

## Лабораторная работа №3

- 1) Создать конфигурационные файлы и запустить docker compose

```
pg-slave | 2024-12-23 23:54:04,212 INFO: no action. I am (postgresl1), the leader with the lock
pg-master | 2024-12-23 23:54:05,969 INFO: no action. I am (postgresl0), a secondary, and following a leader (postgresl1)
```

- 2) Создаем таблицу и записываем данные в мастер ноду

```
pg-slave public *pg-slave Script X
create table test_table (
  id int,
  data varchar(255)
);
insert into test_table values(1, 'тестовая запись')
```

- 3) Убеждаемся, что во второй ноду таблица так же создалась

```
pg-master localhost:5433
  Базы данных
    postgres
      Схемы
        public
          Таблицы
            test_table 8K
```

- 4) Проверяем, что мы не можем записывать в зависимую ноду

```
pg-slave public *pg-slave Script X
insert into test_table values(2, 'test test');

Результат 1 X
insert into test_table values (2, 'test test') | Введите SQL выражение чтобы отфильтровать

SQL Error [25006]: ERROR: cannot execute INSERT in a read-only transaction
Позиция ошибки:
```

- 5) Добавляем балансировщик в docker-compose, перезапускаем контейнеры и проверяем, что все запустилось ок

```
pg-slave | 2024-12-24 00:11:01.454 UTC [22] LOG: received SIGHUP, reloading configuration files
pg-slave | server signaled
pg-slave | 2024-12-24 00:11:01.455 UTC [22] LOG: parameter "synchronous_standby_names" changed to "postgresl0"
pg-slave | 2024-12-24 00:11:01.664 UTC [49] LOG: standby "postgresl0" is now a synchronous standby with priority 1
pg-slave | 2024-12-24 00:11:01.664 UTC [49] STATEMENT: START_REPLICATION SLOT "postgresl0" 0/7000000 TIMELINE 5
pg-slave | 2024-12-24 00:11:03,576 INFO: Synchronous standby status assigned to ['postgresl0']
pg-slave | 2024-12-24 00:11:03,581 INFO: no action. I am (postgresl1), the leader with the lock
postgres_entrypoint | [WARNING] (7) : Server postgres/postgresql_pg_slave_5432 is UP, reason: Layer7 check passed, code: 200, check duration: 8ms.
pg-master | 2024-12-24 00:11:10,603 INFO: Lock owner: postgresl1; I am postgresl0
```

6) Проверим результат

The screenshot shows a database client interface. At the top, a tree view displays the database structure: 'pg-master localhost:5433' > 'pg-slave localhost:5434' > 'postgres localhost:5432' > 'Базы данных' > 'postgres' > 'Схемы' > 'public' > 'Таблицы' > 'test\_table'. Below this, the SQL query 'select \* from test\_table tt;' is entered. The result is displayed in a table with two columns: 'id' and 'data'. The first row contains the values '1' and 'тестовая запись'.

id	data
1	тестовая запись

7) Пробуем отключить текущую мастер ноду. Через некоторое время вторичная нода самопровозгласилась мастером, а балансировщик обновил путь. Работоспособность кластера самостоятельно восстановилась.

The screenshot shows the same database client interface as before. The SQL query 'select \* from test\_table tt;' is entered. The result is displayed in a table with two columns: 'id' and 'data'. The first row contains the values '1' and 'тестовая запись'. A log window is open on the left side, showing the following messages:

```
pg-master | 2024-12-24 08:13:41.764 INFO: promoted self to leader by acquiring session lock
pg-master | server promoting
pg-master | 2024-12-24 08:13:41.766 UTC [29] LOG: received promote request
pg-master | 2024-12-24 08:13:41.766 UTC [29] LOG: redo done at 0/7803888 system usage: CPU: user= 0.00 s, system= 0.00 s, elapsed= 162.16 s
pg-master | 2024-12-24 08:13:41.767 UTC [29] LOG: last completed transaction was at log time 2024-12-24 08:11:01.565192+00
pg-master | 2024-12-24 08:13:41.769 UTC [29] LOG: selected new timeline ID: 6
pg-master | 2024-12-24 08:13:41.806 UTC [29] LOG: archive recovery complete
pg-master | 2024-12-24 08:13:41.808 UTC [27] LOG: checkpoint starting: force
pg-master | 2024-12-24 08:13:41.810 UTC [25] LOG: database system is ready to accept connections
pg-master | 2024-12-24 08:13:42.794 INFO: Lock owner: postgresql0; I am postgresql0
pg-master | 2024-12-24 08:13:42.794 INFO: Reaped pid=129, exit status=0
pg-master | 2024-12-24 08:13:42.801 INFO: no action. I am (postgresql0), the leader with the lock
pg-master | 2024-12-24 08:13:42.809 UTC [27] LOG: checkpoint complete: wrote 23 buffers (0.1%); 0 WAL file(s) added, 0 removed, 0 recycled; write=0.990 s, sync=0.006 s, total=1.001 s; sync files=19, longest=0.004 s, average=0.001 s
pg-master | 2024-12-24 08:13:42.814 UTC [27] LOG: checkpoint starting: immediate force wait
pg-master | 2024-12-24 08:13:42.814 UTC [27] LOG: checkpoint complete: wrote 1 buffers (0.0%); 0 WAL file(s) added, 0 removed, 0 recycled; write=0.001 s, sync=0.001 s, total=0.005 s; sync files=1, longest=0.001 s, average=0.001 s
pg-master | 2024-12-24 08:13:42.822 INFO: Lock owner: postgresql0; I am postgresql0
pg-master | 2024-12-24 08:13:42.826 INFO: Droppes unknown replication slot 'postgresql1'
pg-master | 2024-12-24 08:13:42.827 INFO: no action. I am (postgresql0), the leader with the lock
postgres_entrypoint | [WARNING] (7) : Server postgres/postgresql_pg_master_5432 is UP, reason: Layer7 check passed, code: 200, check duration: 6ms. 1 active and 0 backup servers online. 0 sessions requested, 0 total in queue.
```

### **Ответы на доп вопросы:**

- 1) Разница между директивами заключается в том, что `expose` только уведомляет Docker о портах, которые контейнер слушает, но не делает их доступными извне. Это может быть полезно для внутренней связи между контейнерами в одной сети. Напротив, `ports` пробрасывает порты контейнера на хост-машину, что позволяет к ним подключаться из внешней среды.
- 2) При простом перезапуске контейнера образ не будет собираться заново. Так же изменения конфигов не будут отражены в контейнере, пока образ не будет сбилжен заново. При изменении `Dockerfile` образ все еще не будет автоматически пересобран, пока явно не будет указан флаг `--build` при запуске `compose` или билд не будет запущен вручную.
- 3)