

## Уровень изоляции Read Committed

Создаем таблицу:

```
postgres=# BEGIN;
BEGIN
postgres=# SELECT * FROM t;
 n
----
 42
(1 ÆËÿ)
```

```
postgres=# CREATE TABLE t(n integer);
CREATE TABLE
postgres=# INSERT INTO t VALUES (42);
INSERT 0 1
postgres=#
```

Запрос из первой транзакции:

Удаляем строку во второй транзакции и фиксируем изменения:

```
postgres=# DELETE FROM t;
DELETE 1
```

```
postgres=# SELECT * FROM t;
 n
---
(0 ёёЁёь)
```

Повторим запрос:

```
postgres=# commit;
COMMIT
postgres=# SELECT * FROM t;
   n
---
(0 ёёЁюь)
```

Первая транзакция видит произошедшие изменения.

## Уровень изоляции Repeatable Read

Вернем строку:

```
postgres=# INSERT INTO t VALUES (42);
INSERT 0 1
```

```
postgres=# BEGIN ISOLATION LEVEL REPEATABLE READ;
BEGIN
postgres=# SELECT * FROM t;
      n
-----
    42
(1 row)
```

Запрос из первой транзакции:

```
postgres=# DELETE FROM t;
DELETE 1
```

Удаляем строку во второй транзакции и фиксируем изменения:

Повторим запрос:

На этом уровне изоляции первая транзакция не видит изменений.

```
postgres=# SELECT * FROM t;
 n
----
 42
(1 ёёёёюър)
```