

Язык SQL, дз – 8

Бурмашев Григорий, БПМИ-208

16 декабря 2022 г.

Номер 2

Задание. Модифицируйте сценарий выполнения транзакций: в первой транзакции вместо фиксации изменений выполните их отмену с помощью команды ROLLBACK и посмотрите, будет ли удалена строка и какая конкретно.

Вместо фиксации произвожу откат, код, произведенный в первом терминале:

```
demo=# BEGIN;
BEGIN
demo=# SELECT *
  FROM aircrafts_tmp
  WHERE range < 2000;
aircraft_code |          model          | range
-----+-----+-----
CN1           | Сессна 208 Караван     | 1200
(1 row)

demo=# UPDATE aircrafts_tmp
  SET range = 2100
  WHERE aircraft_code = 'CN1';
UPDATE 1
demo=# UPDATE aircrafts_tmp
  SET range = 1900
  WHERE aircraft_code = 'CR2';
UPDATE 1
demo=# ROLLBACK;
ROLLBACK
demo=#
```

Во втором терминале же получили следующий результат:

```

demo=# BEGIN;
BEGIN
demo=# SELECT *
  FROM aircrafts_tmp
  WHERE range < 2000;
 aircraft_code |          model          | range
-----+-----+-----
  CN1          | Сессна 208 Караван    | 1200
(1 row)

demo=# DELETE FROM aircrafts_tmp WHERE range < 2000;
DELETE 1
demo=# SELECT * FROM aircrafts_tmp;
 aircraft_code |          model          | range
-----+-----+-----
  773          | Боинг 777-300         | 11100
  763          | Боинг 767-300         | 7900
  SU9          | Сухой Суперджет-100   | 3000
  320          | Аэробус A320-200      | 5700
  321          | Аэробус A321-200      | 5600
  319          | Аэробус A319-100      | 6700
  733          | Боинг 737-300         | 4200
  CR2          | Бомбардье CRJ-200     | 2700
(8 rows)

```

Видим, что в результате отката первой транзакции её результаты не сохранились, а следовательно вторая транзакция привела к удалению строки с **model = Сессна 208 Караван**, поскольку у этого самолета изначальная дистанция (до 1 транзакции) меньше 2000

Номер 3

- 3.* Когда говорят о таком феномене, как потерянное обновление, то зачастую в качестве примера приводится операция UPDATE, в которой значение какого-то атрибута изменяется с применением одного из действий арифметики. Например:

```
UPDATE aircrafts_tmp
  SET range = range + 200
  WHERE aircraft_code = 'CR2';
```

При выполнении двух и более подобных обновлений в рамках параллельных транзакций, использующих, например, уровень изоляции Read Committed, будут учтены все такие изменения (что и было показано в тексте главы). Очевидно, что потерянное обновления не происходит.

Предположим, что в одной транзакции будет просто присваиваться новое значение, например, так:

```
UPDATE aircrafts_tmp
  SET range = 2100
  WHERE aircraft_code = 'CR2';
```

А в параллельной транзакции будет выполняться аналогичная команда:

```
UPDATE aircrafts_tmp
  SET range = 2500
  WHERE aircraft_code = 'CR2';
```

283

Глава 9. Транзакции

Очевидно, что сохранится только одно из значений атрибута range. Можно ли говорить, что в такой ситуации имеет место потерянное обновление? Если оно имеет место, то что можно предпринять для его недопущения? Обоснуйте ваш ответ.

Да, в такой ситуации имеет место потерянное обновление, поскольку одна из двух транзакций перезапишет данные, которые были обновлены другой транзакцией, вероятно пользователь хочет видеть от базы данных другого поведения (например запрета выполнения таких транзакций одновременно)