

# Инструкция по установке RusTNG на PostgreSQL

Это пошаговая инструкция по установке базы PostgreSQL и настройка приложения RusTNG для использования.

Вам необходима операционная система на Linux с установленным docker и docker-compose.

## Инструкция

### Создайте папки

1. Создайте папку *tng* и перейдите в созданную папку : `~$ mkdir tng && cd tng`
2. Создайте папку *db* и перейдите в созданную папку: `~tng$ mkdir db && cd db`
3. Создайте папку *data*: `~tng/db$ mkdir data`
4. Создайте папку *init*: `~tng/db$ mkdir init`
5. Вернитесь обратно а папку *tng*: `~tng/db$ cd ..`
6. Создайте папку *web*: `~tng$ mkdir web`

### Извлеките образ docker

1. Извлеките docker образ postgresql: `~$ sudo docker pull postgres:14.6`

```
ubuntu@ubuntu2004:~$ sudo docker pull postgres:14.6
14.6: Pulling from library/postgres
3f4ca61aafcd: Already exists
048d3078d446: Already exists
c6d23b4fe6c1: Already exists
d846f6946dd5: Already exists
76f7157f330d: Already exists
4eacfb0464b2: Already exists
5c197e2b597b: Already exists
2c4576649951: Already exists
a9ab539fce2b: Pull complete
4b8ce16933e9: Pull complete
88e4ae5af1fb: Pull complete
8b507b1498e6: Pull complete
9a3f79ee6c66: Pull complete
Digest: sha256:cc80ac7c45f1518d74f6dbb4a982d9e88a457f8e2fbad90f81b09053c8058eb2
Status: Downloaded newer image for postgres:14.6
docker.io/library/postgres:14.6
```

2. Извлеките docker образ tomcat: `~$ sudo docker pull tomcat:9-jdk8`

```
ubuntu@ubuntu2004:~/tng$ sudo docker pull tomcat:9-jdk8
9-jdk8: Pulling from library/tomcat
6e3729cf69e0: Already exists
96aa423488f0: Pull complete
f34fa275bf0b: Pull complete
28777328b1ab: Pull complete
499862a62942: Pull complete
216c22cb5e2e: Pull complete
3e7bb6450c75: Pull complete
Digest: sha256:8272e11c9f612bc00bdadf39d6c06b19b451a9c3f369879cb3c65d9016d624c9
Status: Downloaded newer image for tomcat:9-jdk8
docker.io/library/tomcat:9-jdk8
```

## Создайте конфигурацию docker-compose:

1. Замените текущую папку на *tng*
2. Создайте файл конфигурации docker-compose: *~tng\$ vim docker-compose.yml*
3. `version: "3.9"`
4. `services:`
5.     `db:`
6.         `image: postgres:14.6`
7.         `environment:`
8.             `POSTGRES_DB: "club"`
9.             `POSTGRES_USER: "club"`
10.             `POSTGRES_PASSWORD: "hrs"`
11.             `PGDATA: "/var/lib/postgresql/data/pgdata"`
12.         `volumes:`
13.             `- .db/init:/docker-entrypoint-initdb.d`
14.             `- .db/data:/var/lib/postgresql/data`
15.         `ports:`
16.             `- "5432:5432"`
17.         `healthcheck:`
18.             `test: ["CMD-SHELL", "pg_isready -U club -d club"]`
19.             `interval: 10s`
20.             `timeout: 5s`
21.             `retries: 5`
22.             `start_period: 10s`
23.         `restart: unless-stopped`
24.         `deploy:`
25.             `resources:`
26.                 `limits:`
27.                     `cpus: '1'`
28.                     `memory: 4G`
29.         `web:`
30.             `depends_on:`
31.                 `- db`
32.             `image: tomcat:9-jdk8`
33.             `volumes:`
34.                 `- ./web/tng3.war:/usr/local/tomcat/webapps/ROOT.war`
35.                 `- ./web/tng3.properties:/usr/local/tomcat/conf/tng3.properties`
36.             `ports:`
37.                 `- '8080:8080'`
38.             `environment:`
39.                 `- JAVA_OPTS=-Xmx4G -Xms1G`

39. Измените директорию на *tng/db/init*: `~tng$ cd db/init`
40. Создайте скрипт инициализации `create_schema.sql` в папке `init`: `~tng/db/init$ vim create_schema.sql`

```
CREATE SCHEMA IF NOT EXISTS club AUTHORIZATION club;
```

41. Измените папку на *tng/web*: `~tng/db/init$ cd ../../web`
42. Создайте файл конфигурации *tng3.properties* (Ниже пример конфигурации):  
`~tng/web$ vim tng3.properties`

```
db.url=jdbc:postgresql://localhost:5432/club
db.username=club
db.password=hrs
db.schema=club
db.hbm2ddl=update
tng.locale=ru
```

43. Скопируйте `tng3.war` с приложение RusTNG в папку *tng/web*
44. Измените папку на *tng*: `~tng/web$ cd ..`
45. Запустите `docker` контейнер: `~tng$ sudo docker-compose up -d`

```
ubuntu@ubuntu2004:~/tng$ sudo docker-compose up -d
/snap/docker/2285/lib/python3.6/site-packages/paramiko/transport.py:33: CryptographyDeprecationWarning: Python 3.6 is no longer supported by the Python core team. Therefore, support for it is deprecated in cryptography and will be removed in a future release.
  from cryptography.hazmat.backends import default_backend
Starting tng_db_1 ... done
Starting tng_web_1 ... done
```