

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

ФАКУЛЬТЕТ БІОМЕДИЧНОЇ ІНЖЕНЕРІЇ
(повна назва інституту/факультету)

кафедра БІОМЕДИЧНОЇ КІБЕРНЕТИКИ
(повна назва кафедри)

КУРСОВА РОБОТА

з дисципліни (кредитного модуля) **Системи баз даних**

спеціальність **122 Комп'ютерні науки**

спеціалізація **Комп'ютерні технології в біології та медицині**

На тему **Розробка програмного застосунку для управління
туристичним агентством**

(тема індивідуального завдання)

**Виконала студентка 3-го курсу гр. БС-01
ЯРОВИХ ДІАНА ОЛЕКСАНДРІВНА**

*Засвідчую, що у роботі немає запозичень з праць
інших авторів без відповідних посилань.*

Студент (-ка) _____

Перевірів ст.викл. Віталій СЕРДАКОВСЬКИЙ

Бали за роботу студента відповідно до РСО за

Оформлення роботи (із 18 балів)	
Обґрунтування (із 12 балів)	
Захист роботи (із 60 балів)	
ВСЬОГО	

(підпис викладачів)

Київ – 2022р.

Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Інститут (факультет) БІОМЕДИЧНОЇ ІНЖЕНЕРІЇ
(повна назва)

Кафедра БІОМЕДИЧНОЇ КІБЕРНЕТИКИ
(повна назва)

ЗАВДАННЯ
на курсову роботу студенту

ЯРОВИХ ДІАНИ ОЛЕКСАНДРІВНІ

(прізвище, ім'я, по батькові)

1. Тема роботи Розробка програмного застосунку для управління туристичним агентством

2. Термін подання студентом роботи 26 грудня 2022 року

3. Вихідні дані до роботи Індивідуальне завдання студента до лабораторних робіт.

4. Зміст роботи 1. Концептуальний рівень (визначення необхідних знань для реалізації) 2. Логічний рівень (аналітичний аналіз курсової роботи бізнес процесів, побудова діаграми «сутність-зв'язок»). 3 Фізичний рівень (реалізація розробки бази даних).

5. Дата видачі завдання 12 вересня 2022 р.

Календарний план

№ з/п	Назва етапів виконання курсової роботи	Термін виконання етапів роботи	Примітка
1	Отримати завдання на КР	12 вересня 2022р..	
2	Оформлення розділу 1	23 жовтня 2022р.	
3	Оформлення розділу 2	13 листопада 2022р.	
4	Оформлення розділу 3	4 грудня 2022р.	
5	Оформлення КР	19 грудня 2022р.	
6	Подання в електронному вигляді КР та анотації до неї на сайт кафедри.	До 26 грудня 2022р	
7	Подання пакету документів по КР до захисту	26 грудня 2022р.	
8	Захист КР	10 січня 2023р	

Студент

(підпис)

Діана ЯРОВИХ

(ініціали, прізвище)

Керівник роботи

(підпис)

Віталій СЕРДАКОВСЬКИЙ

(ініціали, прізвище)

Анотація

«Курсова робота (Курсовий проект)» з дисципліни «Системи баз даних» являється частиною циклу **загальної підготовки** дисциплін першого (бакалаврського) рівня вищої освіти ступеня «**бакалавр**», зі спеціальності 122 «Комп'ютерні науки» за спеціалізацією «Комп'ютерні технології в біології та медицині»).

Загальна трудомісткість освоєння модуля становить 1 кредит (ЕКТС), 30 годин.

Курсову роботу виконала **Ярових Діана Олександрівна** студентка 3 курсу, гр. БС-01 кафедри Біомедичної кібернетики факультету Біомедичної інженерії НТУУ «КПІ ім. Ігоря Сікорського» .

Тема роботи: Розробка програмного застосунку для управління туристичним агентством.

Питання, розглянуті в курсовій роботі: створення інформаційно-логічної моделі даних, проектування фізичної моделі бази даних, розробка бази даних, проектування запитів.

Отримані результати курсової роботи: було отримано навички створення діаграм концептуального та логічного етапів проектування, створення таблиць, створення запитів для отримання необхідних даних, змінення даних, створення представлень, використання індексів для підвищення продуктивності виконання запитів. Наведені приклади створеної бази даних використання для вирішення поставлених задач.

Структура і обсяг роботи: курсова робота складається із вступу, трьох розділів, висновків, списку використаної літератури та додатків. Загальний обсяг курсової роботи становить 61 сторінку, основного тексту (без додатків) – 48 сторінок, ілюстрацій – 22, таблиць – 1.

Abstract

"Coursework (Course project)" in the discipline "Database Systems" is part of the cycle of general training of disciplines of the first (bachelor's) level of higher education of the degree "Bachelor", specialty 122 "Computer Science", specialization "Computer Technologies in Biology and Medicine").

The total labor intensity of the module is 1 credit (ECTS), 30 hours.

The course work was performed by Yarovykh Diana Oleksandrivna, 3rd year student, gr. BS-01 Department of Biomedical Cybernetics, Faculty of Biomedical Engineering, NTUU "KPI".

Theme of work: Development of software application for travel agency management.

Issues considered in the course work: creating an information-logical data model, designing a physical database model, database development, query design.

The results of the course work: the skills of creating diagrams of the conceptual and logical stages of design, creating tables, creating queries to obtain the necessary data, modifying data, creating views, using indexes to improve the performance of queries were obtained. Examples of the created database are used to solve the tasks.

Structure and scope of work: the term paper consists of an introduction, three chapters, conclusions, references and appendices. The total volume of the course work is 61 pages, the main text (without appendices) - 48 pages, illustrations - 22, tables - 1.

ЗМІСТ

СПИСОК СКОРОЧЕНЬ.....	7
ВСТУП	8
РОЗДІЛ 1 ТЕОРЕТИЧНА ЧАСТИНА	9
1.1 Поняття бази даних	9
1.2 Система управління базами даних.....	10
1.3 Реляційні бази даних	11
1.4 SQL.....	12
1.5 Концептуальне проектування	12
1.6 Нормалізація та нормальні форми.....	13
1.7 Представлення (view)	14
1.8 Тригери (triggers).....	15
1.9 Індеси (index)	15
Висновки до розділу 1	17
РОЗДІЛ 2 АНАЛІТИЧНА ЧАСТИНА	18
2.1 Актуальність роботи	18
2.2 Бізнес-процеси.....	18
2.3 Перелік сутностей та їх атрибутів:.....	22
2.4 Діаграма «сутність-зв'язок».....	24
2.5 Функціональні залежності	25
2.6 Характеристика елементів даних.....	27
UML діаграма класів	29
Висновки до розділу 2	29
РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА	30
3.1 Обмеження предметної області.....	30
3.2 Створення таблиць.....	31
Команди для створення коментарів до таблиць та до стовпців:	32
3.3 Автоматично змінюване представлення.....	32

Приклад CREATE VIEW:.....	32
Приклад INSERT:	35
Приклад UPDATE:.....	35
Приклад DELETE:	36
Представлення з використанням функцій агрегації та клаузи HAVING	37
Представлення із зовнішнім з'єднанням	37
Тригер INSTEAD OF для додавання нового рядку	38
3.4 Тригери для таблиць.....	39
Тригери AFTER	39
Тригери BEFORE.....	40
Тригери для представлень.....	40
3.5 Індeksi.....	41
Виконання запитів без використання індeksiв.....	41
Створення індeksiв	43
Виконання запитів з індeksiами	43
Висновки до розділу 3	45
ВИСНОВКИ	46
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	47
Додаток А Оператори DDL	49

СПИСОК СКОРОЧЕНЬ

SQL – structured query language

БД – база даних

ІС – інформаційна система

ПЗ – програмний застосунок

Рис. – рисунок

Табл. – таблиця

СУБД – система управління базами даних

ФЗ – функціональна залежність

продовж. – продовження

ВСТУП

В сучасних умовах нашого життя все більше й більше набувають значення інформаційні системи, що дозволяють забезпечити інформаційну підтримку процесів прийняття рішень. В більшості інформаційних систем одним із ключових елементів є саме бази даних.

Кожна людина зараз хоче подорожувати задля відпочинку та підвищення рівня культурного насичення, тому постає питання у зручності використання послуг туристичних агентств. Саме програмний застосунок може вирішити це питання, адже він має бути зручним для використання для обох сторін.

Зараз туристичне агентство виступає в ролі посередника між клієнтами та туроператорами, що пропонують свій туристичний продукт. Завдання програмного застосунку це спростити зв'язок між туристами та туристичними операторами. Основна мета – це простота, доступність та легкість моніторингу інформації про певні продукти та її порівняння чи придбання.

Невід'ємною частиною програмного застосунку є бази даних, що вміщують в собі різну інформацію про туристичні продукти, що пропонують туроператори, та полегшують бронювання на клієнта певної подорожі.

Актуальність створення бази даних, як частини програмного застосунку досить прозора – це можливість використання додатку людьми, що не мають високий рівень знань у сферах інформаційних технологій, забезпечення швидкого доступу до інформації, що зберігається в базі даних, можливість спільного використання бази даних декількома користувачами, можливість побудови звітів на запит користувача.

Предметною областю даної курсової роботи є розробка бази даних для застосунку управління туристичним агентством.

РОЗДІЛ 1

ТЕОРЕТИЧНА ЧАСТИНА

1.1 Поняття бази даних

База даних — це певний набір даних, які пов'язані між собою спільною ознакою або властивістю, та впорядковані, наприклад, за алфавітом.

Об'єднання великої кількості даних в єдину базу дає змогу для формування безлічі варіації групування інформації — особисті дані клієнта, історія замовлень, каталог товарів та будь-що інше.

Головною перевагою БД є швидкість внесення та використання потрібної інформації. Завдяки спеціальним алгоритмам, які використовуються для баз даних, можна легко знаходити необхідні дані всього за декілька секунд. Також в базі даних існує певний взаємозв'язок інформації: зміна в одному рядку може спричинити зміни в інших рядках — це допомагає працювати з інформацією простіше і швидше.

Класифікація БД за моделлю даних:

- ієрархічні,
- мережеві,
- реляційні,
- об'єктні,
- об'єктно-орієнтовані,
- об'єктно-реляційні.

Класифікація БД за технологією фізичного зберігання:

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

1.2 Система управління базами даних

Система управління базами даних (СУБД) - це комплекс програмно-мовних засобів, що дозволяють створити БД та керувати даними. Іншими словами, СУБД – це набір програм, що дозволяє організовувати, контролювати та адмініструвати БД. Більшість сайтів не можуть працювати без БД, тому СУБД використовується практично повсюдно.

Основні функції СУБД:

- керування даними у зовнішній пам'яті (на дисках);
- керування даними в оперативній пам'яті з використанням дискового кешу;
- журналізація змін (збереження історії), резервне копіювання та відновлення БД після збоїв;
- підтримка мов БД (мова визначення даних, мова маніпулювання даними).

Кожна СУБД ґрунтується на будь-якій моделі даних, це одна з ознак класифікації. За моделлю даних СУБД бувають:

- Ієрархічні. У цій моделі даних використовується уявлення БД як деревоподібної структури, що складається з даних різних рівнів.
- Мережеві. Ця модель є розширенням ієрархічного підходу. Ієрархічна модель передбачає, що запис-нащадок може мати строго одного предка, тоді як у мережевий структурі нащадок може мати будь-яку кількість предків.
- Реляційні. СУБД, орієнтовані на організацію даних як набір пов'язаних записів та атрибутів у двовимірній таблиці.
- Об'єктно-орієнтовані. Для управління БД, що базуються на об'єктній моделі даних. Як правило ґрунтуються на об'єктно-орієнтованих мовах програмування.

- Об'єктно-реляційні. Поєднує у собі концепції реляційної моделі з додатковими об'єктно-орієнтованими можливостями

1.3 Реляційні бази даних

Реляційна база даних - це база даних, побудована на основі реляційної моделі, тобто БД, що має табличний спосіб вистави даних, а на зовнішньому рівні, що задається набором однорідних таблиць. Кожний об'єкт записується рядком у таблиці. Рядок називається записом. Запис складається з полів різного типу.

Реляційна база даних створюється й потім управляється за допомогою спеціальних засобів — реляційних систем керування базами даних (РСУБД).

Коротко особливості реляційної бази даних можна сформулювати наступним чином:

- Дані зберігаються в таблицях, що складаються із стовпців і рядків;
- На перетині кожного стовпця і рядка стоїть одне значення;
- У кожного стовпця є своє ім'я, яке служить його назвою, і всі значення в одному стовпці мають один тип.
- Стовпці розташовуються в певному порядку, який визначається при створенні таблиці, на відміну від рядків, які розташовуються в довільному порядку. У таблиці може не бути жодного рядка, але обов'язково повинен бути хоча б один стовпець;
- Запити до бази даних повертають результат у вигляді таблиць, які теж можуть виступати як об'єкт запитів.
- Рядки в реляційній базі даних невпорядковані - впорядковування проводиться в момент формування відповіді на запит.
- Основним інструментом роботи з реляційними базами даних, є мова SQL

1.4 SQL

Мова структурованих запитів — це універсальна мова для створення, модифікації та управління інформацією, яка входить до складу реляційних баз даних.

Переваги SQL:

- Незалежність від існуючої в даній системі СУБД. Тексти SQL є універсальними для багатьох СУБД.
- Наявність стандартів SQL сприяє «стабілізації» мови.
- Декларативність. Ця перевага полягає в тому, що при роботі з даними, програміст вибирає тільки ту інформацію, яка повинна бути змінена або модифікована. Те, яким чином це буде зроблено, в автоматичному режимі вирішується на програмному рівні самої СУБД.

1.5 Концептуальне проектування

Концептуальне проектування технічних систем — початкова стадія проектування, де приймаються визначальні наступне вигляд рішення, і проводиться дослідження та узгодження параметрів створених технічних рішень із можливою їх організацією. Термін «концепція» застосовується для опису принципу дії у технічних системах, а й у наукових, художніх та інших видів діяльності. "Концепт" (лат.) - зміст поняття, сенс. Таким чином, проектування на концептуальному рівні — на рівні сенсу чи змісту поняття систем.

Основний обсяг завдань концептуального проектування відноситься до ранніх стадій розробки технічних систем (ТС): при постановці задачі на проектування, виробленні масиву варіантів технічних та оформлювальних рішень та в ескізному проектуванні при створенні технічного завдання.

1.6 Нормалізація та нормальні форми

Нормалізація схеми бази даних — покроковий процес розбиття одного відношення або таблиці відповідно до алгоритму нормалізації на декілька відношень на базі функціональних залежностей.

Мета нормалізації — отримати таку структуру даних, яка уникає надмірності, тобто кожна частина інформації з'являється лише один раз в логічно вірному місці. Нормалізація призначена не для збільшення ефективності або для зменшення фізичного об'єму БД, її ціль отримати мінімальну логічну надмірність.

Нормальна форма — це сукупність вимог, яким має задовольняти відношення (таблиця) з метою усунення надмірних функціональних залежностей між атрибутами.

Існують наступні нормальні форми:

- Перша нормальна форма
- Друга нормальна форма
- Третя нормальна форма
- Четверта нормальна форма
- П'ята нормальна форма
- Шоста нормальна форма
- Нормальна форма Бойса-Кодда
- Нормальна форма «домен-ключ»

Перша нормальна форма (1NF) - це сутність, що не має повторюваних стовпців, тобто вони є унікальними.

Друга нормальна форма (2NF) - це сутність, що дорівнює першій нормальній формі, та один з її атрибутів визначається як первинний ключ, від якого залежить решта атрибутів.

Третя нормальна форма (3NF) – це таблиця, яка є у другій нормальній формі, та стовпці таблиці залежать від первинного ключа.

Четверта нормальна форма (4NF) - це таблиця, яка не містить двох або більше незалежних даних, що описують відповідну сутність.

П'ята звичайна форма (5NF) – це таблиця, яка відповідає умовам 4NF і може бути розбита на кілька таблиць без втрати будь-яких даних.

Шоста нормальна форма (6NF) – це таблиця, що знаходиться у 5NF та задовольняє вимозі відсутності нетривіальних залежностей.

Нормальна форма Бойса-Кодда – це відношення, в якому детермінант кожної функціональної залежності є потенційним ключем.

Нормальна форма «домен-ключ» - нормальна форма, вимагає, аби в схемі не було інших обмежень окрім ключів та доменів.

1.7 Представлення (view)

Представлення (view) бази даних є підмножиною бази даних, заснованою на запиті, який виконується в одній або кількох таблицях бази даних. Представлення зберігаються в базі даних як іменовані запити і можуть використовуватися для збереження часто використовуваних складних запитів.

Існує два типи представлень бази даних: динамічні та статичні. Динамічні представлення можуть містити дані з однієї або двох таблиць і автоматично включати всі стовпці з зазначеної таблиці або таблиць. Динамічні представлення оновлюються автоматично, коли створюються або змінюються пов'язані об'єкти або розширені об'єкти. Статичні представлення можуть містити дані з кількох таблиць, і необхідні стовпці з цих таблиць мають бути вказані в SELECT і WHERE статичного подання. Статичні представлення необхідно оновлювати вручну, коли створюються або змінюються пов'язані об'єкти або розширені об'єкти.

Коли створюється динамічне представлення даних з двох таблиць, обидві таблиці повинні мати однакові стовпці `primarykeycolseq` або містити унікальні індекси з однаковою назвою стовпця в тому самому порядку.

Представлення бази даних заповнюються залежно від об'єкта, на якому вони засновані.

1.8 Тригери (triggers)

Тригер (trigger) — це процедурний код, який автоматично виконується у відповідь на певні події в певній таблиці або представленні в базі даних. Тригер в основному використовується для підтримки цілісності інформації в базі даних.

Код, описаний всередині тригеру викликається після, або перед дією, яка модифікує значення стовпців (INSERT, DELETE, UPDATE) відповідно до написаного тригеру AFTER (код всередині тригеру викликається до виконання операції модифікації) або BEFORE (код всередині тригеру викликається після виконання операції модифікації).

1.9 Індекси (index)

Індекси є потужним інструментом, який використовується у фоновому режимі бази даних для прискорення запитів. Індекс надає метод швидкого пошуку запитуваних даних.

Простіше кажучи, індекс - це вказівник на дані в таблиці. Показчик у базі даних дуже схожий на показчик у кінці книги.

Більшість програмного забезпечення для баз даних містить технологію індексування, яка дозволяє підлінійний пошук часу для підвищення продуктивності, оскільки лінійний пошук неефективний для великих баз даних.

Припустимо, база даних містить N елементів даних, і один потрібно отримати на основі значення одного з полів. Проста реалізація отримує та перевіряє кожен елемент відповідно до тесту. Якщо є лише один відповідний елемент, це може зупинитися, коли він знайде цей один елемент, але якщо є кілька збігів, він повинен перевірити все. Це означає, що кількість операцій у середньому випадку дорівнює $O(N)$ або лінійному часу. Оскільки бази даних можуть містити багато об'єктів, а пошук є звичайною операцією, часто бажано підвищити продуктивність.

Індекс — це будь-яка структура даних, яка покращує продуктивність пошуку. Для цього використовується багато різних структур даних. Існують складні компроміси щодо дизайну, що включають продуктивність пошуку, розмір індексу та продуктивність оновлення індексу. Багато дизайнів індексів демонструють логарифмічну ($O(\log(N))$) продуктивність пошуку, а в деяких програмах можна досягти рівної ($O(1)$) продуктивності.

Індекси використовуються для контролю обмежень бази даних, таких як `UNIQUE`, `EXCLUSION`, `PRIMARY KEY` і `FOREIGN KEY`. Індекс може бути оголошений як `UNIQUE`, що створює неявне обмеження для базової таблиці. Системи баз даних зазвичай неявно створюють індекс для набору стовпців, оголошених `PRIMARY KEY`, і деякі можуть використовувати вже існуючий індекс для контролю за цим обмеженням. Багато систем баз даних вимагають, щоб як посилаючі, так і посилаючі набори стовпців у обмеженні `FOREIGN KEY` були проіндексовані, таким чином покращуючи продуктивність вставок, оновлень і видалення таблиць, які беруть участь у обмеженні.

Деякі системи баз даних підтримують обмеження `EXCLUSION`, яке гарантує, що для щойно вставленого або оновленого запису певний предикат не зберігається для іншого запису. Це можна використовувати для реалізації обмеження `UNIQUE` (з предикатом рівності) або більш складних обмежень, наприклад, для забезпечення того, щоб у таблиці не зберігалися часові діапазони,

що перетинаються, або об'єкти геометрії, що перетинаються. Індекс, що підтримує швидкий пошук записів, що задовольняють предикат, потрібен для контролю такого обмеження.

Висновки до розділу 1

В результаті підготовки першого розділу було розглянуто та проаналізовано базову теоретичну інформацію, що пов'язана з подальшою розробкою програмного застосунку та бази даних до неї. Також було засвоєно знання, що знадобляться для проєктування бази даних за обраною тематикою.

РОЗДІЛ 2

АНАЛІТИЧНА ЧАСТИНА

2.1 Актуальність роботи

Зараз туристичне агентство виступає в ролі посередника між клієнтами та туроператорами, що пропонують свій туристичний продукт. Завдання програмного застосунку спростити зв'язок між туристами та туристичними операторами. Основна мета – це простота, доступність та легкість моніторингу інформації про певні продукти та її порівняння чи придбання.

Актуальність створення бази даних, як частини програмного застосунку досить прозора – це можливість використання додатку людьми, що не мають високий рівень знань у сферах інформаційних технологій, забезпечення швидкого доступу до інформації, що зберігається в базі даних, можливість спільного використання бази даних декількома користувачами, можливість побудови звітів на запит користувача.

2.2 Бізнес-процеси

Бізнес-процес – опис кроків користувача для досягнення визначених дій.

Моделювання бізнес-процесу – процес відображення суб'єктивного бачення потоку робіт у вигляді формальної моделі, що складається з взаємопов'язаних операцій.

Прецеденти – це технологія визначення функціональних вимог до системи, опис взаємодії користувача та системи.

Перелік назв прецедентів використання до даної БД:

- ❖ Реєстрація користувача
- ❖ Перегляд актуальної туристичної продукції

❖ Придбання туристичної продукції

Перелік бізнес-процесів до ПЗ зі стислим описом:

Прецедент: Реєстрація користувача

Користувач – людина

Мета: Створити аккаунт для використання програмного додатку

Завдання: Надати дані для створення особистого аккаунту

Користувач вводить особисті дані для реєстрації у додатку, що значно спрощує далі покупку і пошук певної туристичної продукції. Також користувач зв'язує свій аккаунт з поштою щоб отримувати розсилки про знижки або оновлення у застосунку. Я вирішила, що в даному випадку пошта виступає логіном, а пароль користувач придумує самостійно.

Прецедент: Перегляд актуальної туристичної продукції

Користувач – людина

Мета: Переглянути актуальну інформацію про наявність туристичної продукції

Завдання: Задати фільтри відображення інформації для зручного пошуку актуальної інформації про туристичні продукти (місце, термін, дата, готель, ціна тощо)

Користувач може переглядати наявну туристичну продукцію без будь-якої вибірки або застосувати фільтри (місце поїздки, термін перебування, дата відправлення, назва бажаного готелю та його привілеї, ціна подорожі тощо) для відображення відсортованої інформації про певну продукцію. Також можна обрати параметр відображення (за ціною – від меншої до більшої або від більшої до меншої, за попитом)

Прецедент: Перегляд детальної інформації про подорож

Користувач – людина

Мета: Дізнатися детальну інформацію обраного продукту

Завдання: Обрати певну продукцію із запропонованих та переглянути детальну інформацію про неї

Користувач переглядає детальну інформацію про обрану продукцію, тобто місце (країна, місто, адреса) подорожі, дату виїзду (за наявності), ціну (з окремими розрахунками на дорогу), готель, додаткові заохочення (додаткові відвідування пам'яток, парків, театрів тощо, ресторани чи розважальні програми)

Прецедент: Придбання туристичної продукції

Користувач – людина

Мета: Оформити покупку обраної продукції

Завдання: Обрати певну продукцію із запропонованих та оплатити її.

Користувач може купити обрану подорож, що має встановлену дату або замовити на свою дату. Для покупки продукції користувач має бути зарезервований у системі, де введені його реальні дані, задля бронювання готелю та транспорту, що здійснює перевезення на вказане ім'я.

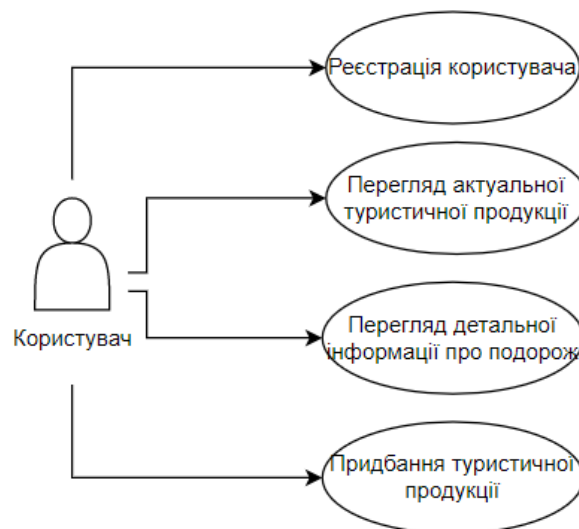


Рис. 1 Діаграма прецедентів

Проведення морфологічного аналізу всіх іменників та дієслів:

Іменники та їх аналіз:

Туристичне агентство (турагентство) – сутність, посередник між туроператором та клієнтом

Туристичний оператор (туроператор) – сутність, надає туристичну продукцію, взаємодіє з клієнтами через туристичне агентство

Туристична продукція – сутність, комплекс туристичних послуг, що необхідні для задоволення потреб клієнта. Частіше всього вміщують в собі послуги перевезення, розміщення, тощо та мають встановлену ціну.

Готель – сутність, що надає послуги проживання на деякий термін.

Клієнт – синонім до слова «користувач»

Аккаунт – атрибут користувача

Наявність – атрибут, існування будь-чого, присутність

Фільтр відображення – атрибут, дає змогу відображати інформацію за заданим відсортуванням

Параметр відображення – атрибут, дає змогу відображати відсортовану інформацію у певному порядку

Термін – атрибут, відрізок часу, протягом якого щось відбувається

Покупка – сутність, що відображає замовлення та оплату певної продукції

Дієслова та їх аналіз:

Користувач створює аккаунт

Користувач вводить дані

Користувач переглядає продукцію

Користувач застосовує фільтри та параметри відображення

Користувач переглядає детальну інформацію про продукцію

Користувач купує певну подорож.

2.3 Перелік сутностей та їх атрибутів:

Клієнтом в даній інформаційній системі виступає людина, яка взаємодіє з системою через інтерфейс програмного додатку.

ID – атрибут даних клієнта, що означає ідентифікатор окремого користувача.

Ім'я – атрибут даних клієнта, який він вносить при заповненні інформації про себе для реєстрації у додатку.

Прізвище – атрибут даних клієнта, який він вносить при заповненні інформації про себе для реєстрації у додатку.

Почтова адреса (логін) – атрибут даних клієнта, що виступає в ролі логіна для реєстрації у додатку.

Пароль – атрибут даних клієнта, що виступає в ролі пароля для реєстрації у додатку.

Номер телефону – атрибут даних клієнта, який необхідний для ідентифікації особистості.

Для замовлення туристичної продукції клієнт також повинен надати інформацію щодо наступних атрибутів:

Дата виїзду – інформація про початок подорожі.

Тип транспорту – клієнт повинен надати інформацію про тип пересування, яким він бажає дістатися готелю

Кількість днів – інформує про термін перебування у готелі після заїзду.

Кількість людей – інформує про кількість людей, що будуть відвідувати подорож

Статус оплати – інформує про оплату певної туристичної продукції за вказаною датою, терміном на кількість людей.

Туристичною продукцією в даній системі виступають туристичні путівки та тури, яку можна відсортувати та знайти необхідну за заданими фільтрами пошуку.

Унікальне ID – атрибут даних туристичної продукції, що виступає в ролі ідентифікатора певної туристичної продукції.

Ціна – атрибут даних туристичної продукції, містить інформацію про ціну поїздки за 1 людину на 7 днів.

Дата – атрибут даних туристичної продукції, може бути у форматі звичайної дати, що буде означати дату початку туру. Дата є тільки у путівках, що розцінюються як «гарячі».

Статус – атрибут даних туристичної продукції, містить інформацію про актуальність туристичної продукції. Може бути актуальна чи неактуальна.

Назва готелю – атрибут даних туристичної продукції, що містить назву готелю, в який буде здійснюватися подорож.

Місце – атрибут даних туристичної продукції, містить інформацію про місце поїздки.

Деталі – атрибут даних туристичної продукції, містить додаткову інформацію про поїздку та готель, таку як: якість обслуговування готелю, статус готелю (“all inclusive” або ні), комплектація номерів, додаткові розваги, програма туру (додаткові поїздки) і тд.

Інформація про готель виступає окремою сутністю, де міститься інформація про певний готель, куди буде здійснюватися поїздка.

Назва – атрибут інформації про готель, містить в собі повну назву готелю.

Ціна – атрибут інформації про готель, що містить ціну номеру за 1 ніч на 1 особу.

Розташування – атрибут інформації про готель, інформує про точне місцезнаходження готелю.

Номер телефону – атрибут інформації про готель, що містить номер телефону рецепції готелю, в який здійснюється подорож.

Унікальне ID – атрибут інформації про готель, надає ідентифікатор готелю.

Готельні номери – атрибут інформації про готель, містить інформацію про комплектацію готельних номерів.

Транспорт виступає сутністю, що містить в собі інформацію від різних транспортних компаній, що здійснюють перевезення пасажирів.

Тип транспорту – атрибут даних транспорту, містить інформацію про способи пересування до місця туру, різні типи транспорту (поїзд, літак, автобус і тд).

Ціна – атрибут даних транспорту, містить інформацію про ціну перевезення 1 особи.

Унікальне ID – атрибут даних транспорту, надає ідентифікатор транспорту.

Дата виїзду – атрибут зв'язку між туристичною продукцією та транспортом, передає дані про дату початку туру.

Унікальне ID – атрибут зв'язку, надає ідентифікатор угоди про резервування транспорту.

Інформація про клієнта – атрибут, що містить необхідну для резервування транспорту.

2.4 Діаграма «сутність-зв'язок»

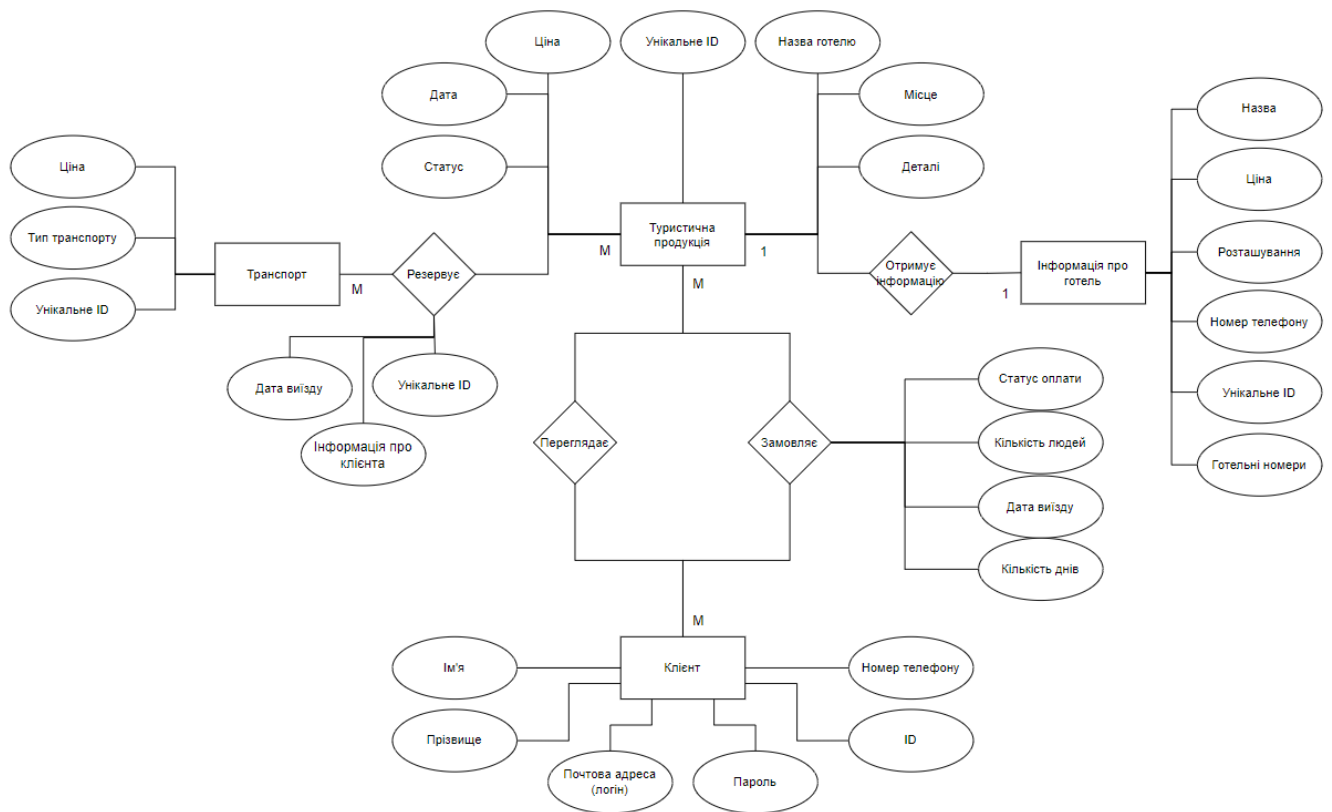


Рис. 2 Діаграма «сутність-зв'язок»

2.5 Функціональні залежності

Функціональна залежність — концепція, що лежить в основі багатьох питань, пов'язаних з реляційними базами даних, включаючи, зокрема, їхнє проектування. Математично являє собою бінарне відношення між множинами атрибутів даного відношення і є, по суті, зв'язком типу «один-до-багатьох». ФЗ забезпечує основу для наукового підходу до розв'язання деяких проблем, оскільки володіє багатим набором цікавих формальних властивостей.

Багатозначна залежність — повне обмеження між двома множинами атрибутів у відношенні. На відміну від функціональної залежності, багатозначна залежність вимагає наявності певних кортежів у відношенні. Отже, багатозначна залежність це особливий випадок кортеж-твірної залежності. Поняття

багатозначної залежності використовується при визначенні четвертої нормальної форми.

Опис (природньою мовою) та перелік (формули) ФЗ та БЗ:

Для проведення декомпозиції, беремо лише фрагмент діаграми «сутність-зв'язок»

1) Клієнт має ідентифікатор, ім'я, прізвище, логін(пошта), пароль та номер телефону:

$cID \longrightarrow cPassword(1), cMailLog(2), cSurname(3), cName(4), cPhone(5)$

2) Клієнт при замовленні вносить свою дату початку туру, кількість днів перебування, кількість людей та статус оплати:

$cID, tID \longrightarrow cDate(6), cTerm(7), cPeople(8), cStatusPay(9)$

3) Туристична продукція має свій код, ціну за 1 людину, назву готелю, дату початку туру (за наявності), статус, тип транспорту, місце, деталі та номер рецепції готелю:

$tID \longrightarrow tPrice(10), tHotelName(11), tStartDate(12), tStatus(13), tTransport(14), tPlace(15), tDetails(16), tHotelNumber(17)$

4) За кодом туру можна знайти код клієнта і навпаки

$tID \longrightarrow cID, cID \longrightarrow tID$

Проведення нормалізації через декомпозицію

$R(cID, cPassword, cMailLog, cSurname, cName, cPhone, cDate, cTerm, cPeople, cStatusPay; tID, tPrice, tHotelName, tStartDate, tStatus, tTransport, tPlace, tDetails, tHotelNumber)$

Ключ на $\{cID, tID\}^+ = \{ cID, cPassword, cMailLog, cSurname, cName, cPhone, cDate, cTerm, cPeople, cStatusPay; tID, tPrice, tHotelName, tStartDate, tStatus, tTransport, tPlace, tDetails, tHotelNumber \}$

ФЗ 1-9 на R

$R1(cID, cPassword, cMailLog, cSurname, cName, cPhone)$

R2(cID, cDate, cTerm, cPeople, cStatusPay, tID, tPrice, tHotelName, tStartDate, tStatus, tTransport, tPlace, tDetails, tHotelNumber)

Ключ на R1: cID+ Ф3 1-9

Ключ на R2: {cID, tID}

Ф3 10-17 на R2

R3(tID, tPrice, tHotelName, tStartDate, tStatus, tTransport, tPlace, tDetails, tHotelNumber)

R4(cID, tID, cDate, cTerm, cPeople, cStatusPay)

Ключ на R3: tID+ Ф3 10-17

Ключ на R4: cID, tID

Результат декомпозиції:

Client(cID, cPassword, cMailLog, cSurname, cName, cPhone) – R1

Tour(tID, tPrice, tHotelName, tStartDate, tStatus, tTransport, tPlace, tDetails, tHotelNumber) – R3

CTour(cID, tID, cDate, cTerm, cPeople, cStatusPay) – R4

2.6 Характеристика елементів даних

В таблиці 3.1 наведено типи даних атрибутів нашої бази даних для подальшого створення коректних таблиць

Сутності та атрибути у БД

№	Сутність	Атрибути	Типи даних	Опис атрибуту
1	Client	cPassword	varchar	пароль клієнта
		cName	varchar	ім'я клієнта
		cSurname	varchar	прізвище клієнта
		cMailLog	varchar	пошта клієнта, що виступає логіном
		cPhone	varchar	номер телефону клієнта
		cID	int	ідентифікатор
2	CTour	cDate	date	дата початку туру
		cTerm	int	кількість днів
		cPeople	int	кількість людей
		cStatusPay	int	статус оплати
3	Tour	tID	int	код туру
		tPrice	int	ціна туру
		tHotelName	varchar	назва готелю
		tStartDate	date	дата початку туру
		tStatus	int	статус туру
		tPlace	varchar	місце розташування
		tDetails	varchar	деталі туру
4	Hotel	hName	varchar	назва готелю
		hPrice	int	ціна номеру
		hLocation	varchar	розташування готелю
		hNumber	varchar	номер рецепції
		hID	int	ідентифікатор
		hRoom	varchar	інформація про номери
5	ReservTransp	rtID	int	ідентифікатор
		rtDate	date	дата початку туру
		rtClient	varchar	інформація про клієнта
6	Transport	trPrice	int	ціна за перевезення
		trType	varchar	тип транспорту
		trID	int	ідентифікатор

UML діаграма класів

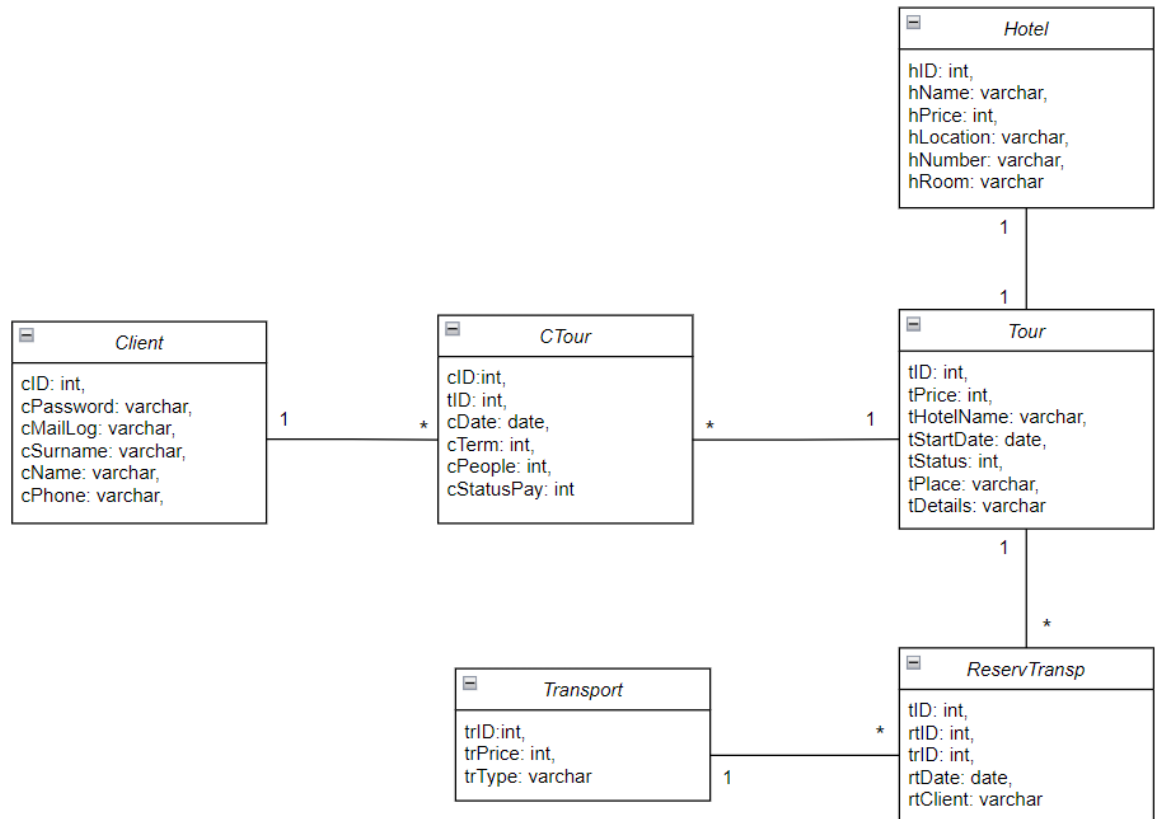


Рис. 3 UML діаграма класів

Висновки до розділу 2

В результаті розробки другого розділу було сформовано актуальність розробки БД, також наведено перелік прецедентів використання програмного застосунку з їх коротким описом. За цією інформацією було розроблено діаграму «сутність-зв'язок», що візуалізує структуру БД та описано сутності та атрибути. Також було отримано та закріплено теоретичні знання з проведення аналізу предметної області. Було спроектовано відношення і, як результат проведення нормалізації через декомпозицію, розроблено UML діаграму класів.

РОЗДІЛ 3

ПРАКТИЧНА ЧАСТИНА

3.1 Обмеження предметної області

- Пароль користувача має бути більший за 8 символів та менший за 16 символів.
- Логін користувача має бути унікальним.
- Ім'я, прізвище та логін користувача обмежено у 30 символів.
- Довжина телефону клієнта та рецепції готелю не має перевищувати за 16 символів
- Статус оплати має приймати лише значення 1 або 0
- Ім'я готелю обмежено у 100 символів та повинне складатись з лише з маленьких літер.
- Статус туристичної продукції має приймати значення лише 1 або 0.
- Ціна має бути не менше 0.
- Видалити позицію з туристичної продукції можна лише якщо вона неактивна (статус = 0)
- Тип транспорту обмежено 50 символами

Перелік відповідних general assertions мовою SQL (приклади):

```
CREATE ASSERTION phone(  
CHECK ((SELECT * FROM Client  
        WHERE length(Client.cPhone) < 16)  
);  
  
CREATE ASSERTION price(  
CHECK ((SELECT * FROM Tour  
        WHERE Tour.tPrice >= 0)));
```

3.2 Створення таблиць

Створення таблиці інформації про клієнта

```
CREATE TABLE Client(  
    cID INTEGER,  
    cPassword varchar(16) CHECK (length(cPassword) > 8),  
    cMailLog varchar(30),  
    cName varchar(30) NOT NULL,  
    cSurname varchar(30) NOT NULL,  
    cPhone varchar(16),  
    CONSTRAINT login UNIQUE (cMailLog),  
    PRIMARY KEY (cID AUTOINCREMENT)  
);
```

Створення таблиці замовлення клієнтом туру

```
CREATE TABLE CTour(  
    tID INTEGER,  
    cID INTEGER,  
    cDate date,  
    cTerm varchar(7),  
    cPeople INTEGER,  
    cStatusPay INTEGER CHECK (cStatusPay = 1 or cStatusPay = 0),  
    FOREIGN KEY (tID) REFERENCES Tour(tID),  
    FOREIGN KEY (cID) REFERENCES Client(cID)  
);
```

Створення таблиці інформації про туристичну продукцію

```
CREATE TABLE Tour(  
    tID INTEGER,  
    tPrice INTEGER CHECK (tPrice >= 0),
```

```

tHotelName varchar(100) NOT NULL,
tStartDate date,
tStatus INTEGER NOT NULL CHECK (tStatus = 1 or tStatus = 0),
tPlace varchar(100) NOT NULL,
tDetails varchar,
PRIMARY KEY (tID AUTOINCREMENT)
);

```

Команди для створення коментарів до таблиць та до стовпців:

```

COMMENT ON TABLE Client 'Таблиця даних про клієнта'
COMMENT ON TABLE Tour 'Таблиця даних туристичної продукції'
COMMENT ON COLUMN Tour.tStatus 'Статус туристичної продукції'
COMMENT ON COLUMN Client.cSurname 'Прізвище клієнта'

```

Оскільки дана робота виконується за допомогою SQLite, додавання коментарів до таблиці та певних стовпців неможливе, лише звичайні коментарі по типу:

```

/*Ім'я клієнта*/
/*Прізвище клієнта*/

```

3.3 Автоматично змінюване представлення.

Приклад CREATE VIEW:

Створення представлення детальної інформації про тур

```

CREATE VIEW Full_info
AS SELECT tID, tPrice, tHotelName, tStartDate, tPlace, hLocation, tDetails,
hNumber, hRoom FROM Tour
JOIN Hotel on tHotelName = hName;

```


Таблица: <div>Full_info</div>									
	tID	tPrice	tHotelName	tStartDate	tPlace	hLocation	tDetails	hNumber	hRoom
	Фи...	Фил...	Фильтр	Фильтр	Фильтр	Фильтр	Фильтр	Фильтр	Фильтр
1	1	15000	golden view beach resort	2022-12-06	Egypt, Sharm El Sheikh	Al Fanar, Street, Om El ...	all-inclusiv, free parking, ...	+20693664694	Rooms are equipped with a...
2	2	25000	malte - astotel	NULL	France, Paris	63 Rue de Richelieu, ...	Dry cleaning, Special diet ...	+33144589494	Rooms provide a flat scree...
3	3	20000	lara barut collection	NULL	Turkey, Antalya	Guzeloba Mahallesi Yasa...	all-inclusiv, free parking, ...	+904449600	Relax in rooms with a flat-...
4	4	46256	padma resort legian	NULL	Indonesia, Bali	Jl. Padma no. 1, Legian ...	Free Wi-Fi, free parking, AL...	+62361752111	Air Conditioning, Additional ...
5	5	32000	park inn by radisson berlin ...	NULL	Germany, Berlin	Alexanderplatz 7, 10178 ...	Free Wi-Fi, Fitness centre, ...	+493023890	Rooms are equipped with a...
6	6	31600	radisson blu hotel	2022-12-24	Germany, Frankfurt	Franklinstr. 65, 60486 ...	Free Internet, Fitness cent...	+49697701550	Radisson Blu Hotel, Frankfu...
7	7	35780	hotel saturnia & international...	NULL	Italy, Venice	Via XXII Marzo, San Mar...	all-inclusiv, free parking, ...	+390415208377	Saturnia Hotel Venice offers...
8	8	108300	royal kona resort	2023-01-15	USA, Hawaii	75-5852 Alii Drive, Kailu...	Swimming pool, Beach, ...	+18083293111	Royal Kona is a family-...
9	9	162000	sun siyam olhuveli	NULL	Maldives	Meedhuffushi Southern ...	All-inclusive, Evening ...	+9606760011	Air conditioning, Private ...

Рис.4 Створення представлення з повною інформацією про тури

Створення представлення інформації про клієнта для готелів

CREATE VIEW Info_for_hotel

as SELECT cSurname, cName, cDate, cTerm, cPeople, tHotelName FROM

Client c

JOIN CTour ct on c.cID = ct.cID JOIN Tour t on t.tID = ct.tID;

Таблиця: Info_for_hotel						
	cSurname	cName	cDate	cTerm	cPeople	tHotelName
	Фильтр	Фильтр	Фильтр	Фильтр	Фильтр	Фильтр
1	Sharpinsky	Andrey	2023-01-04	13 days	2	park inn by radisson berlin alexanderplatz hotel
2	Tsaryk	Svetlana	2023-01-11	7 days	3	lara barut collection
3	Pauchenko	Anna	2023-01-11	7 days	4	lara barut collection
4	Yarovykh	Diana	2023-01-16	10 days	2	hotel saturnia & international venezia
5	Berluta	Darya	2023-01-16	10 days	2	hotel saturnia & international venezia
6	Boichenko	Maria	2023-01-16	10 days	1	hotel saturnia & international venezia
7	Boyko	Valeria	2023-01-02	7 days	2	radisson blu hotel

Рис.5 Результат роботи запиту

Створення представлення інформації про клієнта для транспортних компаній

CREATE VIEW Info_for_transport

as SELECT trType, rtDate, cSurname, cName, cPeople FROM Transport tr

JOIN ReservTransp rt on tr.trID = rt.trID JOIN CTour ct on ct.tID = rt.tID JOIN
Client c on ct.cID = c.cID;

Таблица: Info_for_transport

	trType	rtDate	cSurname	cName	cPeople
	Фильтр	Фильтр	Фильтр	Фильтр	Фильтр
1	airplane "Swiss"	2023-01-04	Sharpinsky	Andrey	2
2	airplane "Turkish Airlines"	2023-01-11	Tsaryk	Svetlana	3
3	airplane "Turkish Airlines"	2023-01-11	Pauchenko	Anna	4
4	bus "LikeBus"	2023-01-16	Yarovykh	Diana	2
5	bus "LikeBus"	2023-01-16	Berluta	Darya	2
6	bus "LikeBus"	2023-01-16	Boichenko	Maria	1
7	airplane "Austrian"	2023-01-02	Boyko	Valeria	2

Рис.6 Результат роботи запиту

Створення представлення з інформацією про готель

CREATE VIEW hotel_info

as SELECT hID, hName, hPrice, hRoom FROM Hotel;

Таблица: hotel_info

	hID	hName	hPrice	hRoom
	Фи...	Фильтр	Фильтр	Фильтр
1	1	golden view beach resort	4007	Rooms are equipped with a flat-scee...
2	2	malte - astotel	5042	Rooms provide a flat screen TV, air ...
3	3	park inn by radisson berlin ...	4543	Rooms are equipped with a flat scree...
4	4	radisson blu hotel	4504	Radisson Blu Hotel, Frankfurt Hotel ...
5	5	hotel saturnia & international venezia	4949	Saturnia Hotel Venice offers rooms ...
6	6	padma resort legian	6677	Air Conditioning, Additional bathroom...
7	7	royal kona resort	15181	Royal Kona is a family-friendly resort ...
8	8	sun siyam olhuveli	31936	Air conditioning, Private beach, Privat...
9	9	lara barut collection	15422	Relax in rooms with a flat-screen TV,...

Рис.7 Результат роботи запиту

Приклад INSERT:

Внесення інформації у представлення інформації про готель

```
INSERT INTO hotel_info VALUES (10, 'Fairmont Grand Hotel Kyiv', 6545, 'The elegant wood-panelled rooms feature Wi-Fi, flat-screen TVs, work desks, seating areas and tea and coffee making facilities. Room service is available. Suites have living rooms with sofas. Superior suites have balconies, kitchens and dining rooms');
```

Таблиця: hotel_info				
	hID	hName	hPrice	hRoom
	Фи...	Фильтр	Фильтр	Фильтр
1	1	golden view beach resort	4007	Rooms are equipped with a flat-scre...
2	2	malte - astotel	5042	Rooms provide a flat screen TV, air ...
3	3	park inn by radisson berlin ...	4543	Rooms are equipped with a flat scree...
4	4	radisson blu hotel	4504	Radisson Blu Hotel, Frankfurt Hotel ...
5	5	hotel saturnia & international venezia	4949	Saturnia Hotel Venice offers rooms ...
6	6	padma resort legian	6677	Air Conditioning, Additional bathroom...
7	7	royal kona resort	15181	Royal Kona is a family-friendly resort ...
8	8	sun siyam olhuveli	31936	Air conditioning, Private beach, Privat...
9	9	lara barut collection	15422	Relax in rooms with a flat-screen TV,...
10	10	fairmont grand hotel kyiv	6545	The elegant wood-panelled rooms ...

Рис.8 Результат внесення інформації

Приклад UPDATE:

Оновлення ціни у представленні інформації про готель

```
UPDATE hotel_info SET hPrice = 7264 WHERE hID = 10;
```

Таблиця:	hotel_info									Filter in any column
	hID	hName	hPrice	hRoom						
	Фи...	Фильтр	Филь...	Фильтр						
1	1	golden view beach resort	4007	Rooms are equipped with a flat-scre...						
2	2	malte - astotel	5042	Rooms provide a flat screen TV, air ...						
3	3	park inn by radisson berlin ...	4543	Rooms are equipped with a flat scree...						
4	4	radisson blu hotel	4504	Radisson Blu Hotel, Frankfurt Hotel ...						
5	5	hotel saturnia & international venezia	4949	Saturnia Hotel Venice offers rooms ...						
6	6	padma resort legian	6677	Air Conditioning, Additional bathroom...						
7	7	royal kona resort	15181	Royal Kona is a family-friendly resort ...						
8	8	sun siyam olhuveli	31936	Air conditioning, Private beach, Privat...						
9	9	lara barut collection	15422	Relax in rooms with a flat-screen TV,...						
10	10	fairmont grand hotel kyiv	7264	The elegant wood-panelled rooms ...						

Рис. 9 Результат оновлення ціни

Приклад DELETE:

Видалення позиції з представлення інформації про готель

DELETE FROM hotel_info WHERE hID = 10;

Таблиця:	hotel_info									Filter in any column
	hID	hName	hPrice	hRoom						
	Фи...	Фильтр	Филь...	Фильтр						
1	1	golden view beach resort	4007	Rooms are equipped with a flat-scre...						
2	2	malte - astotel	5042	Rooms provide a flat screen TV, air ...						
3	3	park inn by radisson berlin ...	4543	Rooms are equipped with a flat scree...						
4	4	radisson blu hotel	4504	Radisson Blu Hotel, Frankfurt Hotel ...						
5	5	hotel saturnia & international venezia	4949	Saturnia Hotel Venice offers rooms ...						
6	6	padma resort legian	6677	Air Conditioning, Additional bathroom...						
7	7	royal kona resort	15181	Royal Kona is a family-friendly resort ...						
8	8	sun siyam olhuveli	31936	Air conditioning, Private beach, Privat...						
9	9	lara barut collection	15422	Relax in rooms with a flat-screen TV,...						

Рис. 10 Результат видалення з представлення

Представлення з використанням функцій агрегації та клаузи HAVING

CREATE VIEW Tour_price

as SELECT tID, tPrice, tHotelName, tDetails FROM Tour

GROUP BY tID HAVING tPrice <= 36000;

Таблица: Tour_price				
	tID	tPrice	tHotelName	tDetails
	Фи...	Фил...	Фильтр	Фильтр
1	1	15000	golden view beach resort	all-inclusiv, free parking, pool, diving
2	2	25000	malte - astotel	Dry cleaning, Special diet menu, Free...
3	3	20000	lara barut collection	all-inclusiv, free parking, pool, diving
4	5	32000	park inn by radisson berlin ...	Free Wi-Fi, Fitness centre, Bar/loung...
5	6	31600	radisson blu hotel	Free Internet, Fitness center, ...
6	7	35780	hotel saturnia & international venezia	all-inclusiv, free parking, pool, diving

Рис. 11 Результат виконання

Представлення із зовнішнім з'єднанням

Створення представлення з інформацією про клієнтів для транспортних компаній

CREATE VIEW Info_for_transport

as SELECT trType, rtDate, cSurname, cName, cPeople FROM Transport tr

JOIN ReservTransp rt on tr.trID = rt.trID JOIN CTour ct on ct.tID = rt.tID JOIN
Client c on ct.cID = c.cID;


```
INSTEAD of INSERT on hotel_info
FOR EACH ROW
BEGIN
    INSERT INTO Hotel VALUES (NEW.hID, NEW.hName, NEW.hPrice,
null, null, NEW.hRoom);
END;
```

3.4 Тригери для таблиць

Тригери AFTER

Тригер на зменшення літер в інформації про готель у таблиці турів

```
CREATE TRIGGER lowerHotelN_tour
AFTER INSERT on Tour
BEGIN
    UPDATE Tour
    SET tHotelName = lower(tHotelName);
END;
```

Тригер на зменшення літер в назві готелю в таблиці готелів

```
CREATE TRIGGER lowerName_Hotel
AFTER INSERT on Hotel
BEGIN
    UPDATE Hotel
    SET hName = lower(hName);
END;
```

Тригери BEFORE

Тригер на перевірку наявної важливої інформації перед видаленням

```
CREATE TRIGGER check_info
BEFORE DELETE on Tour
BEGIN
    SELECT CASE
    WHEN OLD.tStatus = 1
    THEN
        RAISE(ABORT, "Check your information before delete")
    END;
END;
```

Тригер на перевірку паролю користувача

```
CREATE TRIGGER client_password
BEFORE INSERT on Client
BEGIN
    SELECT CASE
    WHEN length(NEW.cPassword) < 8
    THEN
        RAISE(ABORT, "Invalid Password")
    END;
END;
```

Тригери для представлень

Додавання рядка

```
CREATE TRIGGER hotel_info_insert
INSTEAD of INSERT on hotel_info
FOR EACH ROW
```



```
BEGIN
    INSERT INTO Hotel VALUES (NEW.hID, NEW.hName, NEW.hPrice,
null, null, NEW.hRoom);
```

```
END;
```

Оновлення інформації

```
CREATE TRIGGER hotel_info_update
INSTEAD OF UPDATE OF hPrice ON hotel_info
FOR EACH ROW
BEGIN
```

```
    UPDATE Hotel
    SET hPrice = NEW.hPrice
    WHERE hID = OLD.hID;
```

```
END;
```

Видалення інформації:

```
CREATE TRIGGER hotel_info_delete
INSTEAD OF DELETE on hotel_info
FOR EACH ROW
BEGIN
```

```
    DELETE FROM Hotel
    WHERE hID = OLD.hID;
```

```
END;
```

3.5 Індекси

Виконання запитів без використання індексів

Виведення інформація, що передається у транспортні компанії

```
SELECT trType, rtDate, cSurname, cName, cPeople FROM Transport tr
```

JOIN ReservTransp rt on tr.trID = rt.trID JOIN CTour ct on ct.tID = rt.tID JOIN
Client c on ct.cID = c.cID;

	trType	rtDate	cSurname	cName	cPeople
1	airplane "Swiss"	2023-01-04	Sharpinsky	Andrey	2
2	airplane "Turkish Airlines"	2023-01-11	Tsaryk	Svetlana	3
3	airplane "Turkish Airlines"	2023-01-11	Pauchenko	Anna	4
4	bus "LikeBus"	2023-01-16	Yarovykh	Diana	2
5	bus "LikeBus"	2023-01-16	Berluta	Darya	2
6	bus "LikeBus"	2023-01-16	Boichenko	Maria	1
7	airplane "Austrian"	2023-01-02	Boyko	Valeria	2

Рис.14 Результат виконання запиту

```
Execution finished without errors.
Result: 7 строк возвращено за 13мс
```

Рис.15 Швидкість виконання запиту

Виведення популярності напрямків

SELECT tHotelName, count(*) as popularity from Tour t
JOIN CTour ct on ct.tID = t.tID
GROUP BY tHotelName

	tHotelName	popularity
1	hotel saturnia & international venezia	3
2	lara barut collection	2
3	park inn by radisson berlin alexanderplatz hotel	1
4	radisson blu hotel	1

Рис.16 Результат виконання запиту

```
Execution finished without errors.
Result: 4 строк возвращено за 8мс
```

Рис.17 Швидкість виконання

Виведення згрупованої інформації з типом транспорту та місцем поїздки

```

SELECT trType as Type_transport, group_concat(cSurname) as client_info,
tHotelName FROM Transport tr
JOIN ReservTransp rt on tr.trID = rt.trID JOIN CTour ct on ct.tID = rt.tID
JOIN Client c on ct.cID = c.cID JOIN Tour t on ct.tID = t.tID
GROUP BY tHotelName;

```

	Type_transport	client_info	tHotelName
1	bus "LikeBus"	Yarovykh,Berluta,Boichenko	hotel saturnia & international venezia
2	airplane "Turkish Airlines"	Tsaryk,Pauchenko	lara barut collection
3	airplane "Swiss"	Sharpinsky	park inn by radisson berlin ...
4	airplane "Austrian"	Boyko	radisson blu hotel

Рис.18 Результат виконання

```

Execution finished without errors.
Result: 4 строк возвращено за 10мс

```

Рис.19 Швидкість виконання

Створення індексів

Створюємо індекси на об'ємні та найчастіше використовувані колонки:

```

CREATE INDEX client_index on Client (cSurname, cID);
CREATE INDEX ctour_index on CTour (cDate);
CREATE INDEX tour_index on Tour (tHotelName, tID);
CREATE INDEX hotel_index on Hotel (hID, hName);
CREATE INDEX transport_index on Transport (trType);

```

Виконання запитів з індексами

Для кращого порівняння я використала ті ж самі запити, тому для аналізу результатів нам необхідний лише час виконання

Перший запит:

7	airplane "Austrian"	2023-01-02	Boyko
---	---------------------	------------	-------

Execution finished without errors.
Result: 7 строк возвращено за 7мс

Рис. 20 Швидкість виконання з індексами

Другий запит:

4	radisson blu hotel
---	--------------------

Execution finished without errors.
Result: 4 строк возвращено за 4мс

Рис. 21 Швидність виконання другого запиту з індексацією

Третій запит:

3	airplane "Swiss"	Sharpinsky
4	airplane "Austrian"	Boyko

Execution finished without errors.
Result: 4 строк возвращено за 5мс

Рис.22 Швидкість виконання третього запиту

Як можемо бачити швидкість виконання запитів з використанням індиксів зменшилася, особливо це видно на запиті з використанням об'ємних текстових значень.

Звичайно ж для моєї бази даних, де немає досить багато інформації, ця зміна швидкості непомітна, проте якщо база даних заповниться на тисячі рядків, зміна швидкості стане помітною при виконанні запитів.

Висновки до розділу 3

В результаті виконання третього розділу було розроблено практичну частину курсової роботи, де було створено базу даних для управління туристичним агенством за допомогою створення таблиц, тригерів, представлень та індексації.

ВИСНОВКИ

Виходячи з актуальності проблеми для багатьох людей щодо подорожей, в даній курсовій роботі було розроблено базу даних для застосунку управління туристичним агентством.

Оскільки зараз з кожним днем наше життя потроху переноситься у технології, розробка додатку управління туристичним агентством є актуальною, тому що турагенство виступає здебільшого посередником між клієнтами та туристичними операторами, що пропонують свої послуги.

Невід'ємною частиною додатку є база даних, тому я її частково реалізувала. Процес проектування було виконано в 3 етапи. Перший етап виступав підготовкою до розробки та полягав у засвоєнні теоретичного матеріалу. На другому етапі було описано 4 прецеденти використання та наведено опис бізнес-процесів. Також на даному етапі було реалізовано діаграму «сутність-зв'язок», що візуалізує описані всі сутності та атрибути, проведено нормалізацію через декомпозицію де результат я зобразила на діаграмі класів UML. Третій етап був найскладнішим, проте не менш цікавим, оскільки на даному етапі створювалася фізична модель бази даних, що дала змогу набути навички з проектування та використання бази даних.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Що таке база даних? [Електронний ресурс] // [apereps.kpi.ua](http://apereps.kpi.ua/shco-take-basa-danykh). – 2020. – Режим доступу до ресурсу: <http://apereps.kpi.ua/shco-take-basa-danykh>.
2. Класифікація баз даних [Електронний ресурс] // lib.mdpu.org.ua. – 2017. – Режим доступу до ресурсу: <http://lib.mdpu.org.ua/e-book/vstup/L5.htm>.
3. Что такое СУБД [Електронний ресурс] // [ruCenter](https://www.nic.ru/help/chto-takoe-subd_8580.html). – 2021. – Режим доступу до ресурсу: https://www.nic.ru/help/chto-takoe-subd_8580.html.
4. Бази даних [Електронний ресурс] // [UA5.ORG](https://ua5.org/database/189-reljaccjjna-baza-danikh.html). – 2008. – Режим доступу до ресурсу: <https://ua5.org/database/189-reljaccjjna-baza-danikh.html>.
5. Що таке SQL? [Електронний ресурс] // [a-yak](https://a-yak.com/shho-take-sql/). – 2021. – Режим доступу до ресурсу: <https://a-yak.com/shho-take-sql/>.
6. Концептуальное проектирование [Електронний ресурс] // Академик. – 2021. – Режим доступу до ресурсу: <https://dic.academic.ru/dic.nsf/ruwiki/1527943>.
7. SYSTEMS OF TEACHING ENGINEERING WORK ON BASE OF INTERNET TECHNOLOGIES [Електронний ресурс] // International Journal "Information Technologies and Knowledge". – 2007. – Режим доступу до ресурсу: <http://www.foibg.com/ijitk/ijitk-vol01/ijitk01-1-p06.pdf>.
8. База даних [Електронний ресурс] // [wikipedia](https://uk.wikipedia.org/wiki/База_даних). – 6. – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/База_даних.
9. Дейт К. Дж. Введения у системи баз даних. :Пер. з англ. - 6-те вид. - К.: Діалектика, 1998. - 784 с.
10. Карпова Т.С. Бази даних: моделі, розробка, реалізація. - СПб.: Пітер, 2001. - 304 с
11. Date, C. J. Date on Database: Writings 2000–2006. — Apress, 2006. — 566 с.
12. Date, C. J. Database in Depth. — O'Reilly, 2005. — 240 с

13. Дейт К. Дж. Введение в системы баз данных. :Пер. с англ. – 6-е изд. – К.: Диалектика, 1998. – 784 с.
14. Bachman, Charles W. (1973). "The Programmer as Navigator". Communications of the ACM.
15. Connolly, Thomas M.; Begg, Carolyn E. (2014). Database Systems – A Practical Approach to Design Implementation and Management (6th ed.). Pearson

Додаток А Оператори DDL

Створення таблиць:

```
CREATE TABLE Client(  
    cID INTEGER,  
    cPassword varchar(16) CHECK (length(cPassword) > 8),  
    cMailLog varchar(30),  
    cName varchar(30) NOT NULL,  
    cSurname varchar(30) NOT NULL,  
    cPhone varchar(16),  
    CONSTRAINT login UNIQUE (cMailLog),  
    PRIMARY KEY (cID AUTOINCREMENT)  
);  
  
CREATE TABLE CTour(  
    tID INTEGER,  
    cID INTEGER,  
    cDate date,  
    cTerm varchar(7),  
    cPeople INTEGER,  
    cStatusPay INTEGER CHECK (cStatusPay = 1 or cStatusPay = 0),
```

```
FOREIGN KEY (tID) REFERENCES Tour(tID),  
  
FOREIGN KEY (cID) REFERENCES Client(cID)  
  
);  
  
CREATE TABLE Tour(  
  
    tID INTEGER,  
  
    tPrice INTEGER CHECK (tPrice >= 0),  
  
    tHotelName varchar(100) NOT NULL,  
  
    tStartDate date,  
  
    tStatus INTEGER NOT NULL CHECK (tStatus = 1 or tStatus = 0),  
  
    tPlace varchar(100) NOT NULL,  
  
    tDetails varchar,  
  
    PRIMARY KEY (tID AUTOINCREMENT)  
  
);
```

```
CREATE TABLE Hotel(  
  
    hID INTEGER,  
  
    hName varchar(100) NOT NULL,  
  
    hPrice INTEGER,  
  
    hLocation varchar(100),  
  
    hNumber varchar(16),
```

```

        hRoom varchar(225),

        PRIMARY KEY (hID AUTOINCREMENT)

);

CREATE TABLE ReservTransp(

        tID INTEGER,

        trID INTEGER,

        rtID INTEGER,

        rtDate date,

        rtClient varchar(100),

        PRIMARY KEY (rtID AUTOINCREMENT),

        FOREIGN KEY (tID) REFERENCES Tour(tID),

        FOREIGN KEY (trID) REFERENCES Transport(trID)

);

CREATE TABLE Transport(

        trID INTEGER,

        trType varchar(50),

        trPrice INTEGER,

        PRIMARY KEY (trID AUTOINCREMENT)

);

```

Заповнення таблиць даними:

```
INSERT INTO Client VALUES (1, 'sjd214dhf', 'mail@gmail.com', 'Diana',  
'Yarovykh', '+380950927483', null);
```

```
INSERT INTO Client VALUES (2, 'djv3fkj5415', 'alic@gmail.com',  
'Alise','Bybic', '+380503410920', null);
```

```
INSERT INTO Client VALUES (3, '545dfvgrg5', 'sharpynsky@gmail.com',  
'Andrey','Sharpinsky', '+380950425681', null);
```

```
INSERT INTO Client VALUES (4, 'scmnfrgv52', 'j.auksentiev@gmail.com',  
'Jeugen','Auksentiev', '+380635612783', null);
```

```
INSERT INTO Client VALUES (5, 'dfkm66dfvb', 'boyko.l@gmail.com',  
'Valeria','Boyko', '+380981235862', null);
```

```
INSERT INTO Client VALUES (6, 'jfnvkjf6461', 'ivan@gmail.com',  
'Ivan','Ivanovych', '+380968589123', null);
```

```
INSERT INTO Client VALUES (7, 'lfkgm46563', 'svetlana.tsyk@gmail.com',  
'Svetlana','Tsaryk', '+380507240233', null);
```

```
INSERT INTO Client VALUES (8, 'sdsdvnhd52',  
'pauchenkoanna@gmail.com', 'Anna', 'Pauchenko', '+380985628451', null);
```

```
INSERT INTO Client VALUES (9, 'mfk2541dhf5', 'berluta.daria@gmail.com',  
'Darya', 'Berluta', '+380963256145', null);
```

```
INSERT INTO Client VALUES (10, 'kjghn54dgdf',  
'boichenko.maria@gmail.com', 'Maria', 'Boichenko', '+3809545867419', null);
```

```
INSERT INTO Tour VALUES (1, 15000, 'Golden View Beach Resort', '2022-  
12-06', 1, 'Egypt, Sharm El Sheikh', 'all-inclusiv, free parking, pool, diving');
```

INSERT INTO Tour VALUES (2, 25000, 'Malte - Astotel', null, 1, 'France, Paris', 'Dry cleaning, Special diet menu, Free WiFi, Taxi service');

INSERT INTO Tour VALUES (3, 20000, 'Lara Barut Collection', null, 1, 'Turkey, Antalya', 'all-inclusiv, free parking, pool, diving');

INSERT INTO Tour VALUES (4, 46256, 'Padma Resort Legian', null, 1, 'Indonesia, Bali', 'Free Wi-Fi, free parking, ALL-inclusiv, Tennis court, Babysitting');

INSERT INTO Tour VALUES (5, 32000, 'Park Inn by Radisson Berlin Alexanderplatz Hotel', null, 1, 'Germany, Berlin', 'Free Wi-Fi, Fitness centre, Bar/lounge, Taxi service, Conference facilities, Spa');

INSERT INTO Tour VALUES (6, 31600, 'Radisson Blu Hotel', '2022-12-24', 1, 'Germany, Frankfurt', 'Free Internet, Fitness center, Cafeteria, Secured parking, Buffet breakfast');

INSERT INTO Tour VALUES (7, 35780, 'Hotel Saturnia & International Venezia', null, 1, 'Italy, Venice', 'all-inclusiv, free parking, pool, diving');

INSERT INTO Tour VALUES (8, 108300, 'Royal Kona Resort', '2023-01-15', 1, 'USA, Hawaii', 'Swimming pool, Beach, Diving, Pool with a view, Water sport equipment hire, Evening entertainment');

INSERT INTO Tour VALUES (9, 162000, 'Sun Siyam Olhuveli', null, 1, 'Maldives', 'All-inclusive, Evening entertainment, trips to other islands');

INSERT INTO Hotel VALUES (1, 'Golden View Beach Resort', 4007, 'Al Fanar, Street, Om El Seid Hill, South Sinai Governorate, Egypt', '+20693664694', 'Rooms are equipped with a flat-screen TV, a kitchenette and a refrigerator');

INSERT INTO Hotel VALUES (2, 'Malte - Astotel', 5042, '63 Rue de Richelieu, 75002 Paris', '+33144589494', 'Rooms provide a flat screen TV, air conditioning, and a minibar, and you can stay connected with free wifi');

INSERT INTO Hotel VALUES (3, 'Park Inn by Radisson Berlin Alexanderplatz Hotel', 4543, 'Alexanderplatz 7, 10178 Berlin Germany', '+493023890', 'Rooms are equipped with a flat screen TV and air conditioning, and guests can go online at any time thanks to free internet access');

INSERT INTO Hotel VALUES (4, 'Radisson Blu Hotel', 4504, 'Franklinstr. 65, 60486 Frankfurt, Hesse Germany', '+49697701550', 'Radisson Blu Hotel, Frankfurt Hotel features a concierge and room service, to help make your visit even more pleasant. There is also a swimming pool and a lobby. If you are driving to Radisson Blu Hotel, Frankfurt Hotel, parking is available');

INSERT INTO Hotel VALUES (5, 'Hotel Saturnia & International Venezia', 4949, 'Via XXII Marzo, San Marco 2398, 30124 Venice Italy', '+390415208377', 'Saturnia Hotel Venice offers rooms equipped with a flat screen TV, air conditioning and a minibar. Guests can go online at any time thanks to free Wi-Fi. The hotel offers a 24 hour front desk, room service, and a concierge, to make your visit even more pleasant. A free breakfast is also provided');

INSERT INTO Hotel VALUES (6, 'Padma Resort Legian', 6677, 'Jl. Padma no. 1, Legian 80361 Indonesia', '+62361752111', 'Air Conditioning, Additional bathroom, Housekeeping, Interconnecting rooms available, Private balcony, Room service, Refrigerator, Flat-screen TV, Safe-box, Sitting area, Wake-up/wake-up service');

INSERT INTO Hotel VALUES (7, 'Royal Kona Resort', 15181, '75-5852 Alii Drive, Kailua-Kona, Island of Hawaii, HI 96740-1310', '+18083293111', 'Royal Kona is a family-friendly resort where rooms offer a flat screen TV, air conditioning, and a

refrigerator, and you can stay connected with wifi. The resort features a 24 hour front desk, a concierge, and shops');

INSERT INTO Hotel VALUES (8, 'Sun Siyam Olhuveli', 31936, 'Meedhuffushi Southern Nilandhe Atoll, 13100', '+9606760011', 'Air conditioning, Private beach, Private balcony, Room service, Safe, Minibar, Refrigerator, Flat screen TV');

INSERT INTO Hotel VALUES (9, 'Lara Barut Collection', 15422, 'Guzeloba Mahallesi Yasar Sobutay Bulvari,30, Muratpasa, Antalya 07235 Turkey', '+904449600', 'Relax in rooms with a flat-screen TV, air conditioning and a refrigerator. You can also stay connected during your stay as Barut Lara offers guests free wifi. The hotel features room service and a concierge. Plus, Barut Lara Turkey offers a pool and breakfast, providing a pleasant respite from your busy day');

INSERT INTO Transport VALUES (1, 'airplane "Air Ocean Airlines" ', 7210);

INSERT INTO Transport VALUES (2, 'airplane "Turkish Airlines" ', 12204);

INSERT INTO Transport VALUES (3, 'airplane "Austrian" ', 10177);

INSERT INTO Transport VALUES (4, 'airplane "Ryanair" ', 3299);

INSERT INTO Transport VALUES (5, 'airplane "Swiss" ', 9262);

INSERT INTO Transport VALUES (6, 'bus "FlixBus" ', 5310);

INSERT INTO Transport VALUES (7, 'bus "PolskiBus" ', 4360);

INSERT INTO Transport VALUES (8, 'bus "LikeBus" ', 3000);

INSERT INTO Transport VALUES (9, 'bus "InfoBus" ', 1320);

INSERT into CTour VALUES (5, 3, '2023-01-04', '13 days', 2, 1);

```
INSERT into CTour VALUES (3, 7, '2023-01-11', '7 days', 3, 1);  
INSERT into CTour VALUES (3, 8, '2023-01-11', '7 days', 4, 1);  
INSERT into CTour VALUES (7, 1, '2023-01-16', '10 days', 2, 1);  
INSERT into CTour VALUES (7, 9, '2023-01-16', '10 days', 2, 1);  
INSERT into CTour VALUES (7, 10, '2023-01-16', '10 days', 1, 1);  
INSERT into CTour VALUES (6, 5, '2023-01-02', '7 days', 2, 1);
```

```
INSERT INTO ReservTransp VALUES (5, 5, 1, '2023-01-04', 'Andrey  
Sharpinsky');
```

```
INSERT INTO ReservTransp VALUES (3, 2, 2, '2023-01-11', 'Svetlana Tsaryk,  
Anna Pauchenko');
```

```
INSERT INTO ReservTransp VALUES (7, 8, 3, '2023-01-16', 'Diana Yarovykh,  
Darya Berluta, Maria Boichenko');
```

```
INSERT INTO ReservTransp VALUES (6, 3, 4, '2023-01-02', 'Valeria Boyko');
```

Створення представлень:

```
CREATE VIEW Full_info
```

```
AS SELECT tID, tPrice, tHotelName, tStartDate, tPlace, hLocation, tDetails,  
hNumber, hRoom FROM Tour
```

```
JOIN Hotel on tHotelName = hName;
```

```
CREATE VIEW Info_for_hotel
```



```
as SELECT cSurname, cName, cDate, cTerm, cPeople, tHotelName FROM  
Client c
```

```
JOIN CTour ct on c.cID = ct.cID JOIN Tour t on t.tID = ct.tID;
```

```
CREATE VIEW client_hotel
```

```
as SELECT cSurname, cName, cDate, cTerm, cPeople, hName, hPrice FROM  
Client c
```

```
JOIN CTour ct on c.cID = ct.cID JOIN Tour t on t.tID = ct.tID JOIN Hotel h on  
h.hName = t.tHotelName;
```

```
CREATE VIEW Info_for_transport
```

```
as SELECT trType, rtDate, cSurname, cName, cPeople FROM Transport tr
```

```
JOIN ReservTransp rt on tr.trID = rt.trID JOIN CTour ct on ct.tID = rt.tID JOIN  
Client c on ct.cID = c.cID;
```

```
CREATE VIEW Tour_price
```

```
as SELECT tID, tPrice, tHotelName, tDetails FROM Tour
```

```
GROUP BY tID HAVING tPrice <= 36000;
```

```
CREATE VIEW hotel_info
```

```
as SELECT hID, hName, hPrice, hRoom FROM Hotel;
```

Створення тригерів:

```
CREATE TRIGGER lowerHotelN_tour
```

```
AFTER INSERT on Tour
```

```
BEGIN
```

```
UPDATE Tour
```

```
SET tHotelName = lower(tHotelName);
```

```
END;
```

```
CREATE TRIGGER lowerName_Hotel
```

```
AFTER INSERT on Hotel
```

```
BEGIN
```

```
UPDATE Hotel
```

```
SET hName = lower(hName);
```

```
END;
```

```
CREATE TRIGGER client_password
```

```
BEFORE INSERT on Client
```

```
BEGIN
```

```
SELECT CASE
```

```
WHEN length(NEW.cPassword) < 8
```

```
THEN
```

```
RAISE(ABORT, "Invalid Password")
```

```
END;
```

```
END;
```

```
CREATE TRIGGER check_info
```

```
BEFORE DELETE on Tour
```

```
BEGIN
```

```
SELECT CASE
```

```
WHEN OLD.tStatus = 1
```

```
THEN
```

```
RAISE(ABORT, "Check your information before delete")
```

```
END;
```

```
END;
```

```
/*triggers on views*/
```

```
CREATE TRIGGER hotel_info_insert
```

```
INSTEAD of INSERT on hotel_info
```

```
FOR EACH ROW
```

```
BEGIN
```

```
INSERT INTO Hotel VALUES (NEW.hID, NEW.hName, NEW.hPrice, null,  
null, NEW.hRoom);
```

```
END;
```

```
CREATE TRIGGER hotel_info_update
```

```
INSTEAD OF UPDATE OF hPrice ON hotel_info
```

```
FOR EACH ROW
```

```
BEGIN
```

```
UPDATE Hotel
```

```
SET hPrice = NEW.hPrice
```

```
WHERE hID = OLD.hID;
```

```
END;
```

```
CREATE TRIGGER hotel_info_delete
```

```
INSTEAD OF DELETE on hotel_info
```

```
FOR EACH ROW
```

```
BEGIN
```

```
DELETE FROM Hotel
```

```
WHERE hID = OLD.hID;
```

```
END;
```

Створення індексів:

```
SELECT trType, rtDate, cSurname, cName, cPeople FROM Transport tr  
JOIN ReservTransp rt on tr.trID = rt.trID JOIN CTour ct on ct.tID = rt.tID JOIN  
Client c on ct.cID = c.cID;
```

```
SELECT tHotelName, count(*) as popularity from Tour t  
JOIN CTour ct on ct.tID = t.tID  
GROUP BY tHotelName;
```

```
SELECT trType as Type_transport, group_concat(cSurname) as client_info,  
tHotelName FROM Transport tr  
JOIN ReservTransp rt on tr.trID = rt.trID JOIN CTour ct on ct.tID = rt.tID  
JOIN Client c on ct.cID = c.cID JOIN Tour t on ct.tID = t.tID  
GROUP BY tHotelName;
```

```
CREATE INDEX client_index on Client (cSurname, cID);  
CREATE INDEX ctour_index on CTour (cDate);  
CREATE INDEX tour_index on Tour (tHotelName, tID);  
CREATE INDEX hotel_index on Hotel (hID, hName);  
CREATE INDEX transport_index on Transport (trType);
```