**ЛАБОРАТОРНА РОБОТА №7**

**НОРМАЛІЗАЦІЯ ДАНИХ В T-SQL**

Мета: закріпити практичні навички роботи з T-SQL через процес нормалізації даних для покращення структури бази даних та забезпечення ефективного управління даними.

**ТЕОРЕТИЧНІ ОСНОВИ**

Нормалізація даних є ключовим процесом у проектуванні реляційних баз даних, спрямованим на організацію структури даних таким чином, щоб зменшити дублювання та забезпечити цілісність даних. Цей процес включає перетворення великих таблиць з повторюваними даними в кілька менших, взаємопов'язаних таблиць.

Поняття Нормалізації

Нормалізація баз даних включає застосування серії правил або "форм нормалізації", кожна з яких вирішує певні типи проблем дублювання або цілісності. Основною метою нормалізації є усунення аномалій вставки, оновлення, та видалення, які можуть виникнути під час роботи з даними.

Аномалії даних в базах даних виникають, коли структура бази даних дозволяє виникнення непослідовності, некоректності або неповноти даних через операції вставки, видалення або оновлення. Ці аномалії можуть призвести до втрати важливої інформації, дублювання даних та інших проблем, що ускладнюють управління даними. Розглянемо детальніше основні типи аномалій даних:

*1. Аномалія вставки (Insertion Anomaly)*

Виникає, коли обмеження бази даних перешкоджають додаванню нових даних до таблиці без вказівки інших, потенційно непов'язаних або невідомих на момент вставки, даних.

Приклад: у таблиці, яка зберігає інформацію про працівників і відділи, де кожен працівник має бути прив'язаний до відділу, не можна буде додати інформацію про нового працівника, доки не буде створено відділ, до якого цей працівник має бути прив'язаний.

*2. Аномалія оновлення (Update Anomaly)*

Виникає, коли оновлення даних в одному місці таблиці потребує множинних оновлень в інших місцях, щоб зберегти цілісність даних.

Приклад: якщо інформація про відділ дублюється в багатьох записах працівників, зміна назви відділу в одному записі вимагає оновлення всіх записів, що стосуються цього відділу, щоб уникнути непослідовності.

*3. Аномалія видалення (Deletion Anomaly)*

Виникає, коли видалення даних призводить до втрати інших, потенційно важливих, даних.

Приклад у тій же таблиці працівників, видалення останнього працівника відділу може призвести до непрямої втрати інформації про сам відділ, оскільки вона була прив'язана до записів працівників.

Вирішення проблем аномалій відбувається через нормалізацію даних. Нормалізація даних – процес організації структури бази даних, який допомагає уникнути цих аномалій. Через нормалізацію, дані розподіляються по різних таблицях з визначенням відносин між ними, що забезпечує:

- ***Зниження дублювання даних:*** дані зберігаються тільки в одному місці, що усуває потребу в їх множинних оновленнях.

- ***Забезпечення цілісності даних:*** зв'язки між таблицями забезпечують відповідність даних.

- ***Гнучкість у вставці та видаленні даних:*** Структура даних дозволяє додавати та видаляти інформацію без ризику втрати важливих даних.

Процес нормалізації зазвичай включає перехід через кілька "нормальних форм", кожна з яких вирішує певні типи аномалій і покращує структуру бази даних.

**Форми Нормалізації**

**Класичні форми нормалізації**

*Класичний варіант нормалізації баз даних зосереджується на трьох основних формах нормалізації: першій (1НФ), другій (2НФ) та третій (3НФ). Ці форми були розроблені Едгаром Коддом у 1970-х роках і слугували основою для проектування реляційних баз даних. Основна мета цих форм – усунення дублювання даних та забезпечення цілісності баз даних через визначення відносин між даними.*

*1. Перша нормальна форма (1НФ):* вимагає, щоб значення кожного поля були атомарними (нероздільними) і кожен рядок був унікальним. Це забезпечує основу для подальшої нормалізації.

*2. Друга нормальна форма (2НФ):* досягається шляхом усунення часткових залежностей, тобто забезпечення того, щоб неключові поля були залежними від усього первинного ключа, а не лише його частин.

*3. Третя нормальна форма (3НФ):* вимагає відсутності транзитивних залежностей між неключовими полями та первинним ключем. Тобто, кожен атрибут має бути залежним лише від первинного ключа.

**Додаткові форми нормалізації**

*Додаткові нормальні форми виникли як розвиток базових концепцій нормалізації для вирішення специфічних ситуацій, які не повністю покривалися трьома основними формами. Ці додаткові форми нормалізації були запропоновані пізніше, щоб вирішити проблеми, пов'язані з більш складними зв'язками між даними.*

*По мірі розвитку теорії реляційних баз даних та практичного досвіду виявилося, що трьох базових нормальних форм недостатньо для вирішення усіх проблем проектування. Специфічні сценарії, які включають складні зв'язки та залежності між даними, потребували додаткового уточнення та розширення понять нормалізації. Це призвело до розробки 4НФ і 5НФ, а також подальших концепцій, таких як нормальна форма доменів ключів (DKNF), що є ще одним кроком у розвитку теорії нормалізації.*

*Застосування всіх цих форм нормалізації дозволяє розробникам створювати добре структуровані, гнучкі та ефективні схеми баз даних, які забезпечують високий рівень цілісності та продуктивності даних.*

*4. Нормальна форма Бойса-Кодда (БКНФ):* схожа на 3НФ, але вимагає, щоб кожна детермінанта була кандидатом на ключ. Це допомагає уникнути додаткових аномалій, які не покриваються 3НФ.

*5. Четверта нормальна форма (4НФ):* усуває небажані мультивалентні залежності, забезпечуючи, що записи мають однозначні відношення між атрибутами.

*6. П'ята нормальна форма (5НФ):* Вирішує проблеми, пов'язані зі складними взаємозв'язками між таблицями, забезпечуючи, що будь-які залежності можуть бути відтворені через приєднання (join) менших таблиць.

Нормалізація забезпечує кілька ключових переваг для реляційних баз даних:

- Зменшення дублювання даних: зберігання інформації в окремих, але пов'язаних таблицях допомагає уникнути дублювання.

- Підвищення цілісності даних: через визначення чітких зв'язків між таблицями забезпечується точність та послідовність даних.

- Спрощення управління даними: щновлення, вставка та видалення даних стають більш простими та безпечними, зменшуючи ризик аномалій.

Нормалізація — це фундаментальний процес у проектуванні баз даних, який вимагає глибокого розуміння зв'язків між даними. Ретельне планування та застосування принципів нормалізації може значно підвищити ефективність та надійність систем баз даних.

**ЗАВДАННЯ**

Частина 1: аналіз існуючої структури

1. Створення денормалізованої таблиці:

- Створіть денормалізовану таблицю `Orders`, яка містить поля: `OrderID`, `CustomerName`, `ProductName`, `ProductCategory`, `Quantity`, `UnitPrice`, `OrderDate`.

- Додайте в таблицю довільні записи (мінімум 10), які містять повторювані значення в полях `CustomerName`, `ProductName`, і `ProductCategory`.

2. Аналіз проблем денормалізованої таблиці:

- Опишіть потенційні проблеми денормалізованої таблиці, такі як аномалії вставки, оновлення, видалення.

Частина 2: процес нормалізації

3. Перехід до 1НФ (Перша нормальна форма):

- Розробіть структуру для переходу таблиці `Orders` до 1НФ, що передбачає унікальність записів та атомарність значень полів.

- Створіть необхідні таблиці з унікальними первинними ключами.

4. Перехід до 2НФ (Друга нормальна форма):

- Аналізуйте створені таблиці на предмет часткових залежностей. Розробіть структуру для усунення цих залежностей, розділяючи дані на додаткові таблиці.

- Виконайте модифікацію структури бази даних для відповідності 2НФ, створіть необхідні таблиці та визначте зв'язки між ними.

5. Перехід до 3НФ (Третя нормальна форма):

- Виявіть транзитивні залежності у ваших таблицях. Розробіть структуру таблиць, яка усуває транзитивні залежності.

- Оновіть структуру бази даних, щоб відповідати вимогам 3НФ.

Частина 3: реалізація та аналіз

6. Створення нормалізованої бази даних:

- Використовуючи T-SQL, створіть оновлені таблиці відповідно до вашого проекту нормалізації.

- Заповніть таблиці даними, відображаючи зв'язки між ними.

7. Тестування і аналіз:

- Виконайте запити для вставки, оновлення, та видалення даних в нормалізованій структурі. Продемонструйте, як нормалізація вирішує проблеми аномалій даних.

- Проаналізуйте ефективність вашої нормалізованої структури у порівнянні з денормалізованою таблицею.

Вимоги до звіту:

- Опис процесу нормалізації, включаючи початкову структуру та кожен крок нормалізації з обґрунтуванням.

- SQL-скрипти для створення таблиць і вставки даних на кожному етапі нормалізації.

- Приклади запитів для вставки, оновлення, та видалення даних.

- Аналіз отриманих результатів та переваг нормалізованої структури бази даних.

**КОНТРОЛЬНІ ЗАПИТАННЯ**

1. Що таке нормалізація даних? Опишіть загальну мету процесу нормалізації в контексті реляційних баз даних.

2. Які існують форми нормалізації? Наведіть опис кожної нормальної форми, починаючи з 1НФ до 3НФ, і поясніть, які проблеми кожна з них вирішує.

3. Що таке денормалізація даних? Поясніть, чому і в яких випадках може бути вигідно застосувати денормалізацію в базі даних.

4. Які існують типи аномалій даних? Опишіть аномалії вставки, оновлення та видалення, надаючи приклади для кожної.

5. Як нормалізація допомагає уникнути аномалій даних? Продемонструйте, як застосування різних нормальних форм може запобігти появі аномалій вставки, оновлення, та видалення.

6. Чому може бути корисною денормалізація в контексті продуктивності запитів? Надайте аргументи на користь денормалізації для оптимізації читання даних.

7. Які можуть бути наслідки надмірної денормалізації? Обговоріть потенційні ризики або проблеми, пов'язані з денормалізацією даних.

8. Які стратегії можуть бути застосовані для управління аномаліями в денормалізованих базах даних? Розгляньте можливі підходи до управління даними в умовах, коли денормалізація є необхідною.