Лекція 13. Правила нормалізації, ліквідація аномалій вилучення і додавання.

**НОРМАЛІЗАЦІЯ**

###### Постановка задачі

*Нормалізація* – це процедура визначення того, які атрибути зв'язані у відношенні. Одна з головних задач при розробці реляційної БД – об'єднання в одному відношенні тих атрибутів, які зв'язані між собою (між якими є функціональні залежності). Нормалізація являє собою поетапний процес заміни сукупності відношень іншою сукупністю (схемою), в

якій відношення мають просту і регулярну структуру. Результатом нормалізаії є логічна модель БД.

Надлишковість даних в БД є небажаним явищем, оскільки призводить до збільшення об'єму пам'яті, уповільнює роботу БД. Надлишковість даних є результатом в першу чергу дублювання даних. Розрізняють *незбиткове* та *збиткове* дублювання даних. Повністю усувати надлишковість не потрібно, оскільки при цьому неможливо буде підтримувати БД як єдине ціле. Слід тільки мінімізувати надлишковість, залишивши необхідне дублювання даних.

Дублювання даних створює проблеми при виконанні операцій з БД. Ці проблеми виникають при спробі зробити операції: редагування, додавання або вилучення даних.

*Аномаліями* називається така ситуація в БД, яка призводить до протирічь у БД, або суттєво ускладнює обробку даних. Розрізняють аномалії модифікації, додавання і вилучення.

*Приклад*. Розглянемо відношення *Студент* (табл. 7.1).

*Таблиця 7.1*

###### Студент

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Номер залікової книжки | Прізвище | Група | Факультет | Декан |
| 1010 | Бойко | ІТП-31 | АІТ | Барков |
| 2020 | Лемешко | ІУСТ-22 | АІТ | Барков |
| 1030 | Шевченко | ІТП-31 | АІТ | Барков |
| 1121 | Петренко | БМО-32 | АІТ | Барков |
| 2231 | Грицюк | ТБ-21 | БТ | Тимчук |

*Аномалія модифікації* виникає при спробі змінити прізвище декана. В цій ситуації необхідно переглянути всі кортежі. При великих розмірах БД це потребує значного часу,

при цьому можливі помилки (у разі невірного введення прізвища), які порушать цілісність БД.

*Аномалія додавання* виникає при додаванні інформації про нового студента, при цьому необхідно вводити інформацію, яка вже є в БД: назва факультету, прізвище декана. Крім того неможливо створити нову групу поки не існує студентів, які в ній займаються.

*Аномалія вилучення* виникає при спробі вилучити дані про студента, який в групі поки ще один, наприклад Лемешко. В цьому випадку зникне інформація про групу ІУСТ-22.

Виконання декомпозиції наведеного відношення дозволяє позбутися вищеозначених аномалій (табл. 7.2. 7.4).

*Таблиця 7.2*

###### Студент

|  |  |  |
| --- | --- | --- |
| Номер залікової книжки | Прізвище | Група |
| 1010 | Бойко | ІТП-31 |
| 2020 | Лемешко | ІУСТ-22 |
| 1030 | Шевченко | ІТП-31 |
| 1121 | Петренко | БМО-32 |
| 2231 | Грицюк | ТБ-21 |

**Група**

|  |  |
| --- | --- |
| Група | Факультет |
| ІТП-31 | АІТ |
| ІУСТ-22 | АІТ |
| БМО-32 | АІТ |
| ТБ-21 | БТ |

###### Факультет

*Таблиця 7.3*

*Таблиця 7.4*

|  |  |
| --- | --- |
| Факультет | Декан |
| АІТ | Барков |
| БТ | Тимчук |

Процес проектування БД з використанням декомпозиції являє собою процес послідовної нормалізації схем відношень, при цьому кожна наступна ітерація відповідає нормальній формі більш високого рівня і має кращі властивості у порівнянні з попередньою. Кожній нормальній формі (НФ) відповідає деякий набір обмежень. Визначають такі нормальні форми: 1НФ, 2НФ, 3НФ, НФБК (нормальна форма Бойса- Кодда), 4НФ, 5НФ.

При виконанні декомпозиції зберігається множина вихідних функціональних залежностей між атрибутами і виконується зворотність. З*воротність* означає можливість відновлення вихідної схеми. *Функціональні залежності* відображають зв'язки між атрибутами, які властиві реальному об'єкту.

Атрибут *B функціонально* залежить від *A*, якщо кожному значенню *A* відповідає в точності одне значення *B*. Математичний запис функціональної залежності (ФЗ): *A* *B*

*Приклад.* Функціональні залежності:

*Студент* → *Група; Група* → *Факультет; Викладач*, *Студент*, *Дисципліна* → *Оцінка*.

Якщо існує ФЗ *A* → *B*, то це означає, що у всіх кортежах з однаковим значенням атрибуту *A* атрибут *B* буде мати також одне й те ж значення. *A* і *B* можуть складатися з декількох атрибутів.