

Вы подключены к базе данных nebazatis как пользователь Student2025 .
wxze_amanai=#

Имя	Тип данных	Кодировка	Число символов	LC_COLLATE	LC_CTYPE	Число строк	Число строк ICU	Число строк ICU
nebazatis	Student2025	WIN1251	11bc	Russian_Russia.1251	Russian_Russia.1251			
postgres	Student2025	WIN1251	11bc	Russian_Russia.1251	Russian_Russia.1251			
template0	Student2025	WIN1251	11bc	Russian_Russia.1251	Russian_Russia.1251			=c/Student2025 +
template1	Student2025	WIN1251	11bc	Russian_Russia.1251	Russian_Russia.1251			Student2025=CtC/Student2025 +

(4 строк)

wxze_amanai=#

nebazatis=# CREATE DATABASE data_logical;
CREATE DATABASE
wxze_amanai=#c data_logical
Вы подключены к базе данных "data_logical" как пользователь "Student2025".
data_logical=# \l

Имя	Тип данных	Кодировка	Число символов	LC_COLLATE	LC_CTYPE	Число строк	Число строк ICU	Число строк ICU
data_logical	Student2025	WIN1251	11bc	Russian_Russia.1251	Russian_Russia.1251			
nebazatis	Student2025	WIN1251	11bc	Russian_Russia.1251	Russian_Russia.1251			
postgres	Student2025	WIN1251	11bc	Russian_Russia.1251	Russian_Russia.1251			
template0	Student2025	WIN1251	11bc	Russian_Russia.1251	Russian_Russia.1251			=c/Student2025 +
template1	Student2025	WIN1251	11bc	Russian_Russia.1251	Russian_Russia.1251			Student2025=CtC/Student2025 +

(5 строк)

data_logical=#

```

data_logical=# \dn
      имя | тип | тип данных
-----+-----+-----
public | pg_database_owner
(1 запись)

data_logical=# \dn
      имя | тип | тип данных
-----+-----+-----
information_schema | Student2025
pg_catalog | Student2025
pg_toast | Student2025
public | pg_database_owner
(4 записи)

data_logical=# \dn+
      имя | тип | тип данных | тип данных
-----+-----+-----+-----
public | pg_database_owner | pg_database_owner=UC/pg_database_owner+ | standard public schema
(1 запись)

data_logical=# CREATE SCHEMA special;
CREATE SCHEMA
data_logical=# \dn
      имя | тип | тип данных
-----+-----+-----
public | pg_database_owner
special | Student2025
(2 записи)

data_logical=# CREATE TABLE t(n integer);
CREATE TABLE
data_logical=# \dt public.*
      имя | имя | тип | тип данных
-----+-----+-----+-----
public | t | integer | Student2025
(1 запись)

data_logical=# ALTER TABLE t SET SCHEMA special;
ALTER TABLE
data_logical=# \dt public.*
Таблицы с именем "public.*" не найдены.
data_logical=# \dt public.*
Таблицы с именем "public.*" не найдены.
data_logical=# \dt special.*
      имя | имя | тип | тип данных
-----+-----+-----+-----
special | t | integer | Student2025
(1 запись)

```

```

data_logical=# SELECT * FROM special.t;
n
---
(0 ÷÷÷÷÷÷)

data_logical=# SELECT * FROM t;
ОШИБКА: отношение "t" не существует
СТРОКА 1: SELECT * FROM t;
^

data_logical=# SHOW search_path;
search_path
-----
"$user", public
(1 ÷÷÷÷÷÷)

data_logical=# SELECT current_schemas(false);
current_schemas
-----
{public}
(1 ÷÷÷÷÷÷)

data_logical=# SET search_path = public, special;
SET
data_logical=# SELECT * FROM t;
n
---
(0 ÷÷÷÷÷÷)

data_logical=# ALTER DATABASE data_logical SET search_path = public, special;
ALTER DATABASE
data_logical=# \c data_logical
Вы подключены к базе данных "data_logical" как пользователь "Student2025".
data_logical=# SHOW search_path;
search_path
-----
public, special
(1 ÷÷÷÷÷÷)

data_logical=# SELECT dataname FROM pg_database;
ОШИБКА: столбец "dataname" не существует
СТРОКА 1: SELECT dataname FROM pg_database;
^
ПОДСКАЗКА: Возможно, предполагалась ссылка на столбец "pg_database.datname".
data_logical=# SELECT datname FROM pg_database
data_logical=# SELECT datname FROM pg_database;
ОШИБКА: ошибка синтаксиса (примерное положение: "SELECT")
СТРОКА 2: SELECT datname FROM pg_database;
^
data_logical=# SELECT datname FROM pg_database;
datname
-----
postgres
nebazatis
template1
template0
data_logical

```

```

data_logical=# \set ECHO_HIDDEN on
data_logical=# \l
/***** ЗАПРОС *****/
SELECT
  d.datname as "Name",
  pg_catalog.pg_get_userbyid(d.datdba) as "Owner",
  pg_catalog.pg_encoding_to_char(d.encoding) as "Encoding",
  CASE d.dattlocprovider WHEN 'b' THEN 'builtin' WHEN 'c' THEN 'libc' WHEN 'i' THEN 'icu' END AS "Locale Provider",
  d.datcollate as "Collate",
  d.datctype as "Ctype",
  d.dattlocale as "Locale",
  d.daticurules as "ICU Rules",
  CASE WHEN pg_catalog.array_length(d.datacl, 1) = 0 THEN '(none)' ELSE pg_catalog.array_to_string(d.datacl, E'\n') END AS "Access privileges"
FROM pg_catalog.pg_database d
ORDER BY 1;
/*****

```

Имя	Тиффыхы	ИюфшЕютер	ИЕютрщфхЕ юютерш	LC_COLLATE	LC_CTYPE	Юютерш	ИЕртшшр ICU	ИЕртшр фюЕЕЕер
data_logical	Student2025	WIN1251	lihc	Russian_Russia.1251	Russian_Russia.1251			
nebazatis	Student2025	WIN1251	lihc	Russian_Russia.1251	Russian_Russia.1251			
postgres	Student2025	WIN1251	lihc	Russian_Russia.1251	Russian_Russia.1251			
template0	Student2025	WIN1251	lihc	Russian_Russia.1251	Russian_Russia.1251			=c/Student2025 + Student2025=CTc/Student2025
template1	Student2025	WIN1251	lihc	Russian_Russia.1251	Russian_Russia.1251			=c/Student2025 + Student2025=CTc/Student2025

```

(5 строк)

data_logical=# \set ECHO_HIDDEN off
data_logical=# SELECT nspname FROM pg_namespace;
 nspname
-----
 pg_toast
 pg_catalog
 public
 information_schema
 special
(5 строк)

data_logical=# SELECT relname, relkind, relnamespace FROM pg_class WHERE relname = 't';
 relname | relkind | relnamespace
-----+-----+-----
 t       | r       | 16390
(1 строк)

data_logical=# SELECT oid, nspname FROM pg_namespace WHERE oid = 16390;
 oid | nspname
----+-----
 16390 | special
(1 строк)

data_logical=# SELECT relname, relkind, relnamespace::regnamespace::text
data_logical=# FROM pg_class WHERE relname = 't';
 relname | relkind | relnamespace
-----+-----+-----
 t       | r       | special
(1 строк)

```

```

data_logical=# SELECT relname, relkind FROM pg_class
data_logical=# WHERE relnamespace = 'pg_catalog'::regnamespace LIMIT 5;
      relname      | relkind
-----+-----
 pg_statistic      | r
 pg_type           | r
 pg_foreign_table  | r
 pg_proc_oid_index | i
 pg_proc_proname_args_nsp_index | i
(5 строк)

```

```
data_logical=# DROP SCHEMA special;
ОШИБКА: удалить объект схема special нельзя, так как от него зависят другие объекты
ПОДРОБНОСТИ:  таблица t зависит от объекта схема special
ПОДСКАЗКА:  Для удаления зависимых объектов используйте DROP ... CASCADE.
data_logical=# DROP SCHEMA special CASCADE;
ЗАМЕЧАНИЕ:  удаление распространяется на объект таблица t
DROP SCHEMA
```

```
data_logical=# \conninfo
```

Параметр	Значение
БД	data_logical
Подключённый пользователь	Student2025
Узел	localhost
Адрес узла	127.0.0.1
Порт сервера	5432
Параметры	
Версия протокола	3.0
Использован пароль	нет

```
data_logical=# \conninfo
```

Параметр	Значение
БД	data_logical
Подключённый пользователь	Student2025
Узел	localhost
Адрес узла	127.0.0.1
Порт сервера	5432
Параметры	
Версия протокола	3.0
Использован пароль	нет

```
data_logical=# \! chcp 1251
Текущая кодовая страница: 1251
data_logical=# \conninfo
```

Параметр	Значение
БД	data_logical
Подключённый пользователь	Student2025
Узел	localhost
Адрес узла	127.0.0.1
Порт сервера	5432
Параметры	
Версия протокола	3.0
Использован пароль	нет

```
data_logical=# \! chcp 1251
Текущая кодовая страница: 1251
data_logical=# \conninfo
```

Информация о подключении

Параметр	Значение
БД	data_logical
Подключённый пользователь	Student2025
Узел	localhost
Адрес узла	127.0.0.1
Порт сервера	5432
Параметры	
Версия протокола	3.0
Использован пароль	нет
Аутентификация GSSAPI	нет
PID обслуживающего процесса	1976
SSL-подключение	нет
Superuser	on
Hot Standby	off

(13 строк)

```
data_logical=# \c postgres
```

Вы подключены к базе данных "postgres" как пользователь "Student2025".

```
postgres=# \c data_logical
```

Вы подключены к базе данных "data_logical" как пользователь "Student2025".

```
data_logical=# DROP DATABASE data_logical;
```

ОШИБКА: удалить базу данных, открытую в данный момент, нельзя

```
data_logical=# DROP DATABASE data_logical WITH (FORCE);
```

ОШИБКА: удалить базу данных, открытую в данный момент, нельзя

```
data_logical=# DROP DATABASE
```

```
data_logical-# _
```


Практика

Выполните следующие команды в терминале postgres

```
postgres=# CREATE SCHEMA nebasatic;
CREATE SCHEMA
postgres=# CREATE SCHEMA app;
CREATE SCHEMA
postgres=# ALTER USER "Student2025" SET search_path = nebasatic, app, public;
ALTER ROLE
postgres=# \c postgres Student2025
Вы подключены к базе данных "postgres" как пользователь "Student2025".
postgres=# SHOW search_path;
search_path
-----
nebasatic, app, public
(1 ёёёёр)
```

```
postgres=# CREATE TABLE test_auto(id INT);
CREATE TABLE
postgres=# \dt *.*

```

Таблица	Схема	Тип	Пользователь
information_schema	sql_features	Ерсышёр	Student2025
information_schema	sql_implementation_info	Ерсышёр	Student2025
information_schema	sql_parts	Ерсышёр	Student2025
information_schema	sql_sizing	Ерсышёр	Student2025
nebasatic	test_auto	Ерсышёр	Student2025
pg_catalog	pg_aggregate	Ерсышёр	Student2025
pg_catalog	pg_am	Ерсышёр	Student2025
pg_catalog	pg_amop	Ерсышёр	Student2025
pg_catalog	pg_amproc	Ерсышёр	Student2025
pg_catalog	pg_attrdef	Ерсышёр	Student2025
pg_catalog	pg_attribute	Ерсышёр	Student2025
pg_catalog	pg_auth_members	Ерсышёр	Student2025
pg_catalog	pg_authid	Ерсышёр	Student2025
pg_catalog	pg_cast	Ерсышёр	Student2025
pg_catalog	pg_class	Ерсышёр	Student2025
pg_catalog	pg_collation	Ерсышёр	Student2025
pg_catalog	pg_constraint	Ерсышёр	Student2025
pg_catalog	pg_conversion	Ерсышёр	Student2025
pg_catalog	pg_database	Ерсышёр	Student2025
pg_catalog	pg_db_role_setting	Ерсышёр	Student2025
pg_catalog	pg_default_acl	Ерсышёр	Student2025
pg_catalog	pg_depend	Ерсышёр	Student2025
pg_catalog	pg_description	Ерсышёр	Student2025
pg_catalog	pg_enum	Ерсышёр	Student2025
pg_catalog	pg_event_trigger	Ерсышёр	Student2025
pg_catalog	pg_extension	Ерсышёр	Student2025
pg_catalog	pg_foreign_data_wrapper	Ерсышёр	Student2025
pg_catalog	pg_foreign_server	Ерсышёр	Student2025
pg_catalog	pg_foreign_table	Ерсышёр	Student2025
pg_catalog	pg_index	Ерсышёр	Student2025
pg_catalog	pg_inherits	Ерсышёр	Student2025
pg_catalog	pg_init_privs	Ерсышёр	Student2025
pg_catalog	pg_language	Ерсышёр	Student2025
pg_catalog	pg_largeobject	Ерсышёр	Student2025
pg_catalog	pg_largeobject_metadata	Ерсышёр	Student2025
pg_catalog	pg_namespace	Ерсышёр	Student2025
pg_catalog	pg_opclass	Ерсышёр	Student2025
pg_catalog	pg_operator	Ерсышёр	Student2025
pg_catalog	pg_opfamily	Ерсышёр	Student2025
pg_catalog	pg_parameter_acl	Ерсышёр	Student2025
pg_catalog	pg_partitioned_table	Ерсышёр	Student2025
pg_catalog	pg_policy	Ерсышёр	Student2025

```

^Cpostgres=# SELECT spcname FROM pg_tablespace;
    spcname
-----
pg_default
pg_global
(2 rows)

postgres=# \db
      яшёюь ёрсышўэ\i яёюёёёрээёет
      Ъ | тырфыхыЎ | рёяюююцхэшх
-----+-----+-----
pg_default | Student2025 |
pg_global  | Student2025 |
(2 rows)

postgres=# CREATE TABLESPACE ts LOCATION 'C:\PostgreSQL\ts';
CREATE TABLESPACE
postgres=# \db
      яшёюь ёрсышўэ\i яёюёёёрээёет
      Ъ | тырфыхыЎ | рёяюююцхэшх
-----+-----+-----
pg_default | Student2025 |
pg_global  | Student2025 |
ts         | Student2025 | C:\PostgreSQL\ts
(3 rows)

postgres=# CREATE DATABASE data_physical TABLESPACE ts;
CREATE DATABASE
postgres=# \c data_physical
Вы подключены к базе данных "data_physical" как пользователь "Student2025".
data_physical=# CREATE TABLE t(id integer PRIMARY KEY, s text);
CREATE TABLE
data_physical=# INSERT INTO t(id, s)
data_physical=# SELECT id, id::text FROM generate_series(1,100_000) id;
INSERT 0 100000
data_physical=# VACUUM t;
VACUUM
data_physical=# SELECT pg_relation_filepath('t');
      pg_relation_filepath
-----
pg_tblspc/16399/PG_18_202506291/16400/16401
(1 row)

data_physical=# ALTER TABLE t SET TABLESPACE pg_default;
ALTER TABLE
data_physical=# SELECT pg_relation_filepath('t');
      pg_relation_filepath
-----
base/16400/16409
(1 row)

data_physical=# SELECT pg_database_size('data_physical');
      pg_database_size
-----
14972607
(1 row)

```



```

data_physical=# SELECT pg_size_pretty(pg_database_size('data_physical'));
pg_size_pretty
-----
14 MB
(1 ÷)

data_physical=# SELECT pg_size_pretty(pg_total_relation_size('t'));
pg_size_pretty
-----
6568 kB
(1 ÷)

data_physical=# SELECT pg_size_pretty(pg_table_size('t'));
pg_size_pretty
-----
4360 kB
(1 ÷)

data_physical=# SELECT pg_size_pretty(pg_indexes_size('t'));
pg_size_pretty
-----
2208 kB
(1 ÷)

data_physical=# SELECT pg_size_pretty(pg_relation_size('t','main'));
pg_size_pretty
-----
4320 kB
(1 ÷)

data_physical=# SELECT pg_size_pretty(pg_tablespace_size('ts'));
pg_size_pretty
-----
10 MB
(1 ÷)

data_physical=# INSERT INTO t(id, s)
data_physical-# SELECT 0, string_agg(id::text, '.') FROM generate_series(1,5000) AS id;
INSERT 0 1
data_physical=# SELECT pg_size_pretty(pg_table_size('t'));
pg_size_pretty
-----
4416 kB
(1 ÷)

data_physical=# SELECT pg_size_pretty(pg_relation_size('t','main'));
pg_size_pretty
-----
4320 kB
(1 ÷)

```

```

data_physical=# SELECT oid, reltoastrelid::regclass::text FROM pg_class WHERE relname='t';
 oid | reltoastrelid
-----+-----
 16401 | pg_toast.pg_toast_16401
(1         )

data_physical=# SELECT chunk_id, chunk_seq, left(chunk_data::text,45) AS chunk_data
data_physical-# FROM pg_toast.pg_toast_16392;
ОШИБКА: отношение "pg_toast.pg_toast_16392" не существует
СТРОКА 2: FROM pg_toast.pg_toast_16392;
      ^

data_physical=# FROM pg_toast.pg_toast_16401;
ОШИБКА: ошибка синтаксиса (примерное положение: "FROM")
СТРОКА 1: FROM pg_toast.pg_toast_16401;
      ^

data_physical=# SELECT chunk_id, chunk_seq, left(chunk_data::text,45) AS chunk_data
data_physical-# FROM pg_toast.pg_toast_16401;
 chunk_id | chunk_seq | chunk_data
-----+-----+-----
 16412 | 0 | \x545d0000000312e322e332e342e00352e362e372e382
 16412 | 1 | \x392e353161ff31002e3531322e353133002e3531342
 16412 | 2 | \xe215e216e217e218abe219e11a30e11b30e11c30e11
 16412 | 3 | \x11f4aa3611f43611f43611f43611f4aa3611f43611f
 16412 | 4 | \xf43211f4325511f43211f43211f43211f4325511f43
 16412 | 5 | \xf43811f43811f43811f4385511f43811f43811f4381
 16412 | 6 | \x11f4aa3411f43411f43411f43411f4aa3411f43411f
 16412 | 7 | \x0132370132aa370132370132370132370132aa38013
 16412 | 8 | \xf43611f43611f43611f4365511f43611f43611f4361
(9         )

data_physical=# \c postgres
Вы подключены к базе данных "postgres" как пользователь "Student2025".
postgres=# DROP DATABASE data_physical;
DROP DATABASE
postgres=# DROP TABLESPACE ts;
DROP TABLESPACE
postgres=#

```

ПРАКТИКА №2

```

postgres=# CREATE DATABASE testdb;
CREATE DATABASE
postgres=# \c testdb
Вы подключены к базе данных "testdb" как пользователь "Student2025".
testdb=#
testdb=# CREATE TABLESPACE ts LOCATION '/path/to/ts';
ОШИБКА: каталог "/path/to/ts" не существует
testdb=#
testdb=# CREATE TABLE t(
testdb=#     id SERIAL PRIMARY KEY,
testdb=#     name TEXT
testdb=# ) TABLESPACE ts;
ОШИБКА: табличное пространство "ts" не существует
testdb=#
testdb=# INSERT INTO t(name) VALUES
testdb=# ('Ramazan'),
testdb=# ('Gleb'),
testdb=# ('Danil');
ОШИБКА: отношение "t" не существует
СТРОКА 1: INSERT INTO t(name) VALUES
      ^
testdb=#
testdb=# SELECT pg_size_pretty(pg_database_size('testdb')) AS db_size;
      db_size
-----
      8070 kB
(1 ÷)

testdb=# SELECT pg_size_pretty(pg_total_relation_size('t')) AS table_size;
ОШИБКА: отношение "t" не существует
СТРОКА 1: SELECT pg_size_pretty(pg_total_relation_size('t')) AS table_...
      ^
testdb=# SELECT pg_size_pretty(pg_tablespace_size('ts')) AS ts_size;
ОШИБКА: табличное пространство "ts" не существует
testdb=# SELECT pg_size_pretty(pg_tablespace_size('pg_default')) AS default_size;
      default_size
-----
      47 MB
(1 ÷)

testdb=#
testdb=# ALTER TABLE t SET TABLESPACE pg_default;
ОШИБКА: отношение "t" не существует
testdb=#
testdb=# SELECT pg_size_pretty(pg_tablespace_size('ts')) AS ts_size_after;
ОШИБКА: табличное пространство "ts" не существует
testdb=# SELECT pg_size_pretty(pg_tablespace_size('pg_default')) AS default_size_after;
      default_size_after
-----
      47 MB
(1 ÷)

testdb=#
testdb=# DROP TABLESPACE ts;
ОШИБКА: табличное пространство "ts" не существует
testdb=#

```