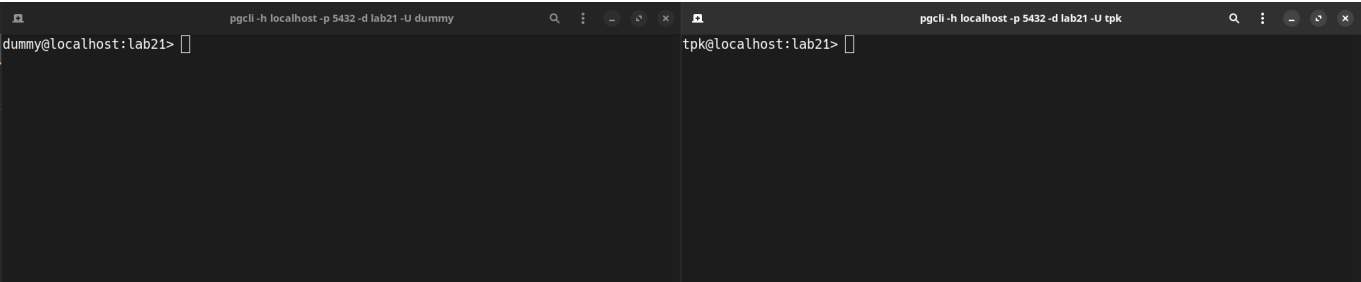


Лабораторная работа №2-2: «Транзакции. Изоляция транзакций»

Выполнил студент группы Б20-505, Полищук Максим.

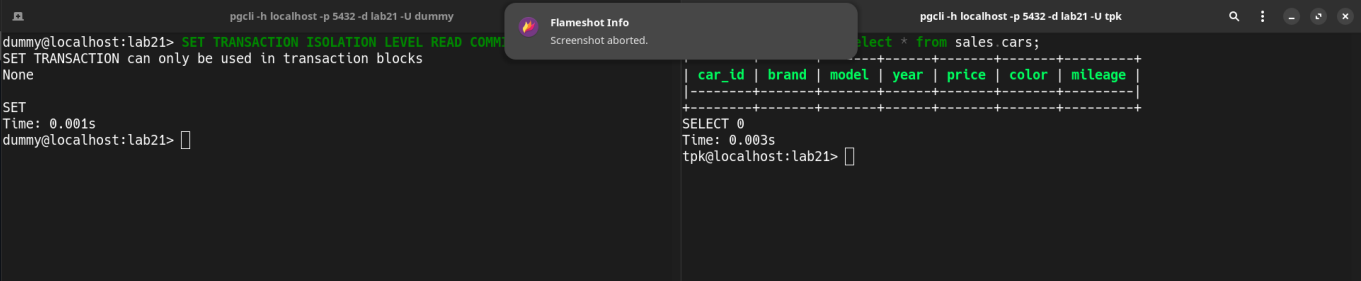
В данной лабораторной работе использовалась та же схема базы данных, что и в лабораторной работе [2_1](#)

Одновременно подключимся к одной и той же БД с двух разных сессий (**dummy** и **tpk**)

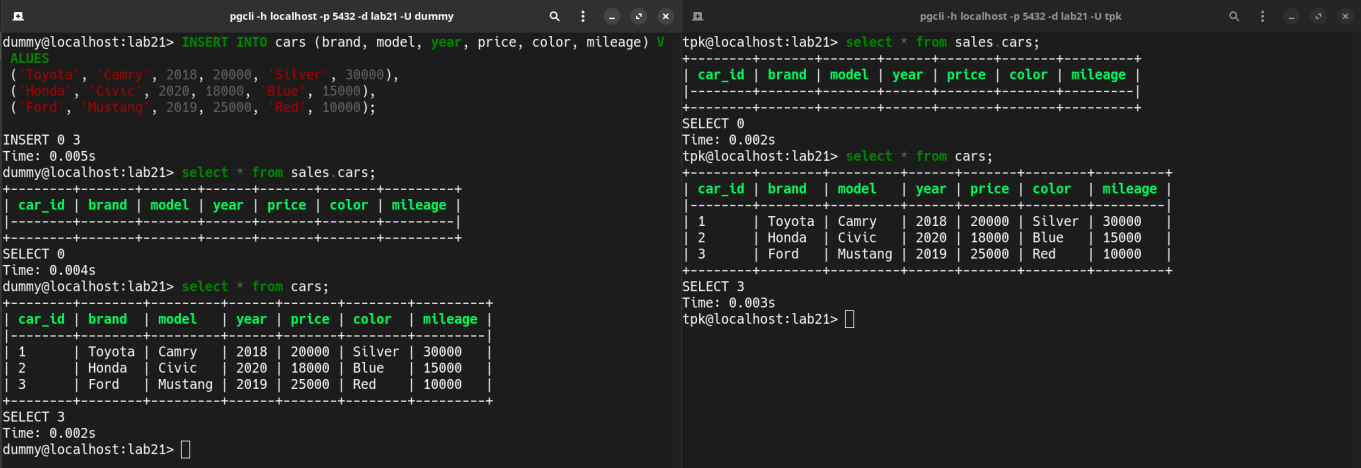


Создать две сессии в Вашей базе данных. Начать транзакцию на уровне изоляции READ COMMITTED в одной из сессий. Изменить и добавить какие-либо данные.

В **dummy** выставлен режим изоляции READ COMMITET, данных в таблице **cars** изначально нет



Дубль 1:



| Ничего не работало т.к. PostgreSQL имеет изначально режим **autocommit**

Дубль 2: Изначально таблица **clients** пустая:

```
pgcli -h localhost -p 5432 -d lab21 -U dummy
dummy@localhost:lab21> START TRANSACTION ISOLATION LEVEL READ COMMITTED;
START TRANSACTION
Time: 0.000s
dummy@localhost:lab21> select * from sales.clients;
+-----+-----+-----+-----+-----+
| client_id | first_name | last_name | phone_number | email |
+-----+-----+-----+-----+-----+
SELECT 0
Time: 0.002s
dummy@localhost:lab21>
```

```
pgcli -h localhost -p 5432 -d lab21 -U tpk
tpk@localhost:lab21> select * from sales.clients;
+-----+-----+-----+-----+-----+
| client_id | first_name | last_name | phone_number | email |
+-----+-----+-----+-----+-----+
SELECT 0
Time: 0.003s
tpk@localhost:lab21>
```

Далее данные вставили только внутри сессии **dummy** потому что выставленный режим изоляции требует прописать **COMMIT**, чтобы сохранить изменения.

```
pgcli -h localhost -p 5432 -d lab21 -U dummy
dummy@localhost:lab21> START TRANSACTION ISOLATION LEVEL READ COMMITTED;
START TRANSACTION
Time: 0.000s
dummy@localhost:lab21> select * from sales.clients;
+-----+-----+-----+-----+-----+
| client_id | first_name | last_name | phone_number | email |
+-----+-----+-----+-----+-----+
SELECT 0
Time: 0.002s
dummy@localhost:lab21> INSERT INTO clients (first_name, last_name, phone_number, email)
VALUES
( 'John', 'Doe', '123-456-7890', 'john.doe@example.com' ),
( 'Jane', 'Smith', '234-567-8901', 'jane.smith@example.com' ),
( 'Emily', 'Johnson', '345-678-9012', 'emily.johnson@example.com' );
INSERT 0 3
Time: 0.001s
dummy@localhost:lab21> select * from sales.clients;
+-----+-----+-----+-----+-----+
| client_id | first_name | last_name | phone_number | email |
+-----+-----+-----+-----+-----+
SELECT 0
Time: 0.002s
dummy@localhost:lab21> select * from clients;
+-----+-----+-----+-----+-----+
| client_id | first_name | last_name | phone_number | email |
+-----+-----+-----+-----+-----+
| 1 | John | Doe | 123-456-7890 | john.doe@example.com |
| 2 | Jane | Smith | 234-567-8901 | jane.smith@example.com |
| 3 | Emily | Johnson | 345-678-9012 | emily.johnson@example.com |
+-----+-----+-----+-----+-----+
SELECT 3
Time: 0.002s
dummy@localhost:lab21>
```

```
pgcli -h localhost -p 5432 -d lab21 -U tpk
tpk@localhost:lab21> select * from sales.clients;
+-----+-----+-----+-----+-----+
| client_id | first_name | last_name | phone_number | email |
+-----+-----+-----+-----+-----+
SELECT 0
Time: 0.003s
tpk@localhost:lab21> select * from clients;
+-----+-----+-----+-----+-----+
| client_id | first_name | last_name | phone_number | email |
+-----+-----+-----+-----+-----+
SELECT 0
Time: 0.002s
tpk@localhost:lab21>
```

Завершаем транзакцию при помощи **ROLLBACK**:

```
pgcli -h localhost -p 5432 -d lab21 -U dummy
dummy@localhost:lab21> ROLLBACK;
ROLLBACK
Time: 0.001s
dummy@localhost:lab21> select * from clients;
+-----+-----+-----+-----+-----+
| client_id | first_name | last_name | phone_number | email |
+-----+-----+-----+-----+-----+
SELECT 0
Time: 0.002s
dummy@localhost:lab21>
```

```
pgcli -h localhost -p 5432 -d lab21 -U tpk
tpk@localhost:lab21> select * from clients;
+-----+-----+-----+-----+-----+
| client_id | first_name | last_name | phone_number | email |
+-----+-----+-----+-----+-----+
SELECT 0
Time: 0.002s
tpk@localhost:lab21>
```

| **ROLLBACK** отменяет изменения внесенные в последней транзакции, поэтому даже из первой сессии данные пропали и вернулись к тому состоянию, которое было до начала транзакции

Завершаем транзакцию при помощи **COMMIT**:

```
pgcli -h localhost -p 5432 -d lab21 -U dummy
dummy@localhost:lab21> INSERT INTO clients (first_name, last_name, phone_number, email)
VALUES
( 'John', 'Doe', '123-456-7890', 'john.doe@example.com' ),
( 'Jane', 'Smith', '234-567-8901', 'jane.smith@example.com' ),
( 'Emily', 'Johnson', '345-678-9012', 'emily.johnson@example.com' );
INSERT 0 3
Time: 0.004s
dummy@localhost:lab21> COMMIT;
there is no transaction in progress
None

COMMIT
Time: 0.000s
dummy@localhost:lab21> select * from clients;
+-----+-----+-----+-----+-----+
| client_id | first_name | last_name | phone_number | email |
+-----+-----+-----+-----+-----+
| 4 | John | Doe | 123-456-7890 | john.doe@example.com |
| 5 | Jane | Smith | 234-567-8901 | jane.smith@example.com |
| 6 | Emily | Johnson | 345-678-9012 | emily.johnson@example.com |
+-----+-----+-----+-----+-----+
SELECT 3
Time: 0.002s
dummy@localhost:lab21>
```

```
pgcli -h localhost -p 5432 -d lab21 -U tpk
tpk@localhost:lab21> select * from clients;
+-----+-----+-----+-----+-----+
| client_id | first_name | last_name | phone_number | email |
+-----+-----+-----+-----+-----+
| 4 | John | Doe | 123-456-7890 | john.doe@example.com |
| 5 | Jane | Smith | 234-567-8901 | jane.smith@example.com |
| 6 | Emily | Johnson | 345-678-9012 | emily.johnson@example.com |
+-----+-----+-----+-----+-----+
SELECT 3
Time: 0.002s
tpk@localhost:lab21>
```

| COMMIT подтверждаем внесенные за время транзакции изменения, поэтому эти данные можно увидеть и из сессии **tpk**

Выполнить те же операции для уровней изоляции REPEATABLE READ и SERIALIZABLE. Объяснить различия

REPEATABLE READ

Не смог понять как это так, но в левой сессии не должны отображаться внесенные изменения, должны отображаться данные на момент начала транзакции

```
> pgcli -h localhost -p 5432 -d lab2 -U tpk
Server: PostgreSQL 14.8
Version: 4.0.1
Home: http://pgcli.com
tpk@localhost:lab2> select * from employees;
+-----+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | position | hire_date | salary |
+-----+-----+-----+-----+-----+-----+
| 3           | Alice     | Brown    | Sales Manager | 2022-05-10 | 50000 |
| 4           | Bob       | White    | Mechanic     | 2021-07-15 | 40000 |
+-----+-----+-----+-----+-----+-----+
SELECT 0
Time: 0.004s
tpk@localhost:lab2> START TRANSACTION ISOLATION LEVEL REPEATABLE READ;
START TRANSACTION
Time: 0.001s
tpk@localhost:lab2> INSERT INTO employees (first_name, last_name, position, hire_date, salary) VALUES
( 'Alice', 'Brown', 'Sales Manager', '2022-05-10', 50000),
( 'Bob', 'White', 'Mechanic', '2021-07-15', 40000);
INSERT 0 2
Time: 0.001s
tpk@localhost:lab2> select * from employees;
+-----+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | position | hire_date | salary |
+-----+-----+-----+-----+-----+-----+
| 3           | Alice     | Brown    | Sales Manager | 2022-05-10 | 50000 |
| 4           | Bob       | White    | Mechanic     | 2021-07-15 | 40000 |
+-----+-----+-----+-----+-----+-----+
SELECT 2
Time: 0.002s
tpk@localhost:lab2> 
```

```
tpk@localhost:lab2> select * from employees;
+-----+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | position | hire_date | salary |
+-----+-----+-----+-----+-----+-----+
| 3           | Alice     | Brown    | Sales Manager | 2022-05-10 | 50000 |
| 4           | Bob       | White    | Mechanic     | 2021-07-15 | 40000 |
+-----+-----+-----+-----+-----+-----+
SELECT 0
Time: 0.002s
tpk@localhost:lab2> select * from employees;
+-----+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | position | hire_date | salary |
+-----+-----+-----+-----+-----+-----+
| 3           | Alice     | Brown    | Sales Manager | 2022-05-10 | 50000 |
| 4           | Bob       | White    | Mechanic     | 2021-07-15 | 40000 |
+-----+-----+-----+-----+-----+-----+
SELECT 0
Time: 0.002s
tpk@localhost:lab2> select * from employees;
+-----+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | position | hire_date | salary |
+-----+-----+-----+-----+-----+-----+
| 3           | Alice     | Brown    | Sales Manager | 2022-05-10 | 50000 |
| 4           | Bob       | White    | Mechanic     | 2021-07-15 | 40000 |
+-----+-----+-----+-----+-----+-----+
SELECT 0
Time: 0.002s
tpk@localhost:lab2> select * from employees;
+-----+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | position | hire_date | salary |
+-----+-----+-----+-----+-----+-----+
| 3           | Alice     | Brown    | Sales Manager | 2022-05-10 | 50000 |
| 4           | Bob       | White    | Mechanic     | 2021-07-15 | 40000 |
+-----+-----+-----+-----+-----+-----+
SELECT 0
Time: 0.002s
tpk@localhost:lab2> select * from employees;
+-----+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | position | hire_date | salary |
+-----+-----+-----+-----+-----+-----+
| 3           | Alice     | Brown    | Sales Manager | 2022-05-10 | 50000 |
| 4           | Bob       | White    | Mechanic     | 2021-07-15 | 40000 |
+-----+-----+-----+-----+-----+-----+
SELECT 0
Time: 0.002s
tpk@localhost:lab2> 
```

Причем уровень изоляции изменяется, но к примеру команду TRUNCATE выполнить не удастся

```
tpk@localhost:lab2> select * from employees;
+-----+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | position | hire_date | salary |
+-----+-----+-----+-----+-----+-----+
| 3           | Alice     | Brown    | Sales Manager | 2022-05-10 | 50000 |
| 4           | Bob       | White    | Mechanic     | 2021-07-15 | 40000 |
+-----+-----+-----+-----+-----+-----+
SELECT 2
Time: 0.002s
tpk@localhost:lab2> SHOW TRANSACTION ISOLATION LEVEL;
+-----+
| transaction_isolation |
+-----+
| repeatable read       |
+-----+
SHOW 1
Time: 0.002s
tpk@localhost:lab2> 
```

```
tpk@localhost:lab2> select * from employees;
+-----+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | position | hire_date | salary |
+-----+-----+-----+-----+-----+-----+
| 5           | Alice     | Brown    | Sales Manager | 2022-05-10 | 50000 |
| 6           | Bob       | White    | Mechanic     | 2021-07-15 | 40000 |
+-----+-----+-----+-----+-----+-----+
SELECT 2
Time: 0.002s
tpk@localhost:lab2> SHOW TRANSACTION ISOLATION LEVEL;
+-----+
| transaction_isolation |
+-----+
| read committed        |
+-----+
SHOW 1
Time: 0.002s
tpk@localhost:lab2> 
```

Ещё одна попытка к изменениям не привела:

pgcli -h localhost -p 5432 -d lab2 -U dummy

employee_id	first_name	last_name	position	hire_date	salary
5	Alice	Brown	Sales Manager	2022-05-10	50000
6	Bob	White	Mechanic	2021-07-15	40000

```
SELECT 2
Time: 0.004s
dummy@localhost:lab2> BEGIN;
BEGIN
Time: 0.000s
dummy@localhost:lab2> SET TRANSACTION ISOLATION LEVEL REPEATABLE READ;
SET
Time: 0.000s
dummy@localhost:lab2> INSERT INTO employees (first_name, last_name, position, hire_date, salary) VALUES
('Max', 'Borsh', 'Sales Manager', '2022-06-21', 66666);
INSERT 0 1
Time: 0.001s
dummy@localhost:lab2> select * from employees;

+-----+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | position | hire_date | salary |
+-----+-----+-----+-----+-----+-----+
| 5           | Alice      | Brown     | Sales Manager | 2022-05-10 | 50000  |
| 6           | Bob        | White     | Mechanic      | 2021-07-15 | 40000  |
| 8           | Max        | Borsh     | Sales Manager | 2022-06-21 | 66666  |
+-----+-----+-----+-----+-----+-----+

SELECT 3
Time: 0.002s
dummy@localhost:lab2> SHOW TRANSACTION ISOLATION LEVEL;

+-----+
| transaction_isolation |
+-----+
| repeatable read      |
+-----+

SHOW 1
Time: 0.002s
dummy@localhost:lab2> 
```

pgcli -h localhost -p 5432 -d lab2 -U tpk

```
tpk@localhost:lab2> select * from employees;

+-----+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | position | hire_date | salary |
+-----+-----+-----+-----+-----+-----+
| 5           | Alice      | Brown     | Sales Manager | 2022-05-10 | 50000  |
| 6           | Bob        | White     | Mechanic      | 2021-07-15 | 40000  |
+-----+-----+-----+-----+-----+-----+

SELECT 2
Time: 0.002s
tpk@localhost:lab2> select * from employees;

+-----+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | position | hire_date | salary |
+-----+-----+-----+-----+-----+-----+
| 5           | Alice      | Brown     | Sales Manager | 2022-05-10 | 50000  |
| 6           | Bob        | White     | Mechanic      | 2021-07-15 | 40000  |
+-----+-----+-----+-----+-----+-----+

SELECT 2
Time: 0.002s
tpk@localhost:lab2> 
```

SERIALIZABLE

При попытке одновременного изменения данных PostgreSQL это запрещает и требует завершения первоначальной транзакции:

pgcli -h localhost -p 5432 -d lab2 -U dummy

```
Time: 0.001s
dummy@localhost:lab2> SHOW TRANSACTION ISOLATION LEVEL;

+-----+
| transaction_isolation |
+-----+
| serializable          |
+-----+

SHOW 1
Time: 0.003s
dummy@localhost:lab2> select * from employees;

+-----+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | position | hire_date | salary |
+-----+-----+-----+-----+-----+-----+
| 5           | Alice      | Brown     | Sales Manager | 2022-05-10 | 50000  |
| 6           | Bob        | White     | Mechanic      | 2021-07-15 | 40000  |
+-----+-----+-----+-----+-----+-----+

SELECT 2
Time: 0.003s
dummy@localhost:lab2> UPDATE employees SET salary = 66666 WHERE first_name = 'Bob' AND last_name = 'White';
You're about to run a destructive command.
Do you want to proceed? [y/N]: y
Your call!
UPDATE 1
Time: 0.002s
dummy@localhost:lab2> select * from employees;

+-----+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | position | hire_date | salary |
+-----+-----+-----+-----+-----+-----+
| 5           | Alice      | Brown     | Sales Manager | 2022-05-10 | 50000  |
| 6           | Bob        | White     | Mechanic      | 2021-07-15 | 66666  |
+-----+-----+-----+-----+-----+-----+

SELECT 2
Time: 0.002s
```

pgcli -h localhost -p 5432 -d lab2 -U tpk

```
> pgcli -h localhost -p 5432 -d lab2 -U tpk
Server: PostgreSQL 14.8
Version: 4.0.1
Home: http://pgcli.com
tpk@localhost:lab2> select * from employees;

+-----+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | position | hire_date | salary |
+-----+-----+-----+-----+-----+-----+
| 5           | Alice      | Brown     | Sales Manager | 2022-05-10 | 50000  |
| 6           | Bob        | White     | Mechanic      | 2021-07-15 | 40000  |
+-----+-----+-----+-----+-----+-----+

SELECT 2
Time: 0.004s
tpk@localhost:lab2> START TRANSACTION ISOLATION LEVEL SERIALIZABLE;
START TRANSACTION
Time: 0.000s
tpk@localhost:lab2> select * from employees;

+-----+-----+-----+-----+-----+-----+
| employee_id | first_name | last_name | position | hire_date | salary |
+-----+-----+-----+-----+-----+-----+
| 5           | Alice      | Brown     | Sales Manager | 2022-05-10 | 50000  |
| 6           | Bob        | White     | Mechanic      | 2021-07-15 | 40000  |
+-----+-----+-----+-----+-----+-----+

SELECT 2
Time: 0.002s
tpk@localhost:lab2> UPDATE employees SET salary = 7777 WHERE first_name = 'Bob' AND last_name = 'White';
You're about to run a destructive command.
Do you want to proceed? [y/N]: y
Your call!
could not serialize access due to concurrent update
Time: 57.136s (57 seconds), executed in: 57.136s (57 seconds)
tpk@localhost:lab2> 
```

Заключение

Знакомство было проведено с тремя ключевыми уровнями изоляции транзакций.

1. READ COMMITTED:
- Уровень изоляции считается базовым.
 - Фиксированные изменения от других транзакций становятся видимыми.
 - Не исключает возможности изменения данных в процессе текущей транзакции.
2. REPEATABLE READ:
- Уровень изоляции выше, чем у READ COMMITTED.
 - Обеспечивает постоянство данных на протяжении транзакции.

- Запрещает доступ к новым данным, которые соответствуют условиям запроса во время транзакции.

3. **SERIALIZABLE:**

- Наивысший уровень изоляции.
- Гарантирует стабильность данных и защищает от изменений, произведенных другими транзакциями во время текущей транзакции.
- Доступ к данным ограничивается их состоянием на момент начала транзакции.