



МИНИСТЕРСТВО НАУКИ  
И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»



**НГТУ  
НЭТИ** | **Факультет прикладной  
математики и информатики**

Кафедра теоретической и прикладной информатики  
Лабораторная работа № 2  
по дисциплине «Администрирование информационных систем»

### ХРАНЕНИЕ ДАННЫХ

Бригада	ХАЙДАЕВ К.Е.
Группа ПМИ-82	ЗЯБЛИЦЕВА У.П.
Вариант	2

Преподаватели      АВРУНЕВ О.Е.

Новосибирск, 2022

## 1 Для б.д. demo получить:

- список схем

psql demo

\dn

```
[dba@centos-7 ~]$ psql demo
psql (14.1)
Type "help" for help.
```

```
demo=# \dn
      List of schemas
  Name | Owner
-----+-----
 bookings | dba
  public  | postgres
(2 rows)
```

- список таблиц в схеме bookings

select \* from pg\_tables where schemaname='bookings';

```
demo=# select * from pg_tables where schemaname='bookings';
 schemaname | tablename      | tableowner | tablespace | hasindexes | hasrules | hastriggers | rowsecurity
-----
 bookings   | ticket_flights | dba        |             | t          | f        | t           | f
 bookings   | boarding_passes | dba        |             | t          | f        | t           | f
 bookings   | aircrafts       | dba        |             | t          | f        | t           | f
 bookings   | flights         | dba        |             | t          | f        | t           | f
 bookings   | airports        | dba        |             | t          | f        | t           | f
 bookings   | seats           | dba        |             | t          | f        | t           | f
 bookings   | tickets         | dba        |             | t          | f        | t           | f
 bookings   | bookings        | dba        |             | t          | f        | t           | f
(8 rows)
```

- список индексов в схеме bookings

select \* from pg\_indexes where schemaname='bookings';

```
demo=# select schemaname,tablename,indexname,tablespace from pg_indexes where schemaname='bookings';
 schemaname | tablename      | indexname                                     | tablespace
-----
 bookings   | aircrafts       | aircrafts_pkey                               |
 bookings   | airports        | airports_pkey                                |
 bookings   | boarding_passes | boarding_passes_flight_id_boarding_no_key    |
 bookings   | boarding_passes | boarding_passes_flight_id_seat_no_key        |
 bookings   | boarding_passes | boarding_passes_pkey                         |
 bookings   | bookings        | bookings_pkey                                 |
 bookings   | flights         | flights_flight_no_scheduled_departure_key    |
 bookings   | flights         | flights_pkey                                  |
 bookings   | seats           | seats_pkey                                    |
 bookings   | ticket_flights  | ticket_flights_pkey                          |
 bookings   | tickets         | tickets_pkey                                  |
(11 rows)

demo=# select indexdef from pg_indexes where schemaname='bookings';
          indexdef
-----
 CREATE UNIQUE INDEX aircrafts_pkey ON bookings.aircrafts USING btree (aircraft_code)
 CREATE UNIQUE INDEX airports_pkey ON bookings.airports USING btree (airport_code)
 CREATE UNIQUE INDEX boarding_passes_flight_id_boarding_no_key ON bookings.boarding_passes USING btree (flight_id, boarding_no)
 CREATE UNIQUE INDEX boarding_passes_flight_id_seat_no_key ON bookings.boarding_passes USING btree (flight_id, seat_no)
 CREATE UNIQUE INDEX boarding_passes_pkey ON bookings.boarding_passes USING btree (ticket_no, flight_id)
 CREATE UNIQUE INDEX bookings_pkey ON bookings.bookings USING btree (book_ref)
 CREATE UNIQUE INDEX flights_flight_no_scheduled_departure_key ON bookings.flights USING btree (flight_no, scheduled_departure)
 CREATE UNIQUE INDEX flights_pkey ON bookings.flights USING btree (flight_id)
 CREATE UNIQUE INDEX seats_pkey ON bookings.seats USING btree (aircraft_code, seat_no)
 CREATE UNIQUE INDEX ticket_flights_pkey ON bookings.ticket_flights USING btree (ticket_no, flight_id)
 CREATE UNIQUE INDEX tickets_pkey ON bookings.tickets USING btree (ticket_no)
(11 rows)
```

**2 Для выбранной таблицы из схемы bookings получить расположение соответствующего файла данных, размер таблицы в страницах, размер соответствующего файла данных (средствами операционной системы).**

2,7	bookings
-----	----------

- Расположение файла bookings

```
select pg_relation_filepath('bookings.bookings');
```

```
demo=# select pg_relation_filepath('bookings.bookings');
pg_relation_filepath
-----
base/16385/16402
(1 row)
```

Каталог base соответствует табличному пространству pg\_default, следующий подкаталог — базе данных, и уже в нем находится интересующий нас файл.

- Размер таблицы

```
select pg_size_pretty(pg_total_relation_size('bookings.bookings'));
```

```
demo=# select pg_size_pretty(pg_total_relation_size('bookings.bookings'));
pg_size_pretty
-----
19 MB
(1 row)
```

Размер 1 страницы 8 кб, тогда количество страниц:

$19 \cdot 1024 / 8 = 2432$  стр.

**Выше указано неверное количество страниц, так как выведен не точный размер таблицы**

```
select pg_total_relation_size('bookings.bookings');
```

```
demo=# SELECT pg_total_relation_size('bookings.bookings');
pg_total_relation_size
-----
19668992
(1 row)
```

$19668992 / 1024 / 8 = 2401$

- Размер файла данных

```
ls -sh /var/lib/pgpro/std-14/data/base/16385/16402
```

```
[dba@centos-7 ~]$ sudo ls -sh /var/lib/pgpro/std-14/data/base/16385/16402
[sudo] password for dba:
14M /var/lib/pgpro/std-14/data/base/16385/16402
```

$14 \cdot 1024 / 8 = 1792$  стр.

**Выше указано неверное количество страниц, так как выведен не точный размер файла**

```
select pg_relation_size('bookings.bookings');
```

```
demo=# SELECT pg_relation_size('bookings.bookings');
pg_relation_size
-----
13713408
(1 row)
```

**13713408/1024/8=1674**

```
SELECT relname, relpages FROM pg_class where relname='bookings';
```

```
demo=# SELECT relname, relpages FROM pg_class where relname='bookings';
relname | relpages
-----+-----
bookings |      1674
(1 row)
```

### 3 Сравнить размер файла и объем таблицы страницах, размер одной страницы равен 8 килобайтам.

Размер файла в страницах: 1674 страниц.

Размер таблицы в страницах: 2401 страниц

### 4 Создать табличное пространство в каталоге, созданном в домашнем каталоге пользователя DBA.

```
mkdir tablespace_1
```

```
sudo chmod 777 /home/dba
```

```
sudo chown postgres:postgres /home/dba/tablespace_1
```

```
psql demo
```

```
CREATE TABLESPACE tablespace_1 OWNER dba LOCATION
'/home/dba/tablespace_1/';
```

```
[dba@centos-7 ~]$ mkdir tablespace_1
[dba@centos-7 ~]$ ll
total 123088
-rw-rw-r--. 1 dba dba 22183920 Nov 11 2016 demo_small_20161013.zip
-rw-rw-r--. 1 dba dba 103857328 Nov 11 2016 demo_small.sql
drwxrwxr-x. 2 dba dba 6 Mar 13 10:17 tablespace_1
[dba@centos-7 ~]$ sudo chmod 777 /home/dba
[dba@centos-7 ~]$ cd
[dba@centos-7 home]$ ll
total 0
drwxrwxrwx. 5 dba dba 207 Mar 13 10:17 dba
[dba@centos-7 home]$ psql demo
psql (14.1)
Type "help" for help.

demo=# CREATE TABLESPACE tablespace_1 OWNER dba LOCATION '/home/dba/tablespace_1/';
ERROR:  directory "/home/dba/tablespace_1" does not exist
demo=# CREATE TABLESPACE tablespace_1 OWNER dba LOCATION '/home/dba/tablespace_1/';
ERROR:  could not set permissions on directory "/home/dba/tablespace_1": operation not permitted
demo=# exit
[dba@centos-7 home]$ cd ~
[dba@centos-7 ~]$ sudo chown postgres:postgres /home/dba/tablespace_1
chown: cannot access '/home/dba/tablespace_1': No such file or directory
[dba@centos-7 ~]$ sudo chown postgres:postgres /home/dba/tablespace_1
[dba@centos-7 ~]$ psql
psql: error: connection to server on socket "/tmp/.s.PGSQL.5432" failed: FATAL:  database "dba" does not exist
[dba@centos-7 ~]$ psql demo
psql (14.1)
Type "help" for help.

demo=# CREATE TABLESPACE tablespace_1 OWNER dba LOCATION '/home/dba/tablespace_1/';
CREATE TABLESPACE
```

chmod 777 'home/dba' – изначально недостаточно прав для работы в домашней директории пользователя dba.

chown postgres:postgres /home/dba/tablespace\_1 – изменяем владельца папки tablespace на пользователь:группа (postgres:postgres).

## 5 Создать отдельную схему

```
CREATE SCHEMA shema_1 AUTHORIZATION dba;
```

```
demo=# CREATE SCHEMA shema_1 AUTHORIZATION dba;  
CREATE SCHEMA
```

## 6 В схеме сделать копию выбранной таблицы, указав в качестве хранения созданное табличное пространство. Заполнить ее данными из исходной.

№ Бригады	Fillfactor, %
1,2	15

```
CREATE TABLE shema_1.bookings_2(LIKE bookings.bookings)  
WITH (fillfactor=15)  
TABLESPACE tablespace_1;
```

```
INSERT INTO shema_1.bookings_2  
SELECT * FROM bookings.bookings;
```

```
demo=# CREATE TABLE shema_1.bookings_2(LIKE bookings.bookings)  
WITH (fillfactor=15)  
TABLESPACE tablespace_1;  
CREATE TABLE  
demo=# INSERT INTO shema_1.bookings_2  
demo=# SELECT * FROM bookings.bookings;  
INSERT 0 262788
```

Like – чтобы создать такие же названия столбцов как в таблице bookings

## 7 Получить для созданной таблицы данные о занятом пространстве и расположении ее файлов. Если количество возвращаемых страниц значительно меньше, чем для исходной, проанализировать таблицу (vacuum analyze) и повторно получить данные.

```
select pg_total_relation_size('shema_1.bookings_2');
```

```
demo=# select pg_total_relation_size('bookings.bookings');  
pg_total_relation_size  
-----  
19668992  
(1 row)  
  
demo=# select pg_total_relation_size('shema_1.bookings_2');  
pg_total_relation_size  
-----  
93650944  
(1 row)
```

## 8 Отключить автоматическую очистку

```
ALTER TABLE shema_1.bookings_2 SET (autovacuum_enabled=off);
demo=# ALTER TABLE shema_1.bookings_2 SET (autovacuum_enabled=off);
ALTER TABLE
```

## 9 Удалить все данные из таблицы, подтвердить транзакцию. Выполнить повторно пункт 7

```
DELETE FROM shema_1.bookings_2;
demo=# DELETE FROM shema_1.bookings_2;
DELETE 262788
demo=# select pg_total_relation_size('bookings.bookings');
pg_total_relation_size
-----
19668992
(1 row)
demo=# select pg_total_relation_size('shema_1.bookings_2');
pg_total_relation_size
-----
93650944
(1 row)
```

Т.к. автоочистка была отключена, размер таблицы после удаления всех данных, остался таким же.

## 10 Выполнить сборку мусора для таблицы (vacuum). Выполнить пункт 7.

```
VACUUM shema_1.bookings_2;
demo=# VACUUM shema_1.bookings_2;
VACUUM
demo=# select pg_total_relation_size('bookings.bookings');
pg_total_relation_size
-----
19668992
(1 row)
demo=# select pg_total_relation_size('shema_1.bookings_2');
pg_total_relation_size
-----
16384
(1 row)
```

Размер таблицы уменьшился до 16 кб.