_ID` INT NOT NULL,

`PurchaseAmount` INT NOT NULL,

`PurchasePrice` INT NOT NULL,

`Bakery\_ID` INT NOT NULL,

**PRIMARY KEY** (`PurchaseId`)

);

**CREATE TABLE** IF NOT EXISTS `mydb`.`Bakery` (

`BakeryId` INT NOT NULL AUTO\_INCREMENT,

`BakeryCity` VARCHAR(15) NOT NULL,

`BakeryStreet` VARCHAR(15) NOT NULL,

`BakeryContactNumber` VARCHAR(15) NOT NULL,

`BakeryEmail` VARCHAR(15) NOT NULL,

**PRIMARY KEY** (`BakeryId`),

**CONSTRAINT** `fk\_Bakery\_Sale1` **FOREIGN KEY** (`BakeryId`)

**REFERENCES** `mydb`.`Sale` (`Bakery\_ID`) **ON DELETE** NO ACTION **ON UPDATE** NO ACTION,

**CONSTRAINT** `fk\_Bakery\_Purchase1` **FOREIGN KEY** (`BakeryId`)

**REFERENCES** `mydb`.`Purchase` (`Bakery\_ID`) **ON DELETE** NO ACTION  **ON UPDATE** NO ACTION

);

**CREATE TABLE** IF NOT EXISTS `mydb`.`Storage` (

`StorageId` INT NOT NULL AUTO\_INCREMENT,

`StorageCity` VARCHAR(15) NOT NULL,

`StorageStreet` VARCHAR(15) NOT NULL,

`StorageContactNumber` VARCHAR(15) NOT NULL,

`StorageRow` VARCHAR(15) NOT NULL,

`StorageShelf` VARCHAR(15) NOT NULL,

**PRIMARY KEY** (`StorageId`),

**CONSTRAINT** `fk\_Storage\_Purchase1` **FOREIGN KEY** (`StorageId`)

**REFERENCES** `mydb`.`Purchase` (`Storage\_ID`) **ON DELETE** NO ACTION **ON UPDATE** NO ACTION

);

**CREATE TABLE** IF NOT EXISTS `mydb`.`Recipe` (

`RecipeId` INT NOT NULL AUTO\_INCREMENT,

`RecipeName` VARCHAR(15) NOT NULL,

`RecipeIngredient` INT NOT NULL,

`RecipeAuthor` VARCHAR(15) NOT NULL,

`RecipeCountry` VARCHAR(15) NOT NULL,

**PRIMARY KEY** (`RecipeId`)

);

**CREATE TABLE** IF NOT EXISTS `mydb`.`RawStuff` (

`RawStuffId` INT NOT NULL AUTO\_INCREMENT,

`RawStuffName` VARCHAR(45) NOT NULL,

`RawStuffPresence` INT NOT NULL,

`RawStuffMaxAmount` INT NOT NULL,

`RawStuffBestBy` VARCHAR(15) NOT NULL,

`RawStuffPrice` INT NOT NULL,

**PRIMARY KEY** (`RawStuffId`),

**CONSTRAINT** `fk\_RawStuff\_Recipe1` **FOREIGN KEY** (`RawStuffId`)

**REFERENCES** `mydb`.`Recipe` (`RecipeIngredient`) **ON DELETE** NO ACTION **ON UPDATE** NO ACTION,

**CONSTRAINT** `fk\_RawStuff\_Purchase1` **FOREIGN KEY** (`RawStuffId`)

**REFERENCES** `mydb`.`Purchase` (`RawStuff\_ID`) **ON DELETE** NO ACTION **ON UPDATE** NO ACTION

);

**Висновки:**

Виконавши дану лабораторну роботу, я завершив моделювання і засобами SQL створив базу даних, що складається з дев’яти таблиць.