



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.04.01 Информатика и вычислительная техника

МАГИСТЕРСКАЯ ПРОГРАММА 09.04.01/07 Интеллектуальные системы анализа,
обработки и интерпретации больших данных

О Т Ч Е Т

по лабораторной работе № 1

Название: Инсталляция СУБД PostgreSQL 9.6 на виртуальной машине,
создание базы данных и работа с ней

Дисциплина: Технология параллельных систем баз данных

Студент

ИУ6-12М

(Группа)

(Подпись, дата)

С.В. Астахов

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

(И.О. Фамилия)

Москва, 2023

Введение

1. Цель работы: формирование следующей компетенции: студент должен получить навыки инсталляции СУБД PostgreSQL в среде ОС Ubuntu. Он должен протестировать её работу, разработать небольшую базу данных и выполнить запросы к ней.

Ход выполнения

2. Создание виртуальной машины

Для выполнения курса лабораторных работ с помощью ПО “Virtula Box” была создана виртуальная машина на базе ОС Ubuntu 22.04 (рисунок 1).

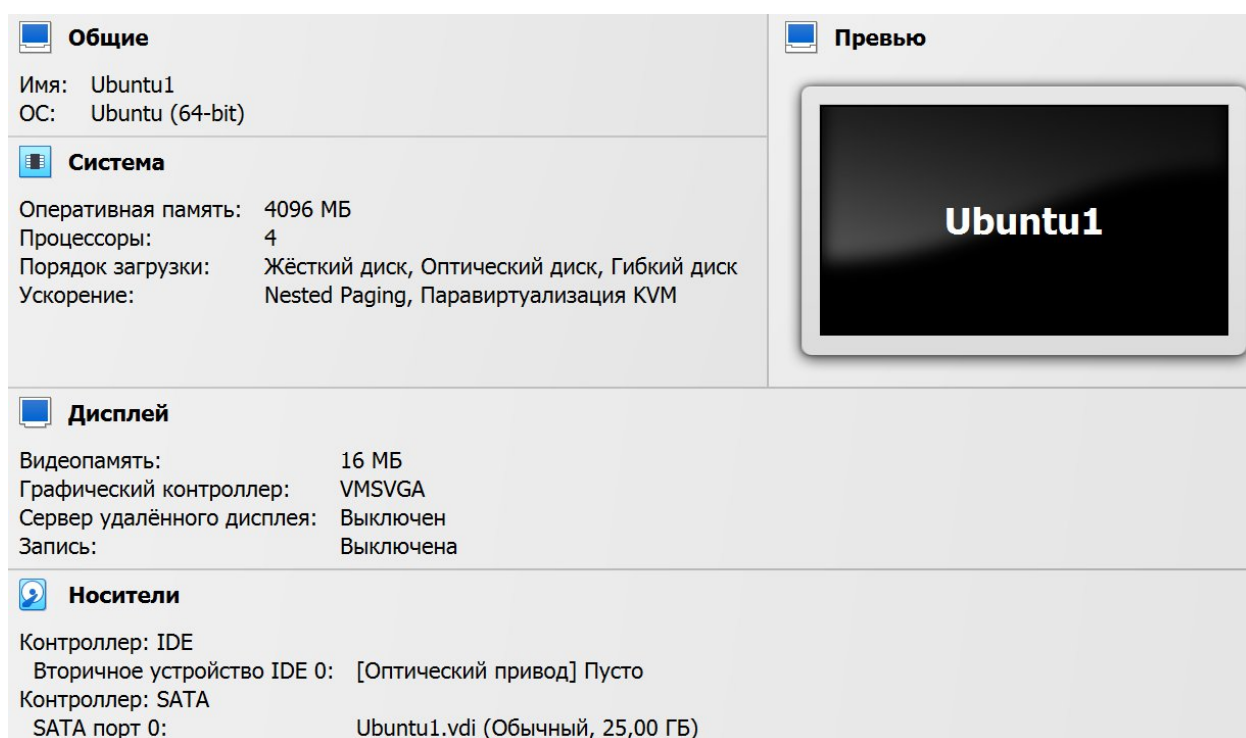


Рисунок 1 — параметры виртуальной машины

3. Инсталляция СУБД PostgreSQL

На рисунке 2 показано добавление ссылки на deb-пакет, получение соответствующего ключа и обновление индекса пакетов. Затем была проведена установка пакета командой “`sudo apt -y install postgresql-9.6`”.

На рисунке 3 показана проверка установки `psql` (запрос пути к исполняемому файлу `psql`), процесс установки пароля, переключение пользователя на `postgres`, создание и подключение к базе данных “`iu6`”.

```

root@Ubuntu1:/home/trickster# sudo sh -c 'echo "deb http://apt.postgresql.org/pub/repos/apt $(lsb_release -cs)-pgdg main" > /etc/apt/sources.list.d/pgdg.list'
root@Ubuntu1:/home/trickster# wget --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo apt-key add -
Warning: apt-key is deprecated. Manage keyring files in trusted.gpg.d instead (see apt-key(8)).
OK
root@Ubuntu1:/home/trickster# sudo apt -y update
Hit:1 http://ru.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://ru.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://ru.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu jammy-security InRelease
Get:5 http://apt.postgresql.org/pub/repos/apt jammy-pgdg InRelease [123 kB]
Get:6 http://apt.postgresql.org/pub/repos/apt jammy-pgdg/main amd64 Packages [288 kB]
Fetched 411 kB in 2s (242 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
52 packages can be upgraded. Run 'apt list --upgradable' to see them.
W: http://apt.postgresql.org/pub/repos/apt/dists/jammy-pgdg/InRelease: Key is stored in legacy trusted.gpg keyring (/etc/apt/trusted.gpg), see the DEPRECATION section in apt-key(8) for details.
N: Skipping acquire of configured file 'main/binary-i386/Packages' as repository 'http://apt.postgresql.org/pub/repos/apt jammy-pgdg InRelease' doesn't support architecture 'i386'
root@Ubuntu1:/home/trickster#

```

Рисунок 2 — загрузка deb-пакета

```

root@Ubuntu1:/home/trickster# which psql
/usr/bin/psql
root@Ubuntu1:/home/trickster# sudo passwd postgres
New password:
Retype new password:
passwd: password updated successfully
root@Ubuntu1:/home/trickster# sudo -i -u postgres
postgres@Ubuntu1:~$ createdb iu6
postgres@Ubuntu1:~$ psql -d iu6
psql (9.6.24)
Type "help" for help.

```

Рисунок 3 — проверка установки и создание базы данных

Создадим пользователя “admin” (рисунок 4). Проверим его существование, подключимся от его имени, создадим последовательность “user_ids” и таблицу “user”. Заполним таблицу и проверим ее содержимое (рисунок 5).

Отберем из таблицы только пользователей с фамилией “Ivanov” (рисунок 6).

```

trickster@Ubuntu1:~$ sudo adduser admin
Adding user `admin' ...
Adding new group `admin' (1001) ...
Adding new user `admin' (1001) with group `admin' ...
Creating home directory `/home/admin' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for admin
Enter the new value, or press ENTER for the default
    Full Name []:
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n] Y

```

Рисунок 4 — создание пользователя admin

```

trickster@Ubuntu1:~$ sed 's/:.*// ' /etc/passwd | grep 'admin'
admin
trickster@Ubuntu1:~$ sudo -u admin psql iu6
could not change directory to "/home/trickster": Permission denied
psql (9.6.24)
Type "help" for help.

iu6=# CREATE SEQUENCE user_ids;
CREATE TABLE users (id INTEGER PRIMARY KEY DEFAULT NEXTVAL('user_ids'), surname VARCHAR(64), firstname VARCHAR(64));
CREATE SEQUENCE
CREATE TABLE
iu6=# INSERT INTO users (surname, firstname) VALUES ('Иванов', 'Иван');
INSERT INTO users (surname, firstname) VALUES ('Петров', 'Пётр');
ERROR:  character with byte sequence 0xd0 0x98 in encoding "UTF8" has no equivalent in encoding "LATIN1"
ERROR:  character with byte sequence 0xd0 0x9f in encoding "UTF8" has no equivalent in encoding "LATIN1"
iu6=# INSERT INTO users (surname, firstname) VALUES ('Ivanov', 'Ivan');
INSERT INTO users (surname, firstname) VALUES ('Petrov', 'Petr');
INSERT 0 1
INSERT 0 1
iu6=# SELECT * FROM users;
SELECT * FROM users WHERE surname='Иванов';
 id | surname | firstname
-----+-----+-----
  1 | Ivanov  | Ivan
  2 | Petrov  | Petr
(2 rows)

```

Рисунок 5 — создание таблицы “users”

```

iu6=# SELECT * FROM users;
SELECT * FROM users WHERE surname='Ivanov';
 id | surname | firstname
-----+-----+-----
  1 | Ivanov  | Ivan
  2 | Petrov  | Petr
(2 rows)

 id | surname | firstname
-----+-----+-----
  1 | Ivanov  | Ivan
(1 row)

```

Рисунок 6 — пример select-запроса

Запросим из БД информацию о таблицах, расширенную информацию о таблицах, информацию о таблице “users”, переключимся между пользователями (рисунок 7).

```

iu6=# \dt
        List of relations
 Schema | Name | Type | Owner
-----+-----+-----+-----
 public | users | table | admin
(1 row)

iu6=# \dt+
                List of relations
 Schema | Name | Type | Owner | Size | Description
-----+-----+-----+-----+-----+-----
 public | users | table | admin | 8192 bytes |
(1 row)

iu6=# \d users
                                Table "public.users"
  Column      |          Type          | Modifiers
-----+-----+-----
 id            | integer                | not null default nextval('users_id_seq'::regclass)
 surname      | character varying(64)  |
 firstname    | character varying(64)  |
Indexes:
    "users_pkey" PRIMARY KEY, btree (id)

iu6=# \c postgres
You are now connected to database "postgres" as user "admin".
postgres=# \c iu6
You are now connected to database "iu6" as user "admin".

```

Рисунок 7 — выполнение полезных команд

4. Работа с PostgreSQL в Python

Установим библиотеку psycopg2 для Python3 (рисунок 8).

```

trickster@Ubuntu1:~$ sudo apt-get install python3-psycopg2
[sudo] password for trickster:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Suggested packages:
  python-psycopg2-doc
The following NEW packages will be installed:
  python3-psycopg2
0 upgraded, 1 newly installed, 0 to remove and 52 not upgraded.
Need to get 133 kB of archives.
After this operation, 487 kB of additional disk space will be used.
Get:1 http://apt.postgresql.org/pub/repos/apt jammy-pgdg/main amd64 python3-psycopg2 a
md64 2.9.6-3.pgdg22.04+1 [133 kB]
Fetched 133 kB in 1s (247 kB/s)
Selecting previously unselected package python3-psycopg2.
(Reading database ... 207549 files and directories currently installed.)
Preparing to unpack .../python3-psycopg2_2.9.6-3.pgdg22.04+1_amd64.deb ...
Unpacking python3-psycopg2 (2.9.6-3.pgdg22.04+1) ...
Setting up python3-psycopg2 (2.9.6-3.pgdg22.04+1) ...
trickster@Ubuntu1:~$ python3

```

Рисунок 7 — установка psycopg2

Создадим файл ex1.py, запрашивающий 10 записей из таблицы “users” и выполним его (рисунки 8-9).

```
GNU nano 6.2 ex1.py
import psycopg2
conn = psycopg2.connect(dbname='iu6', user='admin',
                        password='iu6-magisters', host='localhost')
cursor = conn.cursor()
cursor.execute('SELECT * FROM users LIMIT 10')
for row in cursor:
    print(row)
conn.close()
```

Рисунок 8 — текст программы для выполнения запроса к БД

```
trickster@Ubuntu1:~$ python3 ./ex1.py
(1, 'Ivanov', 'Ivan')
(2, 'Petrov', 'Petr')
```

Рисунок 9 — результат выполнения программы

5. Реализация предметной области

Задание (вариант 1):

Сервис размещения промоутирующих (продвигаемых, рекламируемых) текстов/изображений. Каждый промоутирующий блок может быть привязан к рекламной компании, которая может быть включена/выключена.

Задание:

1. Разработать схему базы данных, состоящую из 3-4 таблиц. Описать её в базе iu6.

2. Включить в каждую таблицу несколько записей.

3. Написать и реализовать в виде одного оператора Select следующий запрос:

«Найти имена сайтов и рекламных компаний, которые продвинули на эти сайты промоутирующие блоки, в текстовой части которых встречается основа слова «автомобиль».

Разработанная схема БД представлена на рисунке 10. В ней представлены следующие таблицы:

- “promotion_blocks” — рекламные блоки;
- “promotion_companies” — компании;

- “promotion_sites” — сайты для размещения рекламы;
- “promotion_bindings” — служебная таблица для реализации связи “многие-ко-многим” между сайтами и компаниями.

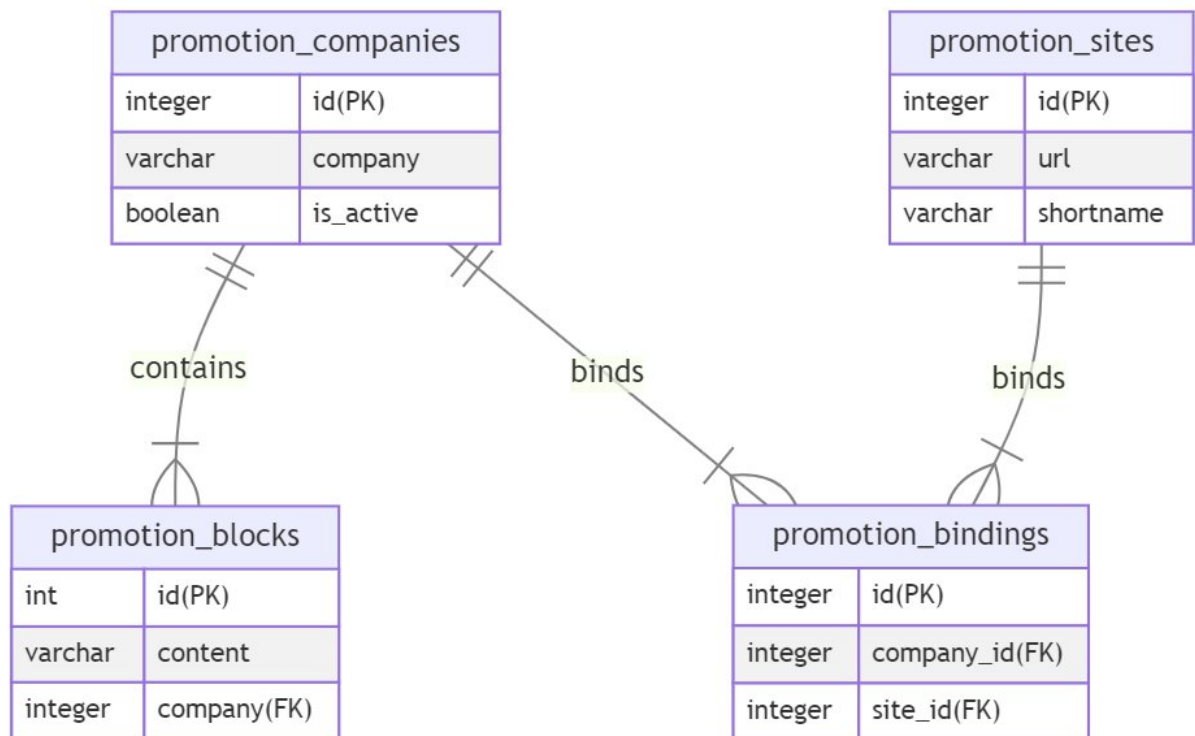


Рисунок 10 — схема БД

Операторы SQL, создающие описанные таблицы и заполняющие их приведены в листинге 1.

Листинг 1 — создание и заполнение таблиц

```

CREATE SEQUENCE promotion_blocks_ids;

CREATE TABLE promotion_blocks (
id INTEGER PRIMARY KEY DEFAULT NEXTVAL('promotion_blocks_ids'),
content VARCHAR(250),
company INTEGER
);

INSERT INTO promotion_blocks (content, company) VALUES ('Toyota - drive your dream', 1);
INSERT INTO promotion_blocks (content, company) VALUES ('Toyota - japanese auto quality', 1);
INSERT INTO promotion_blocks (content, company) VALUES ('Mercedes - das auto', 2);
INSERT INTO promotion_blocks (content, company) VALUES ('Mercedes - deutch auto #1', 2);
INSERT INTO promotion_blocks (content, company) VALUES ('Lada - family automobile', 3);
INSERT INTO promotion_blocks (content, company) VALUES ('Lada - new horizons', 3);
INSERT INTO promotion_blocks (content, company) VALUES ('OcuVue - clear vision', 4);

CREATE SEQUENCE promotion_companies_ids;
    
```

```

CREATE TABLE promotion_companies (
id INTEGER PRIMARY KEY DEFAULT NEXTVAL('promotion_companies_ids'),
company VARCHAR(30),
is_active boolean
);

INSERT INTO promotion_companies (company, is_active) VALUES ('Toyota', TRUE);
INSERT INTO promotion_companies (company, is_active) VALUES ('Mercedez Benz', FALSE);
INSERT INTO promotion_companies (company, is_active) VALUES ('LADA', TRUE);
INSERT INTO promotion_companies (company, is_active) VALUES ('Ocuvue', FALSE);

CREATE SEQUENCE promotion_sites_ids;

CREATE TABLE promotion_sites (
id INTEGER PRIMARY KEY DEFAULT NEXTVAL('promotion_sites_ids'),
url VARCHAR(50),
shortname VARCHAR(15)
);

INSERT INTO promotion_sites (url, shortname) VALUES ('ya.ru', 'yandex');
INSERT INTO promotion_sites (url, shortname) VALUES ('youtube.com', 'youtube');
INSERT INTO promotion_sites (url, shortname) VALUES ('vk.com', 'vKontake');

CREATE SEQUENCE promotion_bindings_ids;

CREATE TABLE promotion_bindings (
id INTEGER PRIMARY KEY DEFAULT NEXTVAL('promotion_bindings_ids'),
company_id INTEGER,
site_id INTEGER
);

INSERT INTO promotion_bindings (site_id, company_id) VALUES (1, 1);
INSERT INTO promotion_bindings (site_id, company_id) VALUES (1, 2);
INSERT INTO promotion_bindings (site_id, company_id) VALUES (2, 2);
INSERT INTO promotion_bindings (site_id, company_id) VALUES (2, 3);
INSERT INTO promotion_bindings (site_id, company_id) VALUES (3, 1);
INSERT INTO promotion_bindings (site_id, company_id) VALUES (3, 3);
INSERT INTO promotion_bindings (site_id, company_id) VALUES (4, 2);

```

Оператор `select`, реализующий требуемый запрос, состоит из объединения всех таблиц по ключам и применения условия для текста рекламного блока. Текст и результат выполнения запроса приведены на рисунке 10.


```

iu6=# select pc.company, shortname as site_name, content
from promotion_blocks pb
join promotion_companies pc ON pb.company = pc.id
join promotion_bindings pbinds ON pbinds.company_id = pc.id
join promotion_sites ps ON pbinds.site_id = ps.id
where content LIKE '%auto%';

```

company	site_name	content
Toyota	yandex	Toyta - japaniese auto qulity
Mercedez Benz	yandex	Mercedes - deutch auto #1
Mercedez Benz	yandex	Mercedes - das auto
Mercedez Benz	youtube	Mercedes - deutch auto #1
Mercedez Benz	youtube	Mercedes - das auto
LADA	youtube	Lada - family automobile
Toyota	vKontake	Toyta - japaniese auto qulity
LADA	vKontake	Lada - family automobile

(8 rows)

Рисунок 10 — запрос и результат его выполнения

Вывод: В ходе выполнения данной лабораторной работы были получены навыки инсталляции СУБД PostgreSQL в ОС Ubuntu. Была протестирована ее работа, а также создана небольшая база данных и разработан запрос к ней в соответствии с вариантом, выданным преподавателем. Задание было успешно выполнено в соответствии с выданным вариантом.