

ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені ІВАНА ФРАНКА

Факультет прикладної математики та інформатики

Бази даних та інформаційні системи

ЛАБОРАТОРНА РОБОТА №8

Віртуальні таблиці SQL (представлення VIEW)

Виконав:

Ст. Яцуляк Андрій

Група ПМІ-21

Оцінка

Перевірила:

доц. Малець Р.Б.

2023

Тема: Віртуальні таблиці SQL.

Мета роботи: Ознайомлення з поняттям Віртуальні таблиці SQL, їх створенням та застосуванням.

Завдання

Розробити базу даних для системи автоматизації шкільної бібліотеки. Система веде облік читачів, які реєструються в бібліотеці і можуть позичати книги. Для читачів зберігається прізвище, ім'я, контактні дані (адреси та телефони) та рейтинг (наскільки вчасно повертають позичені книги). Крім того база даних містить інформацію про книги, які зберігаються в бібліотеці, їх статус (доступна, кому позичена, не видається, в ремонті і т.д.) Бібліотекар може додавати нових читачів, книги і їх деталі, а також здійснювати пошук книг за авторами, видавництвами, назвами, статусом і т.д.

Розв'язання

```
1 CREATE OR REPLACE VIEW mixed_view AS
2 SELECT
3     b.book_id,
4     b.book_name,
5     a.autor_name || ' ' || a.autor_surname AS autor_full_name,
6     p.publisher_name,
7     c.genre
8 FROM
9     book b
10 JOIN
11     book_autor ba ON b.book_id = ba.book_id
12 JOIN
13     autor a ON ba.autor_id = a.autor_id
14 JOIN
15     publisher p ON b.publisher_id = p.publisher_id
16 JOIN
17     catalogue c ON b.catalogue_id = c.catalogue_id;
18
19
20 select * from mixed_view;
```

	book_id integer	book_name character varying (30)	autor_full_name text	publisher_name character varying (30)	genre character varying (15)
1	4	Mathematics	David Shram	Zhenyk	science
2	4	Mathematics	Adam Davies	Zhenyk	science
3	3	Historic	Petro Shchur	Pubproduction	historic
4	5	Literature	Taras Shevchenko	Ababahalamaga	literature
5	5	Literature	Mychailo Dovzenko	Ababahalamaga	literature
6	1	Physics	Mychailo Dovzenko	Nashe	science
7	2	Mathematics	Petro Shchur	Nonprod	science
8	19	Giometrix	Mykola Dniprovskii	Nonprod	science
9	20	Biology	Dmytro Vyshnevetskii	Mypub	science
10	21	Designer	Ivan Franko	Pronto	art
11	22	Kobzar	Taras Shevchenko	Ababahalamaga	literature
12	23	Newera	Dmytro Vyshnevetskii	Litproduct	historic
13	20	Biology	Mykola Dniprovskii	Mypub	science
14	25	Human	Adam Kopernyk	Ababahalamaga	science
15	25	Human	Robert Frost	Ababahalamaga	science
16	25	Human	Maya Angelou	Ababahalamaga	science

Цей код створює вид (view) під назвою `mixed_view`, який поєднає дані з кількох таблиць і надає наступну інформацію:

У виді `mixed_view` вибираються наступні поля з таблиць:

`book_id` з таблиці `book`.

`book_name` з таблиці `book`.

`autor_name` та `autor_surname` з таблиці `autor`, а також об'єднані в одне поле `autor_full_name` за допомогою конкатенації.

`publisher_name` з таблиці `publisher`,

`genre` з таблиці `public.catalogue`.

З'єднання таблиць виконується за допомогою операції `JOIN` з відповідними умовами зв'язку між ними.

Вид `mixed_view` надає змішану (mixed) інформацію про книги, включаючи їх ідентифікатор, назву, повне ім'я автора (об'єднання `autor_name` та `autor_surname`), назву видавця та жанр книги.

Змішані види дозволяють отримувати комплексну інформацію, яка походить з різних таблиць, об'єднаних за певними зв'язками. Вони полегшують отримання цільових даних та спрощують подальші операції з ними, такі як вставка, оновлення та видалення.

```

1 CREATE MATERIALIZED VIEW materialized_view AS
2 SELECT
3     b.book_id,
4     b.book_name,
5     p.publisher_name,
6     c.genre
7 FROM
8     book b
9 JOIN
10    publisher p ON b.publisher_id = p.publisher_id
11 JOIN
12    catalogue c ON b.catalogue_id = c.catalogue_id;
13
14 select * from materialized_view

```

	book_id integer	book_name character varying (30)	publisher_name character varying (30)	genre character varying (15)
1	1	Physics	Nashe	science
2	2	Mathematics	Nonprod	science
3	4	Mathematics	Zhenyk	science
4	19	Giometrix	Nonprod	science
5	20	Biology	Mypub	science
6	21	Designer	Pronto	art
7	22	Kobzar	Ababahalamaga	literature
8	23	Newera	Litproduct	historic
9	5	Literature	Ababahalamaga	literature
10	3	Historic	Pubproduction	historic
11	25	Human	Ababahalamaga	science

Вид materialized_view базується на таблиці book та з'єднується з таблицями publisher та catalogue.

При створенні виду дані витягаються з бази даних і зберігаються в матеріалізованому виді. Це означає, що дані будуть зберігатись фізично в виді та оновлюватись за необхідністю.

Матеріалізовані види надають можливість зберігати результати складних запитів та оновлювати їх при необхідності, що може привести до покращення продуктивності запитів. Проте, варто пам'ятати, що матеріалізовані види займають додатковий

простір на диску та потребують оновлення для відображення оновлених даних у випадку змін в базі даних.

```
1  |-- Створення змінюваного віртуального представлення
2  CREATE OR REPLACE VIEW modifiable_view AS
3  SELECT
4      book_id,
5      book_name,
6      publisher_id,
7      catalogue_id,
8      status,
9      pub_year
10 FROM
11     book;
12
13 -- Додавання правила для вставки даних
14 CREATE OR REPLACE RULE insert_rule AS
15     ON INSERT TO modifiable_view
16     DO INSTEAD
17         INSERT INTO public.book (book_name, publisher_id, catalogue_id, status, pub_year)
18         VALUES (NEW.book_name, NEW.publisher_id, NEW.catalogue_id, NEW.status, NEW.pub_year);
19
20 -- Додавання правила для оновлення даних
21 CREATE OR REPLACE RULE update_rule AS
22     ON UPDATE TO modifiable_view
23     DO INSTEAD
24         UPDATE book
25         SET book_name = NEW.book_name,
26             publisher_id = NEW.publisher_id,
27             catalogue_id = NEW.catalogue_id,
28             status = NEW.status,
29             pub_year = NEW.pub_year
30         WHERE book_id = OLD.book_id;
31
32 -- Додавання правила для видалення даних
33 CREATE OR REPLACE RULE delete_rule AS
34     ON DELETE TO modifiable_view
35     DO INSTEAD
36         DELETE FROM book
37         WHERE book_id = OLD.book_id;
38
39 select * from modifiable_view
```

	book_id integer	book_name character varying (30)	publisher_id integer	catalogue_id integer	status text	pub_year smallint
1	1	Physics	2	2	borrowing	2021
2	2	Mathematics	5	2	borrowing	2022
3	4	Mathematics	4	2	borrowing	2023
4	19	Giometrix	5	2	borrowing	2021
5	20	Biology	7	2	borrowing	2022
6	21	Designer	8	1	borrowing	2023
7	22	Kobzar	1	5	repair	2023
8	23	Newera	6	4	borrowing	2021
9	5	Literature	1	5	borrowing	2021
10	3	Historic	3	4	borrowing	2022
11	25	Human	1	2	free	2020

В даному прикладі створюється змінюване віртуальне представлення `modifiable_view`, яке базується на таблиці `book`. Потім за допомогою правил (rules) визначається поведінка представлення при вставці, оновленні та видаленні даних.

Правило `insert_rule` виконується при спробі вставки даних у віртуальне представлення. Замість вставки даних безпосередньо у представлення, воно вставляє дані в базову таблицю `book`.

Правило `update_rule` виконується при спробі оновлення даних у віртуальному представленні. Замість оновлення даних у представленні, воно виконує оновлення в базовій таблиці `book`.

Правило `delete_rule` виконується при спробі видалення даних з віртуального представлення. Замість видалення даних з представлення, воно видаляє дані з базової таблиці `public.book`.

Віртуальне представлення `modifiable_view` дозволяє виконувати зміни в таблиці `public.book` через відповідні правила. Коли дані вставляються, оновлюються або видаляються у модифікованому представленні, правила замість безпосередньої маніпуляції з даними виконують відповідні операції в базовій таблиці `public.book`.

Правило `insert_rule` виконується, коли спробують вставити дані у віртуальне представлення. Замість вставки даних безпосередньо в представлення, воно вставляє дані в базову таблицю `public.book`.

Правило `update_rule` виконується, коли спробують оновити дані у віртуальному представленні. Замість оновлення даних у представленні, воно виконує оновлення в базовій таблиці `public.book`.

Правило `delete_rule` виконується, коли спробують видалити дані з віртуального представлення. Замість видалення даних з представлення, воно видаляє дані з базової таблиці `public.book`.

Таким чином, використовуючи це віртуальне представлення та правила, можна контролювати зміни в таблиці `public.book` через модифіковане представлення, забезпечуючи більш гнучкий та контрольований спосіб редагування даних.

Висновок: на цій лабораторній роботі я ознайомився з поняттям віртуальних таблиць SQL, їх створенням та застосуванням.