МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

Федеральное государственное автономное

образовательное учреждение высшего образования

«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Лабораторная работа 8

по дисциплине

«Автоматизация процессов жизненного цикла программных средств»

Выполнил:

ст. гр. 2xx-3xx

Иванов И. И.

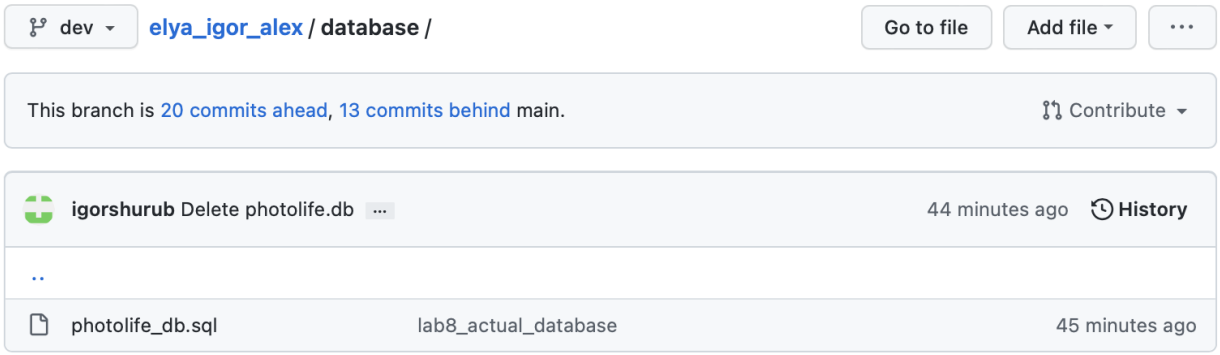
Проверил:

Красников А. С.

Москва – 2024

План выполнения лабораторной работы

1. Создать в репозитории проекта директорию для хранения актуальной схемы базы данных.



2. Провести анализ существующих инструментов статического анализа безопасности SQL-кода, выбрать наиболее удобное, по мнению команды, средство, обосновать выбор.

Анализ был произведен на следующем сайте: https://analysis-tools.dev/tag/sql

И для нашего проекта лучше всего подходит следующий: SQLCheck.

Usage

```
$ sqlcheck -h

Command line options : sqlcheck <options>
   -f --file_name          :  file name
   -r --risk_level         :  set of anti-patterns to check
                           :  1 (all anti-patterns, default)
                           :  2 (only medium and high risk anti-patterns)
                           :  3 (only high risk anti-patterns)
   -c --color_mode         :  color mode
   -v --verbose_mode       :  verbose mode
```

```
$ sqlcheck -f examples/top_mutexes.sql -v

--------------------------------------------------
> RISK LEVEL    :: ALL ANTI-PATTERNS
> SQL FILE NAME :: examples/top_mutexes.sql
--------------------------------------------------
==================== Results ====================

--------------------------------------------------
SQL Statement: with top_mutexes as ( select--+ leading(t1 s1 v1 v2 t2 s2) use_hash(s1)
use_nl(v1) use_hash(s2) materialize t1.hsecs ,s1.* ,s2.sleeps as end_sleeps
,s2.wait_time as end_wait_time ,s2.sleeps-s1.sleeps as delta_sleeps ,t2.hsecs -
t1.hsecs as delta_hsecs --,s2.* from v$timer t1 ,v$mutex_sleep s1 ,(select/*+
no_merge */ sum(level) a from dual connect by level<=1e6) v1 ,v$timer t2
,v$mutex_sleep s2 where s1.mutex_type=s2.mutex_type and s1.location=s2.location
) select * from top_mutexes order by delta_sleeps desc;
```

3. Добавить в CI/CD пайплайны для всех веток шаг, осуществляющий проверку безопасности SQL файлов

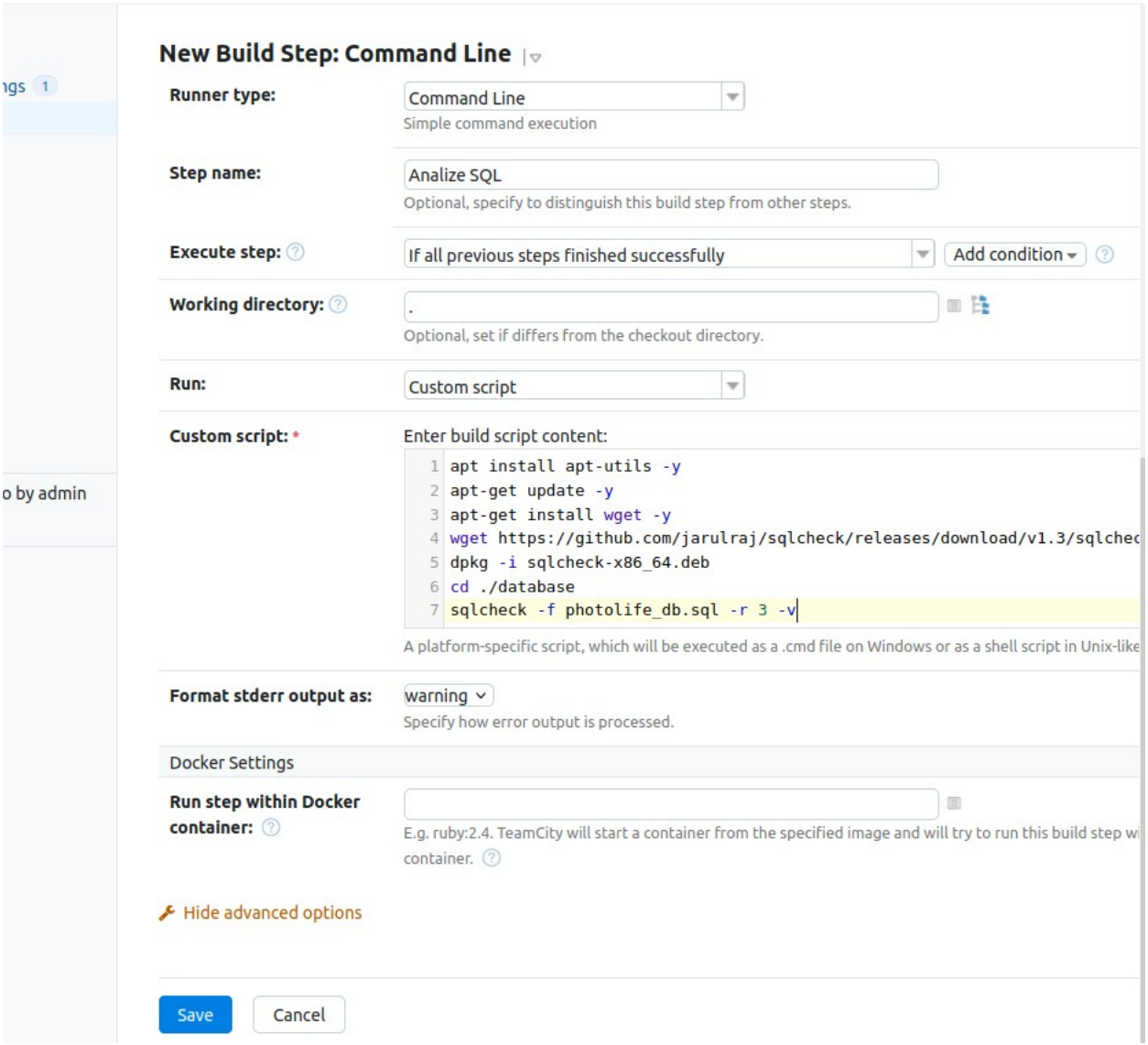
Добавим новый шаг:

Build Steps

In this section you can configure the sequence of build steps to be executed. Each build step is represented by a build runner and provides integration with a specific build or test tool.

+ Add build step | Reorder build steps | Auto-detect build steps

Build Step	Parameters Description	
1. Analize and ConvertAnalysisResults *(disabled)*	Command Line Custom script: apt install apt-utils -y (and 38 more lines) Execute: If all previous steps finished successfully	Edit
2. Unit-testing *(disabled)*	Command Line Custom script: apt install apt-utils -y (and 27 more lines) Execute: If all previous steps finished successfully	Edit
3. Analize SQL	**Command Line** **Custom script: apt install apt-utils -y (and 9 more lines)** **Execute: If all previous steps finished successfully**	Edit
4. Docker build *(disabled)*	Docker Docker build; Dockerfile location: Dockerfile ; context directory: . Execute: If all previous steps finished successfully	Edit
5. Docker push *(disabled)*	Docker docker push photolife/medhelper:latest; remove image(s) after push Execute: If all previous steps finished successfully	Edit
6. Connect SSH *(disabled)*	SSH Exec Target: 192.168.31.4 Port: 22 Commands: cd /home/auto-server docker pull photolife/medhelper docker compose up -d Execute: If all previous steps finished successfully	Edit


New Build Step: Command Line
Runner type:
Command Line
Simple command execution
Step name:
Analize SQL
Optional, specify to distinguish this build step from other steps.
Execute step:
If all previous steps finished successfully
Add condition
Working directory:
.
Optional, set if differs from the checkout directory.
Run:
Custom script
Custom script: *
Enter build script content:
1 apt install apt-utils -y
2 apt-get update -y
3 apt-get install wget -y
4 wget https://github.com/jarulraj/sqlcheck/releases/download/v1.3/sqlchec
5 dpkg -i sqlcheck-x86_64.deb
6 cd ./database
7 sqlcheck -f photolife_db.sql -r 3 -v
A platform-specific script, which will be executed as a .cmd file on Windows or as a shell script in Unix-like
Format stderr output as:
warning
Specify how error output is processed.
Docker Settings
Run step within Docker container:
E.g. ruby:2.4. TeamCity will start a container from the specified image and will try to run this build step wi
container.
Hide advanced options
Save
Cancel

Запустим:

```
[11:23:57]      [Step 3/6] Selecting previously unselected package sqlcheck.
[11:23:57]      [Step 3/6] (Reading database ... 17552 files and directories currently installed.)
[11:23:57]      [Step 3/6] Preparing to unpack sqlcheck-x86_64.deb ...
[11:23:57]      [Step 3/6] Unpacking sqlcheck (1.3) ...
[11:23:57]      [Step 3/6] Setting up sqlcheck (1.3) ...
[11:23:57]      [Step 3/6] +-------------------------------------------------+
[11:23:57]      [Step 3/6] |                   SQLCHECK                      |
[11:23:57]      [Step 3/6] +-------------------------------------------------+
[11:23:57]      [Step 3/6] > RISK LEVEL    :: ONLY MEDIUM AND HIGH RISK ANTI-PATTERNS
[11:23:57]      [Step 3/6] > SQL FILE NAME :: photolife_db.sql
[11:23:57]      [Step 3/6] > COLOR MODE    :: DISABLED
[11:23:57]      [Step 3/6] > VERBOSE MODE  :: ENABLED
[11:23:57]      [Step 3/6] > DELIMITER     :: ;
[11:23:57]      [Step 3/6] -------------------------------------------------
[11:23:57]      [Step 3/6] ==================== Results ====================
[11:23:57]      [Step 3/6] No issues found.
[11:23:57]      [Step 3/6] Process exited with code 0
```

4. Провести анализ существующих инструментов автоматизации миграции схемы БД в CI/CD пайплайне, выбрать наиболее удобное, по мнению команды, средство, обосновать выбор.

Liquibase — это открытая независимая от БД библиотека для отслеживания, управления и применения изменений схемы базы данных. Поддерживает подавляющее большинство БД, включая PostgreSQL, MySQL, Oracle, Sybase, HSQL, Apache Derby. Работает с форматами XML, YAML, JSON, SQL.

Ее преимущества: библиотека Liquibase предоставляет больше возможностей из «коробки» в отличие от того же Flyway — отмена изменений, автогенерация миграций. Имеет dry-run, то есть можно посмотреть, какие SQL-запросы будут выполнены.

В отличие от Flyway, которая поддерживает скрипты миграции только в форматах SQL и Java, Liquibase — это универсальный инструмент. Он позволяет накатывать одни и те же миграции на любые базы данных и абстрагироваться от SQL. Эта библиотека больше подходит для проектов, где необходимо работать с разными окружениями и СУБД.

Установим Liquibase, драйвер (коннектор) PostgreSQL, а также Java на ВМ Test, Stage, Prod:

```
sad201331@sad:/etc$ sudo mkdir liquibase
```

```
sad201331@sad:/etc/liquibase$ sudo wget https://github.com/liquibase/liquibase/releases/download/v4.18.0/liquibase-4.18.0.tar.gz
--2022-12-22 14:33:01--  https://github.com/liquibase/liquibase/releases/download/v4.18.0/liquibase-4.18.0.tar.gz
Распознаётся github.com (github.com)… 140.82.121.3
Подключение к github.com (github.com)|140.82.121.3|:443... соединение установлено.
HTTP-запрос отправлен. Ожидание ответа… 302 Found
Адрес: https://objects.githubusercontent.com/github-production-release-asset-2e65be/2019791/e0cfdf9a-4107-4c72-907c-87eb4e5a7eff?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWNJYAX4CSVEH53A%2F20221222%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20221222T121906Z&X-Amz-Expires=300&X-Amz-Signature=0cf3478079a1712cb641caa0a2a9162650440a2cb604ca6825066c34ffc58166&X-Amz-SignedHeaders=host&actor_id=0&key_id=0&repo_id=2019791&response-content-disposition=attachment%3B%20filename%3Dliquibase-4.18.0.tar.gz&response-content-type=application%2Foctet-stream
[переход]
--2022-12-22 14:33:02--  https://objects.githubusercontent.com/github-production
```

```
sad201331@sad:/etc/liquibase$ sudo tar xvzf liquibase-4.18.0.tar.gz
ABOUT.txt
GETTING_STARTED.txt
LICENSE.txt
README.txt
UNINSTALL.txt
changelog.txt
examples/
examples/yaml/
examples/yaml/blank-changelog.yaml
examples/yaml/example-changelog.yaml
examples/yaml/liquibase.flowvariables.yaml
examples/yaml/example-changeset-yaml.txt
```

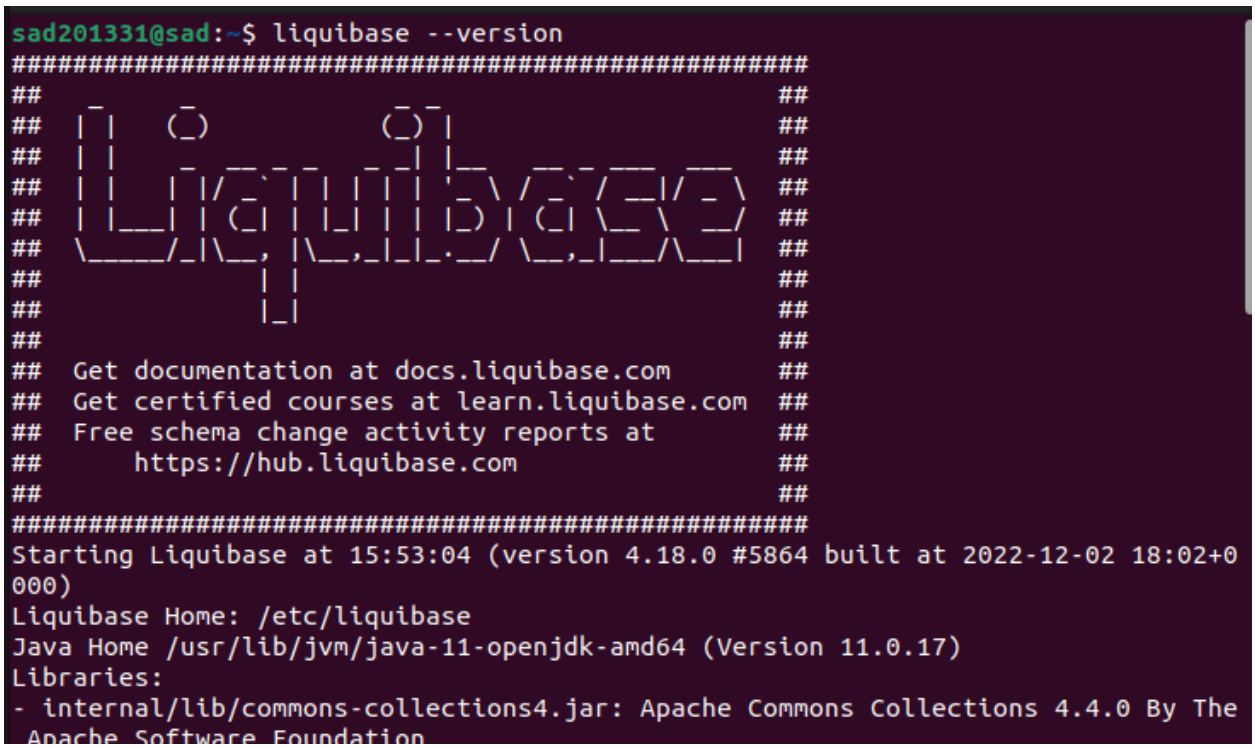
```
sad201331@sad:/etc/liquibase$ sudo wget https://repo1.maven.org/maven2/org/postgresql/postgresql/42.2.5/postgresql-42.2.5.jar -P lib/
--2022-12-22 14:36:07--  https://repo1.maven.org/maven2/org/postgresql/postgresql/42.2.5/postgresql-42.2.5.jar
Распознаётся repo1.maven.org (repo1.maven.org)… 146.75.116.209
Подключение к repo1.maven.org (repo1.maven.org)|146.75.116.209|:443... соединение установлено.
HTTP-запрос отправлен. Ожидание ответа… 200 OK
Длина: 825943 (807K) [application/java-archive]
Сохранение в: ‘lib/postgresql-42.2.5.jar’

postgresql-42.2.5.j 100%[===================>] 806,58K  4,22MB/s    за 0,2s

2022-12-22 14:36:08 (4,22 MB/s) - ‘lib/postgresql-42.2.5.jar’ сохранён [825943/825943]

sad201331@sad:/etc/liquibase$
```

```
sad201331@sad:/etc/liquibase$ sudo apt-get install openjdk-11-jdk
Чтение списков пакетов… Готово
Построение дерева зависимостей… Готово
Чтение информации о состоянии… Готово
Будут установлены следующие дополнительные пакеты:
  ca-certificates-java fonts-dejavu-extra java-common libatk-wrapper-java
  libatk-wrapper-java-jni libice-dev libsm-dev libxt-dev
  openjdk-11-jdk-headless openjdk-11-jre openjdk-11-jre-headless
Предлагаемые пакеты:
  default-jre libice-doc libsm-doc libxt-doc openjdk-11-demo openjdk-11-source
  visualvm fonts-ipafont-gothic fonts-ipafont-mincho fonts-wqy-microhei
  | fonts-wqy-zenhei
Следующие НОВЫЕ пакеты будут установлены:
  ca-certificates-java fonts-dejavu-extra java-common libatk-wrapper-java
  libatk-wrapper-java-jni libice-dev libsm-dev libxt-dev openjdk-11-jdk
  openjdk-11-jdk-headless openjdk-11-jre openjdk-11-jre-headless
```

```
sad201331@sad:/etc/liquibase$ sudo nano ~/.bashrc
```

```
export PATH=$PATH:/etc/liquibase
```

```
sad201331@sad:/home/server$ ls /etc/liquibase/
ABOUT.txt            internal     liquibase                UNINSTALL.txt
changelog.txt        lib          liquibase-4.18.0.tar.gz
examples             licenses     liquibase.bat
GETTING_STARTED.txt  LICENSE.txt  README.txt
sad201331@sad:/home/server$ ls /etc/liquibase/internal/lib
commons-collections4.jar  jaxb-runtime.jar          opencsv.jar
commons-lang3.jar         jaybird.jar              picocli.jar
commons-text.jar          jcc.jar                  postgresql.jar
connector-api.jar         liquibase-commercial.jar snakeyaml.jar
h2.jar                    liquibase-core.jar       snowflake-jdbc.jar
hsqldb.jar                mariadb-java-client.jar  sqlite-jdbc.jar
jaxb-api.jar              mssql-jdbc.jar
jaxb-core.jar             ojdbc8.jar
sad201331@sad:/home/server$
```



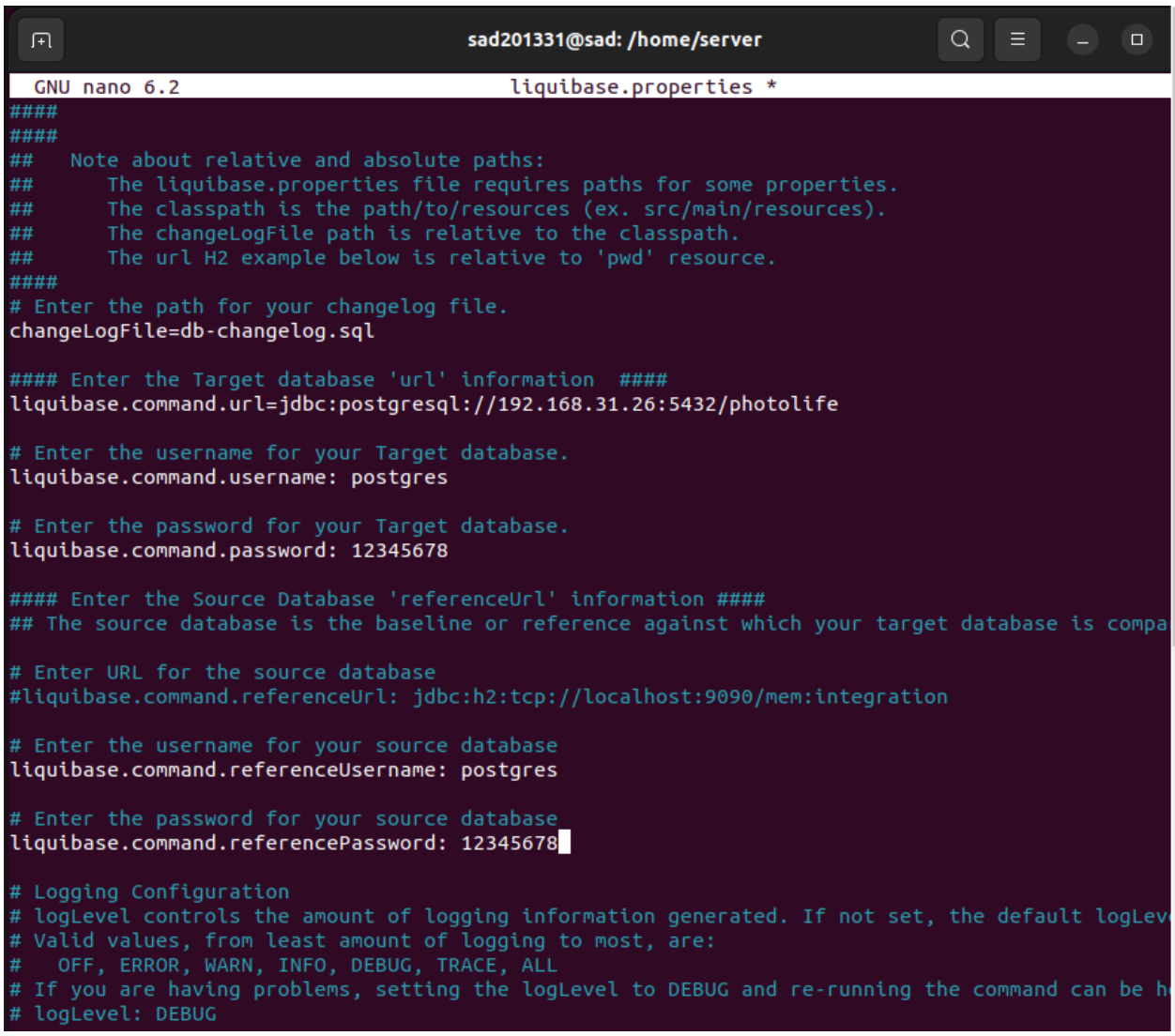
5. Внедрить в CI/CD пайплайн для DEV-ветки шаг(и), осуществляющий автоматическое развертывание новой схемы БД на STAGE-сервере путем предварительной сверки схем БД на TEST-сервере и STAGE-сервере

Перед внедрением (изменения) шага изменим файл liquibase.properties для возможности подключения к локальным и удаленным БД и последующего их сравнения.

Создадим liquibase-проект и отредактируем liquibase.properties на ВМ Test:

```
sad201331@sad:/home/server$ liquibase init project
#####################################################
##  _       _            _ _                     ##
## | |     (_)          (_) |                    ##
## | |      _  __ _ _   _ _| |__   __ _ ___  ___  ##
## | |     | |/ _` | | | | | '_ \ / _` / __|/ _ \ ##
## | |_____| | (_| | |_| | | |_) | (_| \__ \  __/ ##
## \_____/_|\__, |\__,_|_|_.__/ \__,_|___/\___| ##
##             | |                               ##
##             |_|                               ##
##                                               ##
##  Get documentation at docs.liquibase.com      ##
##  Get certified courses at learn.liquibase.com ##
##  Free schema change activity reports at       ##
##      https://hub.liquibase.com                ##
##                                               ##
#####################################################
Starting Liquibase at 15:53:59 (version 4.18.0 #5864 built at 2022-12-02 18:02+0000)
Liquibase Version: 4.18.0
Liquibase Community 4.18.0 by Liquibase
Setup new liquibase.properties, flowfile, and sample changelog? Enter (Y)es with defaults, yes
h (C)ustomization, or (N)o. [Y]:
y
Setting up new Liquibase project in '/home/server/.'...

Created example changelog file '/home/server/example-changelog.sql'
Created example defaults file '/home/server/liquibase.properties'
Created example flow file '/home/server/liquibase.advanced.flowfile.yaml'
Created example flow file '/home/server/liquibase.flowvariables.yaml'
Created example flow file '/home/server/liquibase.endstage.flow'
Created example flow file '/home/server/liquibase.flowfile.yaml'

To use the new project files make sure your database is active and accessible by opening a new
minal window to run "liquibase init start-h2", and then return to this terminal window to run "
uibase update" command.
For more details, visit the Getting Started Guide at https://docs.liquibase.com/start/home.html
Liquibase command 'init project' was executed successfully.
sad201331@sad:/home/server$
```

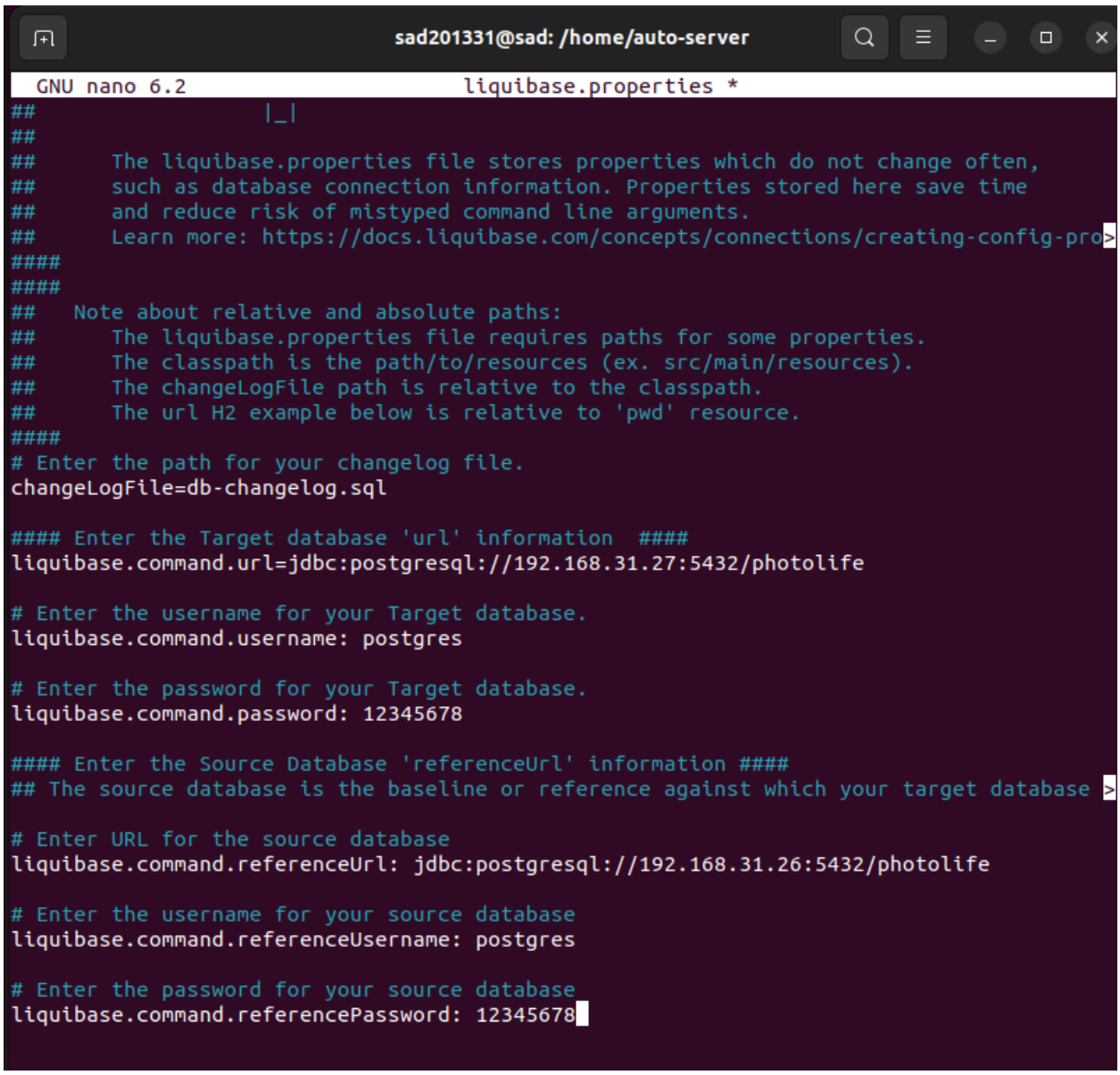
```
sad201331@sad:/home/server$ sudo nano liquibase.properties
```



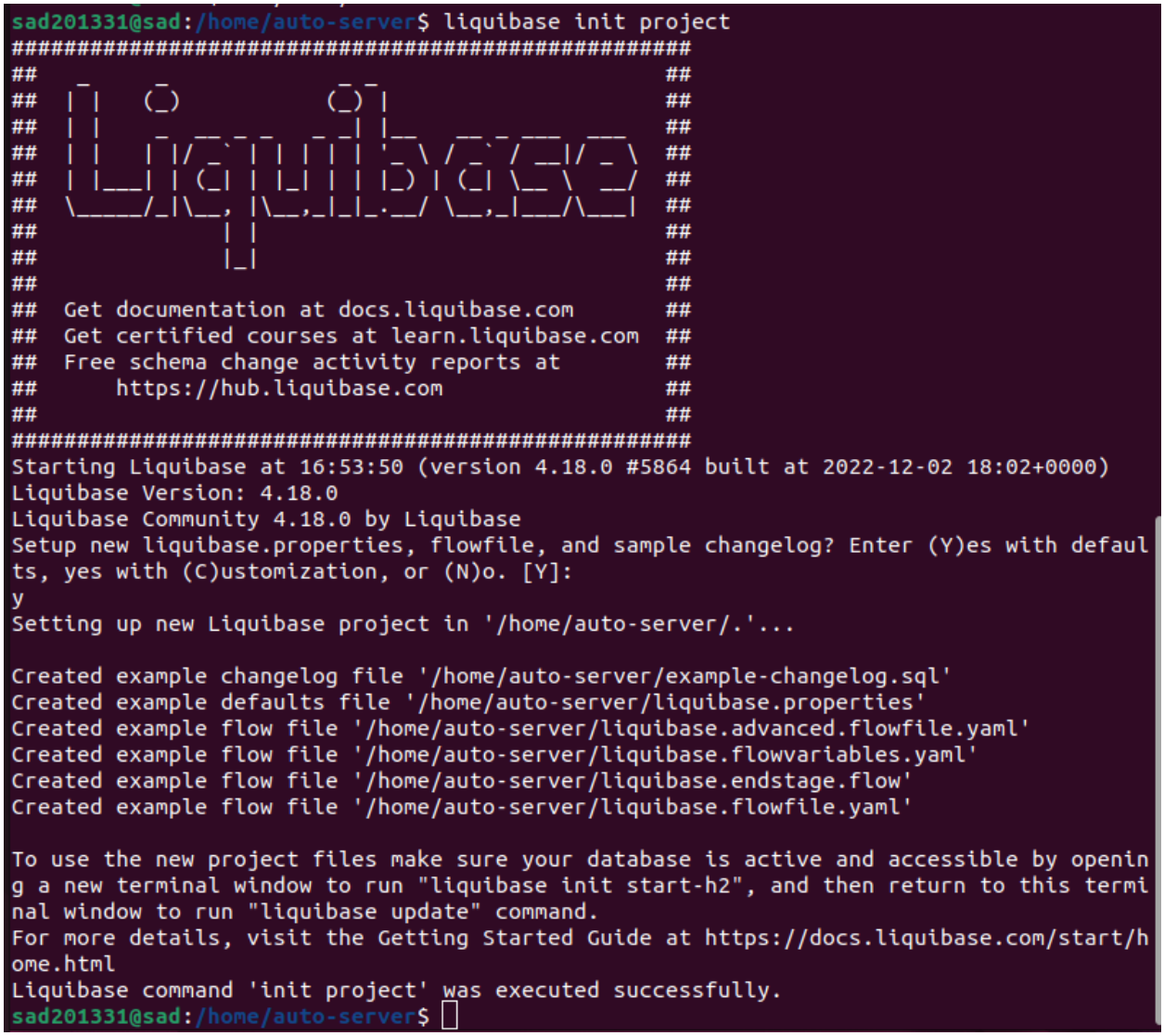
Создадим liquibase-проект и отредактируем liquibase.properties на ВМ Stage:

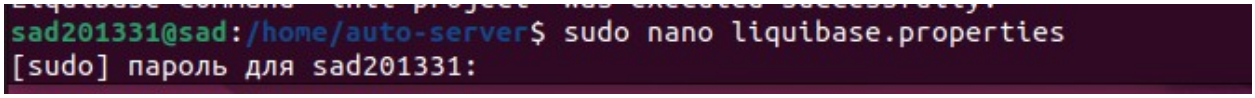
```
sad201331@sad:/home/auto-server$ liquibase init project
#####################################################
##                                                 ##
##  _     _                     _ _                 ##
## | |   (_)                   (_) |                ##
## | |    _  __ _ _   _ _ __ _| |__   __ _ ___  ___ ##
## | |   | |/ _` | | | | '_ \ / _` / __|/ _ \  ##
## | |___| | (_| | |_| | | | | (_) | (_| \__ \  __/ ##
## \_____/_|\__, |\__,_|_|_.__/ \__,_|___/\___| ##
##             | |                                 ##
##             |_|                                 ##
##                                                 ##
##  Get documentation at docs.liquibase.com        ##
##  Get certified courses at learn.liquibase.com   ##
##  Free schema change activity reports at         ##
##      https://hub.liquibase.com                  ##
##                                                 ##
#####################################################
Starting Liquibase at 17:07:28 (version 4.18.0 #5864 built at 2022-12-02 18:02+0000)
Liquibase Version: 4.18.0
Liquibase Community 4.18.0 by Liquibase
Setup new liquibase.properties, flowfile, and sample changelog? Enter (Y)es with default
s, yes with (C)ustomization, or (N)o. [Y]:
y
Setting up new Liquibase project in '/home/auto-server/.'...

Created example changelog file '/home/auto-server/example-changelog.sql'
Created example defaults file '/home/auto-server/liquibase.properties'
Created example flow file '/home/auto-server/liquibase.advanced.flowfile.yaml'
Created example flow file '/home/auto-server/liquibase.flowvariables.yaml'
Created example flow file '/home/auto-server/liquibase.endstage.flow'
Created example flow file '/home/auto-server/liquibase.flowfile.yaml'

To use the new project files make sure your database is active and accessible by opening
 a new terminal window to run "liquibase init start-h2", and then return to this termina
l window to run "liquibase update" command.
For more details, visit the Getting Started Guide at https://docs.liquibase.com/start/ho
me.html
Liquibase command 'init project' was executed successfully.
```

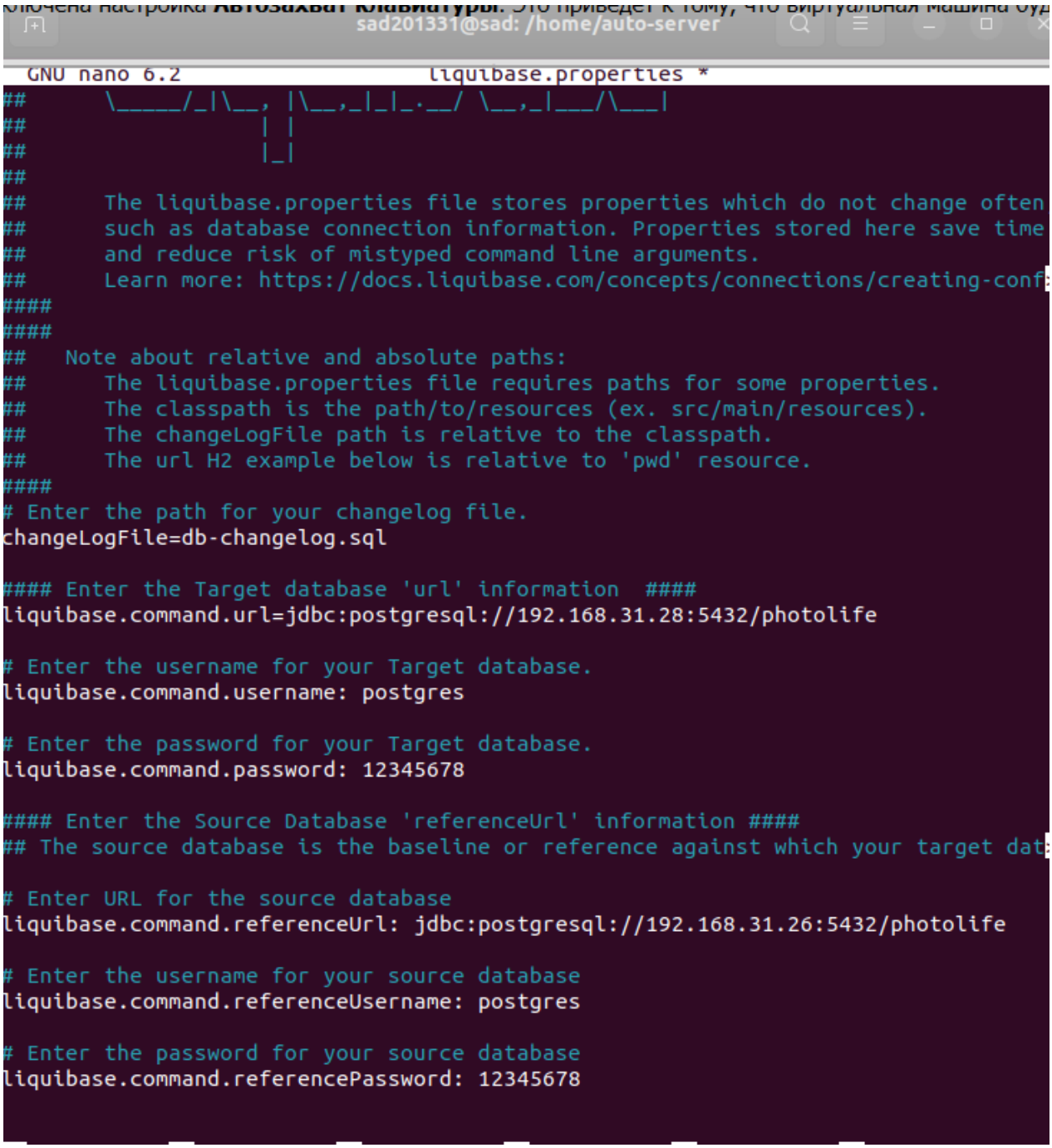
```
zaa201-331@alicia:/home/auto-server$ sudo nano liquibase.properties
```


sad201331@sad: /home/auto-server
GNU nano 6.2 liquibase.properties *
|_|
##
The liquibase.properties file stores properties which do not change often,
such as database connection information. Properties stored here save time
and reduce risk of mistyped command line arguments.
Learn more: https://docs.liquibase.com/concepts/connections/creating-config-pro>
####
####
Note about relative and absolute paths:
The liquibase.properties file requires paths for some properties.
The classpath is the path/to/resources (ex. src/main/resources).
The changeLogFile path is relative to the classpath.
The url H2 example below is relative to 'pwd' resource.
####
# Enter the path for your changelog file.
changeLogFile=db-changelog.sql
Enter the Target database 'url' information
liquibase.command.url=jdbc:postgresql://192.168.31.27:5432/photolife
# Enter the username for your Target database.
liquibase.command.username: postgres
Enter the password for your Target database.
liquibase.command.password: 12345678
Enter the Source Database 'referenceUrl' information
The source database is the baseline or reference against which your target database >
Enter URL for the source database
liquibase.command.referenceUrl: jdbc:postgresql://192.168.31.26:5432/photolife
Enter the username for your source database
liquibase.command.referenceUsername: postgres
Enter the password for your source database
liquibase.command.referencePassword: 12345678

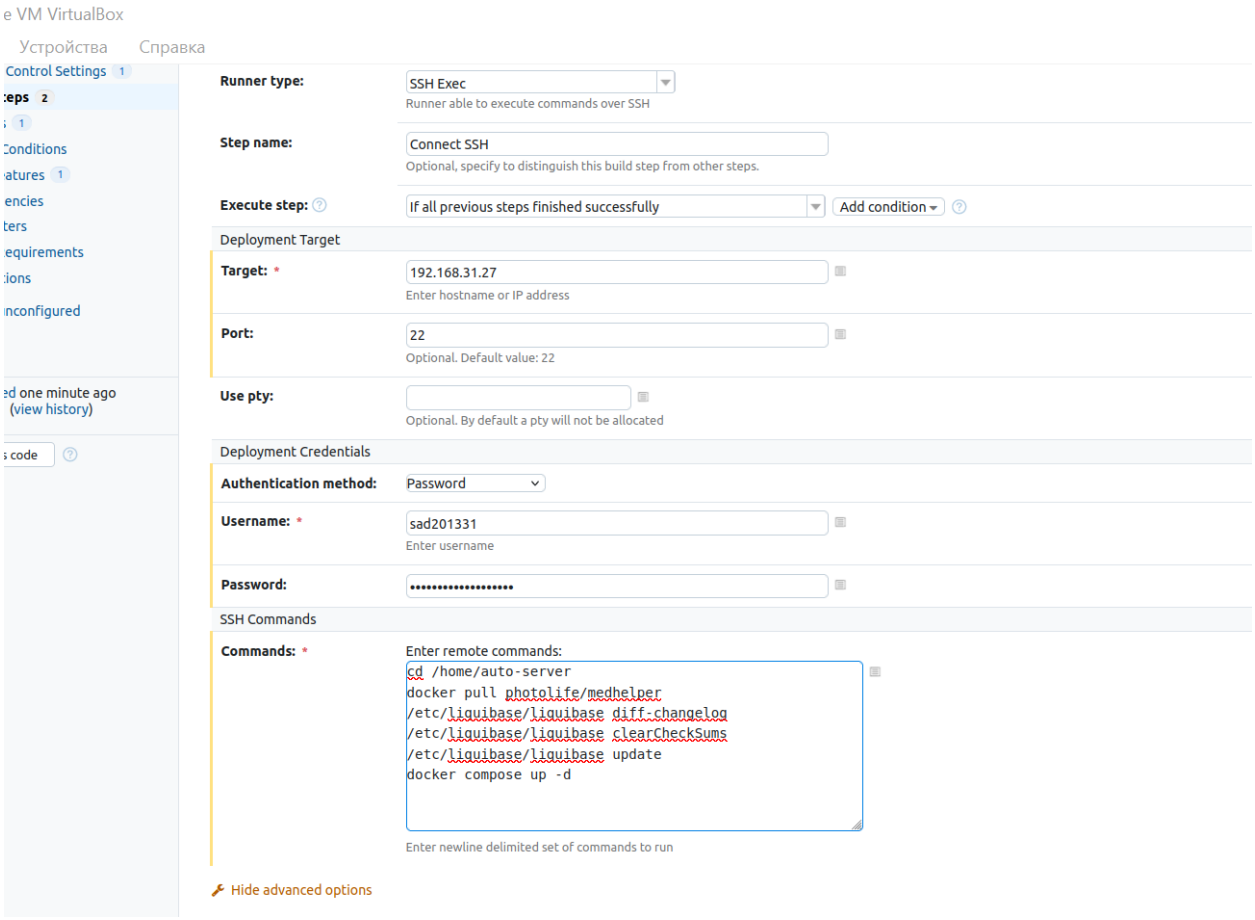
Создадим liquibase-проект и отредактируем liquibase.properties на BM Prod:






sad201331@sad: /home/auto-server
GNU nano 6.2 liquibase.properties *
_____/_|__, |__,_|_|_.__/ __,_|___/___|
| |
|_|
##
The liquibase.properties file stores properties which do not change often
such as database connection information. Properties stored here save time
and reduce risk of mistyped command line arguments.
Learn more: https://docs.liquibase.com/concepts/connections/creating-conf
####
####
Note about relative and absolute paths:
The liquibase.properties file requires paths for some properties.
The classpath is the path/to/resources (ex. src/main/resources).
The changeLogFile path is relative to the classpath.
The url H2 example below is relative to 'pwd' resource.
####
# Enter the path for your changelog file.
changeLogFile=db-changelog.sql
Enter the Target database 'url' information
liquibase.command.url=jdbc:postgresql://192.168.31.28:5432/photolife
# Enter the username for your Target database.
liquibase.command.username: postgres
Enter the password for your Target database.
liquibase.command.password: 12345678
Enter the Source Database 'referenceUrl' information
The source database is the baseline or reference against which your target dat
Enter URL for the source database
liquibase.command.referenceUrl: jdbc:postgresql://192.168.31.26:5432/photolife
Enter the username for your source database
liquibase.command.referenceUsername: postgres
Enter the password for your source database
liquibase.command.referencePassword: 12345678

Изменим шаг подключения и развертывания сервера по SSH на Stage для возможности сверки схем БД:



6. Создать отдельный пайплайн, идентичный пайплайну для DEV-ветки, но осуществляющий доставку и развертывание на PROD-сервере.

Добавим шаг подключения и развертывания сервера по SSH на Prod для возможности сверки схем БД:

6. Connect SSH	SSH Exec Target: 192.168.31.27 Port: 22 Commands: cd /home/auto-server docker pull photolife/medhelper /etc/liquibase/liquibase diff-changelog /etc/liquibase/liquibase clearCheckSums /etc/liquibase/liquibase update docker compose up -d Execute: If all previous steps finished successfully
7. Connect SSH prod	SSH Exec Target: 192.168.31.28 Port: 22 Commands: cd /home/auto-server docker pull photolife/medhelper /etc/liquibase/liquibase diff-changelog /etc/liquibase/liquibase clearCheckSums /etc/liquibase/liquibase update docker compose up -d Execute: If all previous steps finished successfully

7. Внести изменение в базу данных, таким образом, изменив актуальную схему БД.

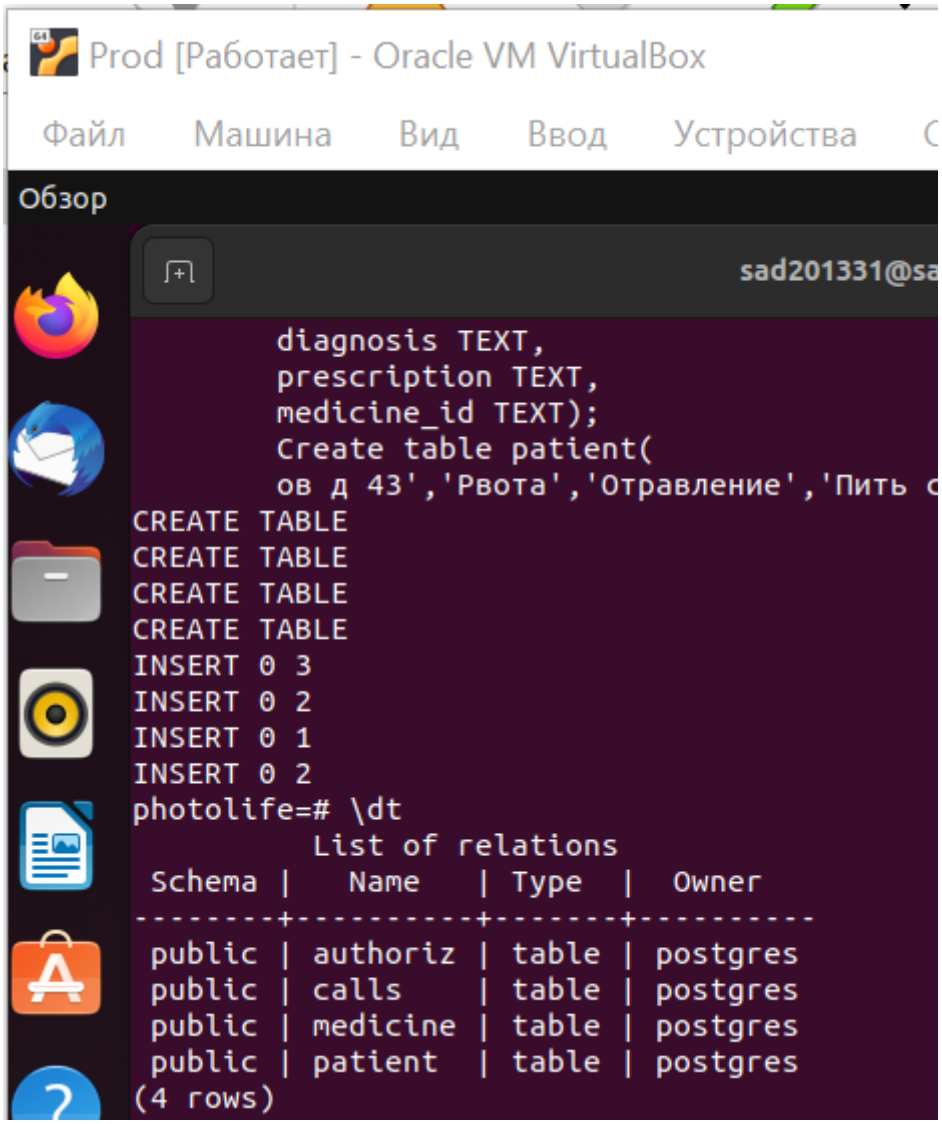
Схемы базы данных до внесения изменений на Test:

```
zaa201-331@alicia:~/server_good$ docker exec -it 3236ea8d3264 bash
root@3236ea8d3264:/# psql --username postgres -d photolife
psql (15.1 (Debian 15.1-1.pgdg110+1))
Type "help" for help.

photolife=# \dt
          List of relations
 Schema |   Name   | Type  |  Owner
--------+----------+-------+----------
 public | authoriz | table | postgres
 public | calls    | table | postgres
 public | medicine | table | postgres
 public | patient  | table | postgres
(4 rows)
```

Схемы базы данных до внесения изменений на Stage:

```
photolife-# \dt
          List of relations
 Schema |   Name   | Type  |  Owner
--------+----------+-------+----------
 public | authoriz | table | postgres
 public | calls    | table | postgres
 public | medicine | table | postgres
 public | patient  | table | postgres
(4 rows)
```

Схемы базы данных до внесения изменений на Prod:

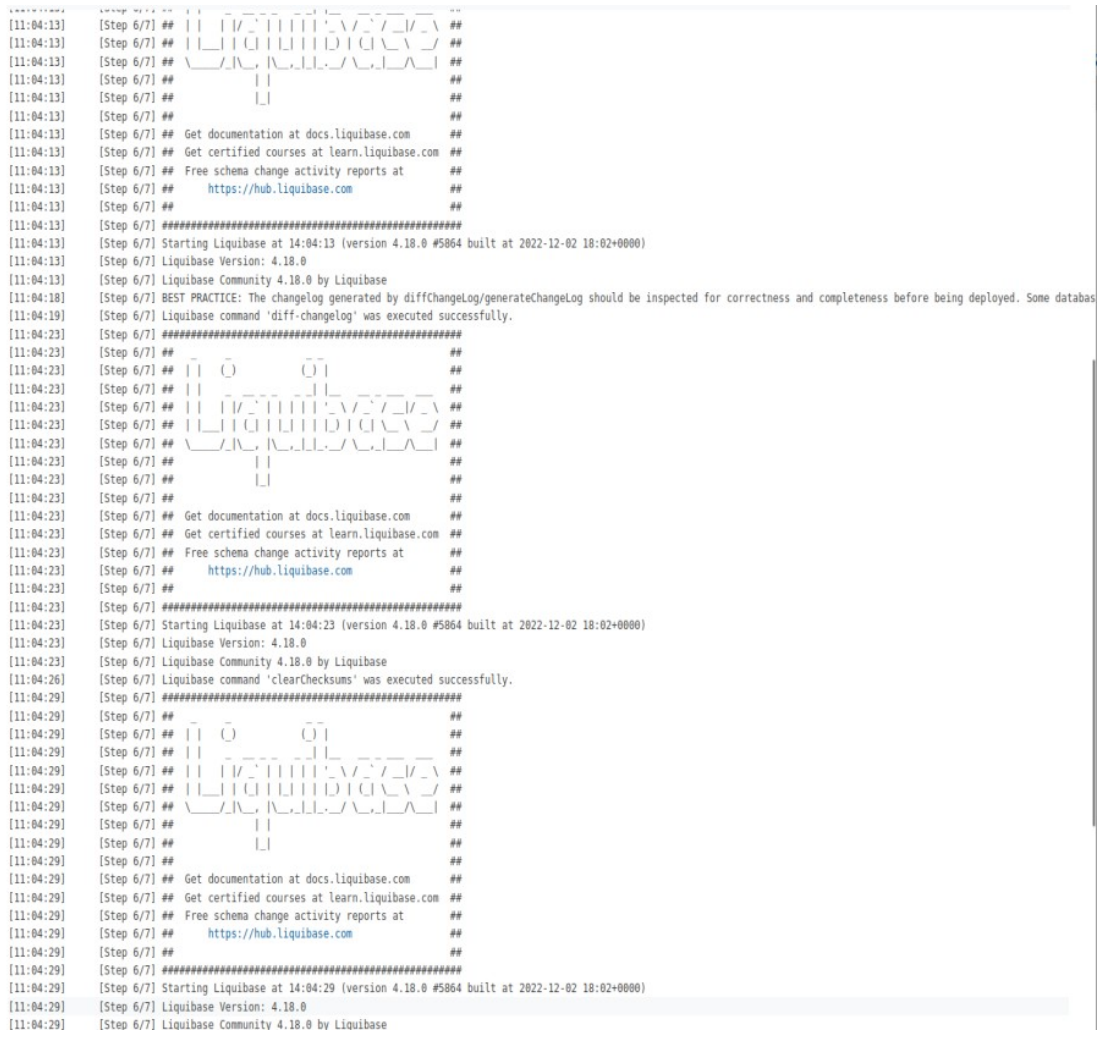


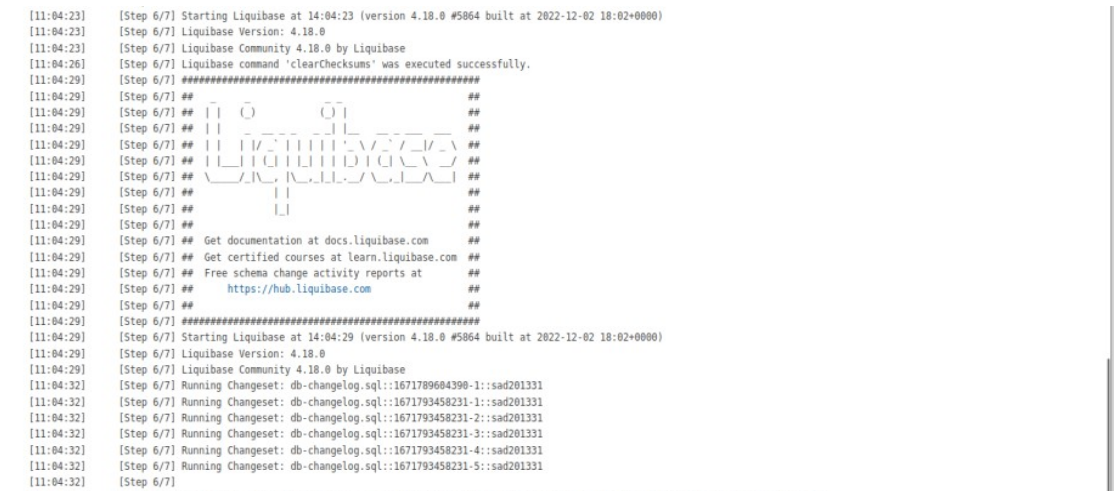
Внесем изменение в базу данных на Test (добавим новую таблицу medicine_new):

```
photolife=# Create table Medicine_new(
photolife(# id serial primary key,
photolife(# name TEXT,
photolife(# descripriom TEXT,
photolife(# effects TEXT);
CREATE TABLE
photolife=# Insert into Medicine_new(id,name,descripriom,effects)
photolife-# values ('1','Teraphlu','Teraphlu can help u if u have hight temperature','No');
INSERT 0 1
photolife=#
```

8. Протестировать работу новых шагов в пайплайне, убедиться, что схемы БД идентичны.

Запустим сборку:





```
[11:04:23]    [Step 6/7] Starting Liquibase at 14:04:23 (version 4.18.0 #5864 built at 2022-12-02 18:02+0000)
[11:04:23]    [Step 6/7] Liquibase Version: 4.18.0
[11:04:23]    [Step 6/7] Liquibase Community 4.18.0 by Liquibase
[11:04:26]    [Step 6/7] Liquibase command 'clearChecksums' was executed successfully.
[11:04:29]    [Step 6/7] ##################################################
[11:04:29]    [Step 6/7] ##  _   _                      _ _                 ##
[11:04:29]    [Step 6/7] ## | | (_)                    (_) |                ##
[11:04:29]    [Step 6/7] ## | |  _  __ _ _   _ _ __ _  _| |__   __ _ ___  ##
[11:04:29]    [Step 6/7] ## | | | |/ _` | | | | '_ \ / _` / __|/ _ \ ##
[11:04:29]    [Step 6/7] ## | |___| | (_| | |_| | | | | (_) | (_| \__ \  __/ ##
[11:04:29]    [Step 6/7] ## \_____/_|\__, |\__,_|_| |_|\__/ \__,_|___/\___| ##
[11:04:29]    [Step 6/7] ##            | |                                 ##
[11:04:29]    [Step 6/7] ##            |_|                                 ##
[11:04:29]    [Step 6/7] ##                                                ##
[11:04:29]    [Step 6/7] ##  Get documentation at docs.liquibase.com       ##
[11:04:29]    [Step 6/7] ##  Get certified courses at learn.liquibase.com  ##
[11:04:29]    [Step 6/7] ##  Free schema change activity reports at        ##
[11:04:29]    [Step 6/7] ##      https://hub.liquibase.com                 ##
[11:04:29]    [Step 6/7] ##                                                ##
[11:04:29]    [Step 6/7] ##################################################
[11:04:29]    [Step 6/7] Starting Liquibase at 14:04:29 (version 4.18.0 #5864 built at 2022-12-02 18:02+0000)
[11:04:29]    [Step 6/7] Liquibase Version: 4.18.0
[11:04:29]    [Step 6/7] Liquibase Community 4.18.0 by Liquibase
[11:04:32]    [Step 6/7] Running Changeset: db-changelog.sql::1671789604390-1::sad201331
[11:04:32]    [Step 6/7] Running Changeset: db-changelog.sql::1671793458231-1::sad201331
[11:04:32]    [Step 6/7] Running Changeset: db-changelog.sql::1671793458231-2::sad201331
[11:04:32]    [Step 6/7] Running Changeset: db-changelog.sql::1671793458231-3::sad201331
[11:04:32]    [Step 6/7] Running Changeset: db-changelog.sql::1671793458231-4::sad201331
[11:04:32]    [Step 6/7] Running Changeset: db-changelog.sql::1671793458231-5::sad201331
[11:04:32]    [Step 6/7]
```

```
[11:12:51]    [Step 7/7] ##      https://hub.liquibase.com                 ##
[11:12:51]    [Step 7/7] ##                                                ##
[11:12:51]    [Step 7/7] ##################################################
[11:12:51]    [Step 7/7] Starting Liquibase at 14:12:51 (version 4.18.0 #5864 built at 2022-12-02 18:02+0000)
[11:12:51]    [Step 7/7] Liquibase Version: 4.18.0
[11:12:51]    [Step 7/7] Liquibase Community 4.18.0 by Liquibase
[11:12:55]    [Step 7/7] BEST PRACTICE: The changelog generated by diffChangeLog/generateChangeLog should be inspected for correctness and completeness before being deployed. Some database ob
[11:12:56]    [Step 7/7] Liquibase command 'diff-changelog' was executed successfully.
[11:12:59]    [Step 7/7] ##################################################
[11:12:59]    [Step 7/7] ##  _   _                      _ _                 ##
[11:12:59]    [Step 7/7] ## | | (_)                    (_) |                ##
[11:12:59]    [Step 7/7] ## | |  _  __ _ _   _ _ __ _  _| |__   __ _ ___  ##
[11:12:59]    [Step 7/7] ## | | | |/ _` | | | | '_ \ / _` / __|/ _ \ ##
[11:12:59]    [Step 7/7] ## | |___| | (_| | |_| | | | | (_) | (_| \__ \  __/ ##
[11:12:59]    [Step 7/7] ## \_____/_|\__, |\__,_|_| |_|\__/ \__,_|___/\___| ##
[11:12:59]    [Step 7/7] ##            | |                                 ##
[11:12:59]    [Step 7/7] ##            |_|                                 ##
[11:12:59]    [Step 7/7] ##                                                ##
[11:12:59]    [Step 7/7] ##  Get documentation at docs.liquibase.com       ##
[11:12:59]    [Step 7/7] ##  Get certified courses at learn.liquibase.com  ##
[11:12:59]    [Step 7/7] ##  Free schema change activity reports at        ##
[11:12:59]    [Step 7/7] ##      https://hub.liquibase.com                 ##
[11:12:59]    [Step 7/7] ##                                                ##
[11:12:59]    [Step 7/7] ##################################################
[11:12:59]    [Step 7/7] Starting Liquibase at 14:12:59 (version 4.18.0 #5864 built at 2022-12-02 18:02+0000)
[11:12:59]    [Step 7/7] Liquibase Version: 4.18.0
[11:12:59]    [Step 7/7] Liquibase Community 4.18.0 by Liquibase
[11:13:02]    [Step 7/7] Liquibase command 'clearChecksums' was executed successfully.
[11:13:05]    [Step 7/7] ##################################################
[11:13:05]    [Step 7/7] ##  _   _                      _ _                 ##
[11:13:05]    [Step 7/7] ## | | (_)                    (_) |                ##
[11:13:05]    [Step 7/7] ## | |  _  __ _ _   _ _ __ _  _| |__   __ _ ___  ##
[11:13:05]    [Step 7/7] ## | | | |/ _` | | | | '_ \ / _` / __|/ _ \ ##
[11:13:05]    [Step 7/7] ## | |___| | (_| | |_| | | | | (_) | (_| \__ \  __/ ##
[11:13:05]    [Step 7/7] ## \_____/_|\__, |\__,_|_| |_|\__/ \__,_|___/\___| ##
[11:13:05]    [Step 7/7] ##            | |                                 ##
[11:13:05]    [Step 7/7] ##            |_|                                 ##
[11:13:05]    [Step 7/7] ##                                                ##
[11:13:05]    [Step 7/7] ##  Get documentation at docs.liquibase.com       ##
[11:13:05]    [Step 7/7] ##  Get certified courses at learn.liquibase.com  ##
[11:13:05]    [Step 7/7] ##  Free schema change activity reports at        ##
[11:13:05]    [Step 7/7] ##      https://hub.liquibase.com                 ##
[11:13:05]    [Step 7/7] ##                                                ##
[11:13:05]    [Step 7/7] ##################################################
[11:13:05]    [Step 7/7] Starting Liquibase at 14:13:05 (version 4.18.0 #5864 built at 2022-12-02 18:02+0000)
[11:13:05]    [Step 7/7] Liquibase Version: 4.18.0
[11:13:05]    [Step 7/7] Liquibase Community 4.18.0 by Liquibase
[11:13:08]    [Step 7/7] Running Changeset: db-changelog.sql::1671793018539-1::sad201331
[11:13:08]    [Step 7/7] Running Changeset: db-changelog.sql::1671793975448-1::sad201331
[11:13:08]    [Step 7/7] Running Changeset: db-changelog.sql::1671793975448-2::sad201331
[11:13:08]    [Step 7/7] Running Changeset: db-changelog.sql::1671793975448-3::sad201331
[11:13:08]    [Step 7/7] Running Changeset: db-changelog.sql::1671793975448-4::sad201331
[11:13:08]    [Step 7/7] Running Changeset: db-changelog.sql::1671793975448-5::sad201331
```

Продублируем новую схему БД на Test:

```
photolife=# Create table Medicine_new(
photolife(# id serial primary key,
photolife(# name TEXT,
photolife(# descripriom TEXT,
photolife(# effects TEXT);
CREATE TABLE
photolife=# Insert into Medicine_new(id,name,descripriom,effects)
photolife-# values ('1','Teraphlu','Teraphlu can help u if u have hight temperature','No');
INSERT 0 1
photolife=#
```

Схема базы данных на Stage:

```
sad201331@sad:~$ docker ps
CONTAINER ID   IMAGE         COMMAND                  CREATED
  STATUS          PORTS                                          NAMES
a3c3a463a697   server-new2   "./server"               About an hour ago
  Up 11 minutes   0.0.0.0:33333->33333/tcp, :::33333->33333/tcp   server
ce1ae65bf175   postgres      "docker-entrypoint.s…"   38 hours ago
  Up 11 minutes   0.0.0.0:5432->5432/tcp, :::5432->5432/tcp       psql
sad201331@sad:~$ docker exec -it ce1ae65bf175  bash
root@db_med:/# psql --username postgres -d photolife
psql (15.1 (Debian 15.1-1.pgdg110+1))
Type "help" for help.

photolife=# \dt
                List of relations
 Schema |         Name          | Type  |  Owner
--------+-----------------------+-------+----------
 public | authoriz              | table | postgres
 public | calls                 | table | postgres
 public | databasechangelog     | table | postgres
 public | databasechangeloglock | table | postgres
 public | medicine              | table | postgres
 public | medicine_new          | table | postgres
 public | patient               | table | postgres
(7 rows)
```

Схема базы данных на Prod: