**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ІВАНА ФРАНКА**

**ФАКУЛЬТЕТ УПРАВЛІННЯ ФІНАНСАМИ ТА БІЗНЕСУ**

**Кафедра цифрової економіки та бізнес-аналітики**

**КУРСОВА РОБОТА**

**з навчальної дисципліни**

**“Проектування та адміністрування БД і СД”**

**на тему:**

**«Інформаційна система для фiрми що виробляє електротехнiку.»**

Спеціальність: 051 «Економіка»

Спеціалізація: «Інформаційні технології в бізнесі»

Освітній ступінь: бакалавр

**Науковий керівник:         Виконавець:**

к.ф.-м.н., доц. Депутат Б.Я.Васько А.Р.        

           (прізвище, ім’я, по-батькові)                                 (прізвище, ім’я, по-батькові)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (підпис)                  УФЕ-31с        група

                                                                                \_\_\_\_\_\_\_\_\_\_\_\_\_\_ (підпис)

**“\_\_\_” \_\_\_\_\_\_\_\_\_\_\_\_\_ 2022 р.                               “\_\_\_” \_\_\_\_\_\_\_\_\_\_\_\_ 2022 р.**

**Загальна кількість балів** \_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис, ПІП членів комісії)

**Львів 2022**

**ЗМІСТ**

[**ВСТУП** 3](#_Toc104434127)

[**РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ** 5](#_Toc104434128)

[**1.1** **Постановка задачі** 5](#_Toc104434129)

[**1.2** **Розробка моделі варіантів використання інформаційної системи** 6](#_Toc104434130)

[**РОЗДІЛ 2. РОЗРОБКА БАЗИ ДАНИХ** 11](#_Toc104434131)

[**2.1 Аналіз засобів реалізації** 11](#_Toc104434132)

[**2.2 Розробка структури бази даних** 14](#_Toc104434133)

[**2.3 Перелік таблиць, полів та типів даних бази даних** 18](#_Toc104434134)

[**2.4 Реалізація SQL-скрипту бази даних** 24](#_Toc104434135)

[**ВИСНОВОК** 41](#_Toc104434136)

[**СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ** 42](#_Toc104434137)

**ВСТУП**

**Актуальність теми дослідження.**

Сьогодні, в еру стрімкого розвитку інформаційних технологій, бази даних стали невід'ємною частиною інформаційних систем та різноманітних програм. Від продуктового магазинчика до транснаціональних мегакорпорацій, у приватних та державних секторах – вони використовуються всюди. Їх необхідність зумовлена зростаючим темпом життя суспільства і аби втриматися на плаву, необхідно швидко і якісно отримувати, зберігати, обробляти та упорядковувати інформацію. Саме для цього й існують інформаційні системи.

**Мета та завдання роботи.**

*Метою* даної роботи є проектування та розробка інформаційної системи для підприємства, що виробляє електротехніку.

Для досягнення цієї мети необхідно виконати наступні *завдання:*

* Коротко охарактеризувати поставлену задачу;
* Проаналізувати предметну область роботи;
* Спроектувати архітектуру бази даних для підприємства, що виробляє електротехніку;
* Обрати та обгрунтувати вибір технологій та інструментарію для розробки;
* Розробити інформаційну систему для підприємства, що виробляє електротехніку;

**Об’єкт дослідження.**

Об’єктом даного дослідження є особливості розробки інформаційної системи для підприємства, що виробляє електротехніку.

**Предмет дослідження.**

Предметом даного дослідження являється використання технологій створення програмноо забезпечення для розробки інформаційної системи підприємства, що виробляє електротехніку.

**Використане програмне забезпечення.**

В процесі розробки будуть використовуватись наступне програмне забезпечення:

* Dbdiagram.io – онлайн-сервіс для дизайну та проектування баз даних;
* MySQL Server – сервер для баз даних;
* MySQL Workbench – система управління базами даних (СУБД);

**Структура роботи.**

Курсова робота складається з двох розділів («Аналіз предметної області» та «Розробка бази даних»), висновків, списку використаних джерел та додатків. Загальний обсяг роботи – сторінок.

**РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ**

* 1. **Постановка задачі**

В сучасності, інформація стала одним з основних ресурсів розвитку суспільства, а інформаційні системи – як засіб підвищення продуктивності праці та ефективності роботи персоналу.

Переробка інформації – одна з найважливіших функцій, без якої немислима успішна діяльність будь-якої організаційно-виробничої системи (підприємства, організації, тощо). Систему, що реалізує функції збору, зберігання, обробки і передачі інформації, називають інформаційною системою. Найважливішими функціями цих систем є облік, аналіз, контроль і регулювання.

Для виконання поставлених задач необхідно реалізувати вищезгадані функції при розробці інформаційної системи для підприємства, що виробляє електротовари.

Для розуміння масштабу роботи необхідно провести певний аналіз нашого підприємства з метою розуміння його бізнес-потреб та структурної організації:

Підприємство Perfect Tools займається виробництвом електроінструменту та електроприладів. На даному етапі свого розвитку, підприємство володіє наступними активами:

* Чотирма магазинами роздрібних продаж - у містах Львів, Київ, Рівне, Житомир;
* Двома фабриками – у містах Львів та Рівне;
* Двома складами – у містах Львів та Київ;

Підприємство сильно централізоване – кадрові рішення та рішення щодо асортименту товарів належать власникові. Підприємство продає товари як оптом так і в роздріб. При оптовій продажі, замовник забирає товар зі складу самовивозом. Матеріали поставляються і доставляються фірмою-постачальником на склад, звідки перерозподіляються по виробництвах.

* 1. **Розробка моделі варіантів використання інформаційної системи**

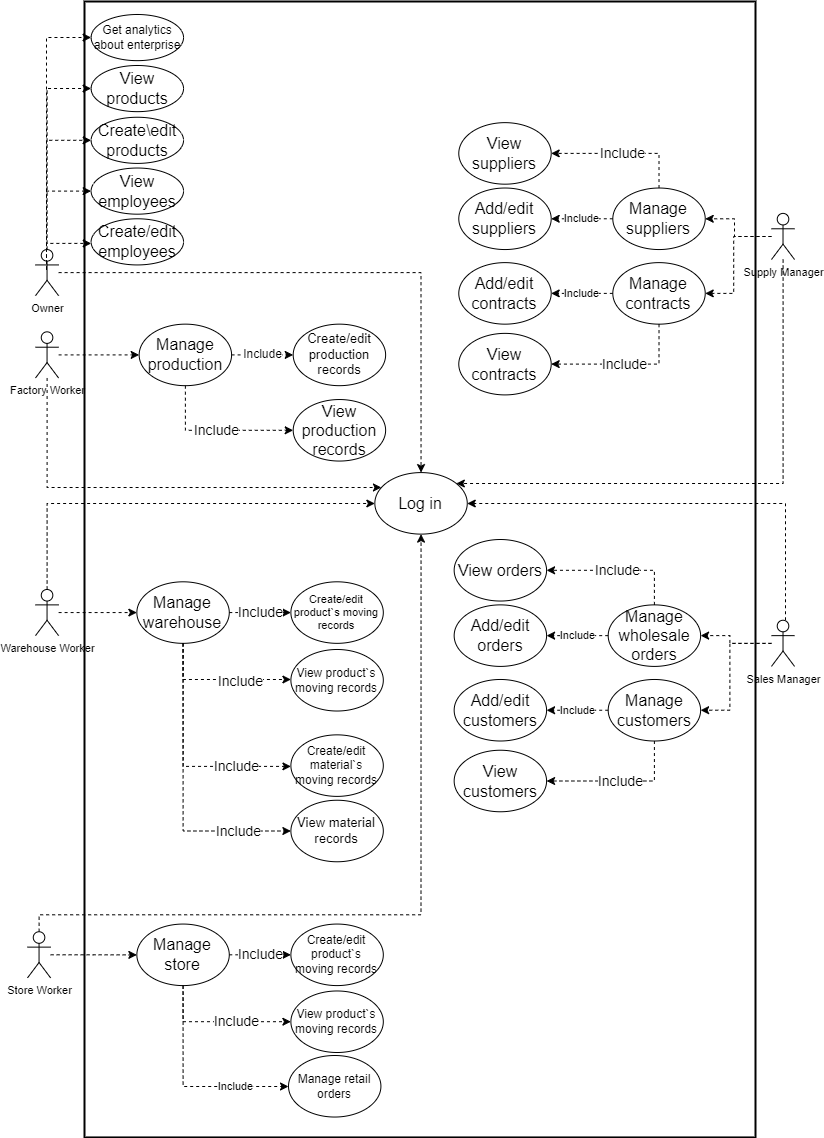


Рис. 1 Use Case Діаграма для підприємства, що виробляє електротехніку

Для розробки успішної інформаційної системи необхідно розуміти хто і як нею буде користуватись. Вирішити це завдання допомагає *Use Case діаграма* (дослівно – діаграма варіантів використання).

Згідно нашої Use Case діаграми (рис. 1) ми можемо бачити наступних користувачів нашої інформаційної системи (так названих - акторів) та їхні способи використання системи:

* **Owner –** власник компанії;
  + **Збирати аналітику про підприємство** – власник повинен отримувати повну аналітичну та статистичну інформацію про підприємство.
  + **Переглядати асортимент товару** – власник має можливість переглядати вже існуючий асортимент товару.
  + **Створювати\Редагувати продукти** – власник власноруч створює та редагує асортимент товарів. Це спричинено централізованим підходом до керування підприємством
  + **Переглядати інофрмацію про працівників** – власник може отримати список працівників із вицерпною інформацією про них.
  + **Додавати\Видаляти працівників** – власник особисто наймає та звільняти працівників. Це спричинено централізованим підходом до керування підприємством.
* **Factory worker –** працівник фабрики;
  + **Вести облік виробництва** – основне завдання працівника фабрики, включає такі підзавдання:
    - **Переглядати записи обліку виробництва** – працівник повинен мати змогу переглядати необхідну йому інформацію про виробництво.
    - **Створювати\Редагувати записи обліку виробництва –** працівник повинен створювати записи про облік виробництва.
* **Warehouse worker –** працівник складу;
  + **Вести складський облік** – основне завдання працівника складу, включає такі підзавдання:
    - **Переглядати записи про переміщення товарів** – працівник повинен володіти інформацією про переміщення товарів по складах та магазинах.
    - **Створювати\Редагувати записи про переміщення товарів** – працівник повинен мати змогу стоврювати записи про переміщення товарів по складах та магазинах.
    - **Переглядати записи про переміщення матеріалів** – працівник повинен володіти інформацією про переміщення матеріалів по складах та фабриках.
    - **Створювати\Редагувати записи про переміщення матеріалів** - працівник повинен мати змогу стоврювати записи про переміщення матеріалів по складах та фабриках.
* **Store worker –** працівникмагазину**;**
  + **Вести облік магазину –** основне завданняпрацівника магазину, включає такі підзавдання:
    - **Переглядати інформацію про переміщення товарів зі складу в магазин –** працівникповинен мати змогу переглядати записи про переміщення товарів зі складів до магазину.
    - **Створювати/Редагувати записи про оприбуткування товарів в магазин –** працівник повинен мати змогу оприбуткувати товари, що надійшли зі складу.
    - **Вести облік роздрібних замовлень –** працівник повинен мати змогу вести облік роздрібних продаж.
* **Supply manager –** менеджер поставок;
  + **Вести облік постачальників-** перше основне завдання працівника, включає в себе такі підзавдання:
    - **Переглядати інформацію про постачальників –** працівник повинен мати змогу переглядати вичерпну інформацію про діючих постачальників.
    - **Створювати\Редагувати список постачальників –** працівник повинен мати змогу створювати\редагувати постачальників.
  + **Вести облік контрактів на поставку –** друге основне завдання працівнка, включає в себе такі підзавдання:
    - **Переглядати список заключених контрактів –** працівник повинен мати змогу переглядати список контрактів та супутню інформацію.
    - **Створювати\Редагувати контракти –** працівник повинен мати змогу створювати записи про нові контракти та редагувати старі.
* **Sales manager –** менеджер продаж;
  + **Вести облік оптових замовленнь –** перше основне завдання працівника, включає в себе такі підзавдання:
    - **Переглядати інформацію про оптові замовлення –** працівник повинен мати змогу переглядати список оптових замовленнь і відповідну аналітику.
    - **Створювати\Редагувати оптові замовлення –** працівник повинен мати змогу створювати та керувати оптовими замовленнями.
  + **Вести облік оптових замовників -** друге основне завдання працівнка, включає в себе такі підзавдання:
    - **Переглядати записи про оптових замовників –** працівник повинен мати змогу отримувати вичерпну інформацію про оптових замовників.
    - **Створювати\Редагувати записи про оптових замовників –** працівник повинен мати змогу створювати та керувати записами про оптових замовників.

**РОЗДІЛ 2. РОЗРОБКА БАЗИ ДАНИХ**

**2.1 Аналіз засобів реалізації**

При виконанні курсової роботи було використано наступне програмне забезпечення:

* **Dbdiagram.io;**
* **MySQL Server;**
* **MySQL Workbench;**

**Dbdiagram.io** – це безплатиний онлайн сервіс для моделювання та дизайну баз даних.

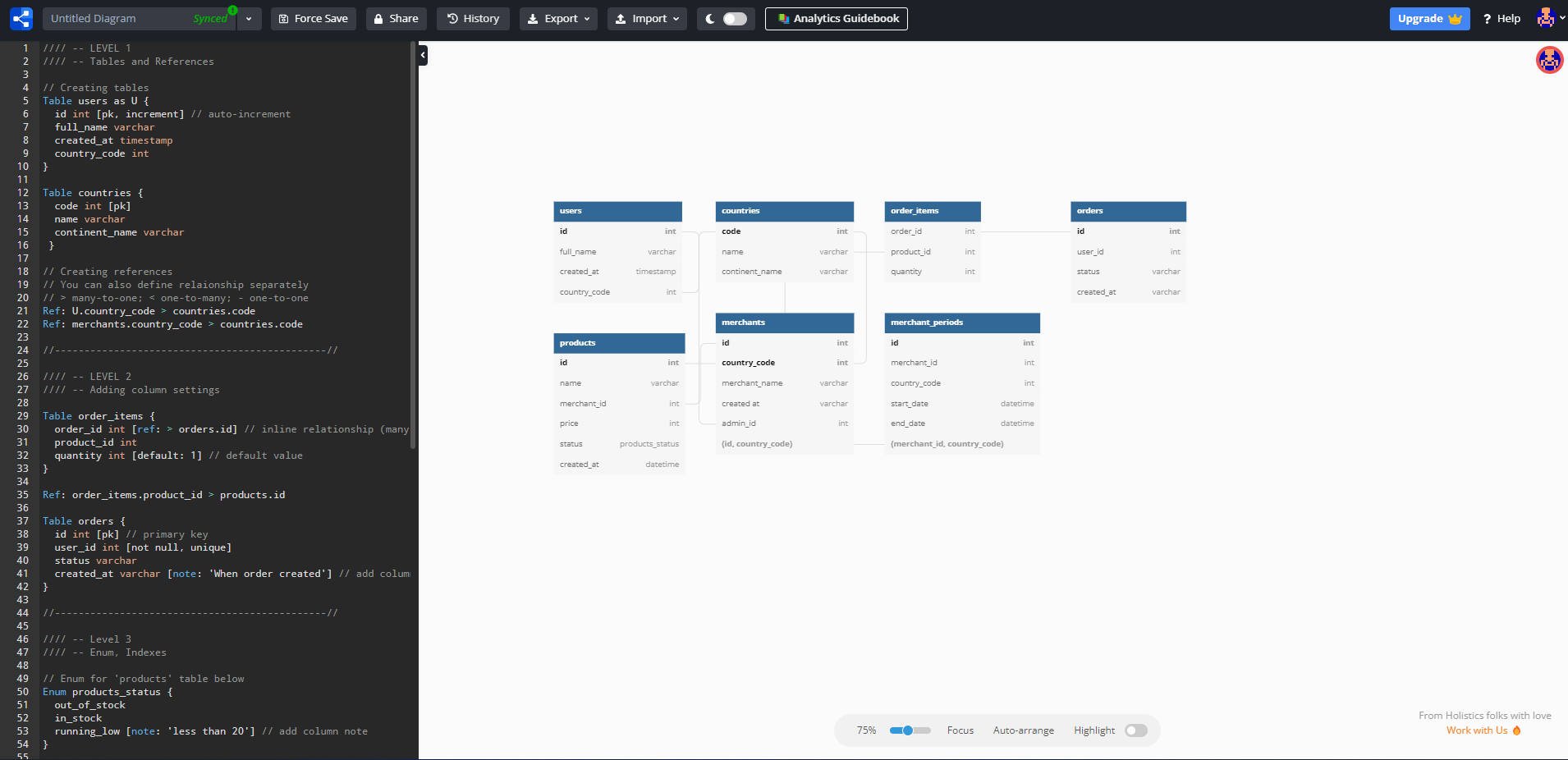


Рис. 2 Зовнішній вигляд онлайн сервісу Dbdiagram.io

dbdiagram.io — це швидкий і простий конструктор баз даних, який допоможе вам малювати діаграми бази даних, використовуючи їх власну доменну мову (Domain-specific language, DSL). Ця мова дуже проста та інтуїтивно зрозуміла, що робить користування нею дуже зручним. Зокрема даний сервіс дозволяє ділитися своїми діаграмами а також працювати в команді.

Після проектування бази даних, сервіс дозволяє експортувати базу у більшість сучасних СУБД – PostgreSQL, MySQL та SQL Server.

**MySQL –** це одна із найпоширеніших реляційних систем управління базами даних. Дана система працює за моделю «клієнт-сервер». Принцип роботи як у будь-яких клієнт-серверних моделях. Один пристрій запитує, а другий відповідає. Запитувачів може бути більше одного, все залежить від сервера, мережі та поставлених завдань.

Технічно дещо інші, але за своєю суттю ідентичні процеси відбуваються в середовищі MySQL:

* Система створює базу даних для зберігання інформації (її сортування, ідентифікації тощо).
* Клієнти (інші комп'ютери мережі) подають запити до бази з допомогою специфічних SQL команд.
* Серверний додаток обробляє запит і видає відповідь клієнту (видає дані, що запитуються).

Для взаємодії з MySQL сервером використовуються відповідні утиліти. Деякі працюють лише у командному рядку. Деякі володіють графічним інтерфейсом. Популярні рішення - WorkBench, SequelPro, SQL Studio, TablePlus. Правда, більшість вебмайстрів віддає перевагу phpMyAdmin та MySQL Workbench, тому що вони входять у LAMP (Linux, Apache, MySQL, PHP/Perl/Python) - що означає один із найпоширеніших програмних стеків для багатьох найпопулярніших веб-додатків.

Перевагами MySQL над іншими системами є:

* Гнучкість та легкість у використанні. На створення та підтримку БД йде менше часу. Потрібний менший рівень компетенції для того, щоб повноцінно працювати з MySQL та реалізувати весь її потенціал.
* Має відкритий вихідний код, тому легко піддається модифікації.
* Підтримується компанією Oracle та спільнотою розробників, що виступають за розвиток opensource додатків.
* Працює швидше конкурентів. Внутрішня структура MySQL дозволяє розгрібати завали з таблиць і рядків за секунди. Незалежно від складності зв'язків між даними та їх кількості, сервер обробляє запити будь-якої складності швидше за інші системи.
* Стала відомим ім'ям і певним стандартом у промисловості. Компанії шукають співробітників, які вміють працювати з MySQL, інтернет наповнений інструкціями по роботі саме з MySQL-серверами.
* Може похвалитися високим рівнем захисту даних завдяки системі видачі прав та просунутій системі керування користувачами. А ще є верифікація на основі хостингу та шифрування.

**2.2 Розробка структури бази даних**

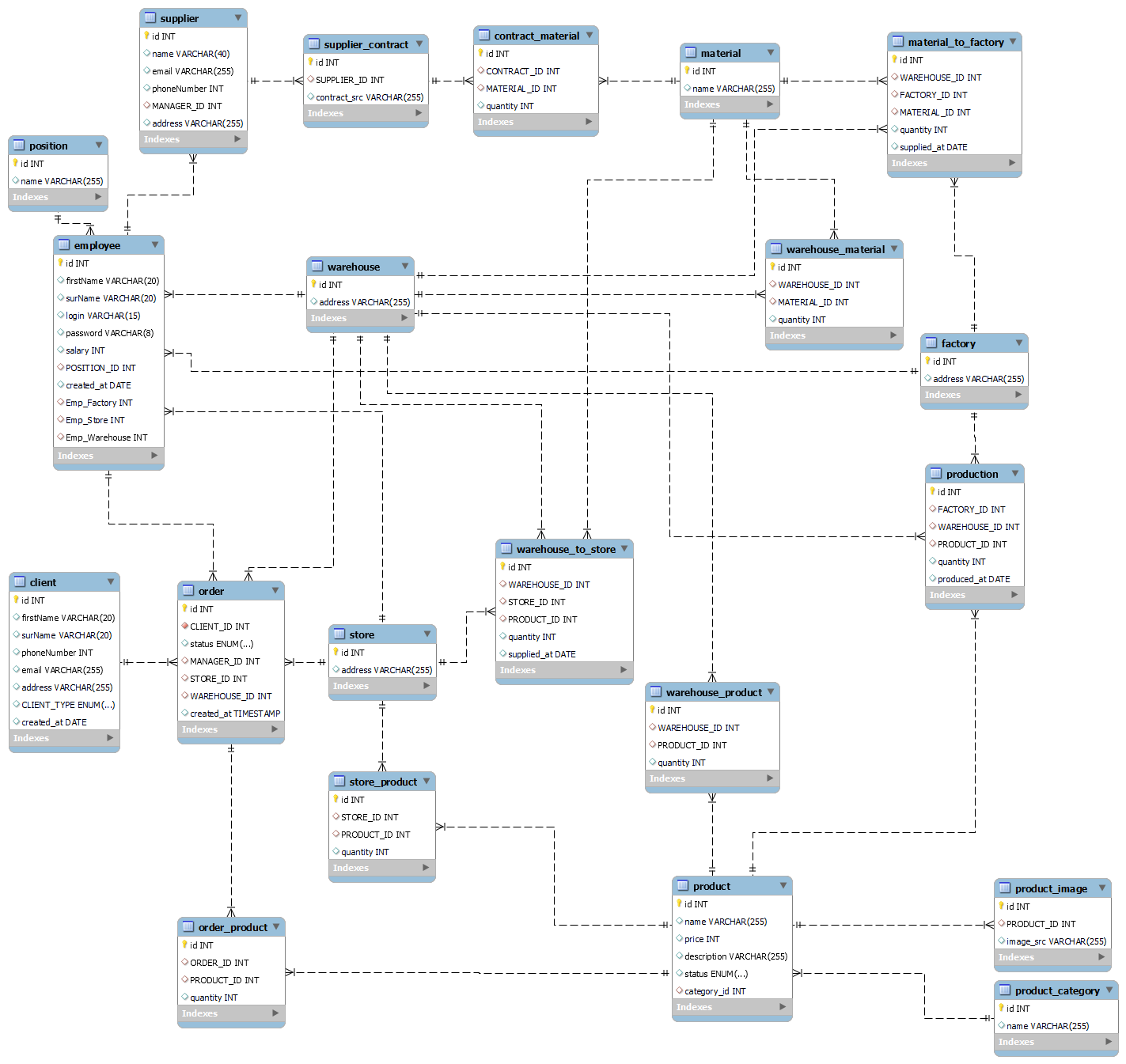


Рис. 3 Entity-Relationship діаграма

На рис. 3 можна побачити діаграму сутностей і зв’язків бази даних підприємства, що виробляє електротовари. Дана база складається із двадцяти таблиць, об’єднанив в єдину систему. На перший погляд вона виглядає доволі складною та комплексною. Для полегшення розуміння діаграми, розіб’ємо її на окремі розділи:

**Облік постачальників та матеріалів:**

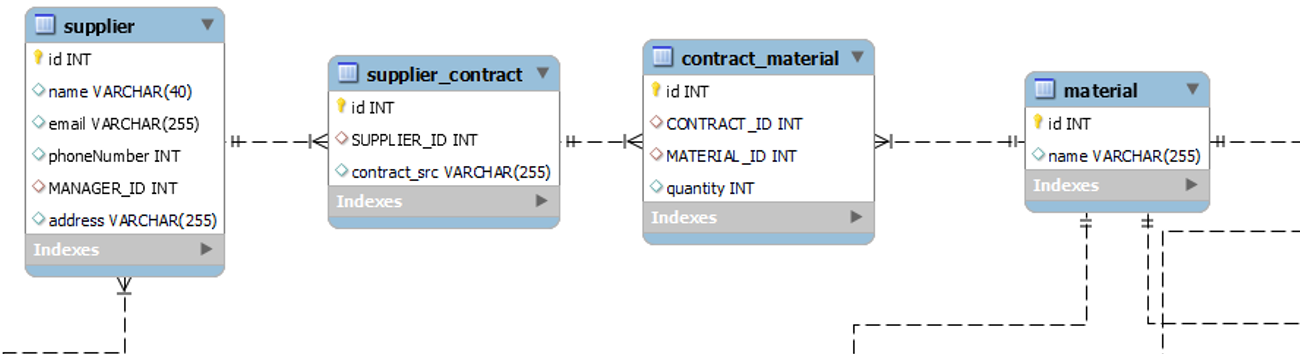


Рис. 4 Таблиці обліку постачальників та матеріалів

В даній групі знаходяться всі таблички, необхідні для обліку постачальників та матеріалів.

**Облік складу:**

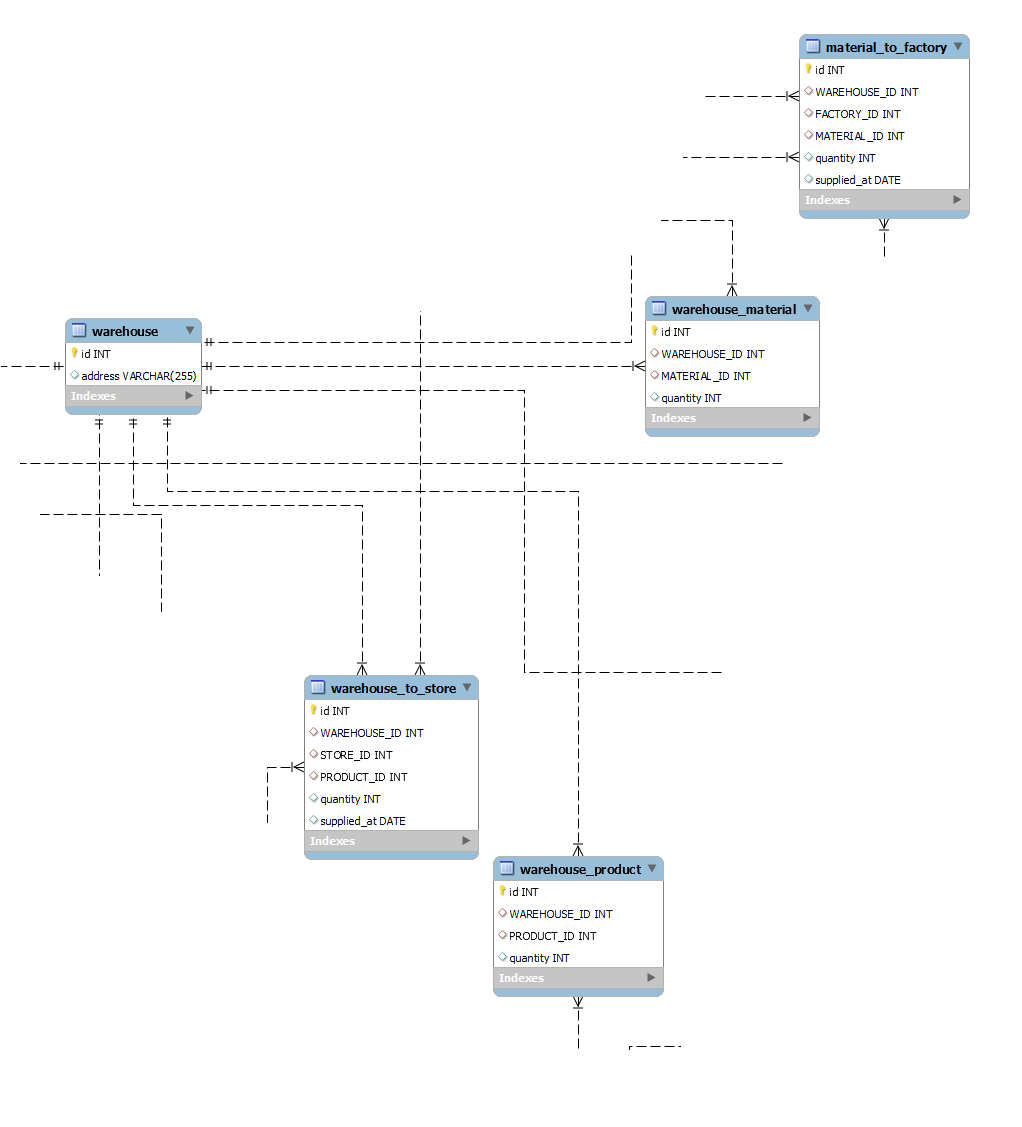


Рис. 5 Таблиці обліку складу

В даній групі знаходяться всі таблички, необхідні для обліку складів.

**Облік магазинів:**

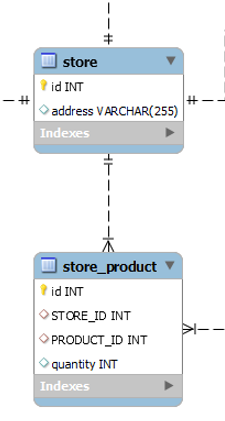
****

Рис. 6 Таблиці обліку магазинів

В даній групі знаходяться всі таблички, необхідні для обліку магазинів.

**Облік виробництва:**

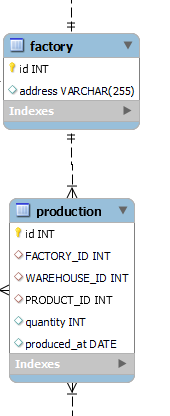
****

Рис. 7 Таблиці обліку виробництва

В даній групі знаходяться всі таблички, необхідні для обліку виробництва.

**Облік замовлень:**

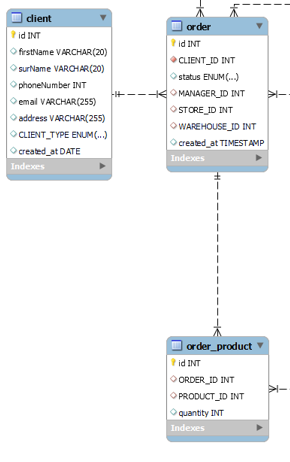
****

Рис. 8 Таблиці обліку замовлень

В даній групі знаходяться всі таблички, необхідні для обліку замовлень.

**Облік працівників:**

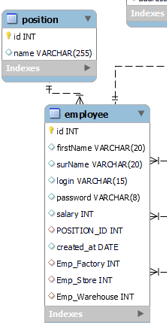


Рис. 9 Таблиці для обліку працівників

В даній групі знаходяться всі таблички, необхідні для обліку працівників.

**Облік асортименту товарів:**

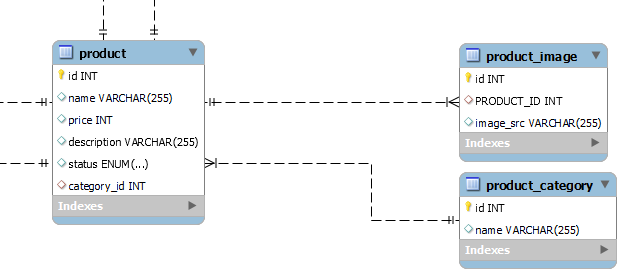
****

Рис. 10 Таблиці обліку асортименту товарів

В даній групі знаходяться всі таблички, необхідні для обліку асортименту товарів.

**2.3 Перелік таблиць, полів та типів даних бази даних**

Як видно із ER-діаграми (рис. 3), наша база даних складається із 20-ти таблиць. В кожній таблиці є свої поля із визначеними типами даних. Основні типи даних, які використовувались при розробці:

1.INT – від -2 147 483 648 до 2 147 483 647;

2.VARCHAR – до 8000 символів;

3.DATE – від 0001-01-01 до 9999-12-31 (YYYY-ММ-DD);

**Таблиці обліку постачальників:**

* Таблиця **Supplier –** таблиця, що містить інформацію про поставищків, містить наступні поля:
  + **Id –** тип даних INT;
  + **Name –** тип даних VARCHAR;
  + **Email –** тип даних VARCHAR;
  + **phoneNumber –** тип даних INT;
  + **MANAGER\_ID –** тип даних INT;
  + **Address –** тип даних VARCHAR;
* Таблиця **supplier\_contract –** таблиця містить інформацію про підписані контракти на поставку матеріалів, містить наступні поля:
  + **Id –** тип даних INT;
  + **SUPPLIER\_ID -** тип даних INT;
  + **contract\_src -** тип даних VARCHAR;
* Таблиця **contract\_material –** таблиця містить інформацію про вид та кількість матеріалу поставленого за певним контрактом, містить наступні поля:
  + **Id –** тип даних INT;
  + **CONTRACT\_ID –** тип даних INT;
  + **MATERIAL\_ID –** тип даних INT;
  + **quantity -** тип даних INT;
* Таблиця **material –** містить інформацію про матеріали, містить наступні поля:
  + **Id –** тип даних INT;
  + **Name –** тип даних VARCHAR;

**Таблиці обліку робітників:**

* Таблиця **employee** – містить інформацію про робітників на підприємстві, містить наступні поля:
  + **Id –** тип даних INT;
  + **firstName –** тип даних VARCHAR;
  + **surName –** тип даних VARCHAR;
  + **login –** тип даних VARCHAR;
  + **password –** тип даних VARCHAR;
  + **salary –** тип даних INT;
  + **POSITION\_ID –** тип даних INT;
  + **created\_at –** тип даних DATE;
  + **Emp\_Factory –** тип даних INT;
  + **Emp\_Store –** тип даних INT;
  + **Emp\_Warehouse –** тип даних INT;
* Таблиця **position –** містить перелік посад, містить наступні поля:
  + **Id –** тип даних INT;
  + **Name –** тип даних VARCHAR;

**Таблиці обліку замовлень:**

* Таблиця **order –** містить інофрмацію про замовлення, містить наступні поля:
  + **Id –** тип даних INT;
  + **CLIENT\_ID –** тип даних INT;
  + **status –** тип даних ENUM, може набувати значень – order\_accepted, order\_completed, order\_in\_process, order\_cancelled;
  + **MANAGER\_ID –** тип даних INT;
  + **STORE\_ID –** тип даних INT;
  + **WAREHOUSE\_ID –** тип даних INT;
  + **created\_at –** тип даних TIMESTAMP;
* Таблиця **client –** містить інформацію про клієнтів, містить наступні поля:
  + **Id –** тип даних INT;
  + **firstName –** тип даних VARCHAR;
  + **surName –** тип даних VARCHAR;
  + **phoneNumber –** тип даних INT;
  + **Email –** тип даних VARCHAR;
  + **Address –** тип даних VARCHAR;
  + **CLIENT\_TYPE -** тип даних ENUM, може набувати значень – retail, wholesale;
  + **created\_at –** тип даних DATE;
* Таблиця **order\_product –** містить назву і кількість товарів куплених у замоленнях:
  + **Id –** тип даних INT;
  + **ORDER\_ID –** тип даних INT;**,**
  + **PRODUCT\_ID –** тип даних INT;
  + **quantity –** тип даних INT;

**Таблиці обліку магазинів:**

* Таблиця **store –** містить інформацію про магазини роздрібних продаж, містить наступні поля:
  + **Id –** тип даних INT;
  + **Address –** тип даних VARCHAR;
* Таблиця **store\_product –** відображує кількість конкретного товару в конкретному магазині, містить наступні поля:
  + **Id –** тип даних INT;
  + **STORE\_ID –** тип даних INT;
  + **PRODUCT\_ID –** тип даних INT;
  + **quantity –** тип даних INT;

**Таблиці обліку складів:**

* Таблиця **warehouse –** відображує інформацію про склади, містить наступні поля:
  + **Id –** тип даних INT;
  + **Address –** тип даних VARCHAR;
* Таблиця **material\_to\_factory –** містить записи про переміщення матеріалів зі складу на виробництво, містить наступні поля:
  + **Id –** тип даних INT;
  + **WAREHOUSE\_ID –** тип даних INT;
  + **FACTORY\_ID –** тип даних INT;
  + **MATERIAL\_ID –** тип даних INT;
  + **quantity –** тип даних INT;
  + **supplied\_at –** тип данихDATE;
* Таблиця **warehouse\_material –** містить інформацію про вид та кількість матеріалів на складах, містить наступні поля:
  + **Id –** тип даних INT;
  + **WAREHOUSE\_ID –** тип даних INT;
  + **MATERIAL\_ID –** тип даних INT;
  + **quantity –** тип даних INT;
* Таблиця **warehouse\_to\_store–** містить записи про переміщення товарів зі складу в магазини, містить наступні поля:
  + **Id –** тип даних INT;
  + **WAREHOUSE\_ID –** тип даних INT;
  + **STORE\_ID –** тип даних INT;
  + **PRODUCT\_ID –** тип даних INT;
  + **quantity –** тип даних INT;
  + **supplied\_at –** тип данихDATE;
* Таблиця **warehouse\_product –** містить інформацію про вид та кількість товарів на складах, містить наступні поля:
  + **Id –** тип даних INT;
  + **WAREHOUSE\_ID –** тип даних INT;
  + **PRODUCT\_ID –** тип даних INT;
  + **quantity –** тип даних INT;

**Таблиці обліку виробництва:**

* Таблиця **factory –** відображує інформацію про фабрики, містить наступні поля:
  + **Id –** тип даних INT;
  + **Address –** тип даних VARCHAR;
* Таблиця **production –** відображує записи про виробництво, містить наступні поля:
  + **Id –** тип даних INT;
  + **FACTORY\_ID –** тип даних INT;
  + **WAREHOUSE\_ID –** тип даних INT;
  + **PRODUCT\_ID –** тип даних INT;
  + **quantity –** тип даних INT;
  + **produced\_at –** тип данихDATE;

**Таблиці обліку асортимунту товарів:**

* Таблиця **product –** містить інформацію про асортимент продуктів, містить наступні поля:
  + **Id –** тип даних INT;
  + **Name –** тип даних VARCHAR;
  + **Price –** тип даних INT;
  + **Description –** тип даних VARCHAR;
  + **Status -** тип даних ENUM, може набувати значень – in\_stock, out\_of\_stock, running\_low;
  + **Category\_id –** тип даних INT;
* Таблиця **product\_image –** містись посилання на фотографії продукції, містить наступні поля:
  + **Id –** тип даних INT;
  + **PRODUCT\_ID –** тип даних INT;
  + **Image\_src –** тип даних VARCHAR;
* Таблиця **product\_category –** містить перелік категорій товарів, містить наступні поля:
  + **Id –** тип даних INT;
  + **Name –** тип даних VARCHAR

**2.4 Реалізація SQL-скрипту бази даних**

Одним із важливих завдань даної роботи – реалізувати способи використання інформаційної системи користувачами а саме – задовольнити їх Use Case, описані в Use Case діаграмі (рис. 1). Нижче наведено лістинг SQL-скрипту який виконує поставлені перед ним завдання:

/\* OWNER \*/

/\*Операції з працівниками\*/

/\* Додавання нового працівника на фабрику \*/

*INSERT INTO employee (firstName, surName, login, password, salary, POSITION\_ID, created\_at, Emp\_Factory) VALUES ('Courtney', 'Hope', 'CourtneyHope', 'Courtney', 16000, 3, '2010-01-02', 1);*

/\* Додавання нового працівника в магазин \*/

*INSERT INTO employee (firstName, surName, login, password, salary, POSITION\_ID, created\_at, Emp\_Store) VALUES ('Saddam', 'Fus', 'SaddamFus', 'Saddam', 21000, 5, '2010-01-05', 1);*

/\* Додавання нового працівника на склад \*/

*INSERT INTO employee (firstName, surName, login, password, salary, POSITION\_ID, created\_at, Emp\_Warehouse) VALUES ('Gans', 'Hose', 'GansHose', 'Gans', 13000, 4, '2010-01-08', 1);*

/\* Додавання нового менеджера продаж \*/

*INSERT INTO employee (firstName, surName, login, password, salary, POSITION\_ID, created\_at) VALUES ('Chloe', 'Doe', 'ChloeDoe', 'Chloe', 31000, 2, '2010-01-04');*

/\* Додавання нового менеджера поставок \*/

*INSERT INTO employee (firstName, surName, login, password, salary, POSITION\_ID, created\_at) VALUES ('Daria', 'Kell', 'DariaKell', 'Daria', 29500, 1, '2010-01-05');*

/\* Видалення працівника \*/

*DELETE FROM employee*

*where firstName = 'Daria';*

/\* Перегляд всіх робітників \*/

*SELECT employee.id as 'Employee Id', firstName as 'First Name', surName as 'Surname', position.name as 'Position', salary as 'Salary'*

*FROM employee*

*JOIN position on position.id = POSITION\_ID*

*ORDER BY employee.id;*

/\* Аналітика по замовленнях \*/

/\* Всі роздрібні замовлення \*/

*SELECT client.firstName as 'Client name', client.address as 'Client address', perfect\_tools.ORDER.id as 'Order Id', perfect\_tools.order.status, perfect\_tools.order.created\_at, product.name as 'Product name', order\_product.quantity, order\_product.quantity \* product.price as 'Total sum'*

*FROM perfect\_tools.ORDER*

*join client on CLIENT\_ID = client.id*

*join order\_product on ORDER\_ID = perfect\_tools.order.id*

*join product on product.id = order\_product.PRODUCT\_ID*

*WHERE client.CLIENT\_TYPE = 'retail';*

/\* Всі оптові замовлення \*/

*SELECT client.firstName as 'Client name', client.address as 'Client address', perfect\_tools.ORDER.id as 'Order Id', perfect\_tools.order.status, perfect\_tools.order.created\_at, product.name as 'Product name', order\_product.quantity, order\_product.quantity \* product.price as 'Total sum'*

*FROM perfect\_tools.order*

*join client on CLIENT\_ID = client.id*

*join order\_product on ORDER\_ID = perfect\_tools.order.id*

*join product on product.id = order\_product.PRODUCT\_ID*

*WHERE client.CLIENT\_TYPE = 'wholesale';*

/\* Сумарний заробіток від замовлень \*/

*SELECT Count(DISTINCT perfect\_tools.order.id) as 'Number of orders' , SUM(product.price \* order\_product.quantity) as 'Total retail sum'*

*FROM order\_product*

*JOIN perfect\_tools.order on order\_product.ORDER\_ID = perfect\_tools.order.id*

*JOIN product on order\_product.PRODUCT\_ID = product.id*

*JOIN client on perfect\_tools.order.CLIENT\_ID = client.id;*

/\* Сумарний заробіток від роздрібних продаж \*/

*SELECT client.CLIENT\_TYPE as 'Clients type', Count(DISTINCT perfect\_tools.order.id) as 'Number of orders' , SUM(product.price \* order\_product.quantity) as 'Total retail sum'*

*FROM order\_product*

*JOIN perfect\_tools.order on order\_product.ORDER\_ID = perfect\_tools.order.id*

*JOIN product on order\_product.PRODUCT\_ID = product.id*

*JOIN client on perfect\_tools.order.CLIENT\_ID = client.id*

*WHERE client.CLIENT\_TYPE = 'retail';*

/\* Сумарний заробіток від оптових продаж \*/

*SELECT client.CLIENT\_TYPE as 'Clients type', Count(DISTINCT perfect\_tools.order.id) as 'Number of orders', SUM(product.price \* order\_product.quantity) as 'Total wholasale sum'*

*FROM order\_product*

*JOIN perfect\_tools.order on order\_product.ORDER\_ID = perfect\_tools.order.id*

*JOIN product on order\_product.PRODUCT\_ID = product.id*

*JOIN client on perfect\_tools.order.CLIENT\_ID = client.id*

*WHERE client.CLIENT\_TYPE = 'wholesale';*

/\* Сумарний заробіток за конкретний місяць \*/

*SELECT Count(DISTINCT perfect\_tools.order.id) as 'Number of orders', SUM(product.price \* order\_product.quantity) as 'Total month sum', monthname(perfect\_tools.order.created\_at) as 'Month', YEAR(perfect\_tools.order.created\_at) as 'Year'*

*FROM order\_product*

*JOIN perfect\_tools.order on order\_product.ORDER\_ID = perfect\_tools.order.id*

*JOIN product on order\_product.PRODUCT\_ID = product.id*

*JOIN client on perfect\_tools.order.CLIENT\_ID = client.id*

*WHERE perfect\_tools.order.created\_at BETWEEN '2010-04-01' AND '2010-04-30';*

/\* Аналітика по складах \*/

/\* Кількість складів \*/

*SELECT id as "ID", address as 'Warehouse address' FROM warehouse;*

/\* Кількість товарів на конкретному складі \*/

*SELECT warehouse.address as 'Warehouse address', product.name as 'Product Name', quantity*

*FROM warehouse\_product*

*JOIN product on warehouse\_product.PRODUCT\_ID = product.id*

*JOIN warehouse on warehouse\_product.WAREHOUSE\_ID = warehouse.id*

*WHERE warehouse\_product.WAREHOUSE\_ID = 1; /\* На першому складі (з двох) \*/*

/\* Товарів у грошах на конкретному складі \*/

*SELECT warehouse.address, SUM(product.price \* warehouse\_product.quantity) as 'Summary value of goods' FROM warehouse*

*JOIN warehouse\_product on warehouse\_product.WAREHOUSE\_ID = warehouse.id*

*JOIN product ON warehouse\_product.PRODUCT\_ID = product.id*

*WHERE warehouse\_product.WAREHOUSE\_ID = 1; /\* На першому складі (з двох) \*/*

/\* Аналітика по магазинах \*/

/\* Кількість магазинів \*/

*SELECT id as "ID", address as 'Store address' FROM store;*

/\* Кількість товарів у конкретному магазині \*/

*SELECT store.address as 'Store address', product.name as 'Product Name', quantity*

*FROM store\_product*

*JOIN product on PRODUCT\_ID = product.id*

*JOIN store on STORE\_ID = store.id*

*WHERE store\_product.STORE\_ID = 1; /\* В першому магазині (з чотирьох) \*/*

/\* Товарів у грошах в конкретному магазині \*/

*SELECT store.address, SUM(product.price \* store\_product.quantity) as 'Summary value of goods'*

*FROM store*

*JOIN store\_product on store\_product.STORE\_ID = store.id*

*JOIN product on store\_product.PRODUCT\_ID = product.id*

*WHERE store\_product.STORE\_ID = 1; /\* В першому магазині (з чотирьох) \*/*

/\* Аналітика по виробництві \*/

/\* Кількість фабрик \*/

*SELECT id as "ID", address as 'Factory address' FROM factory;*

/\* Всі записи по виробництву \*/

*SELECT production.id, factory.address as 'From factory', warehouse.address as 'To warehouse', product.name as 'Product name', production.quantity, production.produced\_at*

*FROM production*

*JOIN factory on production.FACTORY\_ID = factory.id*

*JOIN warehouse on production.WAREHOUSE\_ID = warehouse.id*

*JOIN product on production.PRODUCT\_ID = product.id;*

/\* Останній запис по виробництву \*/

*SELECT max(production.id), factory.address as 'From factory', warehouse.address as 'To warehouse', product.name as 'Product name', production.quantity, production.produced\_at*

*FROM production*

*JOIN factory on production.FACTORY\_ID = factory.id*

*JOIN warehouse on production.WAREHOUSE\_ID = warehouse.id*

*JOIN product on production.PRODUCT\_ID = product.id;*

/\* Запис по виробництву на конкретну дату \*/

*SELECT production.id, factory.address as 'From factory', warehouse.address as 'To warehouse', product.name as 'Product name', production.quantity, production.produced\_at*

*FROM production*

*JOIN factory on production.FACTORY\_ID = factory.id*

*JOIN warehouse on production.WAREHOUSE\_ID = warehouse.id*

*JOIN product on production.PRODUCT\_ID = product.id*

*WHERE produced\_at = '2010-01-12';*

/\* Запис по виробництву в грошах \*/

*SELECT production.id, factory.address as 'From factory', warehouse.address as 'To warehouse', product.name as 'Product name', production.quantity, product.price \* production.quantity as 'Total cost' ,production.produced\_at*

*FROM production*

*JOIN factory on production.FACTORY\_ID = factory.id*

*JOIN warehouse on production.WAREHOUSE\_ID = warehouse.id*

*JOIN product on production.PRODUCT\_ID = product.id;*

/\* Операції з асортиментом продуктів \*/

/\* Прегляд всього асортименту товарів \*/

*SELECT product.id as 'Id', product.name as 'Product name', product.price as 'Product price', product.description as 'Product description', product.status, product\_category.name as 'Category'*

*FROM product*

*JOIN product\_category on category\_id = product\_category.id;*

/\* Добавлення нового товару в асортимент \*/

*INSERT INTO product (name, price, description, status, category\_id) VALUES ('SUPER DRILL', 5000, 'Has a 15 amp motor that delivers 12,000 OPM', 'in\_stock', 1);*

/\* Видалення товару в з асортименту \*/

*DELETE FROM product*

*WHERE name = 'SUPER DRILL';*

/\* Аналітика по поставщиках \*/

/\* Прегляд поставщиків із закріпленими менеджерами \*/

*SELECT supplier.id AS 'Id', supplier.name as 'Supplier name', supplier.email as 'Email', supplier.phoneNumber as 'Phone number', supplier.MANAGER\_ID as 'Manager Id',CONCAT(employee.firstName, ' ', employee.surName) as 'Manager name', supplier.address as 'Supplier address', supplier\_contract.contract\_src as 'Contract', material.name as 'Supplied material'*

*FROM supplier*

*JOIN employee on supplier.MANAGER\_ID = employee.id*

*JOIN supplier\_contract on SUPPLIER\_ID = supplier.id*

*JOIN contract\_material on contract\_material.CONTRACT\_ID = supplier\_contract.id*

*JOIN material on contract\_material.MATERIAL\_ID = material.id;*

/\* FACTORY WORKER \*/

/\* Всі записи по виробництву \*/

*SELECT production.id, factory.address as 'From factory', warehouse.address as 'To warehouse', product.name as 'Product name', production.quantity, production.produced\_at*

*FROM production*

*JOIN factory on production.FACTORY\_ID = factory.id*

*JOIN warehouse on production.WAREHOUSE\_ID = warehouse.id*

*JOIN product on production.PRODUCT\_ID = product.id;*

/\* Створення нового запису про виробництво \*/

*INSERT INTO production (FACTORY\_ID, WAREHOUSE\_ID, PRODUCT\_ID, quantity, produced\_at) VALUES (1, 1, 6, 5, '2010-01-14');*

/\* Редагування запису про виробництво \*/

*UPDATE production*

*SET WAREHOUSE\_ID = 2*

*WHERE production.id = 7;*

/\* WAREHOUSE WORKER \*/

/\* Операції з товарами \*/

/\* Кількість товарів на конкретному складі \*/

*SELECT warehouse.address as 'Warehouse address', product.id, product.name as 'Product Name', quantity*

*FROM warehouse\_product*

*JOIN product on warehouse\_product.PRODUCT\_ID = product.id*

*JOIN warehouse on warehouse\_product.WAREHOUSE\_ID = warehouse.id*

*WHERE warehouse\_product.WAREHOUSE\_ID = 1; /\* На першому складі (з двох) \*/*

/\* Перегляд записів про переміщення товарів в магазин \*/

*SELECT warehouse\_to\_store.id as 'Id', warehouse\_to\_store.WAREHOUSE\_ID as 'Warehouse Id', warehouse.address as 'Warehouse address' , warehouse\_to\_store.STORE\_ID as 'Store Id', store.address as 'Store address', warehouse\_to\_store.PRODUCT\_ID as 'Product Id', product.name as 'Product name', warehouse\_to\_store.quantity, warehouse\_to\_store.supplied\_at as 'Supplied at'*

*FROM warehouse\_to\_store*

*JOIN warehouse on warehouse\_to\_store.WAREHOUSE\_ID = warehouse.id*

*JOIN store on warehouse\_to\_store.STORE\_ID = store.id*

*JOIN product on warehouse\_to\_store.PRODUCT\_ID = product.id;*

/\* Оприбуткування товарів на склад \*/

*INSERT INTO warehouse\_product (WAREHOUSE\_ID, PRODUCT\_ID, quantity) VALUES (1, 6, 5); /\* Новий товар \*/*

*UPDATE warehouse\_product*

*SET quantity = quantity + 3*

*WHERE WAREHOUSE\_ID = 1 AND PRODUCT\_ID = 2;/\* Старий товар \*/*

/\* Запис про переміщення товарів зі складу в магазин \*/

*UPDATE warehouse\_product*

*SET quantity = quantity - 3*

*WHERE WAREHOUSE\_ID = 1 AND PRODUCT\_ID = 2;*

*INSERT INTO warehouse\_to\_store (WAREHOUSE\_ID, STORE\_ID, PRODUCT\_ID, quantity, supplied\_at) VALUES (1, 1, 2, 3, '2010-02-13');*

/\* Списання товару згідно замовлення (Id = 9) \*/

*UPDATE warehouse\_product*

*SET quantity = quantity - 2*

*WHERE WAREHOUSE\_ID = 1 AND PRODUCT\_ID = 1;*

/\* Операції з матеріалами \*/

/\* Кількість матеріалів на конкретному складі \*/

*SELECT warehouse.address as 'Warehouse address', material.id, material.name as 'Material Name', quantity*

*FROM warehouse\_material*

*JOIN material on MATERIAL\_ID = material.id*

*JOIN warehouse on WAREHOUSE\_ID = warehouse.id*

*WHERE WAREHOUSE\_ID = 1;/\* На першому складі (з двох) \*/*

/\* Перегляд записів про переміщення матеріалів зі складу на фабрику \*/

*SELECT material\_to\_factory.id as 'Id', material\_to\_factory.WAREHOUSE\_ID as 'Warehouse Id', warehouse.address as 'Warehouse address', material\_to\_factory.FACTORY\_ID as 'Factory Id', factory.address as 'Factory address', material\_to\_factory.MATERIAL\_ID as 'Material Id', material.name as 'Material name', material\_to\_factory.quantity, material\_to\_factory.supplied\_at as 'Supplied at'*

*FROM material\_to\_factory*

*JOIN warehouse on material\_to\_factory.WAREHOUSE\_ID = warehouse.id*

*JOIN factory ON material\_to\_factory.FACTORY\_ID = factory.id*

*JOIN material on material\_to\_factory.MATERIAL\_ID = material.id*

*ORDER BY material\_to\_factory.supplied\_at;*

/\* Оприбуткування матеріалів на склад \*/

*INSERT INTO warehouse\_material (WAREHOUSE\_ID, MATERIAL\_ID, quantity) VALUES (1, 1, 20); /\* Новий матеріал \*/*

*UPDATE warehouse\_material*

*SET quantity = quantity + 10*

*WHERE WAREHOUSE\_ID = 1 AND MATERIAL\_ID = 3; /\* Старий матеріал \*/*

/\* Запис про переміщення матеріалів зі складу на фабрику \*/

*UPDATE warehouse\_material*

*SET quantity = quantity - 10*

*WHERE WAREHOUSE\_ID = 1 AND MATERIAL\_ID = 3; /\* Зняття зі складу \*/*

*INSERT INTO material\_to\_factory (WAREHOUSE\_ID, FACTORY\_ID, MATERIAL\_ID, quantity, supplied\_at) VALUES (1, 1, 3, 10, '2010-02-08');*

/\* STORE WORKER \*/

/\* Кількість товарів у конкретному магазині \*/

*SELECT store.address as 'Store address', store\_product.PRODUCT\_ID as 'Product id' , product.name as 'Product Name', quantity*

*FROM store\_product*

*JOIN product on PRODUCT\_ID = product.id*

*JOIN store on STORE\_ID = store.id*

*WHERE store\_product.STORE\_ID = 1; /\* В першому магазині (з чотирьох) \*/*

/\* Замовлення виконані в конкретному магазині \*/

*SELECT perfect\_tools.order.id as 'Id', CONCAT(client.firstName, ' ' ,client.surName) as 'Client name', store.address as 'Store address', product.name as 'Product name', order\_product.quantity , product.price \* order\_product.quantity as 'Total sum',perfect\_tools.order.created\_at as 'Created at'*

*FROM perfect\_tools.order*

*JOIN client on perfect\_tools.order.CLIENT\_ID = client.id*

*JOIN store on perfect\_tools.order.STORE\_ID = store.id*

*JOIN order\_product on perfect\_tools.order.id = order\_product.ORDER\_ID*

*JOIN product on order\_product.PRODUCT\_ID = product.id*

*WHERE perfect\_tools.order.STORE\_ID = 1; /\* В першому магазині (з чотирьох) \*/*

/\* Оприбуткування товару в магазин \*/

*UPDATE store\_product*

*SET quantity = quantity + 3*

*WHERE STORE\_ID = 1 AND PRODUCT\_ID = 2;*

/\* Реєстрація нового замовлення \*/

*INSERT INTO CLIENT (firstName, surName, phoneNumber, email, address, CLIENT\_TYPE, created\_at) VALUES ('Alex','Saint', 0234867894, 'alex.saint@mail.com', 'Lviv, Vyhovsky Str', 'retail', '2010-04-17');*

*INSERT INTO perfect\_tools.order (CLIENT\_ID, status, STORE\_ID, created\_at) VALUES (11, 'order\_completed', 1 , '2010-04-17' );*

*INSERT INTO order\_product (ORDER\_ID, PRODUCT\_ID, quantity) VALUES ( 8, 2, 3 );*

/\* Списання товару згідно родрібного замовлення (Id = 8) \*/

*UPDATE store\_product*

*SET quantity = quantity - 3*

*WHERE STORE\_ID = 1 AND PRODUCT\_ID = 2;*

/\* SUPPLY MANAGER \*/

/\* Прегляд поставщиків із закріпленими менеджерами \*/

*SELECT supplier.id AS 'Id', supplier.name as 'Supplier name', supplier.email as 'Email', supplier.phoneNumber as 'Phone number', supplier.MANAGER\_ID as 'Manager Id',CONCAT(employee.firstName, ' ', employee.surName) as 'Manager name', supplier.address as 'Supplier address', supplier\_contract.contract\_src as 'Contract', material.name as 'Supplied material'*

*FROM supplier*

*JOIN employee on supplier.MANAGER\_ID = employee.id*

*JOIN supplier\_contract on SUPPLIER\_ID = supplier.id*

*JOIN contract\_material on contract\_material.CONTRACT\_ID = supplier\_contract.id*

*JOIN material on contract\_material.MATERIAL\_ID = material.id;*

/\* Перегляд поставлених товарів по контрактах \*/

*SELECT supplier.id AS 'Id', supplier.name as 'Supplier name', supplier\_contract.contract\_src as 'Contract', material.name as 'Supplied material', contract\_material.quantity as 'Material quantity'*

*FROM supplier*

*JOIN employee on supplier.MANAGER\_ID = employee.id*

*JOIN supplier\_contract on SUPPLIER\_ID = supplier.id*

*JOIN contract\_material on contract\_material.CONTRACT\_ID = supplier\_contract.id*

*JOIN material on contract\_material.MATERIAL\_ID = material.id;*

/\* Створення нового поставщика \*/

*INSERT INTO supplier (name, email, phoneNumber, MANAGER\_ID, address) VALUES ('SP "CORGY"', 'corgy.company@mail.com', 0984712893, 15, 'US, NY-Brooklyn');*

/\* Створення нового контракту \*/

*INSERT INTO supplier\_contract (SUPPLIER\_ID, contract\_src) VALUES (4, 'CORGY\_contract.pdf'); /\* На основі попреднього INSERT \*/*

*INSERT INTO contract\_material (CONTRACT\_ID, MATERIAL\_ID, quantity) VALUES (4, 4, 100);*

/\* SALES MANAGER \*/

/\* Всі оптові замовлення \*/

*SELECT client.firstName as 'Client name', client.address as 'Client address', perfect\_tools.ORDER.id as 'Order Id', perfect\_tools.order.status, perfect\_tools.order.created\_at, product.name as 'Product name', order\_product.quantity, order\_product.quantity \* product.price as 'Total sum'*

*FROM perfect\_tools.order*

*join client on CLIENT\_ID = client.id*

*join order\_product on ORDER\_ID = perfect\_tools.order.id*

*join product on product.id = order\_product.PRODUCT\_ID*

*WHERE client.CLIENT\_TYPE = 'wholesale';*

/\* Сумарний заробіток від оптового клієнта \*/

*SELECT CONCAT(client.firstName, ' ', client.surName) as 'Client name' , client.CLIENT\_TYPE as 'Clients type', Count(DISTINCT perfect\_tools.order.id) as 'Number of orders', SUM(product.price \* order\_product.quantity) as 'Total wholasale sum'*

*FROM order\_product*

*JOIN perfect\_tools.order on order\_product.ORDER\_ID = perfect\_tools.order.id*

*JOIN product on order\_product.PRODUCT\_ID = product.id*

*JOIN client on perfect\_tools.order.CLIENT\_ID = client.id*

*WHERE client.CLIENT\_TYPE = 'wholesale' AND client.id = 6;*

/\* Реєстрація нового клієнта \*/

*INSERT INTO CLIENT (firstName, surName, phoneNumber, email, address, CLIENT\_TYPE, created\_at) VALUES ('Fernando','Torres', 0234867894, 'fernando.torres@mail.com', 'Lviv, Antonenka Str', 'wholesale', '2010-04-18');*

/\* Реєстрація нового замовлення \*/

*INSERT INTO perfect\_tools.order (CLIENT\_ID, status, WAREHOUSE\_ID, created\_at) VALUES (12, 'order\_accepted', 1 , '2010-04-18' );*

*INSERT INTO order\_product (ORDER\_ID, PRODUCT\_ID, quantity) VALUES ( 9, 1, 2 );*

**ВИСНОВОК**

Під час виконання даної курсової роботи було проведене дослідження теоретичних та практичних аспектів використання мови SQL для розробки баз даних. В процесі виконання курсової роботи було проаналізовано предметну область завдання, а також спроектовано, розроблено та реалізовано інформаційну систему підприємства, що виготовляє електротехніку. Було розроблено абстрактну модель підприємства, в якій було описано його організаційну структуру та особливості. Основною метою цього етапу було визначити масштаб проекту.

На етапі проведення бізнес-аналізу предметної області було розроблено так звану Use Case діаграму (діаграма прецедентів), що вичерпно відображає способи використання інформаційної системи користувачами.

Фінальним етапом виконання курсової роботи було проектування, розробка та реалізація бази даних. Під час цього етапу було створено Entity-Relationship діаграму, що відображає структуру бази даних та зв’язки між її таблицями, а також було реалізовано SQL-скрипти, що відповідають вимогам проекту.

**СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ**

1. MySQL Documentation URL: <https://dev.mysql.com/doc/>
2. SQL Tutorial URL: <https://www.w3schools.com/sql/default.asp>
3. UML Use Case Diagram Tutorial URL: <https://www.lucidchart.com/pages/uml-use-case-diagram>
4. What is Use Case Diagram? URL: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/>
5. What is an Entity Relationship Diagram (ERD)? URL: <https://www.lucidchart.com/pages/er-diagrams>
6. Моделі баз даних URL: <https://uk.wikipedia.org/wiki/Моделі_баз_даних>
7. Business analysis URL: <https://en.wikipedia.org/wiki/Business_analysis>
8. MySQL URL: <https://uk.wikipedia.org/wiki/MySQL>
9. Що таке MySQL-сервер URL: <https://timeweb.com/ru/community/articles/chto-takoe-mysql-server>
10. ІНФОРМАЦІЙНІ СИСТЕМИ ТА ТЕХНОЛОГІЇ НА ПІДПРИЄМСТВІ URL: <http://www.repository.hneu.edu.ua/bitstream/123456789/3112/1/%D0%9A%D0%BE%D0%BD%D1%81%D0%BF%D0%B5%D0%BA%D1%82%20%D0%BB%D0%B5%D0%BA%D1%86%D1%96%D0%B9.%20%D0%86%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%86%D1%96%D0%B9%D0%BD%D1%96%20%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B8%20%D1%82%D0%B0%20%D1%82%D0%B5%D1%85%D0%BD%D0%BE%D0%BB%D0%BE%D0%B3%D1%96%D1%97%20%D0%BD%D0%B0%20%D0%BF%D1%96%D0%B4%D0%BF%D1%80%D0%B8%D1%94%D0%BC%D1%81%D1%82%D0%B2%D1%96.%20%D0%86.%20%D0%9E.%20%D0%A3%D1%88%D0%B0%D0%BA%D0%BE%D0%B2%D0%B0..pdf>

**ДОДАТКИ**

Додаток А

Лістинг коду створення табличок бази даних А.1:

CREATE TABLE `ORDER` (

`id` int PRIMARY KEY NOT NULL AUTO\_INCREMENT,

`CLIENT\_ID` int,

`status` ENUM ('order\_accepted', 'order\_in\_process', 'order\_completed', 'order\_cancelled'),

`MANAGER\_ID` int,

`STORE\_ID` int,

`WAREHOUSE\_ID` int,

`created\_at` timestamp

);

CREATE TABLE `STORE` (

`id` int PRIMARY KEY NOT NULL AUTO\_INCREMENT,

`address` varchar(255)

);

CREATE TABLE `PRODUCT` (

`id` int PRIMARY KEY NOT NULL AUTO\_INCREMENT,

`name` varchar(255),

`price` int,

`description` varchar(255),

`status` ENUM ('out\_of\_stock', 'in\_stock', 'running\_low'),

`category\_id` int

);

CREATE TABLE `PRODUCT\_IMAGE` (

`id` int PRIMARY KEY NOT NULL AUTO\_INCREMENT,

`PRODUCT\_ID` int,

`image\_src` varchar(255)

);

CREATE TABLE `PRODUCT\_CATEGORY` (

`id` int PRIMARY KEY NOT NULL AUTO\_INCREMENT,

`name` varchar(255)

);

CREATE TABLE `CLIENT` (

`id` int PRIMARY KEY NOT NULL AUTO\_INCREMENT,

`firstName` varchar(20),

`surName` varchar(20),

`phoneNumber` int,

`email` varchar(255),

`address` varchar(255),

`CLIENT\_TYPE` ENUM ('retail', 'wholesale'),

`created\_at` date

);

CREATE TABLE `EMPLOYEE` (

`id` int PRIMARY KEY NOT NULL AUTO\_INCREMENT,

`firstName` varchar(20),

`surName` varchar(20),

`login` varchar(15),

`password` varchar(8),

`salary` int,

`POSITION\_ID` int,

`created\_at` date,

`Emp\_Factory` int,

`Emp\_Store` int,

`Emp\_Warehouse` int

);

CREATE TABLE `POSITION` (

`id` int PRIMARY KEY NOT NULL AUTO\_INCREMENT,

`name` varchar(255)

);

CREATE TABLE `SUPPLIER` (

`id` int PRIMARY KEY NOT NULL AUTO\_INCREMENT,

`name` varchar(40),

`email` varchar(255),

`phoneNumber` int,

`MANAGER\_ID` int,

`address` varchar(255)

);

CREATE TABLE `MATERIAL` (

`id` int PRIMARY KEY NOT NULL AUTO\_INCREMENT,

`name` varchar(255)

);

CREATE TABLE `SUPPLIER\_CONTRACT` (

`id` int PRIMARY KEY NOT NULL AUTO\_INCREMENT,

`SUPPLIER\_ID` int,

`contract\_src` varchar(255)

);

CREATE TABLE `WAREHOUSE` (

`id` int PRIMARY KEY NOT NULL AUTO\_INCREMENT,

`address` varchar(255)

);

CREATE TABLE `FACTORY` (

`id` int PRIMARY KEY NOT NULL AUTO\_INCREMENT,

`address` varchar(255)

);

CREATE TABLE `ORDER\_PRODUCT` (

`id` int PRIMARY KEY NOT NULL AUTO\_INCREMENT,

`ORDER\_ID` int,

`PRODUCT\_ID` int,

`quantity` int

);

CREATE TABLE `STORE\_PRODUCT` (

`id` int PRIMARY KEY NOT NULL AUTO\_INCREMENT,

`STORE\_ID` int,

`PRODUCT\_ID` int,

`quantity` int

);

CREATE TABLE `CONTRACT\_MATERIAL` (

`id` int PRIMARY KEY NOT NULL AUTO\_INCREMENT,

`CONTRACT\_ID` int,

`MATERIAL\_ID` int,

`quantity` int

);

CREATE TABLE `WAREHOUSE\_MATERIAL` (

`id` int PRIMARY KEY NOT NULL AUTO\_INCREMENT,

`WAREHOUSE\_ID` int,

`MATERIAL\_ID` int,

`quantity` int

);

CREATE TABLE `WAREHOUSE\_PRODUCT` (

`id` int PRIMARY KEY NOT NULL AUTO\_INCREMENT,

`WAREHOUSE\_ID` int,

`PRODUCT\_ID` int,

`quantity` int

);

CREATE TABLE `MATERIAL\_TO\_FACTORY` (

`id` int PRIMARY KEY NOT NULL AUTO\_INCREMENT,

`WAREHOUSE\_ID` int,

`FACTORY\_ID` int,

`MATERIAL\_ID` int,

`quantity` int,

`supplied\_at` date

);

CREATE TABLE `WAREHOUSE\_TO\_STORE` (

`id` int PRIMARY KEY NOT NULL AUTO\_INCREMENT,

`WAREHOUSE\_ID` int,

`STORE\_ID` int,

`PRODUCT\_ID` int,

`quantity` int,

`supplied\_at` date

);

CREATE TABLE `PRODUCTION` (

`id` int PRIMARY KEY NOT NULL AUTO\_INCREMENT,

`FACTORY\_ID` int,

`WAREHOUSE\_ID` int,

`PRODUCT\_ID` int,

`quantity` int,

`produced\_at` date

);

ALTER TABLE `ORDER` ADD FOREIGN KEY (`CLIENT\_ID`) REFERENCES `CLIENT` (`id`);

ALTER TABLE `ORDER` ADD FOREIGN KEY (`MANAGER\_ID`) REFERENCES `EMPLOYEE` (`id`);

ALTER TABLE `ORDER` ADD FOREIGN KEY (`STORE\_ID`) REFERENCES `STORE` (`id`);

ALTER TABLE `ORDER` ADD FOREIGN KEY (`WAREHOUSE\_ID`) REFERENCES `WAREHOUSE` (`id`);

ALTER TABLE `PRODUCT` ADD FOREIGN KEY (`category\_id`) REFERENCES `PRODUCT\_CATEGORY` (`id`);

ALTER TABLE `PRODUCT\_IMAGE` ADD FOREIGN KEY (`PRODUCT\_ID`) REFERENCES `PRODUCT` (`id`);

ALTER TABLE `EMPLOYEE` ADD FOREIGN KEY (`POSITION\_ID`) REFERENCES `POSITION` (`id`);

ALTER TABLE `EMPLOYEE` ADD FOREIGN KEY (`Emp\_Factory`) REFERENCES `FACTORY` (`id`);

ALTER TABLE `EMPLOYEE` ADD FOREIGN KEY (`Emp\_Store`) REFERENCES `STORE` (`id`);

ALTER TABLE `EMPLOYEE` ADD FOREIGN KEY (`Emp\_Warehouse`) REFERENCES `WAREHOUSE` (`id`);

ALTER TABLE `SUPPLIER` ADD FOREIGN KEY (`MANAGER\_ID`) REFERENCES `EMPLOYEE` (`id`);

ALTER TABLE `SUPPLIER\_CONTRACT` ADD FOREIGN KEY (`SUPPLIER\_ID`) REFERENCES `SUPPLIER` (`id`);

ALTER TABLE `ORDER\_PRODUCT` ADD FOREIGN KEY (`ORDER\_ID`) REFERENCES `ORDER` (`id`);

ALTER TABLE `ORDER\_PRODUCT` ADD FOREIGN KEY (`PRODUCT\_ID`) REFERENCES `PRODUCT` (`id`);

ALTER TABLE `STORE\_PRODUCT` ADD FOREIGN KEY (`STORE\_ID`) REFERENCES `STORE` (`id`);

ALTER TABLE `STORE\_PRODUCT` ADD FOREIGN KEY (`PRODUCT\_ID`) REFERENCES `PRODUCT` (`id`);

ALTER TABLE `CONTRACT\_MATERIAL` ADD FOREIGN KEY (`CONTRACT\_ID`) REFERENCES `SUPPLIER\_CONTRACT` (`id`);

ALTER TABLE `CONTRACT\_MATERIAL` ADD FOREIGN KEY (`MATERIAL\_ID`) REFERENCES `MATERIAL` (`id`);

ALTER TABLE `WAREHOUSE\_MATERIAL` ADD FOREIGN KEY (`WAREHOUSE\_ID`) REFERENCES `WAREHOUSE` (`id`);

ALTER TABLE `WAREHOUSE\_MATERIAL` ADD FOREIGN KEY (`MATERIAL\_ID`) REFERENCES `MATERIAL` (`id`);

ALTER TABLE `WAREHOUSE\_PRODUCT` ADD FOREIGN KEY (`WAREHOUSE\_ID`) REFERENCES `WAREHOUSE` (`id`);

ALTER TABLE `WAREHOUSE\_PRODUCT` ADD FOREIGN KEY (`PRODUCT\_ID`) REFERENCES `PRODUCT` (`id`);

ALTER TABLE `MATERIAL\_TO\_FACTORY` ADD FOREIGN KEY (`WAREHOUSE\_ID`) REFERENCES `WAREHOUSE` (`id`);

ALTER TABLE `MATERIAL\_TO\_FACTORY` ADD FOREIGN KEY (`FACTORY\_ID`) REFERENCES `FACTORY` (`id`);

ALTER TABLE `MATERIAL\_TO\_FACTORY` ADD FOREIGN KEY (`MATERIAL\_ID`) REFERENCES `MATERIAL` (`id`);

ALTER TABLE `WAREHOUSE\_TO\_STORE` ADD FOREIGN KEY (`WAREHOUSE\_ID`) REFERENCES `WAREHOUSE` (`id`);

ALTER TABLE `WAREHOUSE\_TO\_STORE` ADD FOREIGN KEY (`STORE\_ID`) REFERENCES `STORE` (`id`);

ALTER TABLE `WAREHOUSE\_TO\_STORE` ADD FOREIGN KEY (`PRODUCT\_ID`) REFERENCES `MATERIAL` (`id`);

ALTER TABLE `PRODUCTION` ADD FOREIGN KEY (`FACTORY\_ID`) REFERENCES `FACTORY` (`id`);

ALTER TABLE `PRODUCTION` ADD FOREIGN KEY (`WAREHOUSE\_ID`) REFERENCES `WAREHOUSE` (`id`);

ALTER TABLE `PRODUCTION` ADD FOREIGN KEY (`PRODUCT\_ID`) REFERENCES `PRODUCT` (`id`);

ALTER TABLE `perfect\_tools`.`contract\_material`

CHANGE COLUMN `quantity` `quantity` INT UNSIGNED NULL DEFAULT NULL ;

ALTER TABLE `perfect\_tools`.`material\_to\_factory`

CHANGE COLUMN `quantity` `quantity` INT UNSIGNED NULL DEFAULT NULL ;

ALTER TABLE `perfect\_tools`.`order\_product`

CHANGE COLUMN `quantity` `quantity` INT UNSIGNED NULL DEFAULT NULL ;

ALTER TABLE `perfect\_tools`.`production`

CHANGE COLUMN `quantity` `quantity` INT UNSIGNED NULL DEFAULT NULL ;

ALTER TABLE `perfect\_tools`.`store\_product`

CHANGE COLUMN `quantity` `quantity` INT UNSIGNED NULL DEFAULT NULL ;

ALTER TABLE `perfect\_tools`.`warehouse\_material`

CHANGE COLUMN `quantity` `quantity` INT UNSIGNED NULL DEFAULT NULL ;

ALTER TABLE `perfect\_tools`.`warehouse\_product`

CHANGE COLUMN `quantity` `quantity` INT UNSIGNED NULL DEFAULT NULL ;

ALTER TABLE `perfect\_tools`.`warehouse\_to\_store`

CHANGE COLUMN `quantity` `quantity` INT UNSIGNED NULL DEFAULT NULL ;

Лістинг коду заповнення бази даних А.2:

/\* CLIENT INSERT \*/

INSERT INTO CLIENT (firstName, surName, phoneNumber, email, address, CLIENT\_TYPE, created\_at) VALUES ('Andriy','Vasko', 0785604023, 'andriy.vasko@mail.com', 'Lviv, Stepan Bandera Str', 'retail', '2010-04-01');

INSERT INTO CLIENT (firstName, surName, phoneNumber, email, address, CLIENT\_TYPE, created\_at) VALUES ('Oleksiy','Vyatrovych', 0986541223, 'oleksiy.vyatrovych@mail.com', 'Kyiv, Shuknevych Str', 'retail', '2010-04-02');

INSERT INTO CLIENT (firstName, surName, phoneNumber, email, address, CLIENT\_TYPE, created\_at) VALUES ('Olexandr','Peperoni', 0834873296, 'olexandr.peperoni@mail.com', 'Lviv, Vasyl Stus Str', 'retail', '2010-04-02');

INSERT INTO CLIENT (firstName, surName, phoneNumber, email, address, CLIENT\_TYPE, created\_at) VALUES ('Serhiy','Prutyla', 0327894512, 'serhiy.prytula@mail.com', 'Lviv, Ivan Mazepa Str', 'retail', '2010-04-04');

INSERT INTO CLIENT (firstName, surName, phoneNumber, email, address, CLIENT\_TYPE, created\_at) VALUES ('Maria','Stadnyk', 0784589632, 'maria.stadnyk@mail.com', 'Lviv, Sagaidachnyi Str', 'retail', '2010-04-05');

INSERT INTO CLIENT (firstName, surName, phoneNumber, email, address, CLIENT\_TYPE, created\_at) VALUES ('Olesya','Mint', 0845678915, 'olesya.mint@mail.com', 'Zhytomyr, Chornobaivka Str', 'wholesale', '2010-04-10');

INSERT INTO CLIENT (firstName, surName, phoneNumber, email, address, CLIENT\_TYPE, created\_at) VALUES ('Semen','Semenuk', 0651238579, 'semen.semenyk@mail.com', 'Chornobaivka, Zhytomyrska Str', 'wholesale', '2010-04-10');

INSERT INTO CLIENT (firstName, surName, phoneNumber, email, address, CLIENT\_TYPE, created\_at) VALUES ('Vladyslav','Sytnyk', 0847891637, 'vladyslav.sytnyk@mail.com', 'Lviv, Syhivska Str', 'wholesale', '2010-04-11');

INSERT INTO CLIENT (firstName, surName, phoneNumber, email, address, CLIENT\_TYPE, created\_at) VALUES ('Anatoliy','Donych', 0986547328, 'anatoliy.donych@mail.com', 'Zhytomyr, Lvyvska Str', 'wholesale', '2010-04-15');

INSERT INTO CLIENT (firstName, surName, phoneNumber, email, address, CLIENT\_TYPE, created\_at) VALUES ('Tetyana','Ivanuk', 0456878945, 'tetyana.ivanuk@mail.com', 'Rivne, Kyivska Str', 'wholesale', '2010-04-16');

/\* STORE INSERT \*/

INSERT INTO STORE (address) VALUES ('Lviv, Hmelnytskya Str');

INSERT INTO STORE (address) VALUES ('Kyiv, Peremoha Square');

INSERT INTO STORE (address) VALUES ('Rivne, Chornobaivska Str');

INSERT INTO STORE (address) VALUES ('Zhytomyr, Lvivska Str');

/\* WAREHOUSE INSERT \*/

INSERT INTO WAREHOUSE (address) VALUES ('Lviv, Zelena Str');

INSERT INTO WAREHOUSE (address) VALUES ('Kyiv, Promyslova Str');

/\* FACTORY INSERT \*/

INSERT INTO factory (address) VALUES ('Lviv, Zubra Str');

INSERT INTO factory (address) VALUES ('Rivne, Shevchenka Str');

/\* MATERIAL INSERT \*/

INSERT INTO material (name) VALUES ('Engine\_18v');

INSERT INTO material (name) VALUES ('Engine\_24v');

INSERT INTO material (name) VALUES ('Screws');

INSERT INTO material (name) VALUES ('Plactic parts');

INSERT INTO material (name) VALUES ('Electric parts');

/\* PRODUCT\_CATEGORY INSERT \*/

INSERT INTO product\_category (name) VALUES ('Drills');

INSERT INTO product\_category (name) VALUES ('Grinders & Polishers');

INSERT INTO product\_category (name) VALUES ('Compressors');

INSERT INTO product\_category (name) VALUES ('Impact Drivers');

INSERT INTO product\_category (name) VALUES ('Saws');

INSERT INTO product\_category (name) VALUES ('Nailers');

INSERT INTO product\_category (name) VALUES ('Sanders');

/\* PRODUCT INSERT \*/

INSERT INTO product (name, price, description, status, category\_id) VALUES ('24V Brushless Cordless 1/2in. Hammer Drill', 3600, 'Drill/Driver delivers the durability and performance professionals need on jobsite', 'in\_stock', 1);

INSERT INTO product (name, price, description, status, category\_id) VALUES ('18V Brushless Cordless 1/2in. Drill', 3200, 'Drill delivers the durability and performance professionals need on jobsite', 'in\_stock', 1);

INSERT INTO product (name, price, description, status, category\_id) VALUES ('60V MAX\* 7 in. Brushless Cordless Grinder with Kickback Brake Kit', 5400, 'Complete metal and concrete cutting projects', 'in\_stock', 2);

INSERT INTO product (name, price, description, status, category\_id) VALUES ('20 gal. 200 psi Oil Lubed Belt Drive Portable Horizontal Electric Air Compressor', 10000, 'Belt drive air compressor is the ideal choice for tackling a variety of tasks', 'in\_stock', 3);

INSERT INTO product (name, price, description, status, category\_id) VALUES ('Atomic 20V MAX\* 1/4 in. Brushless Cordless 3-Speed Impact Driver Kit', 6300, 'Impact Driver can handle tight applications like narrow cabinetry', 'in\_stock', 4);

INSERT INTO product (name, price, description, status, category\_id) VALUES ('XTREME™ 12V MAX\* 5-3/8 in. Brushless Cordless Circular Saw Kit', 4700, 'Get the power and cutting depth of corded with the portability', 'in\_stock', 5);

INSERT INTO product (name, price, description, status, category\_id) VALUES ('20V MAX\* XR® Cordless Metal Connecting Nailer Kit', 3000, ' Nailer allows for installation of joist hangers', 'out\_of\_stock', 6);

INSERT INTO product (name, price, description, status, category\_id) VALUES ('5" Single Speed Random Orbit Sander - PSA Pad', 4000, 'Has a 3 amp motor that delivers 12,000 OPM', 'out\_of\_stock', 7);

/\* PRODUCT\_IMAGE INSERT \*/

INSERT INTO product\_image (PRODUCT\_ID, image\_src) VALUES (1, '24V Brushless Cordless 1/2in. Hammer Drill\_1.png');

INSERT INTO product\_image (PRODUCT\_ID, image\_src) VALUES (1, '24V Brushless Cordless 1/2in. Hammer Drill\_2.png');

INSERT INTO product\_image (PRODUCT\_ID, image\_src) VALUES (1, '24V Brushless Cordless 1/2in. Hammer Drill\_3.png');

INSERT INTO product\_image (PRODUCT\_ID, image\_src) VALUES (2, '18V Brushless Cordless 1/2in. Hammer Drill\_1.png');

INSERT INTO product\_image (PRODUCT\_ID, image\_src) VALUES (2, '18V Brushless Cordless 1/2in. Hammer Drill\_2.png');

INSERT INTO product\_image (PRODUCT\_ID, image\_src) VALUES (3, '60V MAX\* 7 in. Brushless Cordless Grinder with Kickback Brake Kit\_1.png');

INSERT INTO product\_image (PRODUCT\_ID, image\_src) VALUES (4, '20 gal. 200 psi Oil Lubed Belt Drive Portable Horizontal Electric Air Compressor\_1.png');

INSERT INTO product\_image (PRODUCT\_ID, image\_src) VALUES (4, '20 gal. 200 psi Oil Lubed Belt Drive Portable Horizontal Electric Air Compressor\_2.png');

INSERT INTO product\_image (PRODUCT\_ID, image\_src) VALUES (5, 'Atomic 20V MAX\* 1/4 in. Brushless Cordless 3-Speed Impact Driver Kit\_1.png');

INSERT INTO product\_image (PRODUCT\_ID, image\_src) VALUES (6, 'XTREME™ 12V MAX\* 5-3/8 in. Brushless Cordless Circular Saw Kit\_1.png');

INSERT INTO product\_image (PRODUCT\_ID, image\_src) VALUES (6, 'XTREME™ 12V MAX\* 5-3/8 in. Brushless Cordless Circular Saw Kit\_2.png');

INSERT INTO product\_image (PRODUCT\_ID, image\_src) VALUES (6, 'XTREME™ 12V MAX\* 5-3/8 in. Brushless Cordless Circular Saw Kit\_3.png');

INSERT INTO product\_image (PRODUCT\_ID, image\_src) VALUES (7, '20V MAX\* XR® Cordless Metal Connecting Nailer Kit\_1.png');

INSERT INTO product\_image (PRODUCT\_ID, image\_src) VALUES (7, '20V MAX\* XR® Cordless Metal Connecting Nailer Kit\_2.png');

INSERT INTO product\_image (PRODUCT\_ID, image\_src) VALUES (8, '5" Single Speed Random Orbit Sander - PSA Pad\_1.png');

/\* POSITION INSERT\*/

INSERT INTO position (name) VALUES ('Sales Manager');

INSERT INTO position (name) VALUES ('Supply Manager');

INSERT INTO position (name) VALUES ('Factory Worker');

INSERT INTO position (name) VALUES ('Warehouse Worker');

INSERT INTO position (name) VALUES ('Store Worker');

/\* EMPLOYEE INSERT \*/

INSERT INTO employee (firstName, surName, login, password, salary, POSITION\_ID, created\_at, Emp\_Factory) VALUES ('Finlay', 'Bass', 'FinlayBass', 'Finlay', 15000, 3, '2010-01-01', 1);

INSERT INTO employee (firstName, surName, login, password, salary, POSITION\_ID, created\_at, Emp\_Factory) VALUES ('Abiha', 'Beach', 'AbihaBeach', 'Abiha', 18000, 3, '2010-01-02', 1);

INSERT INTO employee (firstName, surName, login, password, salary, POSITION\_ID, created\_at, Emp\_Factory) VALUES ('Nate', 'Buckner', 'NateBuckner', 'Nate', 17000, 3, '2010-01-02', 1);

INSERT INTO employee (firstName, surName, login, password, salary, POSITION\_ID, created\_at, Emp\_Factory) VALUES ('Nur', 'Barber', 'NurBarber', 'Nur', 14000, 3, '2010-01-01', 2);

INSERT INTO employee (firstName, surName, login, password, salary, POSITION\_ID, created\_at, Emp\_Factory) VALUES ('Alysia', 'Richards', 'AlysiaRichards', 'Alysia', 15500, 3, '2010-01-03', 2);

INSERT INTO employee (firstName, surName, login, password, salary, POSITION\_ID, created\_at, Emp\_Warehouse) VALUES ('Giles', 'Miles', 'GilesMiles', 'Giles', 12000, 4, '2010-01-03', 1);

INSERT INTO employee (firstName, surName, login, password, salary, POSITION\_ID, created\_at, Emp\_Warehouse) VALUES ('Marie ', 'Mullen', 'MarieMullen', 'Marie', 12500, 4, '2010-01-03', 1);

INSERT INTO employee (firstName, surName, login, password, salary, POSITION\_ID, created\_at, Emp\_Warehouse) VALUES ('Zayaan', 'Gregory', 'ZayaanGregory', 'Zayaan', 13000, 4, '2010-01-04', 2);

INSERT INTO employee (firstName, surName, login, password, salary, POSITION\_ID, created\_at, Emp\_Warehouse) VALUES ('Kaci', 'Jarvis', 'KaciJarvis', 'Kaci', 13500, 4, '2010-01-04', 2);

INSERT INTO employee (firstName, surName, login, password, salary, POSITION\_ID, created\_at, Emp\_Store) VALUES ('Idris', 'Pugh', 'IdrisPugh', 'Idris', 20000, 5, '2010-01-04', 1);

INSERT INTO employee (firstName, surName, login, password, salary, POSITION\_ID, created\_at, Emp\_Store) VALUES ('Griff', 'Swift', 'GriffSwift', 'Griff', 21000, 5, '2010-01-04', 2);

INSERT INTO employee (firstName, surName, login, password, salary, POSITION\_ID, created\_at, Emp\_Store) VALUES ('Malaki', 'Aguilar', 'MalakiAguilar', 'Malaki', 20500, 5, '2010-01-05', 3);

INSERT INTO employee (firstName, surName, login, password, salary, POSITION\_ID, created\_at, Emp\_Store) VALUES ('Teigan', 'Pena', 'TeiganPena', 'Teigan', 19000, 5, '2010-01-05', 4);

INSERT INTO employee (firstName, surName, login, password, salary, POSITION\_ID, created\_at) VALUES ('Chaim', 'Dunlop', 'ChaimDunlop', 'Chaim', 30000, 2, '2010-01-02');

INSERT INTO employee (firstName, surName, login, password, salary, POSITION\_ID, created\_at) VALUES ('Mollie ', 'Piper', 'MolliePiper', 'Mollie', 32000, 2, '2010-01-01');

INSERT INTO employee (firstName, surName, login, password, salary, POSITION\_ID, created\_at) VALUES ('Alessio', 'Hogg', 'AlessioHogg', 'Alessio', 30500, 1, '2010-01-04');

INSERT INTO employee (firstName, surName, login, password, salary, POSITION\_ID, created\_at) VALUES ('Darlene', 'Knott', 'DarleneKnott', 'Darlene', 29000, 1, '2010-01-05');

/\* SUPPLIER INSERT \*/

INSERT INTO supplier (name, email, phoneNumber, MANAGER\_ID, address) VALUES ('SP "RAGNAR"', 'ragnar.company@mail.com', 0984712893, 14, 'US, NY-Manhattan');

INSERT INTO supplier (name, email, phoneNumber, MANAGER\_ID, address) VALUES ('SP "DAWTON Electrics"', 'dawton.company@mail.com', 0874692154, 14, 'US, Texas');

INSERT INTO supplier (name, email, phoneNumber, MANAGER\_ID, address) VALUES ('SP "MEKANISM"', 'mekanism.company@mail.com', 0874692154, 15, 'US, Vancouver');

/\* SUPPLIER\_CONTRACT INSERT \*/

INSERT INTO supplier\_contract (SUPPLIER\_ID, contract\_src) VALUES (1, 'RAGNAR\_contract.pdf');

INSERT INTO supplier\_contract (SUPPLIER\_ID, contract\_src) VALUES (2, 'DAWTON\_contract.pdf');

INSERT INTO supplier\_contract (SUPPLIER\_ID, contract\_src) VALUES (3, 'MEKANISM\_contract.pdf');

/\* CONTRACT\_MATERIAL INSERT \*/

INSERT INTO contract\_material (CONTRACT\_ID, MATERIAL\_ID, quantity) VALUES (1, 3, 20);

INSERT INTO contract\_material (CONTRACT\_ID, MATERIAL\_ID, quantity) VALUES (1, 4, 15);

INSERT INTO contract\_material (CONTRACT\_ID, MATERIAL\_ID, quantity) VALUES (2, 5, 10);

INSERT INTO contract\_material (CONTRACT\_ID, MATERIAL\_ID, quantity) VALUES (3, 1, 15);

INSERT INTO contract\_material (CONTRACT\_ID, MATERIAL\_ID, quantity) VALUES (3, 2, 15);

/\* WAREHOUSE\_MATERIAL INSERT \*/

INSERT INTO warehouse\_material (WAREHOUSE\_ID, MATERIAL\_ID, quantity) VALUES (1, 3, 20);

INSERT INTO warehouse\_material (WAREHOUSE\_ID, MATERIAL\_ID, quantity) VALUES (1, 4, 15);

INSERT INTO warehouse\_material (WAREHOUSE\_ID, MATERIAL\_ID, quantity) VALUES (2, 5, 10);

INSERT INTO warehouse\_material (WAREHOUSE\_ID, MATERIAL\_ID, quantity) VALUES (2, 2, 15);

INSERT INTO warehouse\_material (WAREHOUSE\_ID, MATERIAL\_ID, quantity) VALUES (2, 1, 15);

/\* WAREHOUSE\_PRODUCT INSERT \*/

INSERT INTO warehouse\_product (WAREHOUSE\_ID, PRODUCT\_ID, quantity) VALUES (1, 1, 5);

INSERT INTO warehouse\_product (WAREHOUSE\_ID, PRODUCT\_ID, quantity) VALUES (1, 2, 7);

INSERT INTO warehouse\_product (WAREHOUSE\_ID, PRODUCT\_ID, quantity) VALUES (2, 3, 8);

INSERT INTO warehouse\_product (WAREHOUSE\_ID, PRODUCT\_ID, quantity) VALUES (2, 4, 15);

INSERT INTO warehouse\_product (WAREHOUSE\_ID, PRODUCT\_ID, quantity) VALUES (2, 5, 5);

INSERT INTO warehouse\_product (WAREHOUSE\_ID, PRODUCT\_ID, quantity) VALUES (2, 6, 12);

/\* STORE\_PRODUCT INSERT \*/

INSERT INTO store\_product (STORE\_ID, PRODUCT\_ID, quantity) VALUES (1, 1, 30);

INSERT INTO store\_product (STORE\_ID, PRODUCT\_ID, quantity) VALUES (1, 2, 25);

INSERT INTO store\_product (STORE\_ID, PRODUCT\_ID, quantity) VALUES (1, 3, 10);

INSERT INTO store\_product (STORE\_ID, PRODUCT\_ID, quantity) VALUES (2, 1, 13);

INSERT INTO store\_product (STORE\_ID, PRODUCT\_ID, quantity) VALUES (2, 4, 5);

INSERT INTO store\_product (STORE\_ID, PRODUCT\_ID, quantity) VALUES (2, 3, 20);

INSERT INTO store\_product (STORE\_ID, PRODUCT\_ID, quantity) VALUES (3, 5, 17);

INSERT INTO store\_product (STORE\_ID, PRODUCT\_ID, quantity) VALUES (3, 6, 11);

INSERT INTO store\_product (STORE\_ID, PRODUCT\_ID, quantity) VALUES (4, 2, 8);

INSERT INTO store\_product (STORE\_ID, PRODUCT\_ID, quantity) VALUES (4, 3, 27);

/\* MATERIALS TO FACTORY INSERT\*/

INSERT INTO material\_to\_factory (WAREHOUSE\_ID, FACTORY\_ID, MATERIAL\_ID, quantity, supplied\_at) VALUES (1, 1, 3, 10, '2010-02-05');

INSERT INTO material\_to\_factory (WAREHOUSE\_ID, FACTORY\_ID, MATERIAL\_ID, quantity, supplied\_at) VALUES (1, 1, 4, 10, '2010-02-06');

INSERT INTO material\_to\_factory (WAREHOUSE\_ID, FACTORY\_ID, MATERIAL\_ID, quantity, supplied\_at) VALUES (2, 1, 5, 5, '2010-02-06');

INSERT INTO material\_to\_factory (WAREHOUSE\_ID, FACTORY\_ID, MATERIAL\_ID, quantity, supplied\_at) VALUES (2, 1, 1, 5, '2010-02-07');

INSERT INTO material\_to\_factory (WAREHOUSE\_ID, FACTORY\_ID, MATERIAL\_ID, quantity, supplied\_at) VALUES (2, 1, 2, 5, '2010-02-07');

INSERT INTO material\_to\_factory (WAREHOUSE\_ID, FACTORY\_ID, MATERIAL\_ID, quantity, supplied\_at) VALUES (1, 2, 3, 15, '2010-02-05');

INSERT INTO material\_to\_factory (WAREHOUSE\_ID, FACTORY\_ID, MATERIAL\_ID, quantity, supplied\_at) VALUES (1, 2, 4, 15, '2010-02-06');

INSERT INTO material\_to\_factory (WAREHOUSE\_ID, FACTORY\_ID, MATERIAL\_ID, quantity, supplied\_at) VALUES (2, 2, 5, 10, '2010-02-06');

INSERT INTO material\_to\_factory (WAREHOUSE\_ID, FACTORY\_ID, MATERIAL\_ID, quantity, supplied\_at) VALUES (2, 2, 1, 10, '2010-02-07');

INSERT INTO material\_to\_factory (WAREHOUSE\_ID, FACTORY\_ID, MATERIAL\_ID, quantity, supplied\_at) VALUES (2, 2, 2, 10, '2010-02-07');

/\* WAREHOUSE TO STORE INSERT \*/

INSERT INTO warehouse\_to\_store (WAREHOUSE\_ID, STORE\_ID, PRODUCT\_ID, quantity, supplied\_at) VALUES (1, 1, 1, 10, '2010-02-10');

INSERT INTO warehouse\_to\_store (WAREHOUSE\_ID, STORE\_ID, PRODUCT\_ID, quantity, supplied\_at) VALUES (1, 1, 2, 5, '2010-02-10');

INSERT INTO warehouse\_to\_store (WAREHOUSE\_ID, STORE\_ID, PRODUCT\_ID, quantity, supplied\_at) VALUES (2, 1, 3, 15, '2010-02-11');

INSERT INTO warehouse\_to\_store (WAREHOUSE\_ID, STORE\_ID, PRODUCT\_ID, quantity, supplied\_at) VALUES (1, 2, 1, 5, '2010-02-11');

INSERT INTO warehouse\_to\_store (WAREHOUSE\_ID, STORE\_ID, PRODUCT\_ID, quantity, supplied\_at) VALUES (1, 2, 2, 10, '2010-02-11');

INSERT INTO warehouse\_to\_store (WAREHOUSE\_ID, STORE\_ID, PRODUCT\_ID, quantity, supplied\_at) VALUES (2, 2, 5, 3, '2010-02-12');

/\* PRODUCTION INSERT \*/

INSERT INTO production (FACTORY\_ID, WAREHOUSE\_ID, PRODUCT\_ID, quantity, produced\_at) VALUES (1, 1, 1, 5, '2010-01-10');

INSERT INTO production (FACTORY\_ID, WAREHOUSE\_ID, PRODUCT\_ID, quantity, produced\_at) VALUES (1, 1, 2, 7, '2010-01-11');

INSERT INTO production (FACTORY\_ID, WAREHOUSE\_ID, PRODUCT\_ID, quantity, produced\_at) VALUES (2, 2, 3, 8, '2010-01-11');

INSERT INTO production (FACTORY\_ID, WAREHOUSE\_ID, PRODUCT\_ID, quantity, produced\_at) VALUES (2, 2, 4, 15, '2010-01-12');

INSERT INTO production (FACTORY\_ID, WAREHOUSE\_ID, PRODUCT\_ID, quantity, produced\_at) VALUES (2, 2, 5, 5, '2010-01-12');

INSERT INTO production (FACTORY\_ID, WAREHOUSE\_ID, PRODUCT\_ID, quantity, produced\_at) VALUES (2, 2, 6, 12, '2010-01-12');

/\* ORDER INSERT \*/

INSERT INTO perfect\_tools.order (CLIENT\_ID, status, STORE\_ID, created\_at) VALUES (1,'order\_accepted', 1 , '2010-04-01' );

INSERT INTO perfect\_tools.order (CLIENT\_ID, status, STORE\_ID, created\_at) VALUES (2,'order\_in\_process', 1 , '2010-04-02' );

INSERT INTO perfect\_tools.order (CLIENT\_ID, status, STORE\_ID, created\_at) VALUES (3,'order\_in\_process', 1 , '2010-04-02' );

INSERT INTO perfect\_tools.order (CLIENT\_ID, status, STORE\_ID, created\_at) VALUES (4,'order\_in\_process', 2 , '2010-04-04' );

INSERT INTO perfect\_tools.order (CLIENT\_ID, status, STORE\_ID, created\_at) VALUES (5,'order\_in\_process', 2 , '2010-04-05' );

INSERT INTO perfect\_tools.order (CLIENT\_ID, status, MANAGER\_ID, WAREHOUSE\_ID, created\_at) VALUES (6,'order\_completed', 16, 1 , '2010-04-10' );

INSERT INTO perfect\_tools.order (CLIENT\_ID, status, MANAGER\_ID, WAREHOUSE\_ID, created\_at) VALUES (7,'order\_completed', 17, 2 , '2010-04-10' );

/\* ORDER PRODUCT INSERT \*/

INSERT INTO order\_product (ORDER\_ID, PRODUCT\_ID, quantity) VALUES ( 1, 1, 1 );

INSERT INTO order\_product (ORDER\_ID, PRODUCT\_ID, quantity) VALUES ( 2, 2, 2 );

INSERT INTO order\_product (ORDER\_ID, PRODUCT\_ID, quantity) VALUES ( 3, 3, 3 );

INSERT INTO order\_product (ORDER\_ID, PRODUCT\_ID, quantity) VALUES ( 4, 4, 1 );

INSERT INTO order\_product (ORDER\_ID, PRODUCT\_ID, quantity) VALUES ( 5, 3, 2 );

INSERT INTO order\_product (ORDER\_ID, PRODUCT\_ID, quantity) VALUES ( 6, 2, 5 );

INSERT INTO order\_product (ORDER\_ID, PRODUCT\_ID, quantity) VALUES ( 6, 1, 5 );

INSERT INTO order\_product (ORDER\_ID, PRODUCT\_ID, quantity) VALUES ( 7, 6, 4 );

INSERT INTO order\_product (ORDER\_ID, PRODUCT\_ID, quantity) VALUES ( 7, 5, 8 );