

## Практическое задание 8

### PostgreSQL

#### Знакомство с PostgreSQL

**Цель.** Знакомство с установкой и возможностями проверки работоспособности СУБД PostgreSQL от российской компании, разработчика программного обеспечения, PostgresPro. Создание экземпляра базы данных. Приобретение навыков работы в pSQL. Создание пользователей и настройка их прав.

**Задачи:**

- установка СУБД PostgreSQL на персональный компьютер и изучение установленного экземпляра
- проверка работы экземпляра БД
- знакомство с файлами экземпляра БД
- использование приложения SQL Shell (psql)
- создание пользователя и настройка прав доступа
- использование GUI PgAdmin

#### *Краткие теоретические сведения*

Для выполнения лабораторной работы мы используем СУБД от российского производителя – компании Postgres Professional. Этот программный продукт построен на базе глубоко переработанного форка PostgreSQL для высоконагруженных систем. Объем переработок исходного кода превышает 20%, что обеспечивает большой набор уникальной функциональности. Продукт компании полностью отвечает требованиям корпоративных пользователей в части производительности, безопасности и надежности. Компания существует на российском рынке с 2015 года и очень много делает для продвижения СУБД PostgreSQL, включая активную работу с документацией и образовательным сообществом.

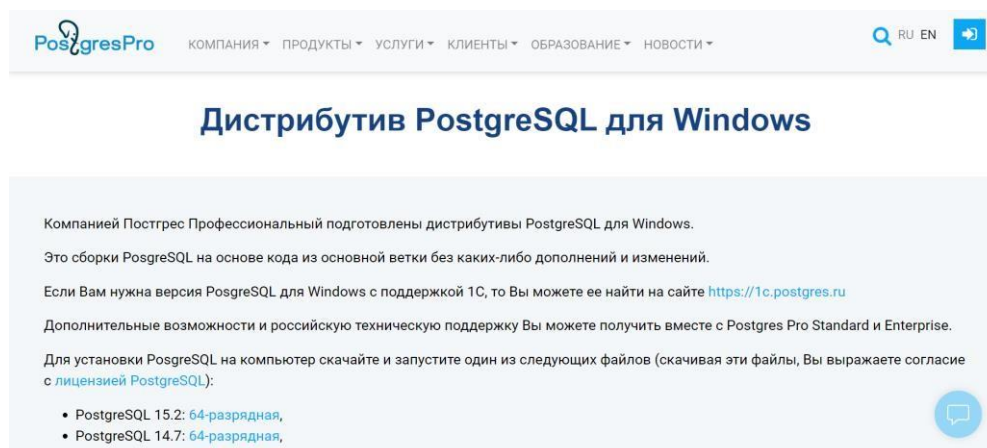
Для более серьезного ознакомления с СУБД PostgreSQL от компании Postgres Professional рекомендуем обратиться к их сайту по адресу [postgrespro.ru](https://postgrespro.ru).

### *Установка экземпляра СУБД*

Зайдите на сайт [postgrespro.ru](https://postgrespro.ru). Скачайте и установите на своем компьютере подходящую вам версию СУБД, не требующую абонентской платы или покупки лицензии.

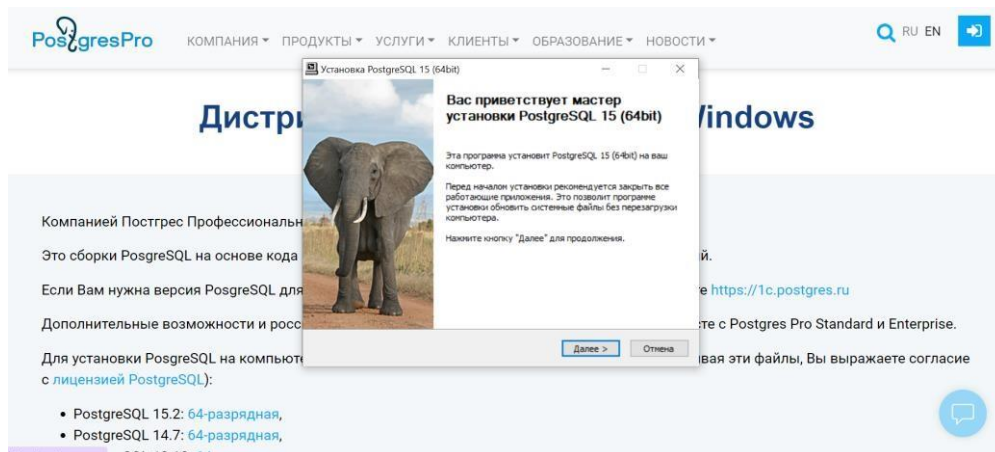


Компания PostgresPro располагает дистрибутивами под все основные типы операционных систем. Например, для Windows, набор дистрибутивов находится по ссылке <https://postgrespro.ru/windows>.

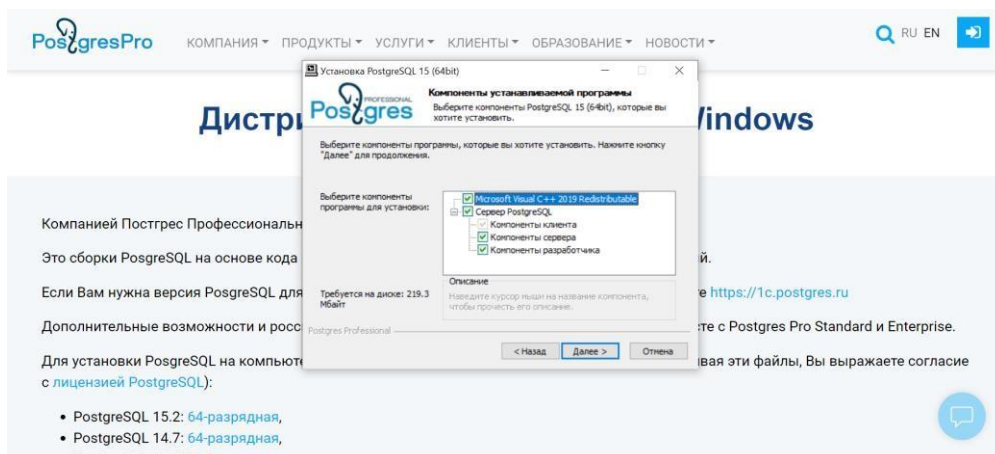


Дальнейшие действия в настоящем пособии будут производиться с дистрибутивом PostgreSQL 15.2: 64-разрядная, однако остальные варианты дистрибутивов демонстрируют ровно такую-же последовательность действий и функционал, поэтому предлагаемые в настоящих методических указаниях задания и методы их решения актуальны и для них.

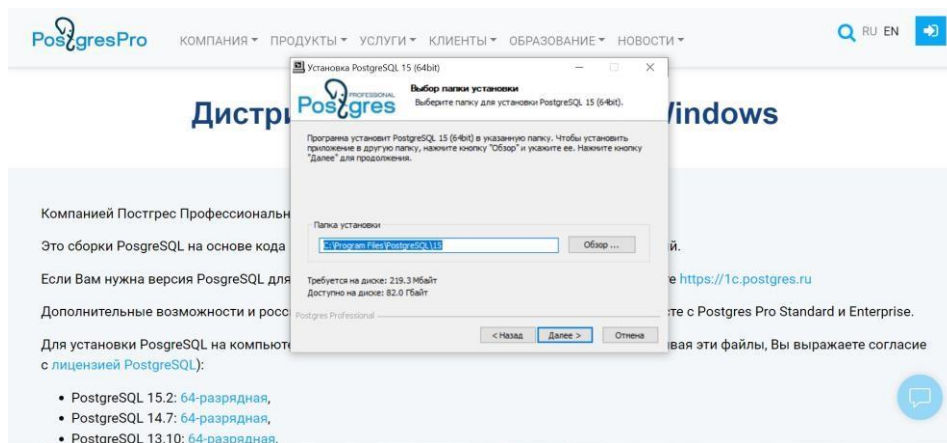
Размер дистрибутива порядка 80 мегабайт, поэтому рассчитывайте время скачивания и способ соединения с сетью Internet.



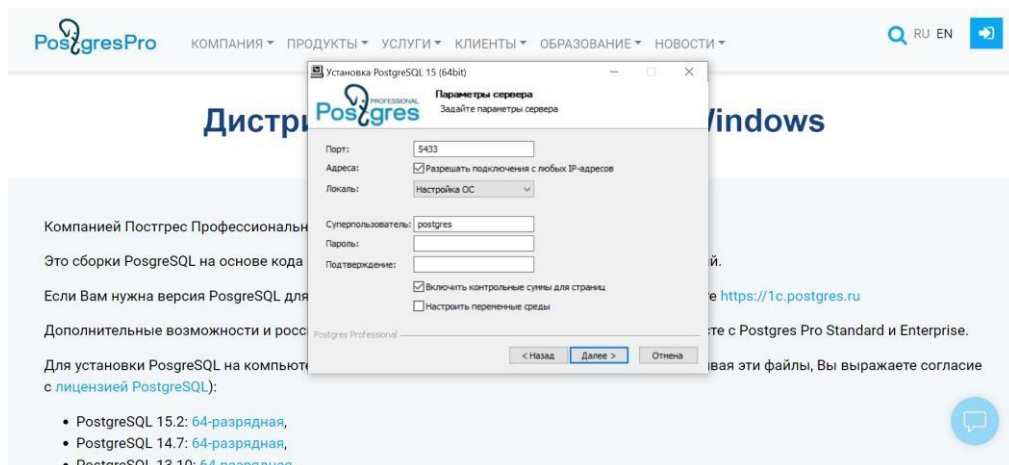
После запуска установки из дистрибутива вам несколько раз предложат выбрать параметры установки. Оптимально действовать по умолчанию, так как это подходит большинству пользователей. Предоставляйте установщику соответствующие разрешения и подтверждайте производимые действия.



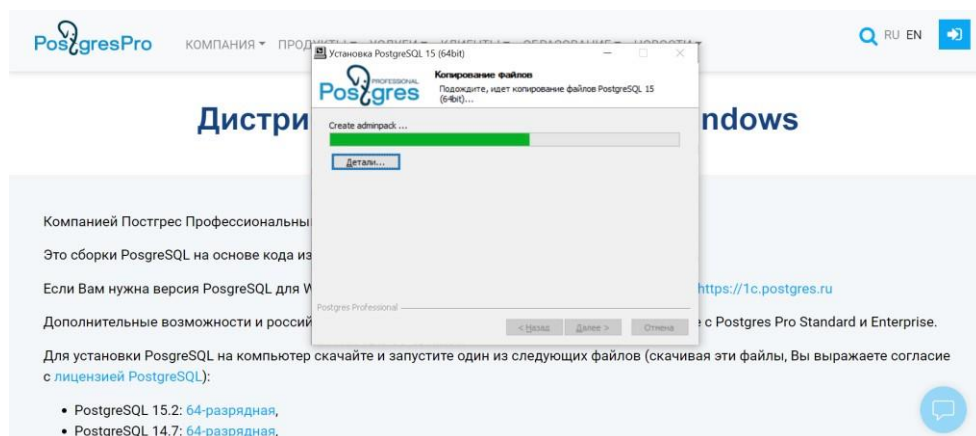
При настройках по умолчанию развернутый на диске дистрибутив займет порядка 220 мегабайт.



Важным шагом является установка пароля суперпользователя. Если вы планируете пользоваться СУБД на персональном компьютере, к которому не имеют доступа другие пользователи, то рекомендуем задать пароль postgres, совпадающий с именем суперпользователя.

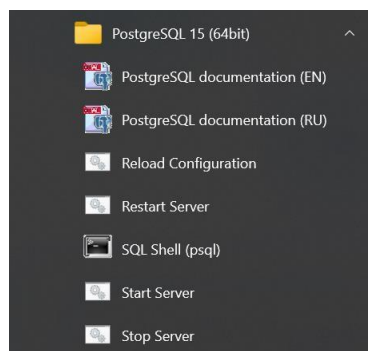


После установки пароля, выбора имени папки в меню Пуск и подтверждения начала процесса копирования, дистрибутив развернет на вашем компьютере экземпляр СУБД PostgreSQL.



### *Изучение установленного экземпляра*

В случае успешного окончания установки, в списке установленных программ меню Пуск появится набор ярлыков для работы СУБД PostgreSQL, включающий в себя документацию к СУБД на русском и английском языках, запуск, остановку и перезапуск сервера баз данных, перезагрузку конфигурации, а также запуск консольного приложения SQL Shell (psql), позволяющего создавать, наполнять и администрировать базы данных в режиме командной строки.



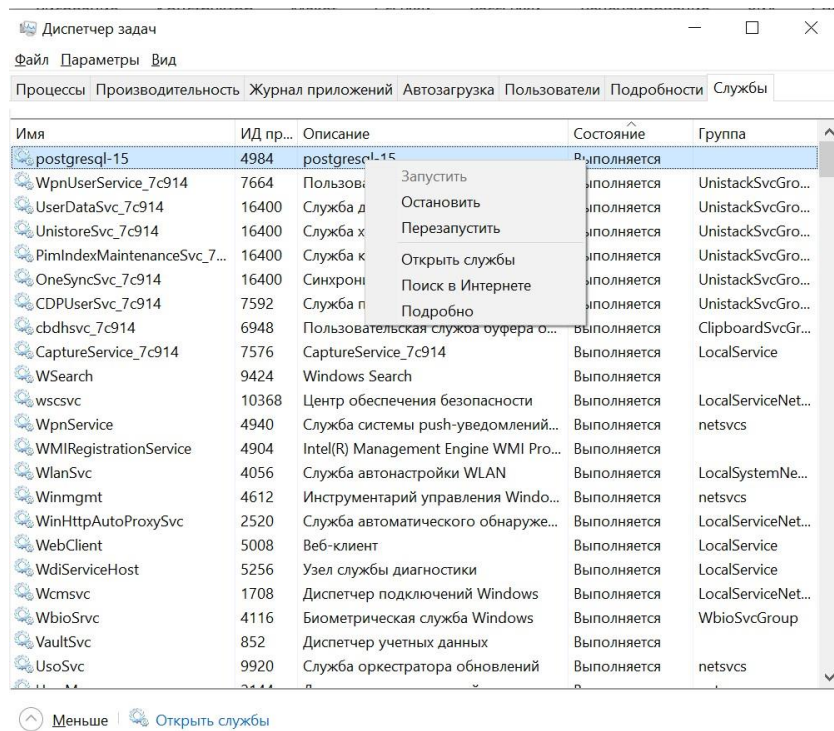
При установке PostgreSQL в вашей системе регистрируется служба «postgresql-15». Она запускается автоматически при старте компьютера под учетной записью Network Service (Сетевая служба). При необходимости вы можете изменить параметры службы с помощью стандартных средств Windows. Чтобы временно остановить службу сервера баз данных, выполните программу «Stop Server» из папки в меню «Пуск», которую вы указали при установке.

Для запуска службы там же находится программа «Start Server». Если при запуске службы произошла ошибка, для поиска причины следует заглянуть в журнал сообщений сервера. Он находится в подкаталоге log каталога, выбранного при установке для баз данных (обычно C:\Program Files\PostgreSQL\15\data\log). Журнал настроен так, чтобы запись периодически переключалась в новый файл. Найти актуальный файл можно по дате последнего изменения или по имени, которое содержит дату и время переключения.

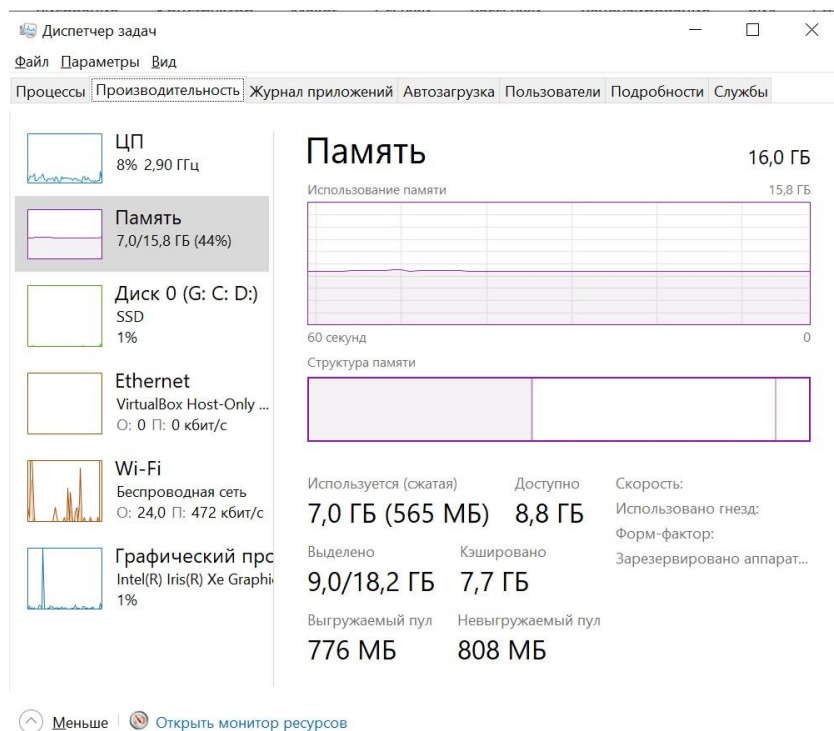
### ***Проверка работы экземпляра БД***

Запустите Диспетчер задач (CTRL+ALT+DEL) и найдите на вкладке «Службы» службу postgresql-15, посмотрите на доступные для нее управляющие команды, попробуйте остановить службу с их помощью.





Перейдите на вкладку «Производительность» и выберите отображение памяти. Попробуйте с помощью ярлыков «Stop server» и «Start server» из меню «Пуск» остановить, а потом запустить сервер баз данных PostgreSQL. Понаблюдайте как меняется объем используемой памяти, сделайте выводы.



*Знакомство с файлами экземпляра БД*

СУБД, созданные при установке и все созданные вами базы данных. Есть несколько важных конфигурационных файлов, которые определяют настройки сервера. Они располагаются в каталоге баз данных. Вам не нужно их изменять, чтобы начать знакомство с PostgreSQL, но в реальной работе они непременно потребуются:

- `postgresql.conf` — это основной конфигурационный файл, содержащий значения параметров сервера;
- `pg_hba.conf` — файл, определяющий настройки доступа. В целях безопасности по умолчанию доступ должен быть подтвержден паролем и допускается только с локального компьютера.

Эти файлы прекрасно документированы, поэтому вы можете в любое время ознакомиться с их содержанием.

Кроме того, процесс установки экземпляра подробно логируется в файл `install.log`, который расположен в каталоге установки экземпляра

Также стоит отметить, что все базы, в том числе и созданные лично вами, находятся в каталоге `C:\Program Files\PostgreSQL\15\data\base` и вы можете самостоятельно в этом убедиться, зафиксировав состояние каталога до создания вами базы и после выполнения этого действия.

### ***Использование приложения SQL Shell (psql)***

Чтобы подключиться к серверу СУБД и выполнить какие-либо команды, требуется программа-клиент. Речь идет о терминальном клиенте `psql`, работа с которым происходит интерактивно в режиме командной строки. Многие скептически относятся к подобным средам управления информационными системами, но у него есть два неоспоримых преимущества:

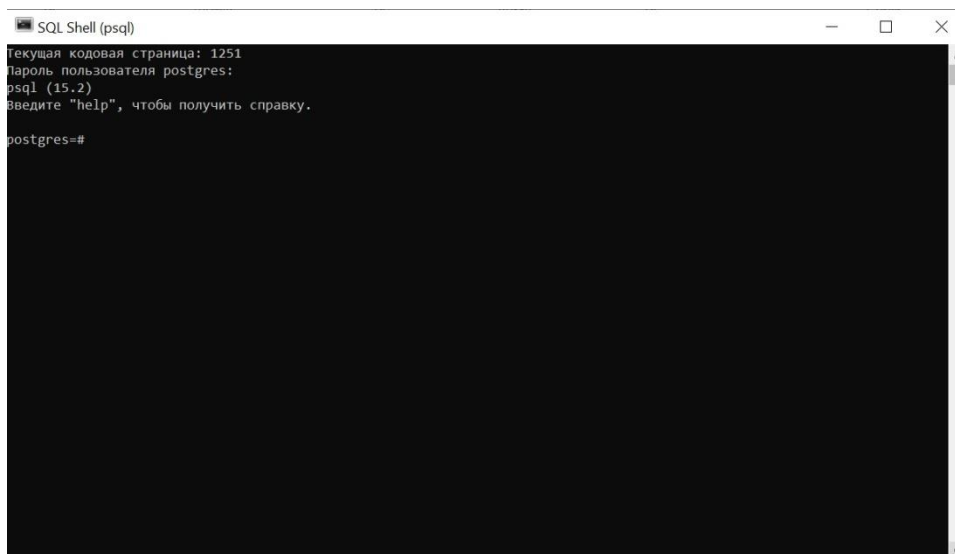
- 1) `psql` — стандартный клиент, он входит в любую сборку PostgreSQL и поэтому всегда под рукой.

2) `psql` очень удобен для решения повседневных задач по администрированию баз данных, для написания небольших запросов и автоматизации процессов, например, для периодической установки изменений программного кода на сервер СУБД. Он имеет собственные команды, позволяющие сориентироваться в объектах, хранящихся в базе данных, и удобно представить информацию из таблиц.

Для активации клиента в ОС Windows запустите программу «SQL Shell (`psql`)» из меню «Пуск».



В ответ на запрос введите пароль пользователя `postgres`, который вы указали при установке PostgreSQL. Обратите внимание, что при вводе пароля в консоли не отображаются вводимые символы или их замена, поэтому по окончании ввода пароля просто нажмите `Enter`.





Пользователи Windows могут столкнуться с проблемой неправильного отображения символов русского языка в терминале. В этом случае убедитесь, что свойствах окна терминала установлен TrueType шрифт (обычно «LucidaConsole» или «Consolas»).

Вы видите одинаковое приглашение `postgres=#`. «Postgres» здесь — имя базы данных, к которой вы сейчас подключены. Один сервер PostgreSQL может обслуживать несколько баз данных, но одновременно вы работаете только с одной из них.

В консольном окне «SQL Shell (psql)» осуществите следующие действия.

Создадим новую базу данных с именем `test`:

```
postgres=# CREATE DATABASE test;  
CREATE DATABASE
```

Не забудьте про точку с запятой в конце команды — пока PostgreSQL не увидит этот символ, он будет считать, что вы продолжаете ввод (так что команду можно разбить на несколько строк). Теперь переключимся на созданную базу:

```
postgres=# \c test  
You are now connected to database "test" as user  
"postgres".  
test=#
```

Как вы видите, приглашение сменилось на `test=#`.

Команда, которую мы только что ввели, не похожа на SQL — она начинается с обратной косой черты. Так выглядят специальные команды, которые понимает только `psql` (поэтому в других управляющих оболочках PostgreSQL необходимо найти аналог).

Команд `psql` довольно много, а полный список с кратким описанием можно получить с помощью команды `test=# \?`

### ***Создание пользователя и настройка прав доступа***

Когда вы открываете окно консоли SQL Shell (`psql`), то вас просят ввести пароль пользователя, а значит есть возможность создавать и

наделять правами собственных пользователей. Выполните следующую команду в консоли:

```
test=# \l
```

Вам выведется список существующих в вашем экземпляре СУБД баз данных.

```
test=# \l
```

Имя		Владелец	Кодировка	LC_COLLATE	Список баз данных LC_CTYPE	локаль ICU	Провайдер локали	Права доступа
postgres		postgres	UTF8	Russian_Russia.1251	Russian_Russia.1251		libc	
template0		postgres	UTF8	Russian_Russia.1251	Russian_Russia.1251		libc	=c/postgres + postgres=CTC/postgres
template1		postgres	UTF8	Russian_Russia.1251	Russian_Russia.1251		libc	=c/postgres + postgres=CTC/postgres
test		postgres	UTF8	Russian_Russia.1251	Russian_Russia.1251		libc	

(4 строки)

Теперь давайте создадим нового пользователя test\_user с паролем

```
CREATE USER test_user PASSWORD 'qwerty';
```

```
test=# CREATE USER test_user PASSWORD 'qwerty';
CREATE ROLE
test=#
```

Давайте проверим, появился ли он в списке пользователей нашего экземпляра. Для этого используем команду \dg в консоли:

```
test=# \dg
```

Имя роли	Список ролей Атрибуты	Член ролей
postgres	Суперпользователь, Создаёт роли, Создаёт БД, Репликация, Пропускать RLS	{}
test_user		{}

После создания пользователя (роли) необходимо определить его права на взаимодействие с базами данных текущего экземпляра СУБД. В зависимости от необходимости вы можете наделить созданного пользователя правами суперпользователя, включив его в роль postgres, а можете использовать варианты ролей, уже существующих в PostgreSQL по умолчанию. Для наделения правами пользователя, используйте команду GRANT.

```
test=# GRANT postgres to test_user;
GRANT ROLE
test=# \dg
```

Имя роли	Список ролей Атрибуты	Член ролей
postgres	Суперпользователь, Создаёт роли, Создаёт БД, Репликация, Пропускать RLS	{}
test_user		{postgres}

Если вы хотите использовать для администрирования базы данных существующие по умолчанию роли – обратитесь к документации по

СУБД, расположенной по адресу:

```
test=# GRANT pg_write_all_data to test_user;  
GRANT ROLE  
test=# \dg
```

Имя роли	Список ролей	Член ролей
postgres	Суперпользователь, Создаёт роли, Создаёт БД, Репликация, Пропускать RLS	{}
test_user		{pg_write_all_data,postgres}

Обратите внимание на изменение перечня ролей пользователя test\_user.

После создания пользователя и при наличии интересующей вас базы данных вы можете подключиться к ней под соответствующим пользователем с помощью консольной команды \с [имя базы] [имя пользователя]:

```
test=# \с postgres test_user  
Пароль пользователя test_user:  
Вы подключены к базе данных "postgres" как пользователь "test_user".  
postgres=> _
```

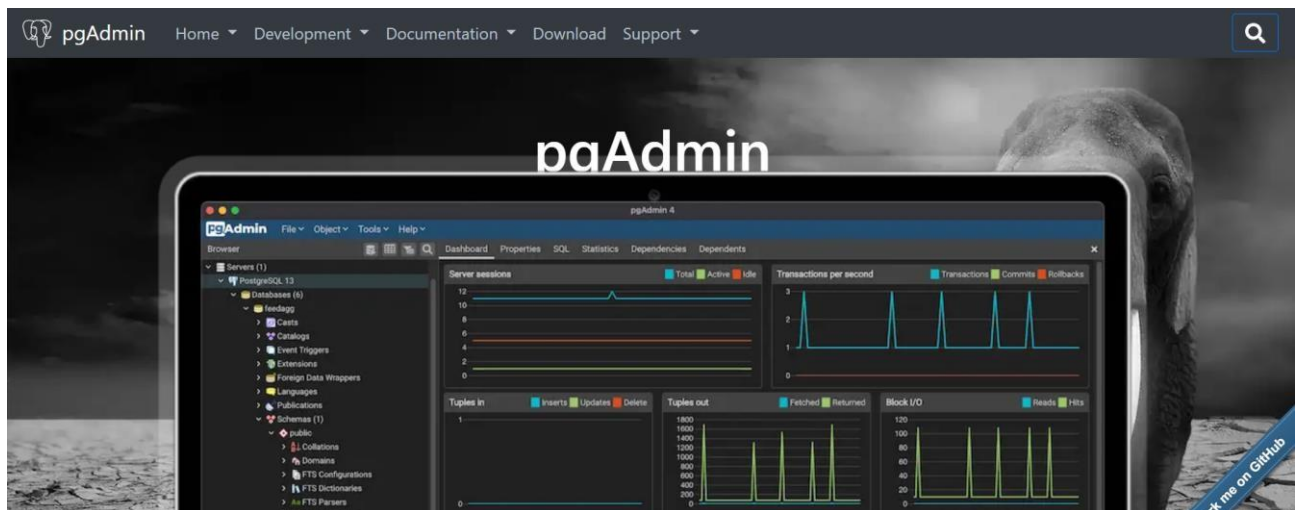
В текущей базе данных можно создать таблицу, в которой будет накапливаться и храниться интересующая вас информация. Для этого вам следует воспользоваться SQL - оператором CREATE TABLE.

```
postgres=> CREATE TABLE klass(  
postgres(> "Id_class" "char" NOT NULL,  
postgres(>     name character varying(15),  
postgres(>     firsting "char");  
CREATE TABLE
```

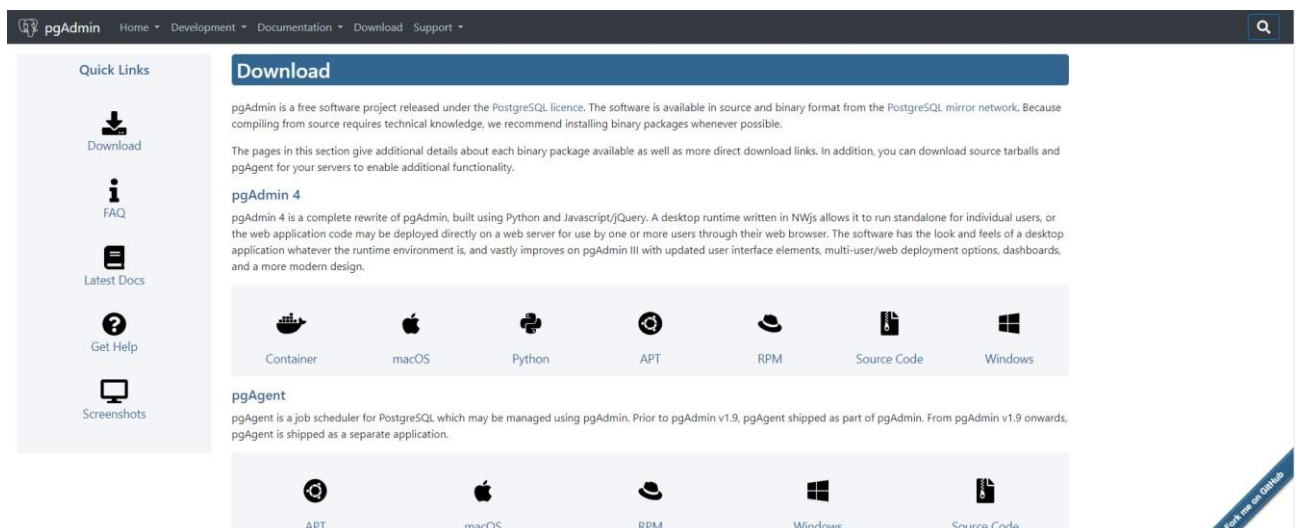
В следующих лабораторных работах мы подробно разберемся с синтаксисом и возможностями SQL-операторов в СУБД PostgreSQL.

### ***Использование GUI PgAdmin***

Для СУБД PostgreSQL существует несколько зарекомендовавших себя графических управляющих оболочек, реализующих большинство функционала СУБД с помощью привычного оконно/вкладочного интерфейса. Одной из самых популярных оболочек является свободно-распространяемое ПО PgAdmin. Для знакомства с этим программным продуктом следует зайти на сайт [pgadmin.org](http://pgadmin.org).

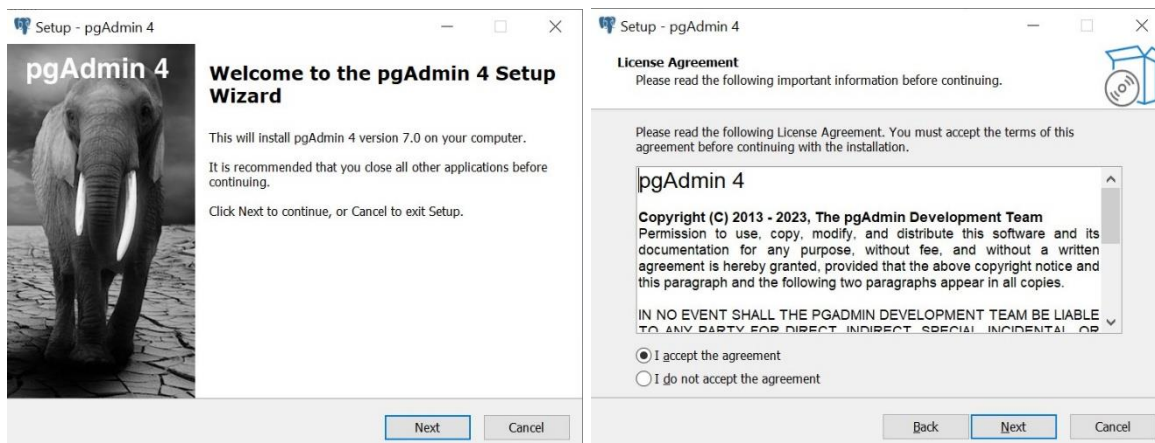


В разделе загрузок по адресу <https://www.pgadmin.org/download/>, необходимо выбрать вашу операционную систему и скачать установочный файл.

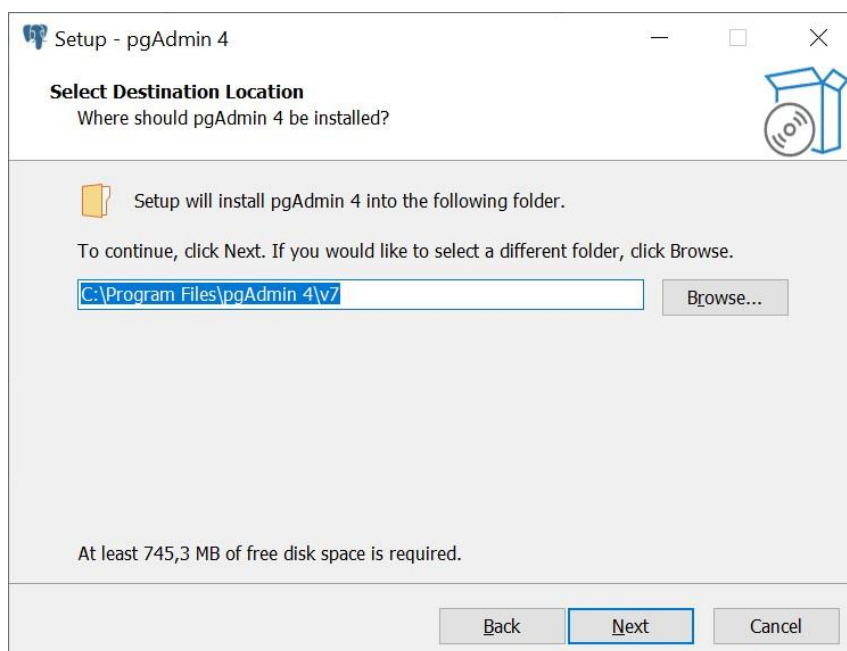


Для операционной системы Windows установочный файл весит порядка 170 мегабайт, поэтому необходимо правильно выбрать способ подключения к сети и аппаратную составляющую для работы с СУБД.

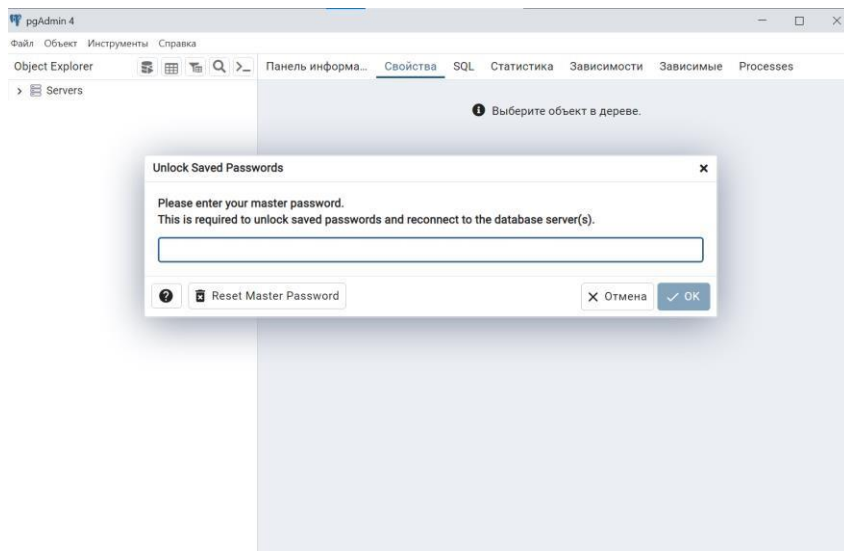
Процесс установки PgAdmin тривиален, большинство диалоговых окон содержат значения по умолчанию, не требующие изменения при стандартных требованиях.



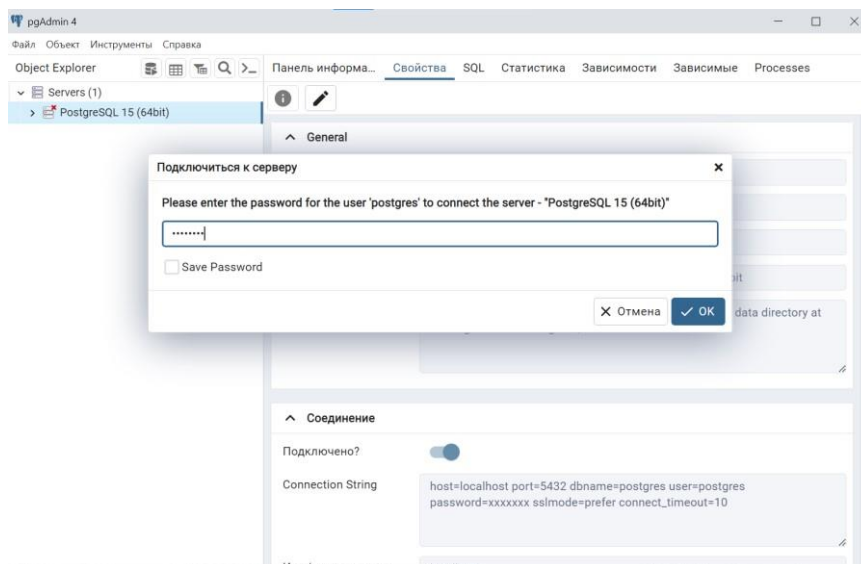
Однако, следует заметить, что в развернутом состоянии пакет PgAdmin займет на диске более 740 мегабайт, поэтому еще раз обращаем ваше внимание на требования к аппаратному обеспечению.



При первом запуске PgAdmin потребует создать мастер-пароль, предоставляющий доступ ко всему функционалу, он может требоваться и в процессе работы программы, поэтому внимательно отнеситесь к этому шагу.

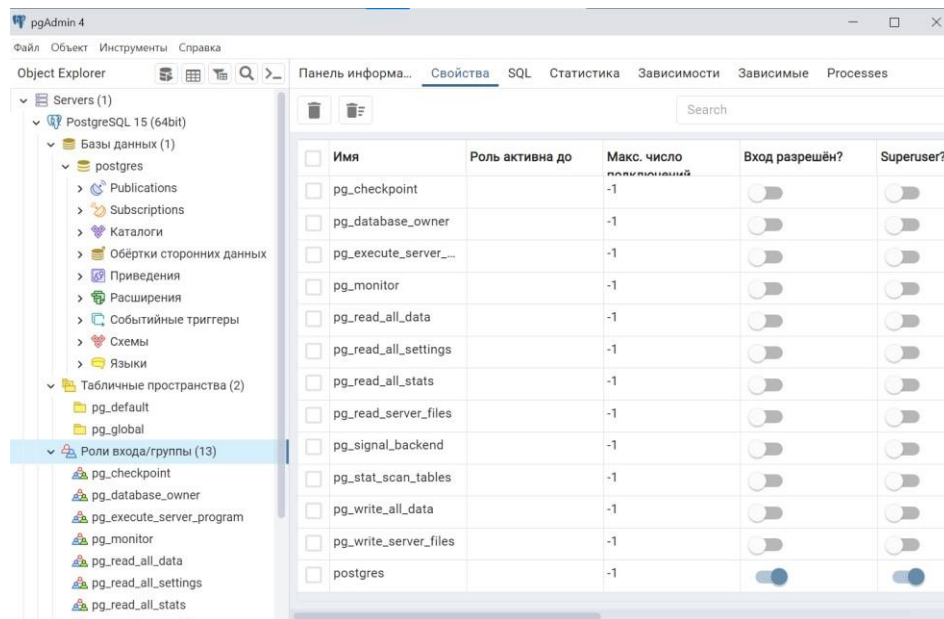


PgAdmin не содержит непосредственно СУБД, поэтому перед его применением необходимо установить PostgreSQL на ваш компьютер. Установленный сервер баз данных отобразится в древовидном каталоге оболочки слева, а при попытке подключиться к нему вам потребуются пароль суперпользователя СУБД, который вы создавали на этапе установки PostgreSQL.

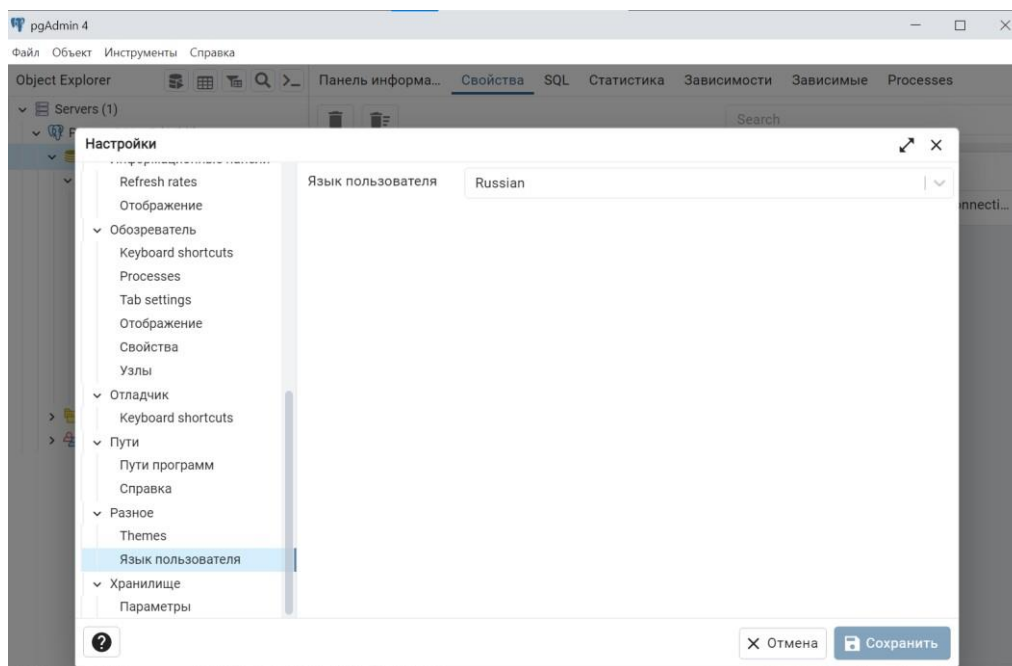


После успешного подключения к серверу баз данных вам станет доступна вся его структура и функционал в визуально удобной форме. В частности, на верхнем уровне проекта, вам будут видны все базы данных, созданные в данном сервере СУБД, все существующие табличные пространства и созданные роли/пользователи.





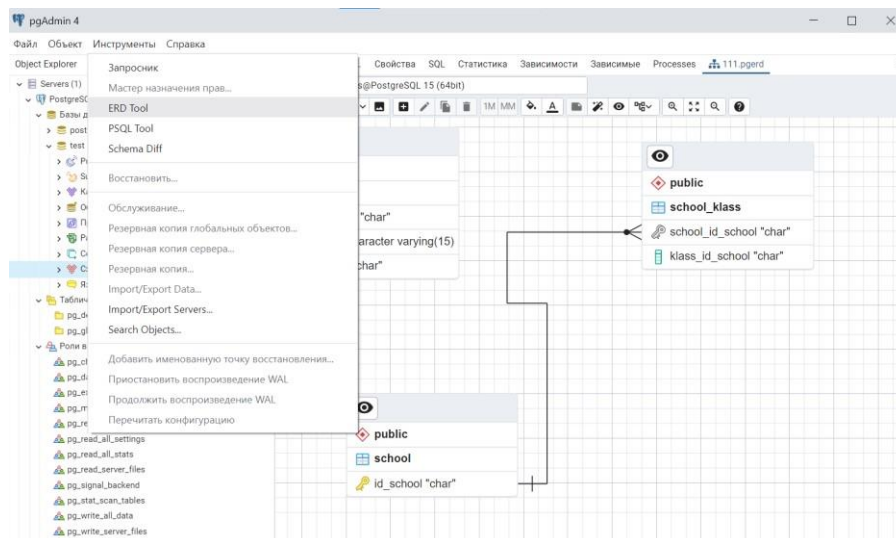
По умолчанию языком программного интерфейса является английский язык, но вы можете самостоятельно установить русский язык в соответствующей части настроек программы.



Для каждого типа объектов, которые могут быть выбраны в левом вертикальном древовидном каталоге, доступен свой набор вкладок, имеющих на горизонтальной панели справа вверху общего окна. Кроме того, этот список вкладок пополняется активными вкладками для различных объектов, с которыми в настоящий момент идет работа в программе. Открытые вкладки могут

быть легко закрыты, но будьте аккуратны при закрытии, чтобы не потерять созданную в процессе работы информацию.

Для нас будут обладать несомненной ценностью дополнительные инструменты работы, которые предоставляет PgAdmin. Например, крайне полезным является инструмент ERD



Этот инструмент позволяет с помощью визуального проектирования создавать EntityRelationshipDiagram – диаграмму схемы базы данных на основе модели «Сущность-Связь». Инструмент позволяет создавать таблицы, наполнять их атрибутами, задавать их тип и свойства. Кроме того, ERD Tool позволяет создавать ограничения и связи типов 1:M и M:M между таблицами. И, наконец, после создания диаграммы, можно сгенерировать SQL код, который позволит в активной базе данных создать все таблицы диаграммы с учетом всех указанных на диаграмме свойств, связей и ограничений.

Для получения базовых навыков работы с GUI PgAdmin, создайте в нем новую базу данных, пользователя и таблицу (с использованием

Названия таблицы и пользователя пусть отличаются от ранее использованных добавлением цифры 1 в конце: test1/test\_user1.