**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”**

**Тема:**

**«Інформаційна система для роботи з клієнтами автосалону»**

**Науковий керівник: Виконавець:**

к.ф.-м.н., доц. Старух А.І. Горбач М.Г.

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

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

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

**‘‘\_\_\_’’\_\_\_\_\_\_\_\_ 2021р. ‘‘\_\_\_’’\_\_\_\_\_\_\_\_ 2021р.**

**Загальна кількість балів**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (підписи, ПІП членів комісії)

**Львів 2020**

**Зміст**

**Зміст**

**Вступ**

**Розділ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ**

* 1. **Постановка завдання**
  2. **Розробка моделі варіантів використання веб сайту**
     1. **Обгрунтування потреби у веб-сайті**
     2. **Проектування веб-сайту**

**Розділ 2. РОЗРОБКА БАЗИ ДАНИХ**

* 1. **Опис моделі даних**
  2. **Нормалізація реляційних відношень**
  3. **Визначення типів даних**
  4. **Обмеження цілісності даних**
  5. **Реалізація SQL-скрипту**

**Висновки**

**Список використаної літератури**

**Додатки**

**ВСТУП**

Функціонування будь-якої системи управління пов'язане з циркуляцією в ній інформації. Будь-який процес управління - перш за все, інформаційний процес, який передбачає виконання функцій збору, передачі, обробки, аналізу інформації та прийняття відповідних рішень. Забезпечення якісного інформаційного обслуговування користувачів шляхом видачі вірогідної, своєчасної та достатньої для прийняття управлінських рішень інформації у формі, яка є зручною для використання, є основним призначенням інформаційного забезпечення.

Основна перевага автоматизації - це скорочення надмірності даних, що зберігаються, а отже, економія об'єму використовуваної пам'яті, зменшення витрат на багатократні операції оновлення надмірних копій і усунення можливості виникнення суперечностей через зберігання в різних місцях відомостей про один і той же об'єкт, збільшення міри достовірності інформації і збільшення швидкості обробки інформації; зайва кількість внутрішніх проміжних документів, різних журналів, тек, заявок і т.д., повторне внесення однієї і тієї ж інформації в різні проміжні документи. Також значно скорочує час автоматичний пошук інформації, який виробляється із спеціальних  екранних форм, в яких указуються параметри пошуку об'єкту.

Задача обліку продажу товарів займає провідне місце в системі обліку. Оскільки облік та контроль за продажем товарів є одним з найскладніших та трудомістких частин блоку обліку товарів та пов'язаний з обробкою великих обсягів облікової інформації та складними розрахунками

Об'єктом дослідження даної роботи є інформаційні процесі на підприємстві, яке займається продажем автомобілів.

На сьогоднішній день комп’ютер став невід’ємною частиною нашого життя. Так як він полегшує нам життя, виконуючи за нас багато роботи, яку ми самі можемо виконувати довгий час. Також комп’ютер став невід’ємною частиною і у автосалоні. Необов’язково наймати на роботу у автосалон консультантів – професіоналів, яким потрібно буде платити високу заробітну плату за те що вони просто будуть розповідати покупцям про автомобілі. Цю роботу за них зможе виконати комп’ютер з базою даних, яка міститиме багато цікавої інформації про автомобілі. Навіть сидячи вдома, кожен охочий зможе переглянути перелік автомобілів які продаються в тому чи іншому автосалоні.  
 Дана тема є досить актуальною, тому що зараз дуже високий попит на автомобілі і в цьому ми можемо переконатись, коли їдучи на роботу чи додому, довго наш транспорт стоїть в пробці. Кількість автосалонів в місті швидко збільшується так як це дуже вигідний бізнес. І автосалон який пропонує широкий вибір сучасних, нових, надійних і стильних автомобілів за помірною ціною, а також може швидко і якісно обслужити покупця завжди буде мати великий прибуток. А також якщо про цей автосалон можна дізнатися з мережі Internet.

**Розділ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ**

* 1. **Постановка завдання**

На сьогодні відбуваються радикальні зміни в соціальній структур світу. Наслідок такого є розширення сфери інформаційної діяльності та послуг. У такій сфері працює приблизно половина від загальної кількості зайнятого населення. До цієї сфери все більше стає наближеною і система автосалонів.  
Однією з найскладніших задач, з якою доводиться стикатися при організаціїї роботи автосалону - це управління послугами, що надає заклад. В більшості випадків ті хто розробляють вирішення подібним задачам, недооцінюють її справжньої складності і виділяють для неї недостатню кількість ресурсів. Поява сучасної обчислювальної техніки стала стимулом для розроблення програмної продукції.  
 Кожний автосалон є індивідуальним, що створює ще одну проблему. Розробити ж універсальне вирішення, включаючи всі вимоги, не так вже і легко.  
Через це і виникає необхідність залучити у процес управління новітні інформаційні засоби. Цей компютеризований підхід дозволяє вирішувати дані завдання з більшою точністю і швидкістю.  
 Для консультантів у автосалоні залучення до компютеризованих систем дозволяє заощадити час, який вони можуть використати для вирішення інших проблем, що не потребують втручання компютера.  
 В свою чергу економічність зумовлена:  
- використанням продуктивних і якісних програмних комплексів;  
- скороченням витрат, що повязані з скороченням кількості консультантів.

**1.2 Розробка моделі варіантів використання веб сайту**

На данний момент у всесвітній павутині розміщено декілька мільйонів веб-сайтів і їх число постійно росте. Усі вони поділяються на різноманітні категорії серед яких можна перелічити наступні:

**Сайт-візитка**

Саме той набір інформації, поширення якої в Інтернеті, безсумнівно, корисно як починаючим, так і акулам бізнесу. Це своєрідний довідник про фірму, який містить всі необхідні контактні дані та інформацію про діяльність компанії. Це повинна бути коротка, чітка і найважливіша інформація про вас, вашу сферу діяльності, інтереси. Не варто перевантажувати ваш сайт-візитку інформацією, яка можливо не знадобиться потенційному клієнтові або партнерові.

Сайт-візитка – це:

• зручний засіб для сканування грунту ринку;

• швидко і якісно;

• вигідний для організацій, у яких бізнес здійснюється не в Інтернеті.

**Kорпоративний сайт**

Більш серйозним виглядом представництва компанії в Інтернеті є корпоративні сайти. Це обов'язковий інструмент успішного бізнесу будь-якої компанії. У першу чергу, сайти подібного роду допомагають формувати позитивний імідж компанії, залучають нових замовників, які можуть не знати про існування вашої фірми на ринку, і відповідно, розширюють ринок збуту компанії. За допомогою корпоративного сайту клієнти та замовники отримують інформацію про ціни на товари і послуги в зручному вигляді. Значно збільшити прибуток можливо шляхом надання можливості робити замовлення вашим клієнтам прямо в Інтернеті, на вашій сторінці.

Отже, корпоративний сайт – це:

• доказ серйозності і перспективності Вашого бізнесу;

• гідне представництво Вашої компанії на ринку;

• найдешевший і наймасовіший вид реклами Вашого бізнесу;

• зручно і комерційно виправдано.

**Промо-сайт**

Прекрасним рекламним інструментом може служити промо-сайт. Особливо, якщо Ви плануєте провести рекламну компанію окремого товару або послуги (культурного заходу. ..). Структура промо-сайту сама по собі повинна виконувати функцію рекламного проспекту або буклету, і привертати увагу клієнтів. Також на такому сайті можна розмістити ту інформацію, яка не поміститься на сторінці журналу, в радіоповідомленні, телевізійному ролику. Обсяг промо-сайту - до 10 сторінок, на яких можна розмістити дані про компанію, напрямок діяльності, просування послуги, контактні дані. Окремо повідомляється інформація про акцію, із зазначенням термінів і місця проведення, умов участі, новин та ін.

Промо-сайт – це:

• використання можливостей інтернету для впливу на цільову аудиторію; • донесення до потенційних клієнтів максимум інформації про товар і послугу;

• потужний й ефективний інструмент реклами компанії та її продуктів.

**Сайт-вітрина**

Бюджетним варіантом реклами вашої компанії є сайт-вітрина. Якщо ви хочете продавати в Інтернеті один конкретний товар або групу товарів, то цей тип сайтів ідеально підходить для вас. На таких сторінках знаходиться вся інформація, необхідна для того, щоб клієнт побачив, зацікавився і купив даний товар, не відволікаючись на новини та посилання. Це структурований каталог продукції вашої компанії. Але цей каталог організований таким чином (фото, опис, ціна), щоб клієнт зробив як можна менше кліків, для того, щоб знайти потрібний товар. Головне призначення сайту-вітрини - продавати.

Сайт-вітрина – це:

• максимальний прибуток при мінімальному вкладенні сил та засобів;

• стабільний дохід;

• позитивний імідж компанії.

**Інтернет-магазин**

На відміну від сайту-вітрини, де клієнт може побачити наявність всіх товарів, за допомогою інтернет-магазину клієнт може ще й зробити замовлення, вибрати варіант розрахунку, спосіб отримання замовлення та одержати рахунок на оплату. Такий сайт має бути зручним і функціональним, щоб відвідувач міг легко знайти те, що йому потрібно, відправити товар у кошик та оформити покупку в кілька кліків. Інтернет-магазин істотно зменшить ваші витрати, так, як не треба орендувати приміщення для магазину, не треба платити заробітну плату співробітникам. Серед інших переваг: величезна аудиторія, можливість розміщувати необмежений асортимент товарів, своєчасно реагувати на зміни на ринку, враховувати потреби клієнтів та інші.

Інтернет - магазин – це:

• нові можливості для бізнесу;

• розширення ринку збуту;

• залучення широкої аудиторії клієнтів;

• скорочення витрат.

Такі сайти в наш час набирають дуже велику популярність, як з боку користувачів, так і з боку підприємців. Це веб-сайти, в яких кожна людина може купити будь-який товар не відходячи від свого комп’ютера.

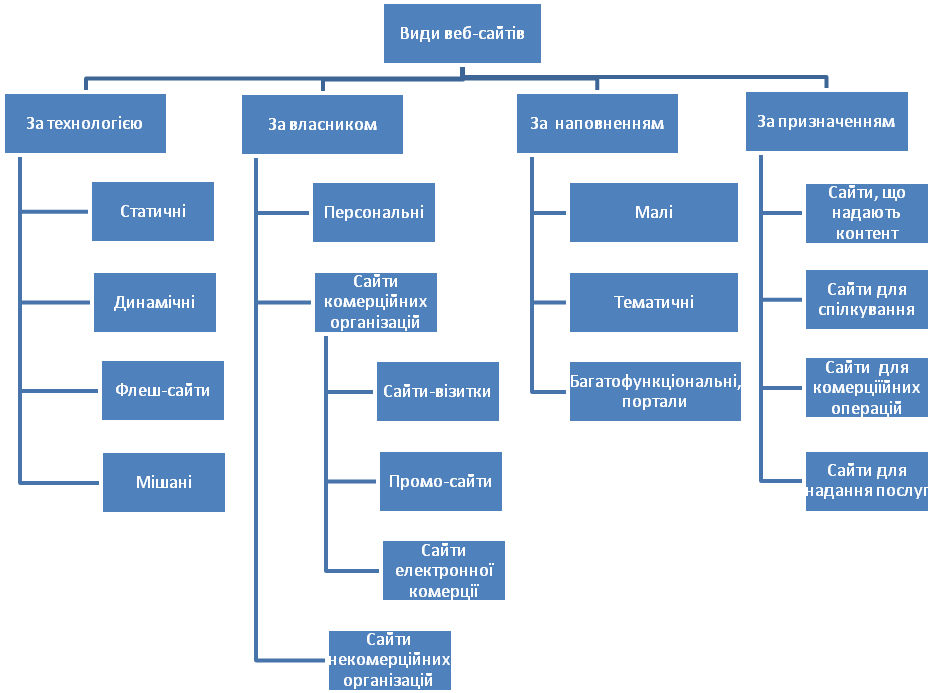


Рис. 1.1 Види вебсайтів

**1.2.1 Обгрунтування потреби у веб-сайті**

На сьогоднішній день практично кожна організація має власний веб-сайт. В умовах використання сучасних інформаційних технологій – це необхідний чинник існування, що дозволяє розширити поле рекламної діяльності і привернути тим самим додаткових клієнтів. Для прикладу значна кількість потенційних клієнтів з більшою долею ймовірності відвідає веб-сайт, щоб ознайомитись з діяльністю компанії, аніж дзвонитиме за номером телефону опублікованому цією компанією в рекламній брошурі. Для того, щоб розмістити на рекламних носіях адресу веб-сайту компанії достатньо лише однієї стрічки, а об’єм поданої інформації на сторінках самого веб-сайту не обмежений і завжди актуальний .

**1.2.2 Проектування веб-сайту**

Проектування і розробка сайтів включає наступні необхідні пункти:

• Затвердження первинного технічного завдання розробки сайту.

• Визначення структурної схеми сайту - розташування розділів, контенту і навігації.

• Веб-дизайн - створення графічних елементів макету сайту, стилів і елементів навігації.

• Розробка програмного коду, модулів, бази даних і інших елементів сайту необхідних в проекті.

• Тестування і розміщення сайту в мережі Інтернет.

# **Розділ 2. Розробка бази даних**

# **2.1. Опис моделі даних**

База даних (БД) — це організована структура, яка призначена для зберігання, зміни та обробки взаємозалежної інформації, переважно великих обсягів. БД використовують для динамічних сайтів з великими обсягами (інтернет-магазин, портал, корпоративний сайт).

БД - упорядкований набір логічно взаємопов'язаних даних, що використовується спільно, та призначений для задоволення інформаційних потреб користувачів. У технічному розумінні включно й система управління БД.

Первинним призначенням бази даних є зберігання масивів даних. Але їх широко використовують і для збереження адміністративної інформації та спеціалізованих даних, наприклад, для інженерних даних чи для економічних моделей.

Прикладами використання баз даних можуть бути:

* автоматизовані системи обліку;
* реєстри та каталоги;
* геоінформаційні системи;
* бази даних транспортних систем;

Бази даних класифікують за різними критеріями.

**За моделлю організації**[**даних**](https://uk.wikipedia.org/wiki/%D0%94%D0%B0%D0%BD%D1%96_(%D0%BE%D0%B1%D1%87%D0%B8%D1%81%D0%BB%D1%8E%D0%B2%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0_%D1%82%D0%B5%D1%85%D0%BD%D1%96%D0%BA%D0%B0)) **розрізняють такі бази даних**:

* ***Ієрархічна***.

Ієрархічна база даних може бути представлена як дерево, що складається з об'єктів різних рівнів.

* ***Об'єктно-орієнтована***.

У базі даних цього виду дані оформляють у вигляді моделей об'єктів.

* ***Мережева***.

Дана БД схожа до ієрархічної, за винятком того, що всі об'єкти можуть мати декілька попередників.

* [***Реляційна***](https://uk.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D1%96%D0%B9%D0%BD%D0%B0_%D0%B1%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85).

[Реляційна база даних](https://uk.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D1%96%D0%B9%D0%BD%D0%B0_%D0%B1%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85) зберігає дані у вигляді таблиць. Найчастіше [СКБД](https://uk.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D0%BA%D0%B5%D1%80%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0%D0%BC%D0%B8_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85) використовують [реляційну модель даних](https://uk.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D1%96%D0%B9%D0%BD%D0%B0_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85).

**За розміщенням даних** виділяють такі види баз:

* ***Локальна***.

Ця база даних підтримується лише на одному комп'ютері.

* ***Розподілена***.

Частини такої бази даних розміщують на різних комп'ютерах мережі.

**За технологією фізичного зберігання** **виділяють**:

* БД у вторинній пам'яті (традиційні).
* БД в оперативній пам'яті (in-memory database).
* БД у третинній пам'яті (tertiary database).

### **Структуровані та неструктуровані БД**

*Структуровані*  бази використовують [структури даних](https://uk.wikipedia.org/wiki/%D0%A1%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B8_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85), тобто структурований опис типу фактів за допомогою схеми даних, більш відомої як [модель даних](https://uk.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85). Модель даних описує об'єкти та взаємовідношення між ними. Існує декілька моделей (чи типів) баз даних, основні: [ієрархічна](https://uk.wikipedia.org/wiki/%D0%91%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85_%D1%96%D1%94%D1%80%D0%B0%D1%80%D1%85%D1%96%D1%87%D0%BD%D0%B0), [мережна](https://uk.wikipedia.org/wiki/%D0%91%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85_%D0%BC%D0%B5%D1%80%D0%B5%D0%B6%D0%BD%D0%B0) та реляційна.

До *неструктурованих*  баз належать повнотекстові бази даних, які містять неструктуровані тексти статей чи книг у формі, що дозволяє здійснювати швидкий пошук

У сучасних інформаційних системах для забезпечення роботи з базами даних використовують системи керування базами даних (СКБД). **Система керування базами даних** — це система, заснована на програмних та технічних засобах, яка забезпечує визначення, створення, маніпулювання, контроль, керування та використання баз даних . Застосунки для роботи з базою даних можуть бути частиною СКБД або автономними. Найпопулярнішими СКБД є MySQL, PostgreSQL, Microsoft SQL Server, Oracle, Sybase, Interbase, Firebird та IBM DB2. СКБД дозволяють ефективно працювати з базами даних, обсяг яких робить неможливим їх ручне опрацювання.

Сучасні [СКБД](https://uk.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D0%BA%D0%B5%D1%80%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0%D0%BC%D0%B8_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85) забезпечують функції щодо керування даними, які можна поділити на такі групи:

* Оголошення даних — створення, зміна та видалення визначень, які описують організацію даних.
* Модифікація даних — додавання даних, їх редагування та видалення.
* Отримання даних — надання даних за запитом застосунку у формі, яка дозволяє їх безпосереднє використання. Дані можуть надаватись або у формі, в якій вони зберігаються у базі даних, або в іншій формі (наприклад, через поєднання різних даних).
* Адміністрування даних — реєстрування та відслідковування дій користувачів, дотримання безпеки роботи з даними, забезпечення надійності та цілісності даних, моніторинг продуктивності, резервне копіювання та відновлення даних тощо.

Бази даних для сайтів дають змогу зберігати інформацію, що виглядає як зв'язані між собою таблиці. Саме в БД зберігаються вся необхідна та корисна інформація для функціонування сайту (клієнтські дані, прайс-лист, список товарів).

Щоб створити запит до бази даних часто використовують Structured Query Language. SQL дає змогу додавати, редагувати та видаляти інформацію, що міститься у таблицях.  
Під час програмування сайтів використовують різні системи управління БД. До основних СУБД, відносять:

* об'єктно-реляційна система управління базами даних Oracle Database;
* вільна система управління базами даних PostgreSQL;
* система керування базами даних Microsoft SQL Сервер;
* вільна система управління базами даних MySQL;

Такі системи управління відрізняються централізованою обробкою запитів, забезпечують надійність, доступність та безпеку БД.

Найбільш популярною системою управління є MySQL, вона дає зручний доступ для управління БД та підтримує велику кількість таблиць різних типів.

Реляційна модель орієнтована на організацію даних у вигляді двовимірних таблиць. Кожна реляційна таблиця являє собою двовимірний масив і має наступні властивості:

* + порядок проходження рядків і стовпчиків може бути довільним.
  + кожен стовпчик має унікальне ім'я;
  + кожен елемент таблиці - один елемент даних;
  + однакові рядки в таблиці відсутні;

**2.2 Нормалізація реляційних відношень**

Під реляційною БД прийнято розуміти сукупність екземплярів кінцевих відношень. Сукупність схем відношень утворює схему реляційної БД.

Схема реляційної БД є логічною моделлю реляційної БД. На основі інформаційної моделі у процесі проектування створюються логічна й фізична моделі даних. Інформаційна модель даних відбиває потреби системи в даних і зв'язку між даними з погляду їх споживачів - користувачів; логічна модель даних є незалежним логічним поданням даних; фізична модель даних містить визначення всіх реалізованих об'єктів у конкретній БД для конкретної СУБД.

Встановлення функціональної залежності й одержання найкращого з погляду мінімальності подання множини функціональних залежностей дозволять побудувати найбільш оптимальний варіант БД, що забезпечує надійність зберігання й обробки даних на основі методів еквівалентних перетворень схем відношень реляційної БД. Процес вирішення такого завдання називається нормалізацією відношень інформаційної моделі ПО й полягає у перетворенні її об'єктів у логічні таблиці БД.

Створення системи, що одночасно задовольняє всім вищезгаданим вимогам, являє собою складну оптимізаційну задачу, що часом не має однозначного вирішення.

Теорія функціональних залежностей дозволяє встановити певні вимоги до схем відношень у реляційній БД. Ці вимоги формулюються у термінах властивостей відношень і називаються нормальними формами схем відношень. Кожна нормальна форма відношень пов'язана з певним класом функціональної залежності, які представлені у відношеннях. Одним з очевидних засобів усунення потенційної суперечливості даних у відношеннях логічної моделі реляційної БД є їх розбиття на двоє або більше відношень, у кожному з яких є присутньою тільки одна функціональна залежність. Нормалізація - це розбивка таблиці на дві або більш, які характеризуються кращими властивостями при доповненні, зміні і вилученні даних.

У теорії реляційних баз даних є така послідовність нормальних форм:

* перша нормальна форма, 1NF (First Normal Form);

- [Перша нормальна форма](https://uk.wikipedia.org/wiki/%D0%9F%D0%B5%D1%80%D1%88%D0%B0_%D0%BD%D0%BE%D1%80%D0%BC%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0_%D1%84%D0%BE%D1%80%D0%BC%D0%B0) (1НФ, 1NF) утворює ґрунт для структурованої схеми бази даних:

Кожна таблиця повинна мати основний ключ: мінімальний набір колонок, які ідентифікують запис.

Уникнення повторень груп (категорії даних, що можуть зустрічатись різну кількість разів в різних записах) правильно визначаючи неключові атрибути.

* друга нормальна форма, 2NF (Second Normal Form);

-[Друга нормальна форма](https://uk.wikipedia.org/wiki/%D0%94%D1%80%D1%83%D0%B3%D0%B0_%D0%BD%D0%BE%D1%80%D0%BC%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0_%D1%84%D0%BE%D1%80%D0%BC%D0%B0) (2НФ, 2NF) вимагає, аби дані, що зберігаються в таблицях із композитним ключем, не залежали лише від частини ключа:

[Схема бази даних](https://uk.wikipedia.org/wiki/%D0%A1%D1%85%D0%B5%D0%BC%D0%B0_%D0%B1%D0%B0%D0%B7%D0%B8_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85) повинна відповідати вимогам першої нормальної форми.

Дані, що повторно з'являються в декількох рядках, виносяться в окремі таблиці.

* третя нормальна форма, ЗNF (Third Normal Form);

- [Третя нормальна форма](https://uk.wikipedia.org/wiki/%D0%A2%D1%80%D0%B5%D1%82%D1%8F_%D0%BD%D0%BE%D1%80%D0%BC%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0_%D1%84%D0%BE%D1%80%D0%BC%D0%B0) (3НФ, 3NF) вимагає, аби дані в таблиці залежали винятково від основного ключа:

[Схема бази даних](https://uk.wikipedia.org/wiki/%D0%A1%D1%85%D0%B5%D0%BC%D0%B0_%D0%B1%D0%B0%D0%B7%D0%B8_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85) повинна відповідати всім вимогам другої нормальної форми.

Будь-яке поле, що залежить від основного ключа та від будь-якого іншого поля, має виноситись в окрему таблицю.

Відношення знаходиться в [НФБК](https://uk.wikipedia.org/wiki/%D0%9D%D0%BE%D1%80%D0%BC%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0_%D1%84%D0%BE%D1%80%D0%BC%D0%B0_%D0%91%D0%BE%D0%B9%D1%81%D0%B0_%E2%80%94_%D0%9A%D0%BE%D0%B4%D0%B4%D0%B0) тоді і лише тоді, коли детермінант кожної функціональної залежності є потенційним ключем. Якщо це правило не виконується, то, щоб привести вказане відношення до НФБК, його слід розділити на два відношення шляхом двох операцій проєкції на кожну функціональну залежність, детермінант якої не є потенційним ключем:

Проекція без атрибутів залежної частини такої функціональної залежності;

Проекція на всі атрибути цієї функціональної залежності.

Визначення НФБК не потребує жодних умов попередніх нормальних форм. Якщо проводити нормалізацію послідовно, то в переважній більшості випадків при досягненні 3НФ автоматично будуть задовольнятися вимоги НФБК. 3НФ не збігається з НФБК лише тоді, коли одночасно виконуються такі 3 умови:

Відношення має 2 або більше потенційних ключів.

Ці потенційні ключі складені (містять більш ніж один атрибут)

Ці потенційні ключі перекриваються, тобто мають щонайменше один спільний атрибут.

* четверта нормальна форма, 4NF (Fourth Normal Form);

- [Четверта нормальна форма](https://uk.wikipedia.org/wiki/%D0%A7%D0%B5%D1%82%D0%B2%D0%B5%D1%80%D1%82%D0%B0_%D0%BD%D0%BE%D1%80%D0%BC%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0_%D1%84%D0%BE%D1%80%D0%BC%D0%B0) (4НФ, 4NF) потребує, аби в схемі баз даних не було нетривіальних багатозначних залежностей множин атрибутів від будь чого, окрім надмножини ключа-кандидата.

Вважається, що таблиця знаходиться у 4НФ тоді і лише тоді, коли вона знаходиться в НФБК та багатозначні залежності є функціональними залежностями. Четверта нормальна форма усуває небажані структури даних — багатозначні залежності.

* п'ята нормальна форма, 5NF (Fifth Normal Form).

- [П'ята нормальна форма](https://uk.wikipedia.org/wiki/%D0%9F%27%D1%8F%D1%82%D0%B0_%D0%BD%D0%BE%D1%80%D0%BC%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0_%D1%84%D0%BE%D1%80%D0%BC%D0%B0) (5НФ, 5NF, PJ/NF) вимагає, аби не було нетривіальних залежностей об'єднання, котрі б не витікали із обмежень ключів.

Вважається, що таблиця в п'ятій нормальній формі тоді і лише тоді, коли вона знаходиться в 4НФ та кожна залежність об'єднання зумовлена її ключами-кандидатами.

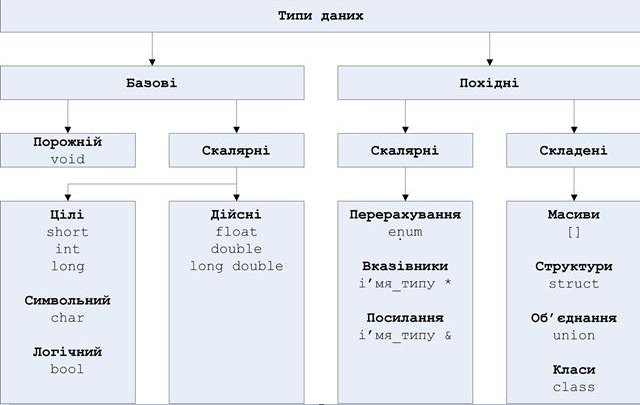
Нормальна форма домен/ключ

Докладніше: [Доменно-ключова нормальна форма](https://uk.wikipedia.org/wiki/%D0%94%D0%BE%D0%BC%D0%B5%D0%BD%D0%BD%D0%BE-%D0%BA%D0%BB%D1%8E%D1%87%D0%BE%D0%B2%D0%B0_%D0%BD%D0%BE%D1%80%D0%BC%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0_%D1%84%D0%BE%D1%80%D0%BC%D0%B0)

Ця нормальна форма вимагає, аби в схемі не було інших обмежень окрім ключів та доменів.

**2.3 Визначення типів даних**

Тип даних — характеристика, яку [явно](https://uk.wikipedia.org/w/index.php?title=%D0%AF%D0%B2%D0%BD%D0%B0_%D1%82%D0%B8%D0%BF%D1%96%D0%B7%D0%B0%D1%86%D1%96%D1%8F&action=edit&redlink=1) чи [неявно](https://uk.wikipedia.org/w/index.php?title=%D0%9D%D0%B5%D1%8F%D0%B2%D0%BD%D0%B0_%D1%82%D0%B8%D0%BF%D1%96%D0%B7%D0%B0%D1%86%D1%96%D1%8F&action=edit&redlink=1) надано об'єкту ([змінній](https://uk.wikipedia.org/wiki/%D0%97%D0%BC%D1%96%D0%BD%D0%BD%D0%B0), [функції](https://uk.wikipedia.org/wiki/%D0%A4%D1%83%D0%BD%D0%BA%D1%86%D1%96%D1%8F_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F)), полю запису, [константі](https://uk.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BD%D1%81%D1%82%D0%B0%D0%BD%D1%82%D0%B0), [масиву](https://uk.wikipedia.org/wiki/%D0%9C%D0%B0%D1%81%D0%B8%D0%B2_(%D1%81%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85)) тощо). Тип даних визначає множину припустимих значень, формат їхнього збереження, розмір виділеної пам'яті та набір операцій, які можна робити над даними. На рисунку 2.1. зображено взаємозв’язок типів даних.



**Діапазони числових типів даних**

Кожний числовий тип даних має мінімальне та максимальне значення, яке називають діапазон значень. Важливо знати діапазон значень, особливо, коли працюєш з «маленькими» типами даних, оскільки у них можна зберігати лише значення у вузькому діапазоні. Спроба внести число, більше за доступний діапазон може призвести до помилок періоду компіляції/виконання, або до неправильних підрахунків (через відкидання) залежно від мови програмування, яка використовується.

**2.4 Реалізація SQL скрипту**

### **2.4.1 Опис предметної області**

Практичним завданням курсової роботи є розробка бази даних для інформаційної системи магазину про продажу автомобілів

Метою роботи є, створення програмного продукту, який дозволить зберігати, обновляти та видаляти інформацію про продукцію магазину, адреси офісів тощо.

У ході створення інформаційної системи робиться уклін на базу даних.

У ході створення інформаційної системи було використанно MySql.

MySQL — [вільна](https://uk.wikipedia.org/wiki/%D0%92%D1%96%D0%BB%D1%8C%D0%BD%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BD%D0%B5_%D0%B7%D0%B0%D0%B1%D0%B5%D0%B7%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%BD%D1%8F) [система керування реляційними базами даних](https://uk.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D0%BA%D0%B5%D1%80%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F_%D1%80%D0%B5%D0%BB%D1%8F%D1%86%D1%96%D0%B9%D0%BD%D0%B8%D0%BC%D0%B8_%D0%B1%D0%B0%D0%B7%D0%B0%D0%BC%D0%B8_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85), яка була розроблена компанією «ТсХ» для підвищення швидкодії обробки великих баз даних. Ця система керування базами даних ([СКБД](https://uk.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D0%BA%D0%B5%D1%80%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0%D0%BC%D0%B8_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85)) з відкритим кодом була створена як альтернатива комерційним системам. MySQL з самого початку була дуже схожою на [mSQL](https://uk.wikipedia.org/w/index.php?title=MSQL&action=edit&redlink=1" \o "MSQL (ще не написана)), проте з часом вона все розширювалася і зараз MySQL — одна з найпоширеніших систем керування базами даних. Вона використовується, в першу чергу, для створення динамічних [веб-сторінок](https://uk.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D1%82%D0%BE%D1%80%D1%96%D0%BD%D0%BA%D0%B0), оскільки має чудову підтримку з боку різноманітних [мов програмування](https://uk.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B2%D0%B0_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F).

Одною із самих використовуваних систем управління базами даних MySQL з 2010 року володіє Oracle Corporation. Продукт розповсюджується під GNU General Public License (ліцензія на вільне програмне забезпечення) або під власною комерційною ліцензією.

MySQL була розроблена шведською компанією MySQL АВ. В 2008 році MySQL АВ була куплена американською компанією Sun Microsystems. Починаючи з квітня 2009 року корпорація Oracle Corporation почала процес придбання Sun Microsystems. Процес злиття двох компаній закінчився в січні 2010 року.

MySQL часто використовується в бесплатних програмних проектах, яким потрібна система управління базами даних з можливістю повнотекстового пошуку, таких як програмне забезпечення WordPress, програмне забезпечення phpBB і багато інших.

Також використовується для крупних продуктів таких як Wikipedia Google Facebook.

MySQL — компактний [багатопотоковий](https://uk.wikipedia.org/wiki/%D0%91%D0%B0%D0%B3%D0%B0%D1%82%D0%BE%D0%BF%D0%BE%D1%82%D0%BE%D0%BA%D0%BE%D0%B2%D1%96%D1%81%D1%82%D1%8C" \o "Багатопотоковість) [сервер баз даних](https://uk.wikipedia.org/w/index.php?title=%D0%A1%D0%B5%D1%80%D0%B2%D0%B5%D1%80_%D0%B1%D0%B0%D0%B7_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85&action=edit&redlink=1). Характеризується високою швидкістю, стійкістю і простотою використання.

MySQL вважається гарним рішенням для малих і середніх застосувань. [Сирцеві коди](https://uk.wikipedia.org/wiki/%D0%A1%D0%B8%D1%80%D1%86%D0%B5%D0%B2%D1%96_%D0%BA%D0%BE%D0%B4%D0%B8) [сервера](https://uk.wikipedia.org/wiki/%D0%A1%D0%B5%D1%80%D0%B2%D0%B5%D1%80) компілюються на багатьох платформах. Найповніше можливості сервера виявляються в [UNIX](https://uk.wikipedia.org/wiki/UNIX)-системах, де є підтримка [багатопоточності](https://uk.wikipedia.org/wiki/%D0%91%D0%B0%D0%B3%D0%B0%D1%82%D0%BE%D0%BF%D0%BE%D1%82%D0%BE%D0%BA%D0%BE%D0%B2%D1%96%D1%81%D1%82%D1%8C" \o "Багатопотоковість), що підвищує продуктивність системи в цілому.

Можливості сервера MySQL:

* простота у встановленні та використанні;
* підтримується необмежена кількість користувачів, що одночасно працюють із БД;
* кількість рядків у таблицях може досягати 50 млн;
* висока швидкість виконання команд;
* наявність простої і ефективної системи безпеки.

Багото веб–додатків використовують MySQL в якості компонентів програмного забезпечення. Популярність використання веб–додатків тісно повязана з популярністю РНР і можливістю його роботи разом з MySQL.

Багато сайтів з високим трафіком ( в тому числі Flickr. Facebook. Wikipedia. Google. Nokia. YouTube) використовують MySQL для збереження даних і реєстрації користувачів.

MySQL і Linux

Найбільш розпоширений спосіб установки і використання MySQL на платформі  Linux, хоча база даних працює і на інших платформах, наприклад Microsoft Windows.

MySQL працює на багатьох платформах різних систем, включає AIX. BSDI. FreeBSD. HP-UX. i5/OS. Linux. Mac OS X. NetBSD. Nowel NetWare. OpenBSD. OpenSolaris. eComStation. OS/2 Warp. QNX. IRIX. Solaris. Symbian. SunOS. SCO OpenSerwer. SCO UnixWare. Sanos. Tru64. Microsoft Windows.

Існує також порт MySQL для OpenVMS.

Всі мови програмування, в яких присутній API (application programming interface — інтерфейс прикладного програмування), включають бібліотеки для доступу до баз даних MySQL.

Крім того інтерфейс ODBC (Open Database Connectivity — програмний інтерфейс для доступу до баз даних) який називають MyODBC, дозволяє додатковим мовам програмуванням таким як ASP або ColdFusion в цілях підтримки інтерфейса ODBC взаємодіяти з базою даних MySQL.

Сервер MySQL і бібліотеки являються більшою частиною додатками до ANSI C/ANCI C++

Для керування базою даних MySQL можна використовувати інструмент командного рядка (команди MySQL  і mysqladmin).  Користувачі можуть скачати з сайта MySQL інструменти адміністрування

GUI (Graphical user interfase– графічний користувацький інтерфейс) MySQL Administrator MySQL Migration Toolkit i MySQL Query Browser. Інструменти GUI обєднані зараз в один пакет під назвою MySQL GUI Tools.

Додам до цієї публікації ще один важливий елемент, це веб– інтерфейс для адміністрування управління базами даних MySQL написаний на мові РНР вільний веб– додаток з відкритим кодом phpMyAdmin.

З  допомогою phpMyAdmin через браузер користувача можно здійснювати адміністрацію сервера., запускати команди SQL і передивлятись вміст таблиці і даних.

Додаток завоював свою популярність у веб програмістів в звязку з тим що надав можливість керувати SQL без посередньо без введення SQL команд шляхом доступу до інтерфейсу.

**2.4.2 Розробка інформаційної системи**

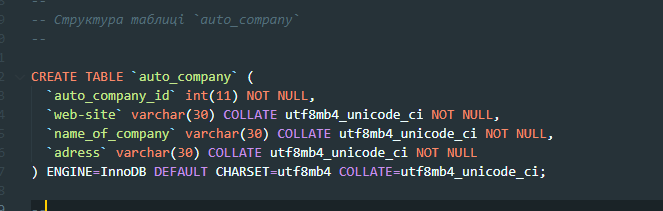
Під час розробки інформаційної системи для магазину автомобілів було створено такі таблиці:

* auto\_company;
* services;
* ordering;
* office;
* bill;

При створенні таблиці auto\_company було задано такі поля:

* id типу int
* web\_site varchar
* name\_of\_company varchar
* address varchar

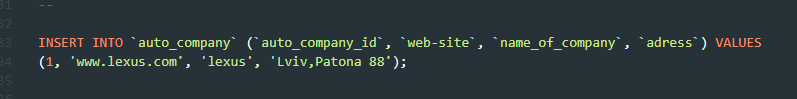
*лістінинг 2.4.1*



Лістінинг 2.4.1 структура таблиці auto\_company

Одразу після створення таблиці та оголошення полів реалізовано наповнення таблиці

*лістінинг 2.4.2*

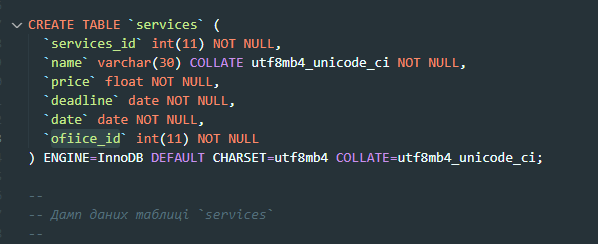


Лістінинг 2.4.2 наповнення таблиці auto\_company\

При створенні таблиці services ми задаємо такі поля:

* services\_id типу int
* name varchar
* price float
* date date
* ofiice\_id int

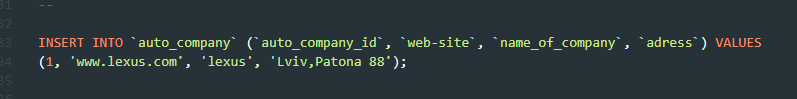
*лістінинг 2.4.3*



Лістінинг 2.4.3 структура таблиці services

Одразу після створення таблиці та оголошення полів реалізовано наповнення таблиці

*лістінинг 2.4.4*

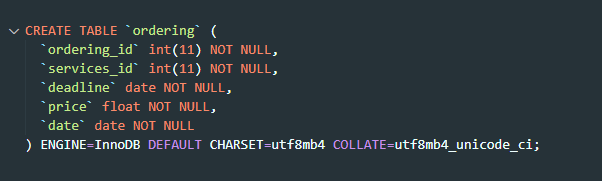


Лістінинг 2.4.4 наповнення таблиці services

При створенні таблиці ordering ми задаємо такі поля:

* ordering\_id типу int
* services\_id int
* deadline date
* price float
* date date

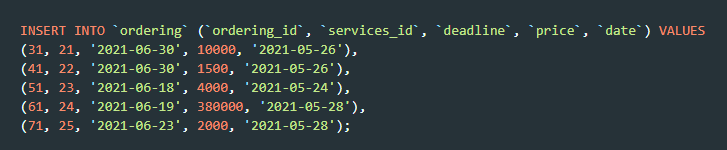
*лістінинг 2.4.5*



Лістінинг 2.4.5 структура таблиці ordering

Одразу після створення таблиці та оголошення полів реалізовано наповнення таблиці

*лістінинг 2.4.6*

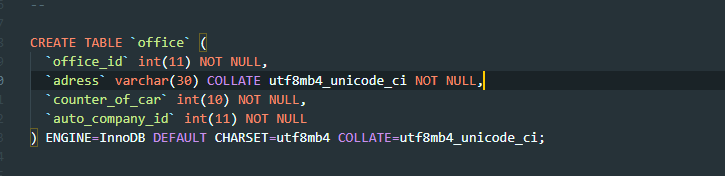


Лістінинг 2.4.6 наповнення таблиці ordering

При створенні таблиці office ми задаємо такі поля:

* office\_id типу int
* adress varchar
* counter\_of\_car int
* auto\_company\_id int

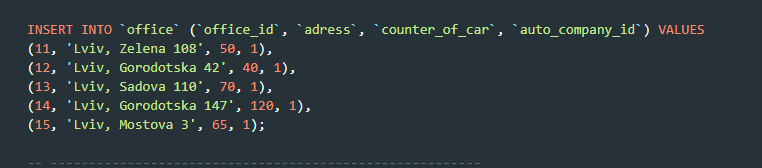
*лістінинг 2.4.7*



Лістінинг 2.4.7 структура таблиці office

Одразу після створення таблиці та оголошення полів реалізовано наповнення таблиці

*лістінинг 2.4.8*

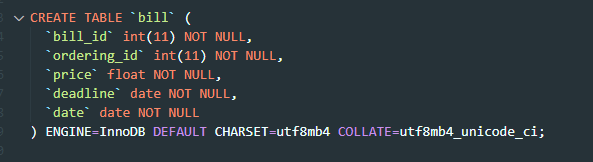


Лістінинг 2.4.8 наповнення таблиці office

При створенні таблиці bill ми задаємо такі поля:

* bill\_id типу int
* ordering\_id типу int
* price float
* deadline типу date
* date date

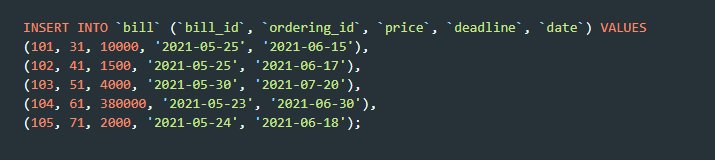
*лістінинг 2.4.9*



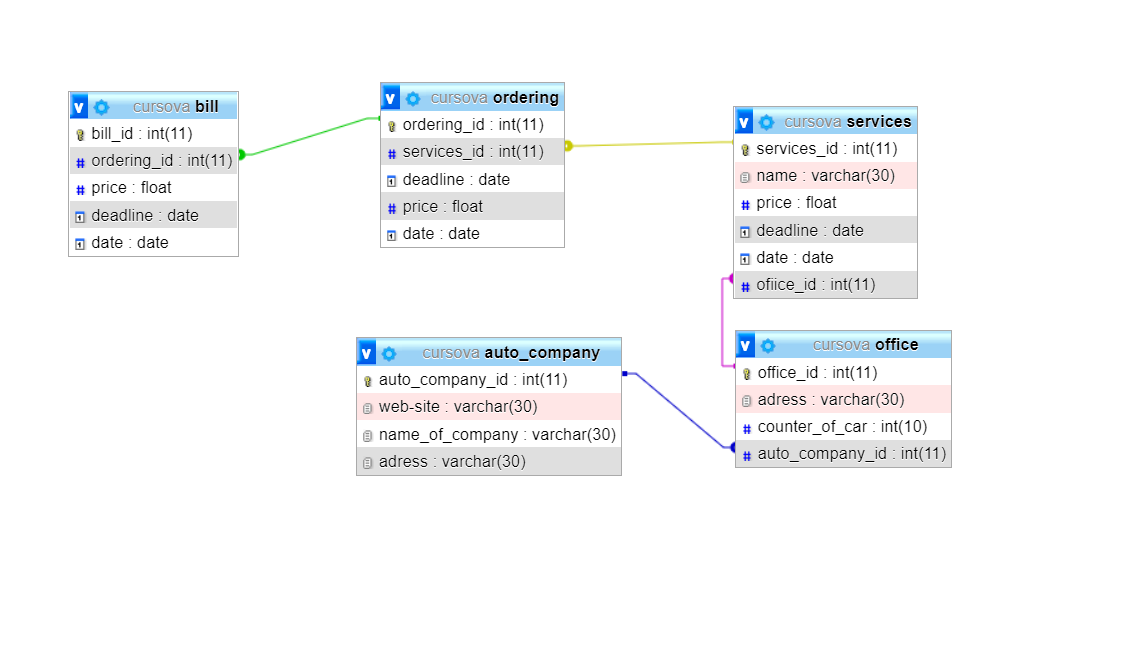
Лістінинг 2.4.9 структура таблиці bill

Одразу після створення таблиці та оголошення полів реалізовано наповнення таблиці

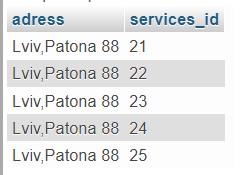
*лістінинг 2.4.10*



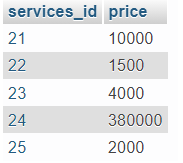
Лістінинг 2.4.10 наповнення таблиці bill



Лістинг 2.4.11візуалізовані таблиці бази даних



Лістинг 2.4.12.1 Запити користувача



Лістинг 2.4.12.2 Запити користувача

**ВИСНОВОК**

На сьогоднішній день майже кожна людиа використовує веб-сайти чи веб-додатки для своїх потреб і кількість таких людей зростає щороку. Цей прогресс відбувся доволі швидко, так, якщо проаналізувати розвиток інтернету за останіх 10-15 років то можна помітити, як різко розвивались веб технології та скільки різних можливостей добавилось з плином часу. Тому на перший погляд мережа Internet подібна до звичайних глобальних мереж. Але насправді вона має ряд абсолютно специфічних особливостей і унікальну структуру.

# Під час курсового проектування було спроектовано інформаційну систему “Автосалону”. За допомогою якої можна отримати інформацію про автомобілі та клієнтів. Вдосконалив свої навички в роботі з MySQL Workbranch. В результаті проектування була розроблена БД для автосалону, призначена для того, щоб полегшити працю персоналу, часу, який витрачається на паперову роботу. АІС була створена в програмі MySQL Workbranch, яка є однією з найпопулярніших серед настільних програмних систем управління базами даних.

# В даному курсовому проекті було проведено аналіз методів автоматизації обробки облікової інформації і, як результат, було розроблено БД для швидкого пошуку інформації про авто, замовлення та клієнтів, яка відповідає основним вимогам до сучасних програмних розробок, для забезпечення автоматизації роботи, зручний інтерфейс. Це дає нам право говорити про конкурентоспроможність розробленої програми і можливість практичного її використання для вирішення реальних завдань.

Проаналізувавши вище зазначену інформацію, можна зробити висновки що у моїй курсовій роботі:

1. Було проаналізовано предметну область для розроблюваного продукту. Досліджено структуру та напрями діяльності об’єкту управління.Також були проаналізовані усі бізнес-процеси, які відбуваються у даному об’єкті управління.

2. Спроектовано та описано структуру БД, її таблиці та записи. Визначено типи даних та функціональні залежності. Розроблено запити та визначено атрибути полів. Для кращого розуміння архітектури продукту було розроблено діаграми, які показують взаємодію користувачів та їх функцій між собою та у системі.

**СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ**

1. Переваги MySQl [Електронний ресурс]. – Режим доступу: https://studfile.net/preview/560354/page:3/ .
2. Типи даних MySQL [Електронний ресурс]. – Режим доступу: [https://uk.wikibooks.org/wiki/SQL/%D0%A2%D0%B8%D0%BF%D0%B8\_%D0%B4%D0%B0%D0%BD%D0%B8%D%85\_MySQL](https://uk.wikibooks.org/wiki/SQL/%D0%A2%D0%B8%D0%BF%D0%B8_%D0%B4%D0%B0%D0%BD%D0%B8%25D%85_MySQL).
3. НОРМАЛІЗАЦІЯ ВІДНОШЕНЬ [Електронний ресурс]. – Режим доступу: <https://life-prog.ru/ukr/1_331_normalizatsiya-vidnoshen.html>
4. Переваги PhP [Електронний ресурс]. – Режим доступу: https://blog.ithillel.ua/ua/articles/chomu-vybyraiut-7-perevah-movyprohramuvannia-php.
5. Моделі даних [Електронний ресурс]. – Режим доступу: https://uk.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB% D1%96\_%D0%B1%D0%B0%D0%B7\_%D0%B4%D0%B0%D0%BD%D0%B8 %D1%85.
6. НОРМАЛІЗАЦІЯ БАЗ ДАНИХ — ВІКІПЕДІЯ [Електронний ресурс]. – Режим доступу: <https://cutt.ly/tyTCqT8>
7. Мартин Грабер :SQL; Видавництво «Лори», 2016. -643 с.
8. Типи даних MySQL [Електронний ресурс]. – Режим доступу: [https://uk.wikibooks.org/wiki/SQL/%D0%A2%D0%B8%D0%BF%D0%B8\_%D0%B4%D0%B0%D0%BD%D0%B8%D%85\_MySQL](https://uk.wikibooks.org/wiki/SQL/%D0%A2%D0%B8%D0%BF%D0%B8_%D0%B4%D0%B0%D0%BD%D0%B8%25D%85_MySQL).
9. НОРМАЛІЗАЦІЯ ВІДНОШЕНЬ [Електронний ресурс]. – Режим доступу: <https://life-prog.ru/ukr/1_331_normalizatsiya-vidnoshen.html>
10. НОРМАЛІЗАЦІЯ БАЗ ДАНИХ — ВІКІПЕДІЯ [Електронний ресурс]. – Режим доступу: <https://cutt.ly/tyTCqT8>
11. HTML ACADEMY: ИНТЕРАКТИВНЫЕ ОНЛАЙН-КУРСЫ ПО HTML, CSS И JAVASCRIPT [Електронний ресурс] – Режим доступу: <https://htmlacademy.ru/>